COMPUTER VISION-BASED ASSESSMENT OF COASTAL BUILDING STRUCTURES DURING HURRICANE EVENTS

by

ZIXIANG ZHOU

A Dissertation Submitted to the

School of Graduate Studies

Rutgers, the State University of New Jersey

In partial fulfillment of the requirements

For the degree of

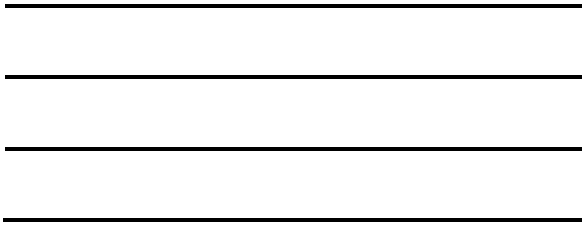
Doctor of Philosophy

Graduate Program in Civil and Environmental Engineering

Written under the Direction of

Jie Gong

And approved by



New Brunswick, New Jersey

JANUARY, 2018

ABSTRACT OF THE DISSERTATION

COMPUTER VISION-BASED ASSESSMENT OF COASTAL BUILDING STRUCTURES DURING HURRICANE EVENTS

By ZIXIANG ZHOU

Dissertation Director:

Jie Gong

Severe hurricane events have been occurring across the United States, threatening both highly developed urban areas and distressed island communities. Assessment of building damages due to hurricane events is a critical element in disaster management as it supports not only search and recues operations but also provides insights into the performance of existing planning and building practices. But unlike many other extreme weather events, hurricanes can topple an entire region for an extended amount of time, creating a daunting task for traditional foot-on-ground building damage assessment approaches.

The overarching goal of this study is to create and test a computational framework to leverage big spatial data acquisition and processing technologies for automated building damage assessment. The specific aims of this study include: (1) formulating a cohesive and multi-scale damage assessment approach that considers the continuously evolving data sources and damage assessment needs during different phases of disaster management; (2) developing algorithms for rapid building damage assessment with airborne lidar data, which are typically collected immediately after the landing of hurricane events; (2) developing algorithms for component-level building damage assessment with high-resolution ground-based lidar data; (3) charactering the

performance of image based 3D reconstruction for damage assessment; (4) developing robust image alignment algorithms for geo-registering post-disaster image data from varied sources to realize the fusion of heterogeneous point cloud and image data for comprehensive damage assessment. The proposed methods were applied on several geospatial data sets collected during Hurricane Sandy. The results are compared with the ground truth which was created by a manual labeling process. The results show that the proposed methods are capable of conducting damage assessment of building structures autonomously and at different resolution and extracting useful damage information to support building performance modeling. Future research of this study will be focused on leveraging high performance computing capabilities to accelerate the damage assessment process.

Acknowledgements

The journey towards Ph.D. at Rutgers has become an unforgettable memory in my life. And it has always been my honor to be advised by Prof. JIE GONG, and worked with him during the past wonderful years. It was his guidance and encouragement that broadened my horizon and knowledge in terms of professional aspects. His kindly support as a mentor and a friend, also led me through the struggle of research.

I would also like to, sincerely, thank my committee members, Prof. FRANKLIN MOON, Prof. HUSAM NAJM, and Prof. ZHIGANG ZHU. For their valuable and supportive guidance throughout this dissertation research. And the thoughtful discussion during my life of Ph.D. at Rutgers.

I would also like to thank XUAN HU, YI YU, MENGYANG GUO, ZHAOJIN WANG, and XIAOYANG WU. Without their support and help, it would be not possible for me to complete my research.

Also, I would like to express my sincere appreciation to all my friends. It is them who made my life at Rutgers wonderful and memorable.

Finally, I would like to dedicate my sincere and cordial appreciation to my loving parents, Mr. DONGFENG ZHOU, Ms. XIAO CHEN. And also special appreciation to HUIQING SUN. For their endless love and support.

Table of Contents

List of Illustrations

List of Tables

List of Algorithms

Chapter 1 : Introduction

1.1 Research Background

In the past decade, humankind has suffered tremendous loss due to the increasing amount, intensity, and frequency of extreme weather events. Sadly, the effects of increased occurrence and ferocity of events are disproportionately felt by the poorest and most vulnerable members of our global population. Among many types of extreme weather events, tropical cyclone is one of the most devastating events affecting many heavily populated areas in the United States and around the world. For example, the recent Hurricane events in the United State alone as shown in Table 1.1 have caused hundreds of billions of dollars of damages to properties and business and have claimed thousands of human lives. One often severely impacted structure during these events are buildings, especially residential homes.

Table 1.1. Recent Major Hurricane Events in U.S.

Name	Date	Total Deaths	Total Damages	Building Damages
Hurricane Matthew	Oct, 2016	15	$6 Billion	102 Destroyed; 100000 Affected
Hurricane Sandy	Oct, 2012	159	$62 Billion	500 Destroyed; 5000 Major Damage; 24000 Minor Damage
Hurricane Katrina	Aug, 2008	1833	$108 Billion	300000 Affected
Hurricane Ike	Sept, 2008	195	$38 Billion	323800 (Affected in Cube)

In the aftermath of large coastal storm events, in fact, any type of large-scale natural disasters, damage assessment is an absolutely essential task as it impacts nearly every aspects of disaster

management. Essential disaster management functionalities including sheltering, temporary housing, search and rescue, debris removal, reestablishment of major transportation links, utilities and communication restoration, hazard mitigation, demolition, and redevelopment planning all depend on the information from damage assessment in order to make informed decisions.

After the occurrence of natural hazard, two types of damage assessment protocols are conducted sequentially (FEMA 2016). Immediate after a disaster, rapid damage assessment is launched primarily for gathering information that can be used to quickly determine whether a structure is safe to enter or not, or to determine whether a structure is collapsed or not. In addition, topographic change and debris distribution are also quickly evaluated to support the decision of routing of evacuation, rescue, and post-hazards evaluation. After the rapid assessment, detailed damage assessment is carried out primarily for repair estimation and damage modeling. In this stage, the extent of structure damages is quantitatively estimated so that recovery teams would be able to determine the damage status of each individual objects.

Conventional post-disaster building damage relies on manual damage assessment by ground assessment teams (Chiu et al. 1999, Crandell et al. 2005, Gurley et al. 2006, Curtis et al. 2012). These approaches are generally labor intensive and are difficult to be scaled up to cover large disaster areas without incurring significant cost. This makes it impractical for large-scale rapid assessment. With the progressive development in remote sensing technologies, remote sensing aided damage assessment has become a feasible alternative solution to manual damage assessment. In particular, modern big spatial data acquisition and processing technologies such as laser scanning from airborne, mobile, or static platforms and dense 3D reconstruction using photos from a variety of sources are playing an increasingly important role for decision support in disaster preparation, response, and recovery operations as they greatly expand the ability of collecting disaster data. For instance, in recent years, because state and federal agencies have made airborne LiDAR (Light Detection and Ranging) data collection a priority, post-storm LiDAR collection is now routine after large surge event and vast amounts of disaster data are now

freely available online (e.g. NOAA's Digital Coast). In another example, emerging high resolution sensing systems such as terrestrial/mobile LiDAR have also been deployed for damage data collection during recent events such as Superstorm Sandy, generating an unprecedented amount of 3D geospatial data. Lastly, volunteered geographic information (VGI), such as geo-tagged disaster photos, is a new breed of disaster data which further enriches but also complicates disaster data analysis. The geo-tagged photos themselves are not 3D. But 3D information can be recovered by either running Structure-from-Motion followed by multi-view stereo on these unstructured images or registering these images with LiDAR data.

Significant research effort have been devoted in the past to post-disaster damage using either imagery-based approaches (Barnes et al. 2007, Vijayaraj et al. 2008) or point cloud-based approach (Elberink et al. 2012). Compared with conventional field survey practices, remote sensing-based approaches have the following advantages:

1) Safety: Compared with manual field survey, which requires the survey team to approach each building one by one, remote sensing technologies can observe objects and extract condition data in distance.
2) Speed: Compared with manual field survey, remote sensing technologies often involve the use of advanced sensing technologies which are mounted on fast moving platforms such as satellite, man and unmanned airplanes, mobile vehicles, or boats. This makes it feasible to collect data on large areas in a short amount of time.
3) Perception: Remote sensing technologies provide large-scale perceptions of the extent of damage, offering a big picture of what is going on in the impacted area. Manual field survey provides rich detail, but lacks the capability in facilitating comprehension of situational awareness at a large-scale.
4) Information: Compared with field survey, remote sensing technologies are able to reconstruct the geo-referenced 3D model of the evaluated objects with actual sizes. This

brings alternative solution to manual damage assessment as engineers and agencies can revisit the damaged areas virtually in the office. Meanwhile, there is no need to return to the field to reevaluate buildings if some components are omitted at the first round of evaluation.

As shown in Figure 1.1, the comparison of remote sensing-based and field survey-based assessment is summarized in radar plot manner. The comparison is conducted qualitatively. As observed, remote sensing-based approach overwhelms the field survey approach in safety, speed, and perception aspects significantly.

Use of remote sensing technologies to assess disaster impacts is not a new topic. A particularly well-studied area is building damage assessment following major disasters such as earthquake, tsunami, and tornado. The type of data used in these studies include satellite imagery, airborne imagery, airborne lidar, InSAR, oblique imagery, point clouds reconstructed from overlapping images, and terrestrial lidar. There are two dominant approaches used in these studies to detect and quantify damages. The first is based on classifying damages based on damage features such as surface smoothness, local features, intensity distribution, and height information extracted from mono-temporal data (Khoshelham et al. 2013, Kashani et al. 2014, Galarreta et al. 2015, Kashani et al. 2015, Kashani et al. 2015, Vetrivel et al. 2015, Vetrivel et al. 2016). The second is based on detecting and characterizing changes between multi-temporal data sets (Vögtle et al. 2004, Huang et al. 2014, Lucieer et al. 2014, Pang et al. 2014, Coulter et al. 2015, Vetrivel et al. 2016).

Table 1.2 summarizes the research conducted regarding the building damage assessment and change detection using multiple remote sensing technologies.

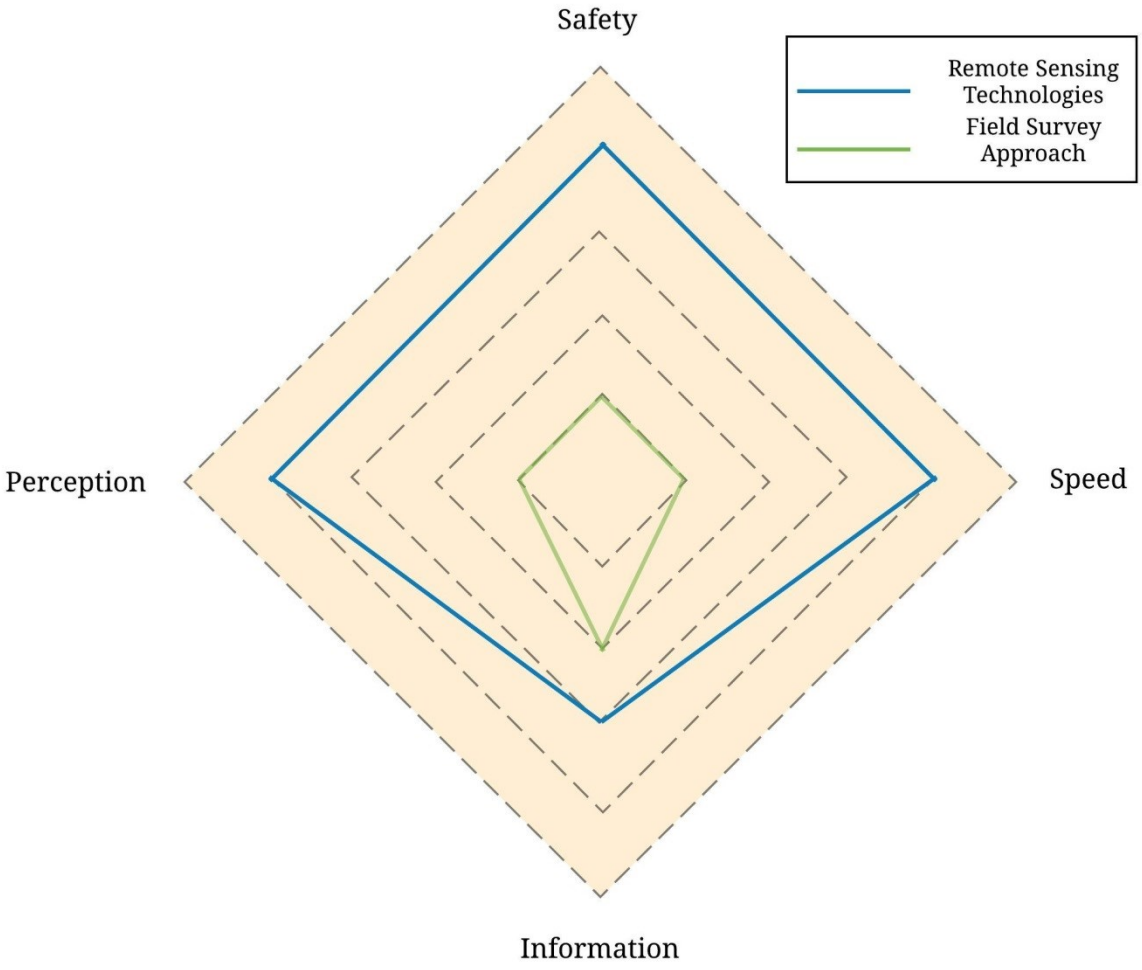


Figure 1.1. Remote Sensing Technologies for Damage Assessment

Table 1.2. Research about Damage Assessment using Remote Sensing Techniques

Research	Technique	Damage	Category	Methodology	Features
Schweier et al. (2004), (2006)	ALS	Earthquake	▪ Inclined layer; ▪ Pancake collapse; ▪ Debris heaps; ▪ Overturn collapse; ▪ Overhanging elements	Comparison between pre-event and post-event data	1) Height difference; 2) Volume reduction; 3) Footprints; 4) Inclination change
Rehor et al. (2008)	ALS	Earthquake		Comparison between pre-event and post-event plane segments	Plane extraction: 1)RANSAC; 2)Region-growing
Yonglin et al. (2010)	ALS	Earthquake	▪ Roof inclination	Compute the roof normal and axis line direction	Group roofs into: 1) Symmetric roof; 2) Partial symmetric roof; 3) Asymmetric roof
Elberink et al.	ALS		▪ Building collapse	Machine learning-based	1) Number of points per

(2012)				approach with post-event data only	segment; 2) Mean height; 3) Planarity; 4) Standard deviation of intensity
Khoshelham et al. (2013)	ALS	Earthquake	▪ Intact buildings; ▪ Damage buildings	Machine learning-based approach with post-event data only	1) Number of points per segment; 2) Mean height; 3) Planarity; 4) Standard deviation of intensity
He et al. (2016)	ALS	Earthquake	▪ Inclined layers; ▪ Pancake collapse; ▪ Debris heaps; ▪ Overturn collapse; ▪ Overhanging elements	Use shape descriptors to detect damage	1) Shape contours; 2) Chaos index

Kashani et al. (2015)	TLS	Tornado	▪ Roof damage	Statistically count the distribution of intensity value of different roof materials	1) Point cloud intensity
Kashani et al. (2015)	TLS	Tornado	▪ Roof damage	Cluster the damage patterns from the colored point cloud	1) Point cloud intensity; 2) Image color
Zhou et al. (2016)	MLS	Hurricane/Storm Surge	▪ Pipeline Risk	Use the building and terrain change to estimate the buried and aboveground pipeline risks	1) Oriented point-to-point distance; 2) Building component level change detection
Gong et al. (2014)	MLS & ALS	Hurricane/Storm Surge	▪ Building damage; ▪ Debris	Comparison between pre-event ALS and post-event MLS	1) Point to point distance
Zhou et al.	Image & MLS	Hurricane/Storm	▪ Building components	Use image-based 3D	1) Plane segmentation;

(2015)		Surge	damage	reconstruction to detect the building components damage	2) Boundary points extraction; 3) of plane fitting; 4) Normal of planes
Kashani et al. (2014), (2016)	TLS & Aerial Imagery	Tornado	▪ Roof damage	Comparison between pre-event aerial imagery and post-event TLS data	1) Point cloud intensity; 2) Image color
Barnes et al. (2007)	Satellite Imagery	Hurricane / Storm Surge	▪ Building roof; ▪ Parking lot; ▪ Vehicle; ▪ Road	Use σ-tree-based structure to detect the wind-induced damage	Multiple image features: 1) Building roof edge; 2) Building roof corner: 3) Ship cargo containers; etc.
Chen et al. (2016)	Aerial Imagery			Generate RGB-D map from aerial imagery, then depth difference is extracted to quantify the	1) SHOT descriptor; 2) HoG; 3) DSTD

				change	
Galarreta et al. (2015)	TLS/MLS & Oblique Imagery		▪ Wall façade; ▪ Roof facet; ▪ Rubble piles	Use UAV image to assist the building damage (addressing the damage level ambiguity)	
Geiß et al. (2015)	Satellite / Aerial Imagery	Earthquake	▪ Masonry; ▪ Reinforced concrete low rise; ▪ Reinforced concrete high rise; ▪ Unreinforced masonry; ▪ Timber frame residential; ▪ Timber frame non-residential; ▪ Steel frame	Machine learning-based approach uses fusion of multi-remote sensing data for seismic building structural types determination	Multiple features including: 1) Extent; 2) Shape; 3) Color; 4) Texture; 5) Spatial Context; 6) 3D characteristics
Hua et al.	Oblique	Earthquake	▪ Intact building;	Detect building collapse	1) HoG

(2016)	Imagery		▪ Collapsed buildings	from UAV-based oblique imagery	
Janalipour et al. (2016)	Satellite Imagery	Earthquake	▪ Very heavy damage; ▪ Heavy damage; ▪ Negligible to moderate damage	Integrate a decision-making system and adaptive network-based fuzzy inference system to extract building damage degree	1) Area; 2) Length; 3) Solidity; 4) Convexity; 5) Roundness; 6) Form factor; 7) Rectangle fit
Kahraman et al. (2016)	Satellite Imagery		▪ Intact buildings; ▪ Damage buildings	Use local self-similarity descriptor to assess damage with pre-event and post-event EQ image	1) Self-similarity descriptor
Tian et al. (2014)	Satellite Imagery		▪ Positive change; ▪ Negative change; ▪ Rebuilt	Use joint of height changes and Kullback-Leibler divergence similarity to detect the	1) Height change; 2) Similarity measurement

				building changes based on stereo matching and DSM generation	
Li et al. (2015)	Aerial Imagery	Earthquake	▪ Roof hole	Gradient Map is first generated, and a Chinese franchise model is used to learn an unsupervised model to detect roof holes	1) Image color features; 2) Image shape features
Tong et al. (2012)	Satellite Imagery	Earthquake	▪ Intact building; ▪ Collapsed building	Construct the 3-D model using stereo matching, and detect the collapsed buildings through height change and area reduction	1) Height difference; 2) Area reduction of DSM
Tong et al. (2013)	Satellite Imagery	Earthquake	▪ Intact building; ▪ Collapsed building	Based on footprint and height data, the 3-D model is generated first. Then the theoretical	1) Theoretical shadow polygon; 2) Actual shadow

				shadow area is computed as reference to detect the collapsed buildings	polygon
Vetrivel et al. (2015)	Oblique Imagery	Earthquake	▪ Debris; ▪ Rubble piles; ▪ Heavy spalling	Train a Visual-Bag-of-Words to classify pixel-wise features to detect structural damage	1) HoG; 2) Gabor wavelets; 3) SURF
Torok et al. (2013)	Image		▪ Concrete crack	Use image-based 3D reconstruction to detect the concrete crack	1) Mesh normal
Hatzikyriakou et al. (2015), Xian et al. (2015)	Image	Hurricane / Storm Surg	▪ Wall collapse; ▪ Roof collapse; ▪ Foundation scouring; ▪ Wall cladding	Use ground-based imagery data to assist the field survey and evaluation	
Yeum (2016)	Image	Earthquake	▪ Building collapse; ▪ Cracks	Use computer vision based techniques to detect	1) Multiple color space; 2) HoG;

			visual patterns from imagery. Use deep learning techniques to detect cracks	3) High-level features
Cha et al. (2017)	Image	▪ Concrete crack	Use convolutional neural network to learn and detect the concrete cracks from	1) High-level features

In contrast to this plethora of studies devoted to building damage assessment after earthquake, tsunami, and tornado events, use of remotely sensed data, in particular high-resolution imagery or point cloud data, to assess building damages after hurricane events has drawn much less attention. For the very limited amount of studies that have investigated hurricane damage assessment with remotely sensed data, virtually all of them rely on manual interpretation of imagery or point cloud (Gong et al. 2014, Hatzikyriakou et al. 2015, Xian et al. 2015). Furthermore, none of these studies has investigated the use of airborne lidar data for hurricane damage assessment.

The pattern of seismic-induced damage varies significantly from the pattern of hurricane-induced damage. As illustrated in Figure 1.2, earthquake induced building damage is mainly caused by the large horizontal movement of the earth and the huge lateral shear force caused by ground acceleration. Tornado events impact the building primarily through wind effects, and tsunami events damage buildings primarily through wave and storm surge actions. In contrast to these events, damages to buildings during hurricanes are often due to the combined wind, storm surge, and wave action effects. In addition to building damage, storm surge depth is also a dominant factor in influencing the extent of building property damages. While the studies in earthquake, tornado, and tsunami damage assessment have shown the promising capabilities associated with remote sensing technologies, it is reasonable to expect that use of remote sensing technologies for hurricane damage assessment requires dedicated methodology development because of the complexity in hurricane damage mechanisms.

Figure 1.2. Pattern of Seismic-induced and Flooding Surge-induced Building Damage.

Figure 1.2 (a) is the collapsed building in Wenchuan earthquake, 2008[1]; (b) is the collapsed building in Hurricane Sandy, 2012[2]; (c) is the damaged roof in Andrew, 1992[3]; (4) is the building damage due to storm surge in Hurricane Sandy, 2012[4].

Crowd-Sourcing and Volunteered Geographic Information (VGI) are widely considered potential game changers in damage assessment. The rise of social media has provided an unconventional way of sensing disaster damages through a so-called social sensing mechanism.

[1] http://agnesngoy.blogspot.com/2014/11/case-study-2008-sichuan-earthquake.html
[2] http://www.nbcphiladelphia.com/news/national-international/Sandy-Retired-from-List-of-Hurricane-Names-202549971.html?_osource=AMP
[3] http://www.floridadisaster.org/hrg/content/risks/risks_index.asp
[4] https://weather.com/news/news/storm-surge-hurricane-sandy-isaac-20121227#/1

Table 1.3 summarizes the studies regarding crowd-sourcing-based and VGI-based damage assessment. Some of these studies have focused on object-based seismic-induced damage assessment by the integration of crowd-sourcing imagery and VGI data. It also can be noticed that great attention has been paid on mapping the flooding surge at global scale. Inundation impact is a critical concern to people who are impacted by hurricane events. Unlike building damage which is statically remained until the recovery procedure begins, the flooding surge height varies quickly and is hardly captured by organized assessment team. To address this issue, a crowd sourcing-based (Howe 2008), or called volunteered geographic information-based concept was introduced in (Goodchild et al. 2007). While many of these studies have clearly demonstrated the potential of social media and VGI-based approach, most of these studies rely on geotags in the social media and VGI data to localize these data. Use of unstructured VGI or crowd sourced image data for damage assessment is an open research field.

Table 1.3. Research about Storm Surge Measurement using Crowd-sourcing and VGI data

Research	Technique	Category	Scale	Damage	Methodology
Lane et al. (2003)	▪ Point Cloud DTM; ▪ Imagery	Flood height measurement	Large scale	Flood Surge	Compare the following methodologies: 1) Point cloud segmentation based upon image analysis; 2) Point cloud segmentation based upon digital flood outlines using watermark line identification; 3) Imagery analysis
De Albuquerque et al. (2015)	▪ Social media text-tweets; ▪ Authoritative spatial data	Explore relation between social media events and disaster events	Large scale	Flood Surge	Relate crowd sourcing data (imagery, tweets) to authoritative spatial data to explore the statistical pattern in the occurrence of flood-related tweets and the associated events
Barrington et al. (2012)	▪ Crowd-sourcing aerial imagery	Earthquake-induced damage detection	Object-based scale	Earthquake	Case study regarding multiple extreme earthquake disaster

Crooks et al. (2013)	▪ Social media text-tweets; ▪ Authoritative spatial data	Explore relation between social media events and disaster event	Large scale	Earthquake	Leveraging social media text as a distributed sensor system to explore the relation between the social media response and actual hazards event
Schnebele et al. (2013)	▪ Social media imagery; ▪ Tweets; ▪ Geo-located videos; ▪ Authoritative data	Flood mapping	Large scale	Hurricane & Flood Surge	Integrate the non-authoritative crowd sourcing data with authoritative geospatial data to conduct the flood damage mapping
Schnebele (2013)	▪ Social media imagery; ▪ Remotely sensed imagery; ▪ DEM' ▪ Authoritative data	Flood mapping	Large scale	Flood Surge	Multi-temporal remotely sensed imagery data are registered. The social media image are then used to improve the classification of water mark pixels.
McDougall	▪ Volunteered	Flood mapping	Large	Flood Surge	Individual imagery is collected and

(2011)	geographic information; ▪ Social media data		scale		the flood height is measured. The flood map is then obtained through integration of VGI and social media data
Triglav-Čekada et al. (2013)	▪ Raster flood maps; ▪ Volunteered geographic information	Flood mapping	Large scale	Flood Surge	The VGI data is obtained through public call. The web-based search is integrated. Then the 3-D flood mapping is obtained through manual fitting approach
Clinch et al. (2012)	▪ Multi-temporal topographic data; ▪ Storm data such as peak water elevation	Flood mapping	Large scale	Flood Surge	Integration of topographic data and storm data is used to estimate the flooding map using 1) temporal dune cross section analysis; 2) dune ridge analysis; 3) wrack line surge estimation; and 4) inundation visualization
Schnebele et al.	▪ Volunteered geographic	Flood mapping	Large scale	Flood Surge	Spatial temporal damage assessment is firstly conducted by fusion of social

(2014)	information; ▪ Social media dtaa				media data and VGI data. Then the authoritative data is integrated to improve the performance, and finally the road network layer is overlaid for the mapping
Rosser et al. (2017)	▪ Social media imagery; ▪ Point cloud data; ▪ Topographic model	Flood mapping	Large scale	Flood Surge	Social media imagery is firstly integrated with georeferenced data. The point cloud DTM is used to generate the fuse model. And finally a Bayesian probabilistic method is adopted to explore the relation between evidence variables

1.2 Research Objectives and Research Questions

The central research question to be addressed in the dissertation is: Can big spatial data acquisition and processing technologies including laser scanning from aerial and mobile platforms and 3D dense reconstruction provide a systematic and automated way to assess building damages in the aftermath of major hurricane events? To answer this question, this dissertation is structured to address the following challenges:

1) After the occurrence of hurricane events, an initiative wave of airborne lidar data collection is often conducted to assess topographical changes. These data sets often cover large geographic regions. It is challenging to automatically and rapidly process these data sets to detect building objects and assess their conditions. It is also unknown that to what extent can building damage be assessed using airborne lidar data which tend to be very sparse.
2) Following rapid response, a detail damage assessment is often required. Mobile lidar has greatly expanded the data collection capability during natural disasters. Unlike airborne lidar data, mobile lidar data have very high resolution. This makes mobile lidar a useful tool for detail damage assessment. However, the complexities in building configuration make detail damage assessment a challenging task for computers. Therefore, how to process mobile lidar data efficiently and intelligently to detect building damage in details is a question to be answered in this research.
3) Images are indispensable sources of information during hurricane events. Immediately following hurricane events, photos of impacted buildings start coming in from many different channels such as first responders, residents, and social media outlets. These photo data are extremely helpful for rapid and later on detailed damage assessment. Technologically, it is now feasible to recreate a virtual reality of impacted sites to

facilitate damage assessment. However, to what extent model reconstruction from these photos can be used for assessing hurricane damages is unknown.

4) Many of the photos of damaged buildings from different sources are not georeferenced, which makes it difficult to figure out the correspondence between images and building objects. The difficulty in tagging images to relevant building structures leads to missed opportunities in assessing damages in a timely, accurate, and comprehensive fashion. A fundamental challenge is image-to-image and image-to-3D data alignment for fusing multi-sourced visual disaster data. If this can be accomplished, it opens doors to many innovative applications such as image based storm surge height measurement and recreation of inundation scenarios.

1.3 Research Methodology

To address the aforementioned research challenges, a framework for processing data collected by big spatial data acquisition technologies for damage assessment is shown in Figure 1.3. In the figure, the dashed arrow indicates the sequence of different stages, and the solid arrow represents the operation at a start point is applied to the operation at an end point. Immediately after the occurrences of hurricane events, building objects are detected and extracted from airborne lidar point cloud using a convolutional neural network-based approach (Figure 1.4 (a)). A rapid building damage assessment is conducted using multi-temporal airborne lidar data sets (Figure 1.4 (b)). The detection is conducted through comparison of multiple features computed from multi-temporal point cloud data. Later on, an automated semantic parsing and damage detection framework is designed to conduct component level damage assessment (Figure 1.4 (c)). To integrate VGI- or crowdsourcing- based image data into the framework, a component level damage assessment approach using fusion of point cloud and image-based reconstruction model is designed (Figure 1.4 (d)). In this approach, 3D reconstruction techniques are employed to generate dense point cloud data, and their potential for supporting component-level damage is

evaluated. The last stage of the proposed framework is a framework of using the fusion of geo-tagged street-level image and point cloud data to estimate the component-level building damage (Figure 1.4 (e)), which address the fundamental challenges in image-to-image and image-to-model alignment in the disaster science context.

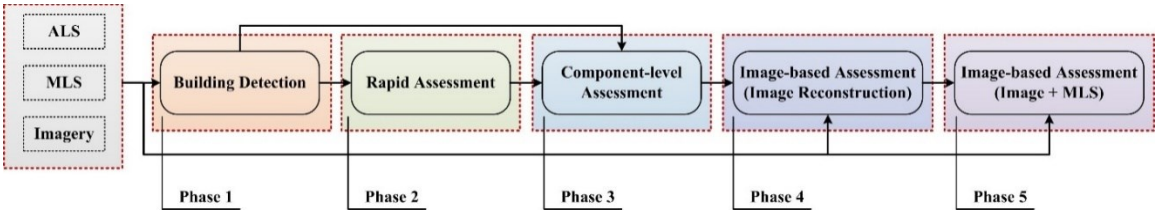


Figure 1.3. The Overview of Framework

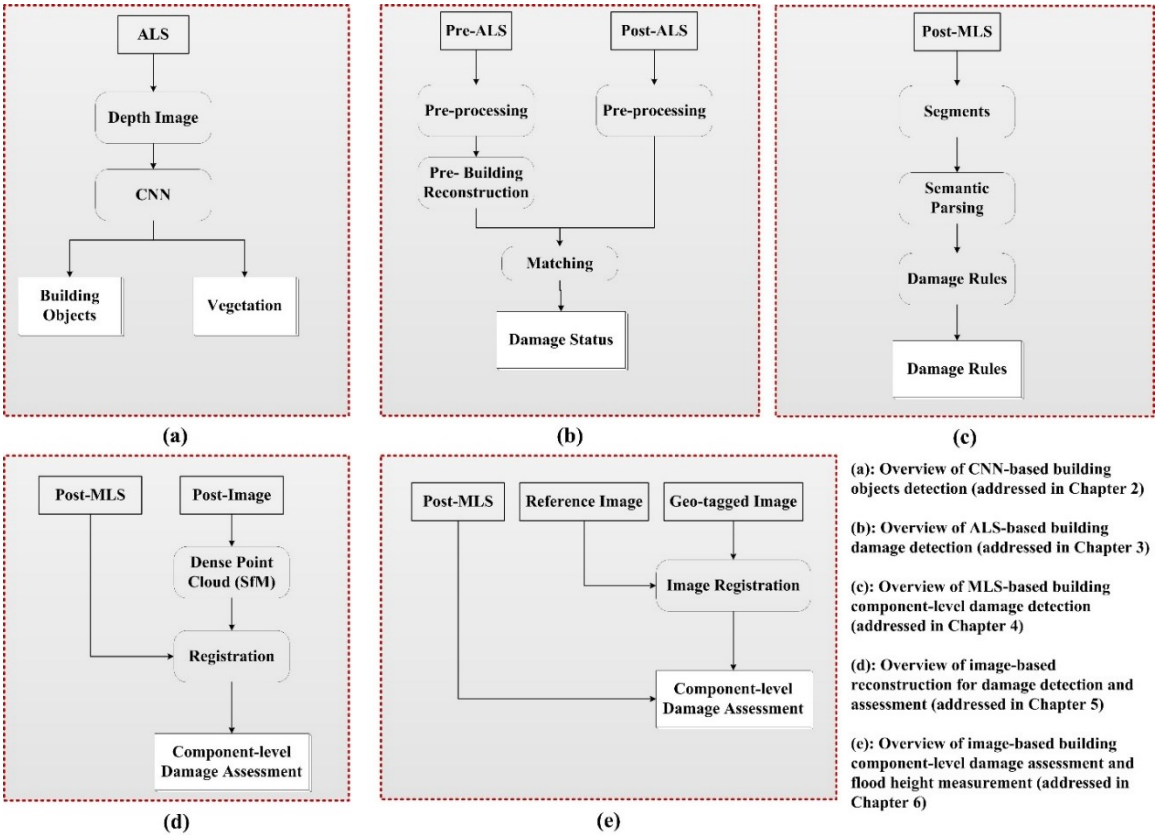


Figure 1.4. Overview of Components of the Proposed Framework

Comprehensive case studies are carried out to validate the proposed system framework. The airborne lidar data collected before and after Hurricane Sandy, the mobile lidar collected in the

aftermath of Hurricane Sandy, and the image data collected by different damage assessment teams are used as the training and testing data sets in this research. All these data sets are collected at Ortley Beach and Seaside Heights in New Jersey, which are among the most severely damaged areas during Hurricane Sandy.

1.4 Research Contributions

The overall contribution of this dissertation research is designing, developing, and validating a unified framework and algorithms for fusing and interpreting emerging spatial disaster data sets for automated and intelligent damage assessment in the aftermath of major hurricane events.

The individual components of this research provide further contributions along the following fronts:

1) The design of a high performance building detection approach that requires minimum preprocessing of airborne LiDAR data (i.e. no ground detection or any other kind of classification is needed), works on both pre- and post-disaster data, and is autonomous and highly efficient. The study also represents the first study in using deep neural networks for building detection in airborne LiDAR data sets.
2) The design of a hierarchical framework that is able to 1) characterize the extent of building damage; 2) deal with the scenario that roof facets are severely damaged so that they are not able to be detected; and 3) propose a framework that is able to detect and assess the hurricane-induced building damage at residential area with rich vegetation.
3) The design of a fully automated method that is able to 1) semantically parse dense point cloud data of residential buildings, regardless of whether they are damaged or not, into major building components including roof, exterior walls, columns, and handrails; 2) apply to post-event point cloud data only, because the proposed methodology is based on rule-based topological assumptions.

4) The design of an imagery reconstruction-based approach that is able to assess the component level building damage including wall cladding failure, balcony inclination, opening size, etc.
5) The design of an efficient framework that is able to
   i. align the arbitrary image with point cloud and retrieve the projection correspondence between the image and point cloud; and
   ii. measure the storm surge height from image data which could not be addressed from point cloud data.

Chapter 2 : Automated Residential Building Detection From Airborne Lidar Data With Deep Neural Networks

Detection of building objects in airborne LiDAR data is an essential task in many types of geospatial data applications such as urban reconstruction and damage assessment. Traditional approaches used in building detection often rely on shape primitives that can be detected by 2D/3D computer vision techniques. These approaches require carefully engineered features which tend to be specific to building types. Furthermore, these approaches are often computationally expensive with the increase of data size. In this paper, I propose a novel approach that employs a deep neural network to recognize and extract residential building objects in airborne LiDAR data. This proposed approach does not require any pre-defined geometric or texture features, and it is applicable to airborne LiDAR data sets with varied point densities and with damaged building objects. The latter makes this approach particularly useful in damage assessment applications. The research results show that the proposed approach is capable of achieving the state-of-the-art accuracy in building detection in these different types of point cloud data sets.

2.1 Introduction

LiDAR is a remote sensing technology that measures distance by illuminating a target with a laser and analyzing the reflected light. Among other applications, LiDAR is useful for detailed mapping of terrain, elevation, structures, and change detection in disaster management at several levels. The field is rapidly maturing in capabilities, applications, and utility. In recent years, because state and federal agencies have made airborne LiDAR data collection a priority, LiDAR collection is now routine after large surge event and vast amounts of point cloud data are now freely available online (e.g. NOAA's *Digital Coast*). LiDAR data have rich use cases in city management and damage assessment applications. One crucial processing task in these LiDAR applications is building detection.

Automated building detection from airborne LiDAR data sets is a challenging task. Airborne LiDAR systems collect primarily nadir LiDAR observations, leading to sharp drops in data density from the rooftops to the vertical facades. Because of the lack of LiDAR observations on the vertical facades, building detection has to rely on detection of roof objects (Wang et al. 2009, Tseng et al. 2016). Nevertheless, because of the attitude of airborne LiDAR data collection, most airborne LIDAR systems produce sparse point cloud data sets, posing challenges to detect planar roof objects and their boundaries. In addition to these challenges, detection of building objects from airborne LiDAR data sets that were collected after major disasters is further complicated by damaged or collapsed roof objects. In these kinds of scenarios, specialized features need to be designed in order to realize automated detection of buildings based on roof objects. However, use of special features that are hand engineered for different types of data sets inevitably limits the usefulness of these building detection methods in broader applications.

With the rapid emerging of deep learning techniques, multiple types of deep learning frameworks have been developed and applied in visual recognition and classification tasks (Sermanet et al. 2013, He et al. 2016, Liu et al. 2016). It has been shown that Convolutional Neural Network (CNN) is capable of recognizing high-level objects by automatically learning a set of abstract features. Recently, Hu et al. (2016) proposed a CNN-based approach to extract

Digital Terrain Model (DTM) from airborne point cloud data sets. In their research, the point clouds were first converted to feature images which were fed into a CNN to learn the abstract features of ground and non-ground points. Ghamisi et al. (2016) proposed a deep learning based framework to extract the spatial information from the fusion of hyperspectral and LiDAR data. In this research, I propose a CNN-based approach to automatically detect building objects from point clouds with different densities and building conditions (damaged, intact, or collapsed). More specifically, the point cloud is first converted to a gray-scale image with the value of each pixel representing the height of the corresponding point. A deep CNN is then trained to learn the features in the converted images. The experiment shows that the proposed approach is able to extract the building objects very reliably from airborne LiDAR data.

The major contributions of this study include the design of a high performance building detection approach that requires minimum preprocessing of airborne LiDAR data (i.e. no ground detection or any other kind of classification is needed), works on both pre- and post-disaster data, and is autonomous and highly efficient. The study also represents the first study in using deep neural networks for building detection in airborne LiDAR data sets. This new approach is capable of achieving the state-of-the-art accuracy (93%) in building detection in these different types of point cloud data sets.

2.2 Related Work

Building detection from airborne LiDAR data has been extensively studied in the context of urban reconstruction and modeling, damage assessment, and many other city management applications. The major approaches used in building detection studies include, but are not limited to, the use of shape descriptors, contours, and prior model data (such as building footprint databases and roof geometric models).

Buildings as man-made structures have strong geometric cues that can be exploited for building detection. For example, Dorninger et al.(2008) proposed an automatic building extraction approach which relies on the detection of plane segments. For community-scale

building extraction, this approach becomes computationally very expensive. To address this issue, the authors proposed to use candidate seeds that are initialized first using hierarchical clustering and to detect building objects around the seeds. Lafarge et al.(2011) developed a large-scale city modeling system which leverages a combination of generic shape descriptors such as plane, sphere, cylinder, etc., to model the building objects. However, this approach requires dense airborne point cloud as inputs to generate Level-of-Detail (LoD) models. Awrangjeb et al.(2014) developed a framework for automatic building extraction. In their approach, a removal of wall points was implemented before the roof segmentation. And then the height and local neighborhood are used to cluster the roof points. And Sun et al. (2013) proposed an automated building object extraction framework in which point cloud data were first classified into ground and non-ground using the graph-cut algorithm, then the non-ground points were segmented using a distance-based clustering method, and eventually the building objects were detected if connected planes were detected. Niemeyer et al. (2014) proposed to use conditional random field to obtain the probabilistic classification of the point cloud. In this way, the contexture relationship among points could be taken advantage.

Another major approach used in building detection from airborne LiDAR data is the use of contours, a geometric representation that could facilitate the detection of building objects. For example, a Markov Random Field-based approach for detecting the building contour is presented in (dos Santos Galvanin et al. 2012). This study leverages MRF to retrieve the regularized contours lines of each building objects, and does not require pre-initialized or recognized building candidates as inputs. However, the MRF-based contours optimization will be hampered if the buildings are located in densely vegetated areas. Similarly, Wu et al.(2017) proposed an approach using contours to represent the building objects. Although the approach performs well on metropolitan area datasets, residential areas with low-rise homes and buildings will likely pose challenges to this approach. This is because as for metropolitan area, the elevations and sizes of

buildings are often remarkably higher and larger than other objects such as vegetation, which makes it relatively easy to distinguish building objects from objects like vegetation. In addition, it is challenging to generate enclosed contours in residential areas if the point cloud data points are relatively sparse. To this end, Zhou et al.(2013) proposed an approach that is able to reconstruct residential buildings in areas with rich vegetation. Their approach utilized the multi-passes of laser scanning at trees to identify buildings from vegetation. However, this approach requires dense point cloud. For low point density scenario, the proposed approach will have problem in detecting 2.5D characteristics (roofs and vertical walls).

Use of prior data as such as building footprints and knowledge about building forms to inform building detection algorithms is another approach that has been studied in the past. A roof ridge decomposition-based roof facets extraction approach was introduced by (Fan et al. 2014). However, their approach requires the use of building footprints retrieved from the OpenStreetMap database to clip out each building object before extracting roof objects. This requirement restricts the use of their approach only in scenarios where building footprints are available and there are no major changes to building footprints. This reduces the applicability of their method in post-disaster scenarios. Wang et al. (2006) proposed a system to reconstruct the building footprints by extracting initial footprints using neighbor-searching, then the regularized footprint was optimized using a Bayesian MAP algorithm. Huang et al. (2013) proposed a statistical approach that does not require the building footprints as input. In their approach, the pre-defined roof primitives are used Markov Chain Monte Carlo is developed to reconstruct the building roofs. However, this approach might fail to detect a building object if the candidate contains too few data points.

The above approaches have focused mostly on intact buildings. Detection of damaged buildings after major disasters is of particular interest to emergency management. Detection of damaged buildings is significantly more challenging than detection of intact buildings, because the damaged buildings often have more complex and un-defined features. Nevertheless, the state-

of-the-art techniques of building detection from damaged area are still built on the detection of shape primitives. For example, Yonglin et al. (2010) proposed a plane-based building damage detection framework. In their research, the building roofs were detected and classified into pre-defined categories at first, and the geometric axis line of the building was then extracted. However, in cases building roofs have been severely damaged or collapsed, the rooftops can hardly be detected using conventional plane detection algorithms. In a more recent study, He, M. et al. (2016) proposed a building detection approach for damaged buildings by extracting building contours followed by the use of chaos index to describe the shape of building roofs. A noticeable issue in this approach is how to distinguish the contours of damaged buildings from vegetation. Due to destruction, the boundaries of a damaged building will hardly preserve features like straight lines. Consequently, this introduces difficulties in classifying vegetation from buildings.

An apparent competition to point cloud based building detection is the use of satellite and airborne imagery for building detection, in particular considering many airborne LiDAR systems now collect imagery along with point cloud data during data collection flights. There is also a resurging of interest in using imagery data for various object detection tasks, largely fueled by the rise of deep neural networks. With deep learning, it is possible to let computers learn features themselves instead of imposing the tasks of extracting human-designed features such as plane segmentation and boundary extraction. Along this line, Vakalopoulou et al. (2015) proposed to use convolutional neural network to detect building objects from very high-resolution satellite imagery. A MRF optimizer was employed to retrieve the optimal labeling results. Huang et al. (2016) also proposed a deep learning-based approach in which multiple sources of imagery data were fused and the deep CNN was fine-tuned to output saliency map. Sherrah (2016) leveraged the Fully Convolutional Neural network (FCN) to achieve the pixel-wise semantic labeling. In their study, the fully connected layers were replaced by the convolutional layers, and the deconvolution kernels were stacked to retrieve the outputs that have size equaling to the size of inputs. Then, the Digital Surface Model (DSM) was concatenated with the high-resolution

imagery during training and inference. Yuan (2016) introduced the use of signed distance functions in a CNN to output the building boundaries. In the study, the GIS data were also used to assist the training phase, and remarkable performance was obtained.

The above research studies are summarized and categorized in Table 2.1. It can be noted that the studies focusing on airborne LiDAR point cloud data use primarily the engineered features and pre-defined shape primitives, such as plane, cylinder, contours, etc. However, a major limitation of the engineered features-based approaches is that they require sufficient point cloud density. This is because the shape primitives are detected via either fitting neighbors to model primitives, or extracting the geometric features of the local neighbor of each single point. Low spatial resolution creates significant challenges for reliable primitive fitting. In addition, these studies focus only on mono-temporal data, meaning only pre-event (intact) datasets or post-event (damaged) datasets are used. In disaster management, multi-temporal data sets offer distinctive advantages over mono-temporal data sets as they enable rapid change detection. However, it is possible that the resolutions of multi-temporal datasets can vary significantly from data set to data set due to data collection approaches, devices, weather condition, etc. These multi-temporal data sets will likely cause challenges to the existing approaches. Another group of relevant studies started investigating the use of emerging deep learning techniques on building detection. These studies demonstrated promising results in building detection applications. However, these studies focused only on imagery data sets, leaving a research gap on using deep learning techniques to detect and extract building objects from airborne LiDAR point data.

Table 2.1. Summary of Related Research Efforts

Research Efforts	Approaches	Limitations
Dorninger et al.(2008), Yonglin et al. (2010), Lafarge et al.(2011), Sun et al. (2013), Fan, Yao et al. (2014), Niemeyer et al. (2014), He, M. et al. (2016), Wu et al.(2017),	Detection of shape primitives: • plane; • cylinder; • contours, etc.	• Require sufficient point cloud density; • Models only applied on either pre-event or post-event datasets
Vakalopoulou et al. (2015), Huang et al. (2016), Sherrah (2016), Yuan(2016)	Deep learning-based approaches: • convolutional neural network; • fully convolutional neural network, etc.	• Model only applied on satellite/aerial imagery data

The scope of this research is to develop a novel approach – convolutional neural network based approach – for the application of building object detection in both pre- and post-disaster scenarios. The learned features are able to delineate buildings of various scenarios. This is a notable gap in existing literature to be filled. A particular important application of this study is assessing building damages in coastal areas after major coastal storm events.

2.3 Research Methodology

In this study, I propose a deep neural network based approach for building detection from airborne LiDAR data sets (Figure 2.1). In this approach, point cloud data are firstly converted to gray-scale images, in which the value of each pixel represents the relative elevation of the

corresponding points. A convolutional neural network is engineered and trained to learn the features that are capable of distinguishing building objects from other objects such as vegetation. During the prediction phrase, I propose to apply morphological operations on the grey-scale images converted from large point cloud data sets to propose object candidates in order to avoid the use of exhaustive sliding windows. Following morphological operations, each object candidate is classified through the trained neural network. The following provides a detail description of each stage in this proposed approach.

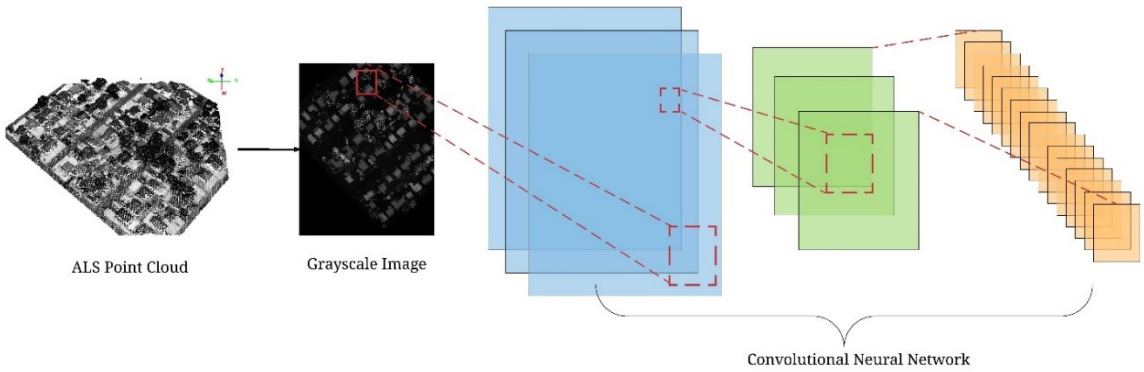


Figure 2.1. The Proposed Deep Neural Network based Building Detection Approach

2.3.1 Engineering Point Cloud into Grey-Scale Images

The first step of this proposed approach is to convert point cloud data into gray-scale images. A key challenge in this step is that the images directly converted from point cloud data with the native resolution contains little information that the CNN can learn and leverage. This has become a focus point of issues to be addressed in this step.

In this proposed approach, the x and y dimensions of the point cloud data are multiplied by a positive scalar factor, s_{xy}. This is because the original airborne point cloud is relatively sparse with point density approximately of 1-4pt/m^2 for pre-event dataset, and $< 1pt/m^2$ for post-event dataset. If the conversion is conducted using original resolution, each building will be converted to the image patch with approximately $(10 \times 10)px^2$, which is considerably low in

image resolution. Because the pixel values of a gray-scale image fall in a range between 0 and 255 and the height values of normal residential areas have a range of 0 to 10 meters, I multiply the height values of the point cloud data by a scalar factor, s_z to normalize the height values such that they fall into between the range of 0 and 255. In the case that more than one point is mapped to a given pixel cell, I pick the elevation of the highest point to represent the pixel value. In summary, the conversion of point cloud data from a three-dimensional coordinate system to a two-dimensional pixel format could be expressed as:

$$\begin{aligned} I_x &= \lfloor s_{xy} \cdot (X - min(X)) \rceil \\ I_y &= \lfloor s_{xy} \cdot (Y - min(Y)) \rceil \\ I(x,y) &= \sum_{X,Y \in (I_x, I_y)} \lfloor s_z \cdot (Z - min(Z)) \rceil \end{aligned} \tag{2.1}$$

here the X, Y, and Z are the point cloud coordinate values, I_x, I_y are the image pixel coordinate values, and $I(x,y)$ is the height value of the image at location I_x and I_y. The original value is shifted by the corresponding minimum value in order to convert the value from the global coordinate to a local coordinate. And $\lfloor f(x) \rceil$ denotes the nearest integer of the function value. The conversion from point cloud to image coordinate is shown Figure 2.2 (b) and (c).

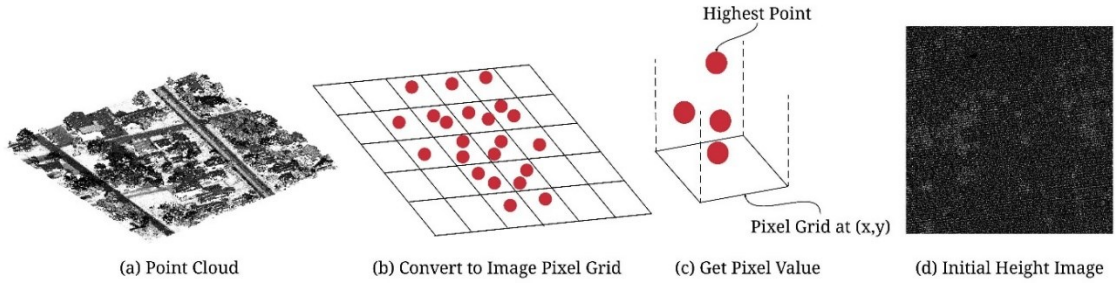


Figure 2.2. Generation of Initial Height Images

Due to the sparsity of airborne LiDAR data points in a given spatial region, the images generated using the above approach inevitably have many holes (pixels with no height values) (Figure 2d). To overcome this issue, a triangulation procedure is used in this study. As shown in Figure 2.3 (a), the triangulation is constructed on the initial height image, where each vertex of

the triangle is the image pixel that has corresponding point cloud data points. In order to keep the fidelity of data interpolation, the triangles with edges longer than a pre-defined threshold are excluded so that every triangle connects points belonging to the same object or adjacent objects (Figure 2.3 (b)). Within each valid triangle, a linear interpolation is conducted. As the triangles might connect points belonging to adjacent objects, such as building and vegetation, or building and ground, it is important to avoid the interpolation between different objects to preserve edges. For each triangle, I denote its vertices as $(V_i^d, i = 1,2,3, d = x, y, height)$. If $\nexists\left(V_i^{height} - V_j^{height}\right) > \delta$, here δ is a pre-defined threshold, I use a bilinear interpolation to estimate the pixel values inside the triangle defined by these three vertices. A visual interpretation of this process is shown in Figure 2.3 (c). A plane is fitted in 3D coordinate system, where the pixel value is interpreted as z-dimension. Then for all pairs of $\{x, y$ inside the triangle enclosed by $\{V_i^x, V_i^y\}$, the corresponding z value is estimated via solving the equation $\boldsymbol{n} \cdot \left(\tilde{V} - V_i^d\right) = 0$, where $\boldsymbol{n} = (V_2^d - V_1^d) \times (V_3^d - V_1^d)$ is the normal of the plane in 3-D space, $\tilde{V} = \{x, y, height\}$ is the point to be interpolated, and V_i^d is the vertices of the triangle.

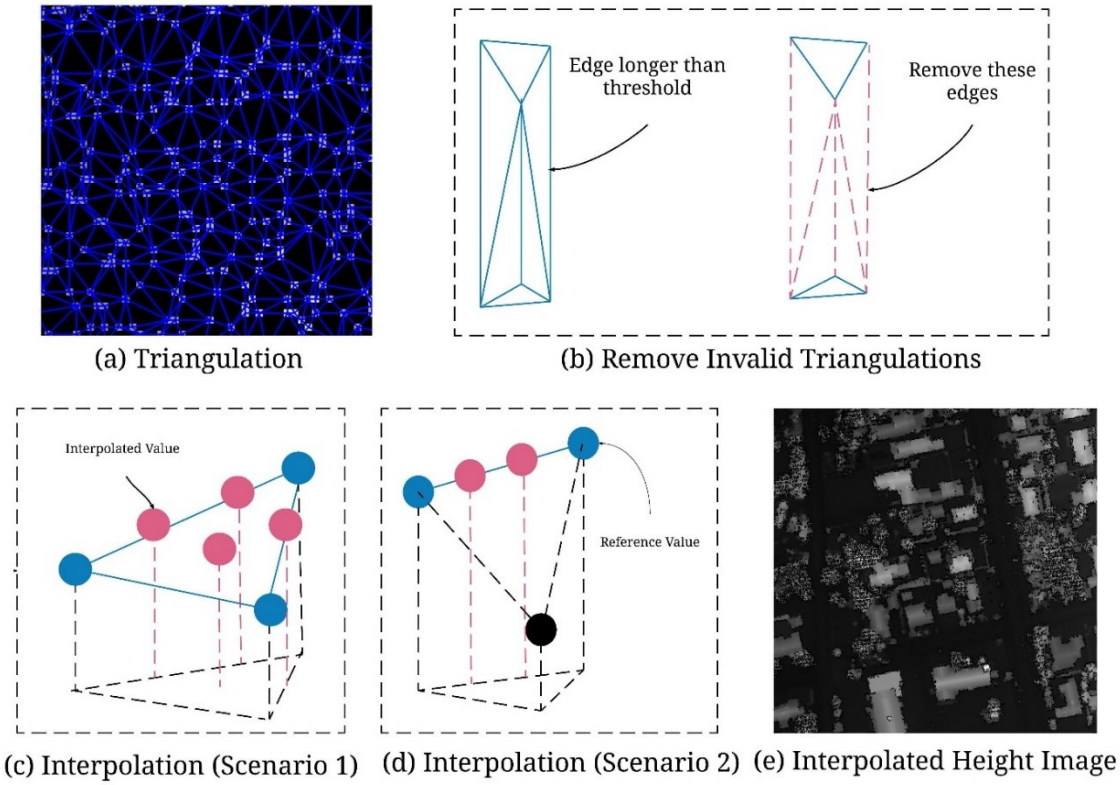


Figure 2.3. Image Interpolation

If $\exists\left(V_i^{height} - V_j^{height}\right) > \delta$ and $\exists\left(V_i^{height} - V_k^{height}\right) < \delta$, where $i, j, k = 1,2,3$, and $i \neq j \neq k$, I use linear interpolation to estimate the pixel values that are very close the line segment connected by $\{V_i^{x,y}, V_k^{x,y}\}$. As shown in Figure 2.3 (d), I only take the vertices having similar height values into consideration to preserve edges and avoid smoothening of the image. In Figure 2.3 (c) and (d), the blue circles represent the vertices that are considered in the interpolation, the pink circles represent the interpolated pixels, and the black circles represent the point not considered when interpolating the hole. The interpolation result is illustrated in Figure 2.3 (e). Compared with the image before interpolation (Figure 2.3 (d)), the proposed interpolation strategy is capable of filling the image holes and maintaining the edge sharpness at the same time.

2.3.2 Design of CNN Architecture

Deep learning has been at the forefront of artificial intelligence research since its initial success in visual recognition tasks (Krizhevsky et al. 2012). In particular, CNN-based architectures have been extensively studied to design new network architecture and improve learning and inference performance. Deep learning is one type of machine learning methods, which maps the inputs to the outputs through very complex linear or non-linear transformations. Deep convolutional neural network, in particular, mimics the behavior of neurons through many shared neurons called convolutional filters. Different filters have different responses on certain feature patterns which could be very high level and very abstract. The bottom neural layers learn the low level features with small reception fields, such as edge features or corner features, while the top layers learn very abstract features with comparatively large reception field. A series of intuitive explanation of deep neural network could be found at (Olah 2017). Among various CNN architectures, VGG net (Simonyan et al. 2014), Google LeNet (Szegedy et al. 2015), and (He et al. 2016) are some of the most popular network architectures. In this research, I adopt a network that has the similar architecture as the VGG net, but has different dimensions of input and output layers. In the VGG net, the input images have the size of 224 x 224 pixels and the output layer is vector with 1000 elements. In this study, the input images are resized to 128 x 128 pixels and the output layer is a vector with two elements as I am only interested in binary classification. As shown in Figure 2.4, the network is designed as multilayer of convolution operation followed by multiple fully connected layers. The configuration of each layer is listed in Table 2.2. As indicated in the table, all the convolution kernels have the same size in x and y dimension, but the different convolution layers have different convolution kernels in depth. The max pooling layers are stacked with the convolution layers to reduce the dimension without losing the maximum response to each kernels.

I use batch normalization (Ioffe et al. 2015) to accelerate the training process. For a mini-batch of variables $\mathcal{B} = \{x_1, x_2, \ldots, x_m\}$, the batch normalization learns two parameters, $\mu =$

$\frac{1}{m}\sum_{i=1}^{m} x_i$, $\sigma^2 = \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu)^2$ to normalize and transform the variables. The new distribution of variables after batch normalization is give as: $\hat{x}_i = (x_i - \mu)/\sqrt{\sigma^2 + \epsilon}$, $y_i = \gamma\hat{x}_i + \beta$. After the last convolution layer, I reshape the output feature map from a 4-D tensor to a 1-D vector, so that the following fully connected layer can be explicitly written in algebraic formation as: $y_j = \boldsymbol{W}_{ij} \cdot x_i + b_j$. In this equation, $\boldsymbol{W}_{ij}$ is the weight matrix of shape $\{M, N\}$, where M stands for the number of variables output from previous layer, and N stands for the number of variables of current layer, and b_j is the bias. For the loss, a softmax cross-entropy as shown below is used to measure the loss.

$$L(w) = -\frac{1}{N}\sum_{i=1}^{N}[y_i \, log \, \hat{y}_i + (1 - y_i) \, log(1 - \hat{y}_i)] \qquad (2.2)$$

where $\hat{y}_i$ is the predicted value and y_i is the ground truth. The AdaGrad (Duchi et al. 2011) optimization algorithm was employed during training.

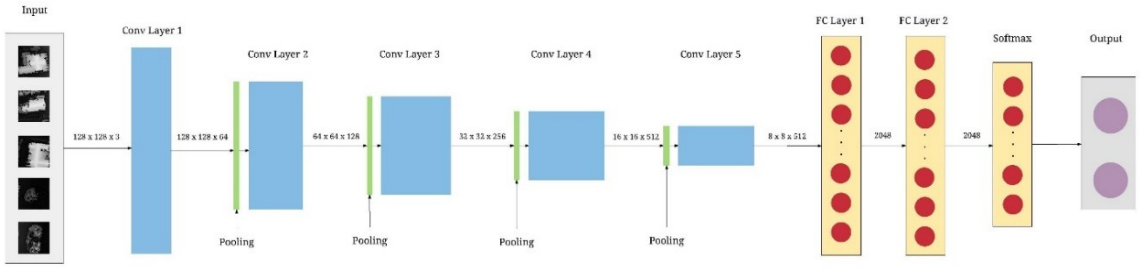


Figure 2.4. Network Architecture

Table 2.2. Network Configuration

Input (batchsize × size-x × size-y × depth)	
N × 128 × 128 × 1	
Conv Layer 1 (filtersize-x × filtersize-y × in-depth × out-depth)	
3 × 3 × 3 × 64	3 × 3 × 64 × 64
Max Pooling 1 (batchsize × size-x × size-y × depth)	
N × 64 × 64 × 64	

Conv Layer 2 (filtersize-x × filtersize-y × in-depth × out-depth)	
3 × 3 × 64 × 128	3 × 3 × 128 × 128
Max Pooling 2 (batchsize × size-x × size-y × depth)	
N × 32 × 32 × 128	
Conv Layer 3 (filtersize-x × filtersize-y × in-depth × out-depth)		
3 × 3 × 128 × 256	3 × 3 × 256 × 256	3 × 3 × 256 × 256
Max Pooling 3 (batchsize × size-x × size-y × depth)		
N × 16 × 16 × 256		
Conv Layer 4 (filtersize-x × filtersize-y × in-depth × out-depth)		
3 × 3 × 256 × 512	3 × 3 × 512 × 512	3 × 3 × 512 × 512
Max Pooling 4 (batchsize × size-x × size-y × depth)		
N × 8 × 8 × 512		
Conv Layer 5 (filtersize-x × filtersize-y × in-depth × out-depth)		
3 × 3 × 512 × 512	3 × 3 × 512 × 512	3 × 3 × 512 × 512
Fully Connected Layer 1 (in-depth × out-depth)		
(N × 32768) × 2048		
Fully Connected Layer 2 (in-depth × out-depth)		
2048 × 2048		
Soft-max Layer (in-depth × out-depth)		
2048 × 1000		

2.3.3 Accelerating Network Inference with Object Proposals

Unlike image classification, object detection requires not only classification of an image, but also the localization of objects in the images. Largely due to the rise of CNN, multiple successful deep learning based object detection approaches have been proposed. Notable ones are OverFeat

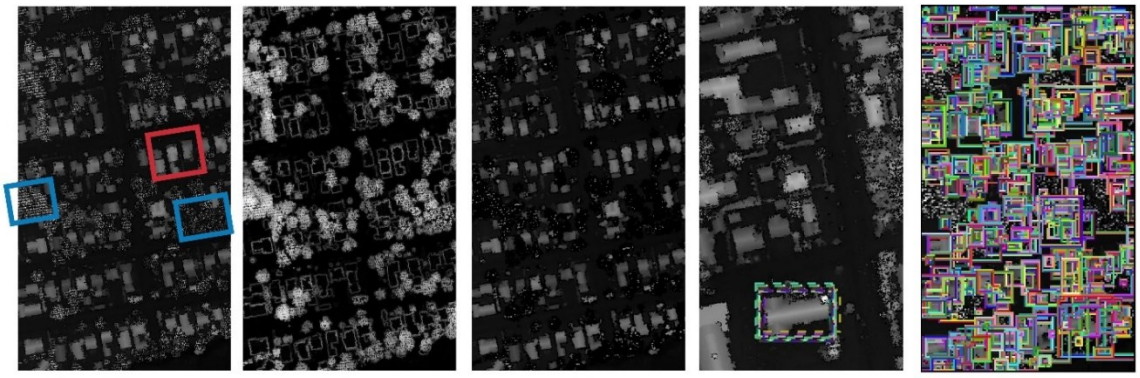
(Sermanet et al. 2013), R-CNN (Girshick et al. 2014), Fast R-CNN (Girshick 2015), Faster R-CNN (Ren et al. 2015), YoLo (Redmon et al. 2016), and SSD (Liu et al. 2016). These approaches use image proposals (Alexe et al. 2012, Carreira et al. 2012, Uijlings et al. 2013) to avoid redundant and exhaustive sliding boxes in order to speed up the detection procedure. Because in this research the images are synthetic gray-scale images, morphological operations rather than object proposal techniques were used to retrieve the object candidates. As shown in Figure 2.5 (a), the synthetic images have two main patterns: the pixels enclosed in the red block represent building objects, and the pixels enclosed in blue block represent the non-building objects such as vegetation (Figure 2.5 (a)). A series of images are generated from original image I by applying the morphological operations.

$$\begin{aligned} I^d &= I \oplus g \\ I^e &= I \ominus g \\ \hat{I} &= I^d - I^e \\ \tilde{I} &= (I \ominus g) \oplus g \end{aligned} \tag{2.3}$$

where $\oplus$ denotes dilation operation, $\ominus$ denotes erosion operation, and g denotes a morphological operator. $\hat{I}$ is an image that preserves possible building objects (Figure 2.5 (c)), while $\tilde{I}$ is an image that preserves non-building objects (Figure 2.5 (b)). All these images are binarized by passing their pixel values through a step function defined as:

$$\mathcal{F}(x,y) = \begin{cases} 1 & \text{if } I(x,y) \geq \delta \\ 0 & \text{if } I(x,y) < \delta \end{cases} \tag{2.4}$$

where $\delta = \Delta + \min_{x,y \in \mathbb{S}} I(x,y)$. Δ is a constant value set to be 3 meters because most of the non-building and non-vegetative objects in residential areas are below this height, and $\min_{x,y \in \mathbb{S}} I(x,y)$ is the minimum pixel value within the region $\mathbb{S}$ centered at pixel (x,y). This equation is adopted based on the assumption that the change of ground elevations in small region (e.g. $10(\text{m}) \times 10(\text{m})$) in residential areas is relatively small, and the ground elevation within such a small region could be approximately represented by a single value. Here I use the minimum elevation of the small region to represent the ground elevation.



(a) Height Image (b) (c) (d) Bounding Box (e) Object Proposals

(a) is the image after interpolation, (b) is the image after subtracting the erosion operation from dilation operation, (c) is the image after opening operation, (d) multiple bounding box over one cluster, and (e) shows the bounding boxes of each object proposals.

Figure 2.5. Object Proposal

For both images, $\hat{I}$ and $\tilde{I}$, I cluster pixels with value $\hat{I}(x,y) = 1$ and $\tilde{I}(x,y) = 1$ using (Ester et al. 1996). The reason to cluster these two images is to obtain object proposals from the binary images. Here the clustering algorithm takes two parameters, k and ϵ, where k stands for the number of neighbor points within a certain radius, and ϵ stands for the searching radius. In this research, these parameters are set as $k = 2, \epsilon = 2$. After the clustering, the centroid of each cluster $\{x_c, y_c\}$ is used as a seed to generate multiple bounding boxes according to the following method: (1) The bounding box of each cluster is first determined as $\{w, h\}$, here $w = x_{max} - x_{min}$, and $h = y_{max} - y_{min}$; and (2) If a bounding box is expressed as: $\mathfrak{B}(x_c, y_c, w, h)$, I then generate multiple bounding boxes around $\{x_c, y_c\}$ as $\mathfrak{B}(x_c + \varepsilon_x, y_c + \varepsilon_y, w + \varepsilon_w, h + \varepsilon_h)$, where ε_x, ε_y, ε_w, ε_h are random positive integers to shift and scale the original bounding box $\mathfrak{B}(x_c, y_c, w, h)$ (Figure 2.5 (d)).

Before conducting inference, the testing images are clipped by the above generated bounding boxes, and the clipped images are resized to the dimension of 128 pixels x 128 pixels x 1 channel. The trained CNN assigns a label to each of those images, which can be mapped back to the original images. If an image clipped by the bounding box $\mathfrak{B}$ has been classified as $y = j$, the pixels of the original image inside the bounding box $\mathfrak{B}$ are then assigned to $e^{\boldsymbol{X}^T \boldsymbol{W}_j} / \sum_{k=1}^{K} e^{\boldsymbol{X}^T \boldsymbol{W}_k}$.

As shown in Figure 2.6 (b) and (c), the probabilities of each pixel being assigned to a building object or a non-building object are plotted as a heat map. The final classification map is shown in Figure 2.6 (d). In the map, the positive value represents the probability of a pixel being classified as a building object, and the negative value represents the probability of a pixel being classified as a non-building object.

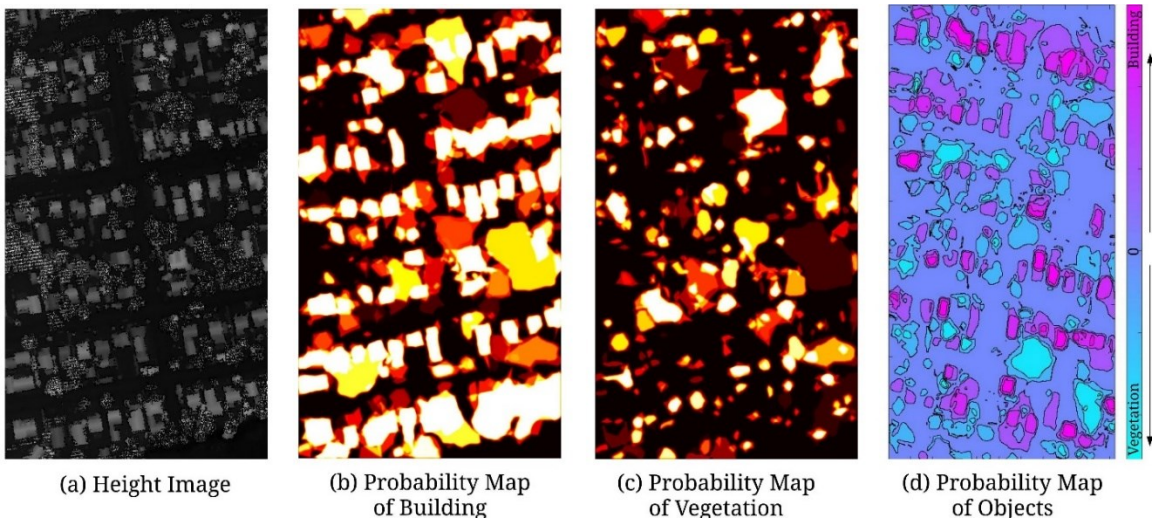


Figure 2.6. Probability Map

2.3.4 From Building Detection to Building Extraction

It can be noticed from Figure 2.6 (d) that although the network outputs pixel-wise labels, the boundary of a blob of pixels with the same labels is not usually aligned perfectly with the true boundary of a building object. Therefore, the output label map cannot be directly used to extract the building objects from the original point cloud data because the map in some cases can clip one building object into two parts, or can include erroneous surrounding objects. To address this issue, a new approach is introduced in this study to retrieve building objects using the label maps.

In Section 2.3.1, I mentioned the binarization of the height image using function $\mathcal{F}(\mathrm{x}, \mathrm{y})$. Denoting the binary image as $I_{\mathcal{B}}$ and the labelled image as $I_{\mathcal{L}}$, I cluster the pixels with value $I_{\mathcal{B}}(x, y) = 1$ into multiple groups. For each cluster $C_i \in I_{\mathcal{B}}$, I count the occurrence of pixel: $I_{\mathcal{L}}(x, y) \in C_i$ assigned to label $\mathcal{L} = \mathrm{l}$ as:

$$C(\mathcal{L} = l) = \sum_{(x,y)\in C_i} I_{\mathcal{L}}^{l}(x, y) \cdot \mathcal{I}(\mathcal{L} = l) \qquad (2.5)$$

where $\mathcal{I}(\mathcal{L} = \mathrm{l}) = 1$, and $\mathcal{I}(\mathcal{L} \neq \mathrm{l}) = 0$. For binary case, if $C(\mathcal{L} = 1) > C(\mathcal{L} = 2)$, I assign the entire cluster to label 1, otherwise, the entire cluster is assigned to label 2. This simple counting is able to retrieve the edge-aware object map with the label assignment.

Figure 2.7 presents the results of building extraction using the edge-aware cluster maps. Figure 2.7 (a) – (c) are the results of extraction on the pre-event airborne LiDAR data set, and Figure 2.7 (d) shows the results on data captured from the post-event data set. It can be observed that the edge-aware cluster map is capable of extracting entire building objects while minimizing the inclusion of erroneous of surrounding objects.

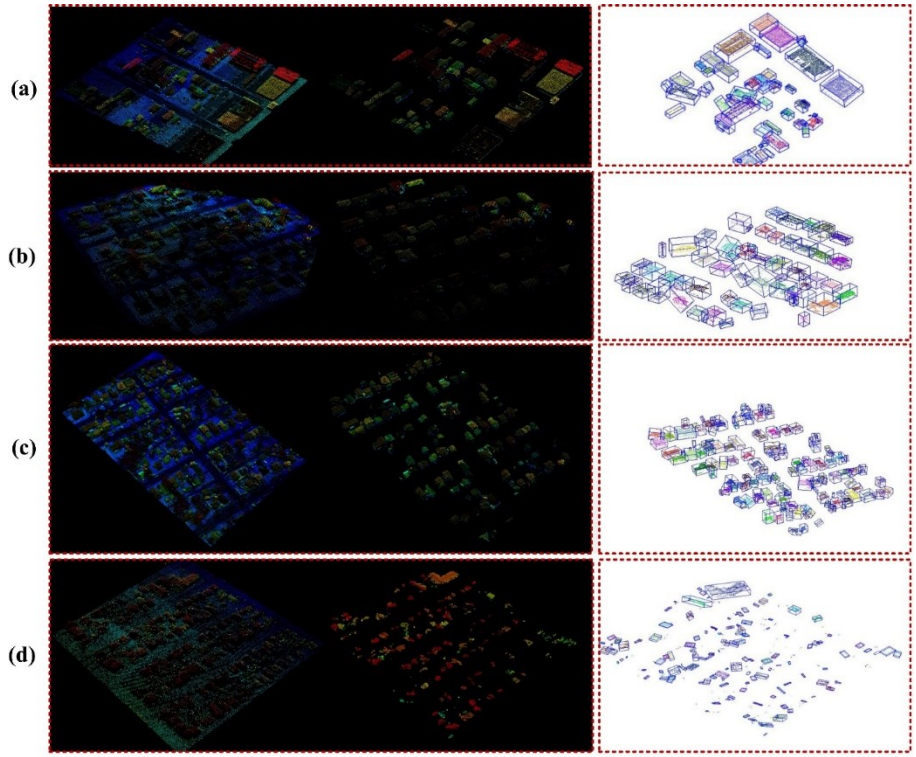


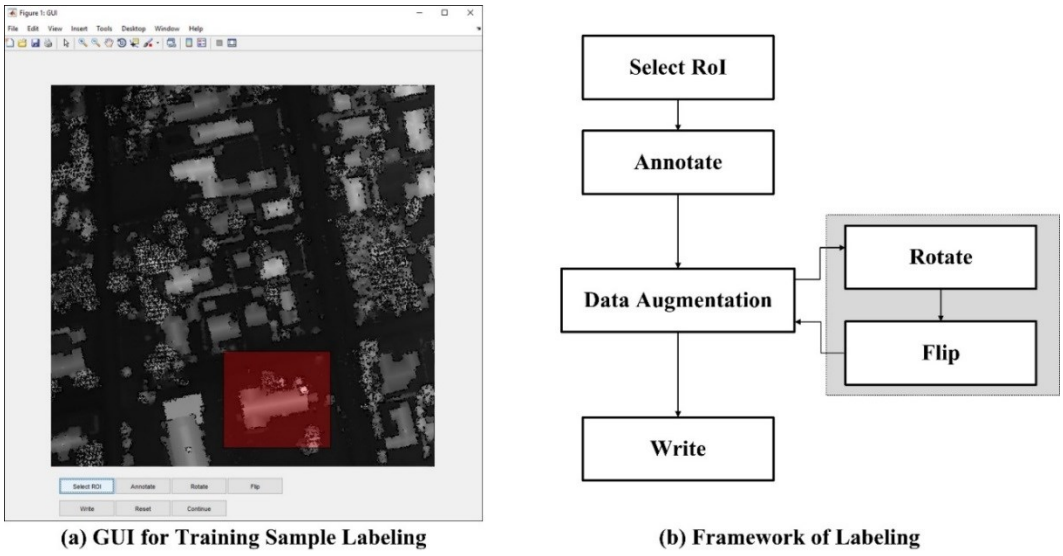
Figure description: (a), (b), (c) are airborne laser scanning point cloud captured at three intact regions. (d) is the airborne laser scanning point cloud captured at damaged region.

Figure 2.7. Extraction of Building Objects

2.4 Model Training, Testing, and Discussion

2.4.1 Model Training

The neural network was trained using an airborne LiDAR data set collected along the Northeast coastline in the United States. These data can be downloaded from the Digital Coast, an online geospatial data repository maintained by the National Ocean and Atmosphere Administration (NOAA). The point cloud data contain mostly low-rise residential buildings in the coastal communities. The average height of the building objects is less than 10 meters. To convert the point cloud data into height images, the scalar factors descripted in Eq. (1) are set as: $s_{xy} = 5.12$, $s_z = 25$. In image interpolation, I set $\delta = 30$ as the boundary between the bilinearity and linearity interpolation modes. 10,000 training samples are generated by manually labeling the point cloud data.



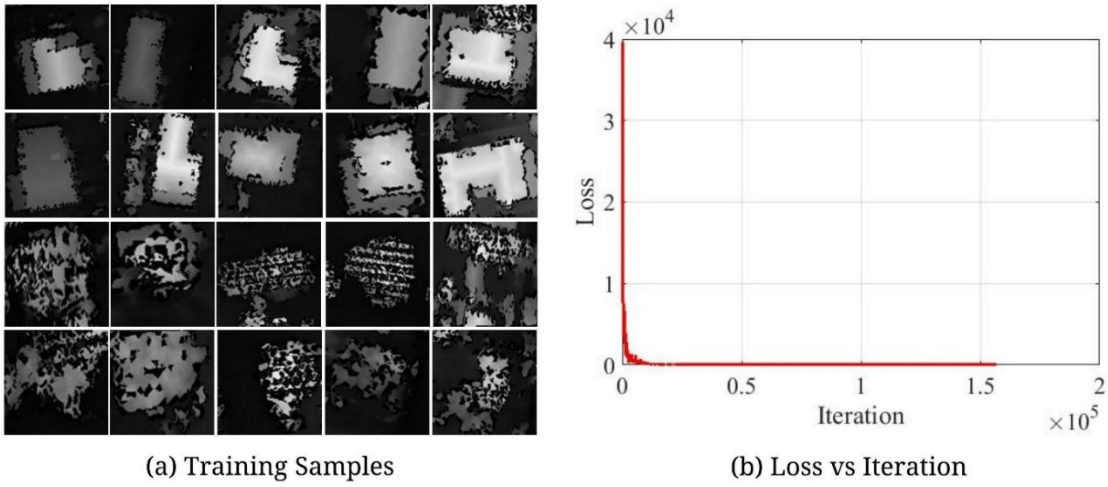
(a) GUI for Training Sample Labeling (b) Framework of Labeling

Figure 2.8. Labeling Process

To improve the labeling efficiency, a GUI is developed to assist the labeling process (Figure 2.8 (a)). The labeling is conducted on the converted gray-scale image. The overview of the labeling framework is also given in Figure 2.8 (b). At the first step, a region-of-interest (RoI) is manually selected via drawing a bounding box on the image. Then the category of the RoI is annotated. As known, training a CNN model consumes tremendous amount of training data, which is extraordinarily tedious and labor intensive if each every one of the training sample is

manually labeled. Therefore, data augmentation technique is introduced at the next step to produce more annotated data. In my GUI, data augmentation is mainly conducted via randomly rotating and flipping an annotated RoI. After the augmentation, the annotated images are saved as training samples.

Among these training samples, 5000 samples are building objects and the other 5000 samples are non-building objects such as vegetation. The convolution neural network was trained using TensorFlow (Abadi et al. 2016). The training was limited to 1.5×10^5 iterations. As shown in Figure 2.9 (b), the loss of the network drops significantly after the first 10,000 iterations.



(a) Training Samples (b) Loss vs Iteration

Figure 2.9. Training Sample and Loss

2.4.2 Model Testing

The performance of the trained network was evaluated using pre-event airborne lidar data and post-event airborne lidar data respectively. The pre-event data was collected two years before the Hurricane Sandy (in 2010), and the post-event data was captured right after the extreme natural hazard. The selected two datasets have significantly different point resolution. As shown in Figure 2.10, the point resolution is evaluated by the average nearest point-to-point distance, and the average number of points per area. As shown in Figure 2.10 (a), the average nearest point-to-point distance of pre-event data is approximately less than half of the post-event dataset. In

addition, Figure 2.10 (b) and (c) show the average number of points per $1(ft^2)$ and $4(ft^2)$, respectively. These metrics show that the resolution of pre-event data is remarkably higher than that of post-event data.

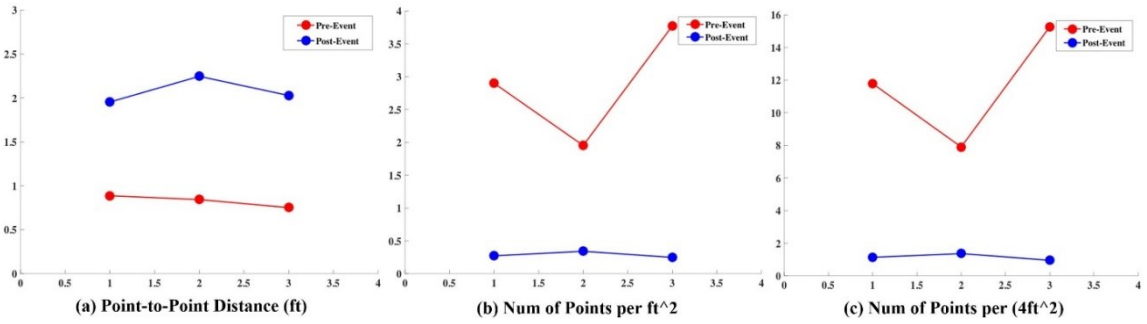


Figure 2.10. Evaluation of Data Resolution

The evaluation was implemented on CPUs, and each image took in average 0.35 sec to inference. The area where the testing data sets were collected is far away from the area where training data were collected. For both pre- and post-event datasets, I further divide them into three blocks to speed up the inference and detection process. Each of these blocks contains 60~90 building objects. The building detection results are shown in Figure 2.11. The yellow regions represent the objects have high probability been recognized as buildings, and the blue regions represent the objects hive high probability been recognized as vegetation. For building objects, the brighter the color is, the higher the probability they are predicted as buildings, and for vegetation, the darker the color is, the higher the probability they are recognized as vegetation objects. As can be shown in the figure, the upper row (a, b, c) are the dataset of three blocks collected prior to the occurrence of natural hazard, and the lower row (d, e, f) are the dataset of three blocks collected after the hazard. Intuitively, the proposed approach performs well on the pre-event dataset, and the performance drops slightly on the post-event dataset. The drop of performance on post-event dataset is in part because that the building patterns in post-event dataset have high randomness compared with the pre-event case. Because in post-event scenario,

the building objects do not necessarily preserve the engineered configuration. For example, buildings in post-event scenario may be collapsed or inclined. Furthermore, the vegetation may be wiped to cover the building roofs because of the extreme wind force. Because the proposed network is trained using the pre-event dataset, all of these unexpected scenarios will pose challenges to the inference stage.

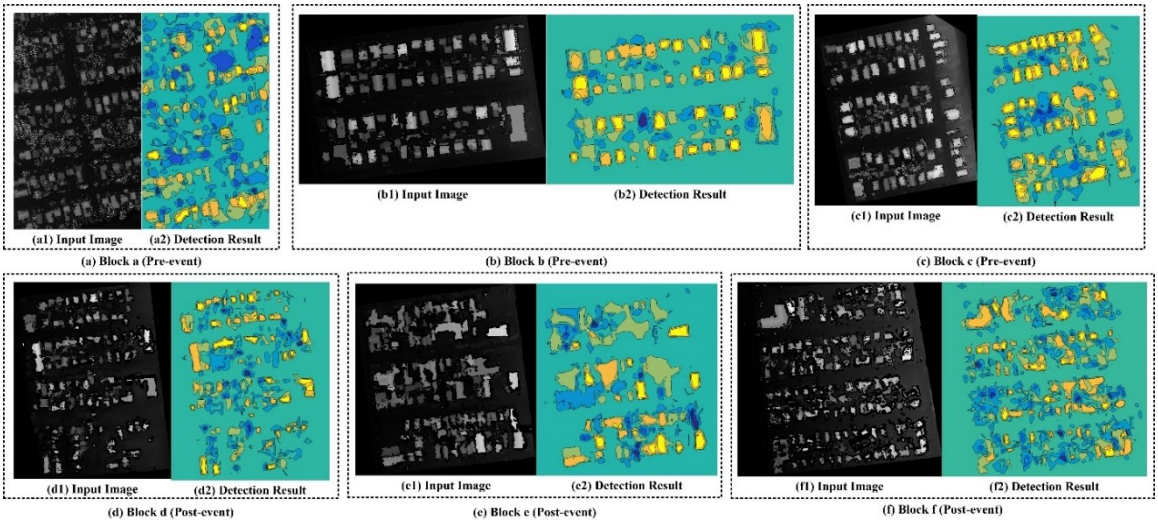


Figure 2.11. Building Detection Results

From the inference results, I identify several typical failure patterns. As shown in Figure 2.12 (a), this approach might recognize the building partially covered by vegetation as vegetation. Another typical failure is shown in Figure 2.12 (b), where the network classifies a building with complex roof configuration as vegetation. I believe there are two main reasons for these kinds of detection failures. The first reason is that the training samples are mostly residential homes, which have similar shape and size. Consequently, the network fails to detect the objects with significantly different patterns and sizes. Another reason is because the sparsity of point cloud data introduces noises during image interpolation. Although the interpolation procedure in this research is designed to preserve object edges, it may smooth the edge when the point cloud data are too sparse. Another failure example is given in Figure 2.12 (c). In this case, the building

enclosed in the red bounding box is recognized as vegetation with very high probability. This is likely because the low point density makes it fit to the abstract features of vegetation.

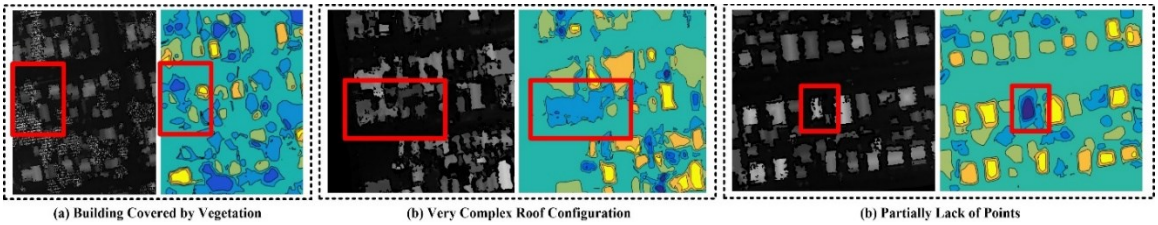


Figure 2.12. Typical Failure Cases

2.4.3 Performance Comparison

To further validate the performance of this approach, I compare my proposed approach with LAStools, a widely used LiDAR data processing tool. To further validate the performance of this approach, I compare this proposed approach with LAStools, a widely used LiDAR data processing tool. For LAStools, I tune the parameters to achieve optimal classification accuracy. In this study, the parameters are set as 1) search area size: 4; 2) building planarity: 0.2; 3) forest ruggedness: 0.4; and 4) ground offset: 2. The results are shown in Figure 12, the figures captioned with mine v.s. GT represents the comparison between the result obtained by proposed approach and the ground truth, and the figures captioned with LAStools v.s. GT stands for the comparison between the result obtained by LAStools and ground truth. In the figure, the dots and circles represent the centroids of detected building object. As commonly used for accuracy measurement, I use True Positive (TP), True Negative(TN), and False Positive(FP), where specifically, True means the object is predicted as building, False means the object is predicted as vegetation, and Negative means the object is actually vegetation. In each sub-figure, the red and black solid dot represent TP, the black circles represent TN, and the red circles represent FP, respectively. It is noticed from the figure that the proposed approach correctly recognizes most of the building objects from both pre- and post-event dataset. Although the detection accuracy obtained by CNN-based approach drops slightly from pre-event to post-event dataset, it still outperforms the

LAStools. Figure 2.13 (a-c) show the comparison between two approaches on pre-event data, and Figure 2.13 (d-f) present that of on post-event data. The comparison on Figure 2.13 (a) shows that both approach performs well, while for the rest of the dataset, the performance of LAStools drops significantly. This is because LAStools tris to classify points as building objects if the neighbors of each of them form planar object with respect to certain parameters. Therefore, the data resolution matters in terms of the neighbor finding and model fitting. As discussed previously, the testing dataset has very low data resolution especially for post-event dataset, this impedes the robust detection using LAStools. In contrast, the proposed approach shows more robust performance against the spatial resolution issues.

To quantitatively compare the results, the Precision and Recall are calculated. The recall and precision are shown in Figure 2.14. As can be seen in Figure 2.14 (a) and (b), the precision comparison does not yield significant differences. What can be noted is that the precision on the pre-event data is slightly higher than on the post-event data. This is largely due to that the post-event data have more clutters and irregular building objects. The comparison on recalls shows significant differences. The recall performance obtained by the proposed approach significantly outperforms that of LAStools. This is because LAStools tend to fail to detect many building objects which are with small sizes and have low point density. In contrast, the proposed approach still performs well and detects most of the building objects. This indicates that by learning very abstract and high level features, the challenges brought by low spatial resolution and clutters and irregularity in post-disaster scenes can still be dealt with to a great extent. The exact precision and recall values are listed in Table 2.3.

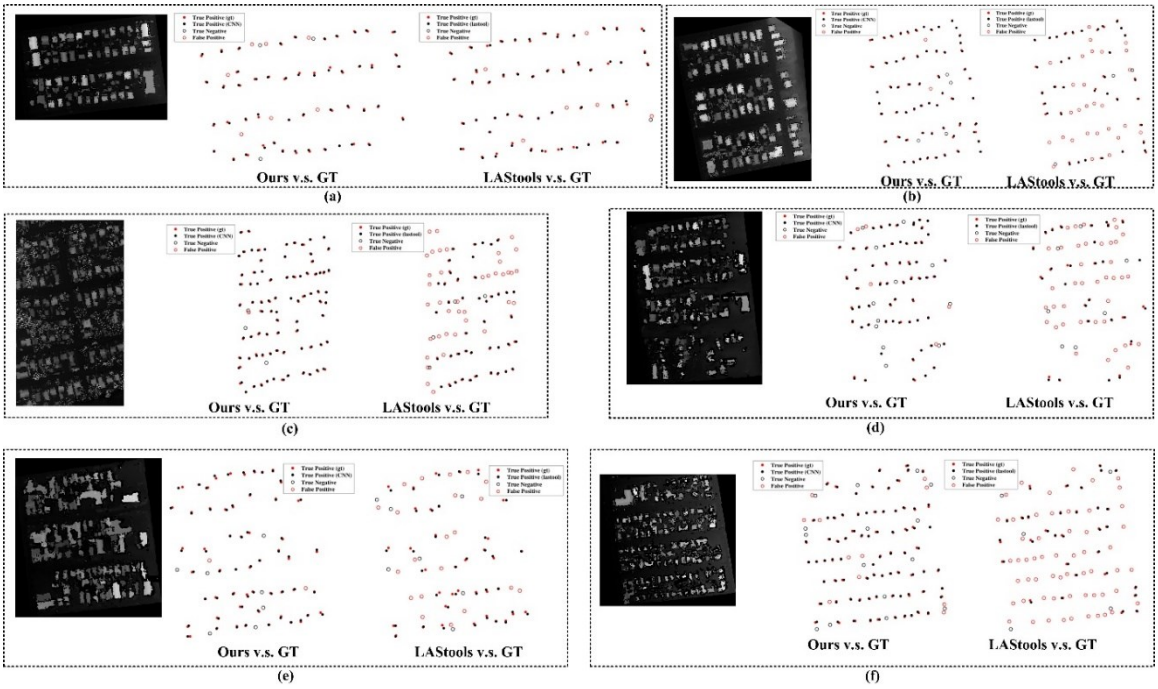


Figure 2.13. Comparison between Proposed Approach and LAStools

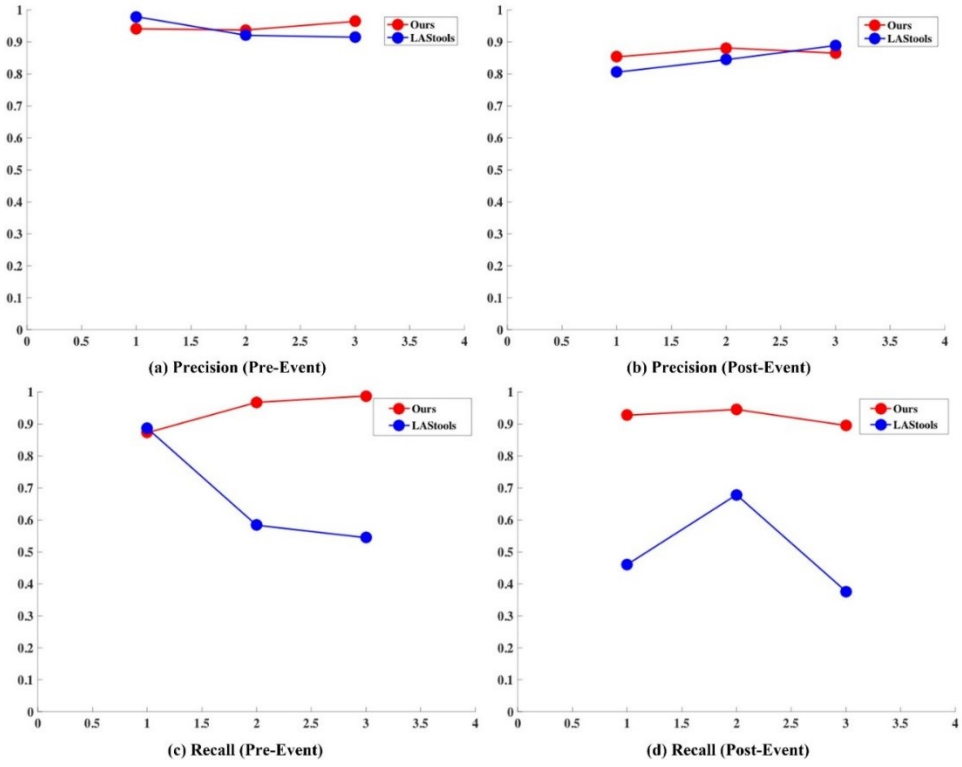


Figure 2.14. Precision and Recall

Table 2.3. Precision and Recall

		Pre-event			Post-event		
Ours	Precision	0.941	.0937	0.964	0.853	0.881	0.865
	Recall	0.873	0.967	0.987	0.927	0.945	0.895
LAStools	Precision	0.979	0.921	0.915	0.805	0.844	0.888
	Recall	0.886	0.583	0.544	0.460	0.678	0.376

Another important metric in measuring the performance of building detection is the area of detected building footprint. To evaluate the performance of this approach in detecting complete building objects, I choose one testing dataset (Figure 2.16 (a)) and compute the area of each building footprints detected by both methods (Ours and LAStools). Figure 2.15 shows the area of each detected building footprints. It can be noticed that most of the building objects detected by this proposed approach have larger footprints than the ones detected by LAStools. Specifically, for 77 building objects inside the testing region, the area of footprints of 69 building objects detected by this approach are larger than those extracted by LAStools. More specifically, in average for each building, the footprint area extracted by LAStools is approximately 63.2% of that detected by this approach. One such example is shown in Figure 2.16. In addition, this approach in many cases produces more complete rooftop boundaries than LAStools does. Furthermore, I observed that in cases where LAStools found buildings with larger footprints than what this approach did, one main reason is that LAStools often merges two adjacent buildings into one building (Figure 2.16 (d)-(f)).

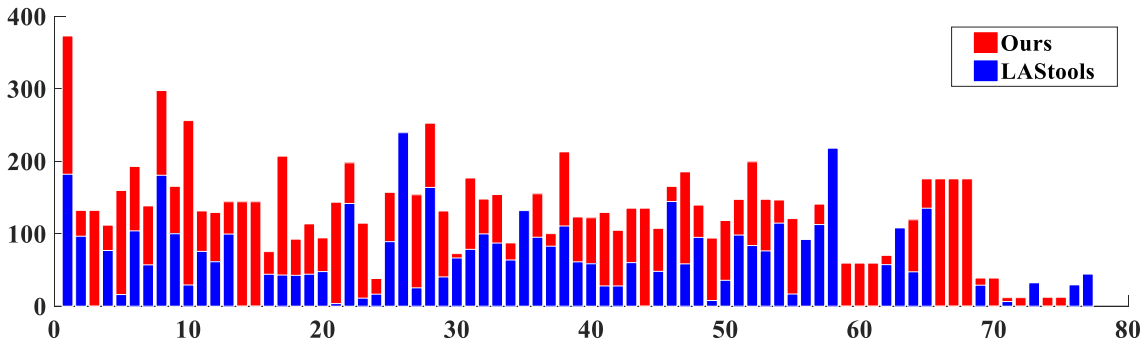


Figure 2.15. Area of Buildings Detected by Proposed Approach and LAStools

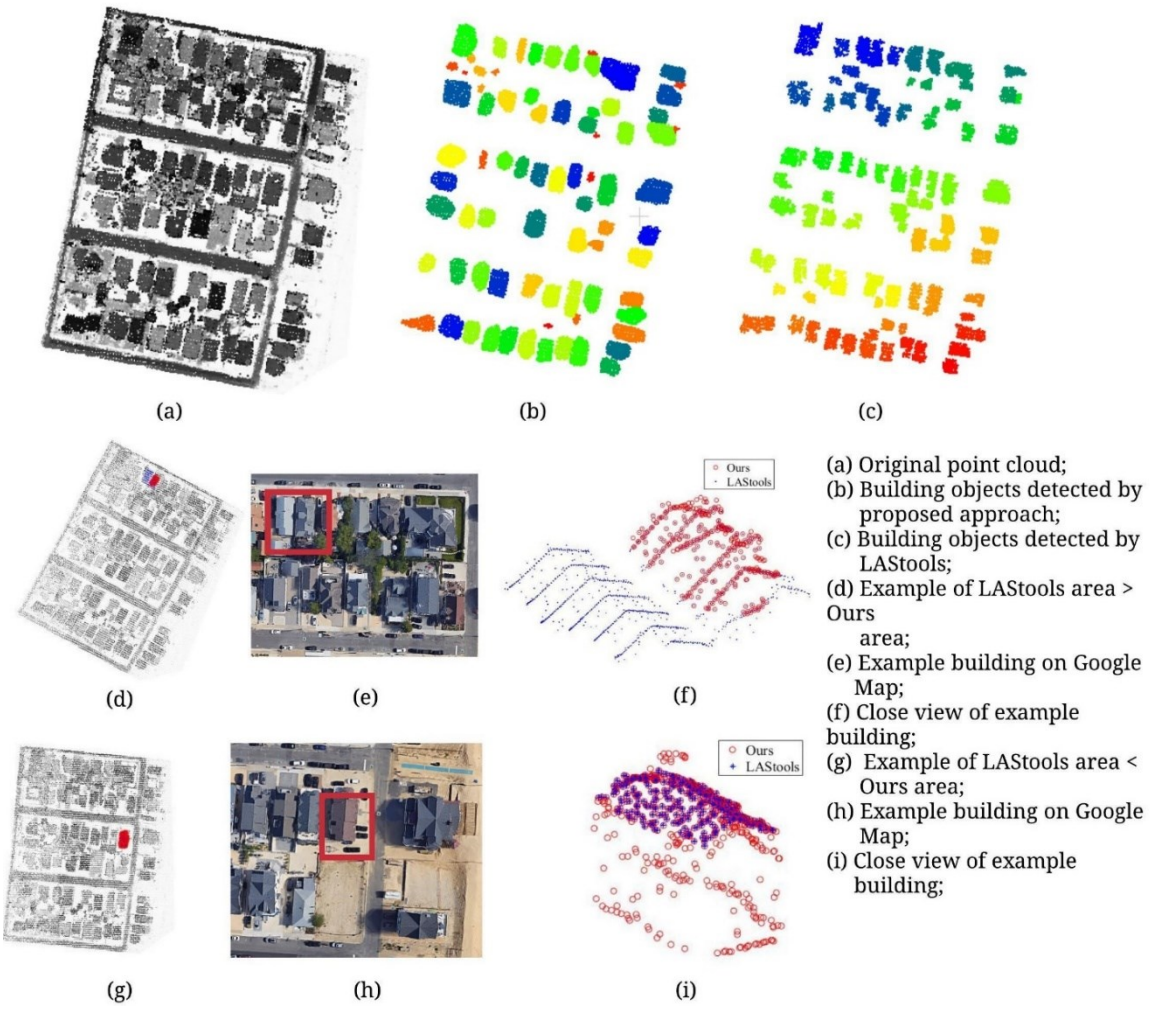


Figure 2.16. Area Completeness

2.5 Conclusion

In this study, I proposed a deep neural network based approach for building detection and extraction from airborne LiDAR data sets. Unlike other computer vision based approaches, the proposed approach does not require carefully engineered features for the purpose of pattern recognition and object detection. More specifically, this proposed approach does not require the computation of surface normal and sample consensus, which are often computationally expensive but key to shape primitive-based methods. Because of this, the proposed approach is applicable to

building object extraction from very large-scale point cloud data sets. The proposed approach is also capable of processing both pre-disaster and post-disaster airborne LiDAR data sets with the same trained neural network to extract building objects, regardless their extent of damage. Finally, the proposed approach is capable of extracting building objects with high-quality boundaries due to its use of edge-aware map to ensure the integrity and compactness of object extraction. The trick of using edge-aware cluster map is still largely an ongoing research topic in CNN-based 2D and 3D semantic segmentation research. Future work of this research includes inclusion of more commercial and high-rise building data sets for expanding the capability of this trained network as well as studying the use of big data infrastructure such as Spark to accelerate the image conversion and inference process.

Chapter 3 : Automated Assessment Of Residential Home Damages During Hurricane Events With Multi-Temporal Airborne Lidar Data

Rapid building damage assessment is a critical task following the landing of major hurricane events. Use of remotely sensed data to support building damage assessment is a logic choice considering the difficulty of gaining ground access to the impacted areas immediately after hurricane events. However, such remote sensing based damage assessment approach is often limited to detect severely damaged buildings. In this study, an efficient airborne lidar-based damage assessment approach is proposed. In the proposed approach, building models are reconstructed at first, then two features are computed between multi-temporal airborne point cloud to indicate the damage status, and finally three property-wise damage patterns are identified based on these features. The results of this study suggest that this proposed approach is capable of 1) reconstructing building models; 2) extracting damage features; and 3) characterizing the extent of damage to individual building properties.

3.1 Introduction

The past decades have witnessed increasing catastrophic natural hazards, such as 2008 Wenchuan earthquake, 2011 Japan earthquake, 2012 Hurricane Sandy, and the two super storms Harvey and Irma in 2017. After the hazard, rapid damage status assessment is critical in terms of immediate post-hazard response and decision making. In recent years, rapid development in remote sensing technology has successfully expanded the capability of rapid assessment by mean of rapid data collection in very large scale. Within the sufficient amount of data, a critical challenge of using these data to support ongoing decision making during extreme events is lack of efficient data processing methods that can extract useful information from these data sets within a realistic time bound.

Use of computer vision methods to interpret remotely sensed disaster data is not a new topic. A particularly well-studied area is building damage assessment following major disasters such as earthquake, tsunami, and tornado. The type of data used in these studies include satellite imagery,

airborne imagery, airborne lidar, InSAR, oblique imagery, point clouds reconstructed from overlapping images, and terrestrial lidar. There are two dominant approaches used in these studies to detect and quantify damages. The first is based on classifying damages based on damage features such as surface smoothness, local features, intensity distribution, and height information extracted from mono-temporal data (Khoshelham et al. 2013, Kashani et al. 2014, Galarreta et al. 2015, Kashani et al. 2015, Vetrivel et al. 2016). The second is based on detecting and characterizing changes between multi-temporal data sets (Vögtle et al. 2004, Huang et al. 2014, Pang et al. 2014, Vetrivel et al. 2016). In contrast to this plethora of studies devoted to building damage assessment after earthquake, tsunami, and tornado events, use of 3D disaster data to assess building damages after hurricane events has drawn much less attention. Airborne lidar data have been recently used more frequently for dune and beach assessment following hurricane events (Woolard et al. 2002, Zhang et al. 2005, Gesch 2009, Claudino-Sales et al. 2010, Rego et al. 2010, Sherman et al. 2013), but much less so for building damage assessment (Friedland 2009). Studies on interpreting building component-level damages with 3D disaster data are scarce (Gong et al. 2014, Hatzikyriakou et al. 2015, Zhou et al. 2015). It appears that most of the building damage assessment studies after major hurricane events have predominantly relied on the foot-on-ground approach. But given the fact that a major hurricane often causes damages to hundreds of thousands of homes spanning multiple states, such assessment approaches would incur tremendous cost and leave little time to assessors for conducting detailed field measurement. Therefore, there is inevitably a loss of valuable information for long-term research that could otherwise lead to improved understanding of severe wind and flood effects on the built environment.

In this study, I propose a novel post-hurricane damage assessment approach that can automatically process multi-temporal airborne lidar data sets to provide fine-scale estimate of building damages. The proposed approach first extracts building objects from both pre-event, and an efficient building reconstruction method is developed to reconstruct the 3D polyhedron

building models. Finally, building damage estimation is conducted based on fusion of two indicators, 1) outlier scores, and 2) average inlier area ratio.

The contribution of the proposed approach lies in its focus on addressing several limitations in existing studies. First, in many studies, airborne lidar based change detection is often conducted at the global level. This means such analysis is capable of detecting whether a building is changed or not, but falls short of characterizing the extent of damage. Second, current state-of-the-art methods depend on detection of roof segments (Schweier et al. 2006). This could be problematic if the rooftops of a damaged building were severely collapsed and deformed so that no plane segments could be identified. Thirdly, current state-of-the-art methods focus mostly on the seismic-induced damage scenario, while less attention has been paid on hurricane/storm surge-induced damage. In fact, during hurricane/storm surge hazard, what is more vulnerable is coastal residential houses. Those buildings are short in height, and distributed sparsely. In addition, they are likely to locate in rich vegetation region, makes it difficult to differentiate from vegetation. Lastly and most importantly, this approach is able to handle the case where post-event LiDAR data is of extreme low resolution, this makes the proposed approach applicable when the post-event data collection has to be carried out in a very short amount of time and, hence, with very poor data quality.

3.2 Related Works

Typically, damage assessment could be grouped into two categories: 1) using mono-temporal data, namely only the post-event data, and 2) using multi-temporal data, specifically pre- and post-event data.

3.2.1 Building Damage Assessment with Mono-Temporal Data

Using mono-temporal data for damage assessment is challenging, because it is difficult to quantitatively define damage. Schweier et al. (2006) presented a framework to describe the seismic-induced building damage patterns. They classified the damage patterns into multiple

categories, including inclined plane, multi-layer collapse, pancake collapse, etc. Shen et al. (2010) proposed to use point cloud segment for building damage assessment. In their approach, rooftops were firstly extracted, then the mean roof plane is estimated and employed to indicate whether the building is inclined or not. Elberink et al. (2012) and Khoshelham et al. (2013) proposed to use statistical learning strategies for damage detection. In their studies, the roof planes were segmented at the first stage. Then multiple features were extracted from each segment. To figure out the mathematical models fitting the features and damage patterns, a machine learning-based strategy was employed to learn the complex models. He et al. (2016), instead of using point cloud segments, proposed to use the shape descriptors to detect the damage patterns. In their approach, the shape descriptor is computed as chaos index of each contour lines of the point cloud.

In addition to the use of LiDAR point cloud, aerial imagery is another major data source in terms of building damage assessment. Balz et al. (2010) proposed to use the appearance feature, including building surface appearance and corner appearance features, to identify damage patterns. Similarly, Tong et al. (2013) proposed a novel approach that is able to detect collapsed building from their shadows using post-event aerial imagery. In their approach, the actual shadows were extracted from the post-event imagery, and the theoretical shadows were estimated using the height information and footprint information of each building objects. The damage was identified by comparing the actual shadow polygons with the theoretical shadow polygons. Nex et al. (2014) proposed a post-event data only approach for building damage detection. They first generated DSM from post-event airborne images, then building objects were extracted using MRF-based inference process. Then the buildings are classified into intact and damaged according to the fusion of multiple features, including spectral features and local planarity. Vetrivel et al. (2016) proposed an approach that uses Bag-of-Words classifier to learn the model fed with multiple features extracted from aerial imagery including histogram of gradient orientation, gradient magnitude, etc. Ye et al. (2017) proposed a novel approach that leverages the interior roofs and building edges to determine the damage status. In their approach, two new

features, including edge significance and local gradient orientation entropy, were introduced to describe the intactness of building objects. Vetrivel et al. (2017) investigated the use of integrating CNN-extracted image features and geometric point cloud features to detect building damages. The fusion of multi-source data shows encouraging performance in detecting damages such as roof collapse and building collapse.

3.2.2 Building Damage Assessment with Multi-Temporal Data

Multi-temporal data has been widely used in damage detection and change detection. Among those research efforts, they can be categorized as 1) use of multi-temporal LiDAR data, 2) use of multi-temporal imagery, and 3) use of multi-temporal and multi-source data.

For the first category, Chen et al. (2010) proposed to use pre-change 3D building models and post-change LiDAR and aerial imagery data for building change detection. In this research, the old building model and new airborne LiDAR was registered, and the points inside the building model were selected. After removing the facet points, the change detection is conducted according to the height difference and area difference. In this research, the change detection is conducted in structural-level, means this is not a binary change detection, but more detailed change analysis is able to be extracted. However, this approach relies 3D building models as an input, which constrains the applicability of this approach. To address this issue Awrangjeb (2015) proposed an approach using pre-change building database and post-change airborne LiDAR point cloud to detect the building change. To generate the pre-change building database, a graphic user interface (GUI) was developed so that building database could be created from building footprints. And the building model could be generated with human editing. During change detection phase, the post-change airborne LiDAR was used to extract new building footprint, and the comparison is carried out between this new footprint and the old building models. Trinder et al. (2011), (2012) proposed to use pre- and post-event airborne LiDAR and airborne imagery data to detect building damage. In their approach, four change detection methods, namely image differencing, PCA, minimum noise fraction, and post-classification are evaluated, and majority voting scheme was

employed to produce the final damage map. The experimental analysis shows that the fusion of imagery and LiDAR improves the detection accuracy. Xu et al. (2015) proposed an approach using multi-temporal airborne LiDAR point cloud to detect the urban environment change. After co-registration of the LiDAR, octree was constructed from the non-ground points, and the change detection is carried out efficiently using octree-based structure. Bloch et al. (2016) introduced a novel approach to identify the interior space of a damaged building caused by earthquake disaster. In their approach, a pre-event as-built BIM model is used and multiple potential collapse patterns were estimated. Then a post-event as-damaged model was generated from terrestrial laser scanning. And this as-damaged model was used to search for the similar damage pattern. This approach could be employed to identify structural-level damage. However, because the damage potential damage patterns may vary according to different building structure and materials, in addition, the use of terrestrial laser scanning is constrained by accessibility issue, which makes this approach less applicable in large scale response.

For the second category, Tong et al. (2012) assessed the building damage caused by Wenchuan earthquake using pre- and post-event satellite stereo-imagery. In this study, the height change was used as one major indicator of building collapse. Tian et al. (2014) proposed to use multi-temporal stereo imagery for building change detection. In this study, stereo imagery were firstly matched to DSM, then the height change and the similarity measurement between imagery were carried out. Susaki (2015) proposed an approach using multi-temporal aerial imagery to find individual tsunami-induced buildings. In the study, stereo-images were firstly matched to generate DSM, then the height change was calculated to detect the damaged buildings. In addition to the height change, the horizontal displacement caused by land deformation was also detected using the SIFT descriptor. Kahraman et al. (2016) explored the use of multi-temporal satellite imagery and footprints in building damage detection. After the building objects been extracted, adaptive self-similarity descriptor was employed to measure the layout of individual building object, and the damage was detected by computing the difference of self-similarity descriptor of

pre- and post-event data. In addition to imagery, synthetic-aperture radar (SAR) is also widely employed in damage assessment field. Li et al. (2012) proposed a SAR-based approach for collapsed building extraction. In this research, an entropy-average scattering mechanism-circular polarization correlation coefficient method is developed to extract the spatial distribution of collapsed building from SAR imagery. Sharma et al. (2017) proposed a novel approach, earthquake damage visualization, to detect the building damage caused by seismic attack. The pre-seismic coherence and post-seismic coherence were computed using pair of pre- and post-seismic SAR imagery, respectively. The damage was detected at the pixel level where low coherence value indicates high possibility of damage, while the region with high coherence value is assumed to represent intact building objects. Karimzadeh et al. (2017) used multi-sensor SAR to assess the earthquake-induced building damage. They used 1) differential coherence values, 2) differential backscattering intensity values of pre- and post-event images, and 3) binary damage map of the pre- and post-event imagery.

For the third category, Rastiveis et al. (2015) proposed an approach that integrates post-event airborne LiDAR and pre-event vector map to extract and map the damaged buildings. In their approach, LiDAR points were classified into debris, intact building, and unclassified categories using texture feature and nDSM. And the buildings are categorized into damaged and undamaged. Du et al. (2016) proposed a novel approach that leverages the dense point cloud generated from aerial imagery and the airborne LiDAR point cloud data to detect the urban environment change. They first generate dense point cloud from earlier imagery. After co-registration, height change and grey-scale similarity are computed to indicate the urban environment change. And in this study, the change is categorized into positive (indicates newly built or taller building), and negative (demolished or lower buildings). Vetrivel et al. (2016) investigated the potential of aerial oblique imagery data and point cloud data for detailed building damage evaluation. In their research, two approaches were developed. The first approach leverages the aerial imagery data and pre-event point cloud data to detect the structural damages.

To compensate the noisy point cloud on vertical façade, the second approach using oriented oblique imagery was introduced. GIS data also plays valuable role in damage assessment and asset management. For example, Tu et al. (2016) presented a framework that uses aerial imagery and GIS data to detect various types of building damages, including intact, trivial, severely, and totally damage. More specifically, the post-event imagery was registered with pre-event GIS model to extract the height change, area reduction, and rooftop texture. A supervised learning strategy was then employed to classify damage into different types.

The aforementioned approaches have the following issues. 1) For imagery based approaches, because the depth information is not delivered, the detection result does not reveal elevation change, which in fact plays a significant role in damage assessment. As a result, imagery-based approach can hardly identify partial damage, as for this case, the damage occurs as elevation change or inclination. 2) For LiDAR based approaches, the state-of-the-arts mono-temporal-based approaches rely on the features of point cloud segments and histogram distribution of the points. Although experiments show promising results, the computation of geometric features and histogram distribution requires consistent data resolution. While this is not guaranteed in real application. As will be shown in the experiment study section, the resolution of post-event data is significantly lower and inconsistent compared with pre-event data. In this research, I proposed a novel approach that uses multi-temporal airborne LiDAR data to detection the hurricane-induced building damage. The proposed approach is able to handle the extreme case that the data resolution is extremely low and inconsistent.

3.3 Methodology

The workflow of the proposed framework is presented in Figure 3.1 The framework contains five major stages. Given the pre- and post-event airborne point cloud dataset, the pre-processing procedure is carried out at first to identify the building objects. The second stage is implemented on pre-event dataset only. At this stage, an efficient roof model reconstruction approach is introduced, which is able to reconstruct the building model from the airborne LiDAR point cloud

with various data resolution. Compared with the conventional approaches, which implement the change (or damage) detection directly on the multi-temporal point cloud, I identify two issues which are associated to the performance of detection with respect to various point cloud resolution. As shown in Figure 3.2, the computation of point-to-point distance is severely affected by the point cloud resolution. The point cloud in Figure 3.2 (a) and (b) are the same dataset but with different resolution, it is noticed that the point-to-point distance for low-resolution scenario is larger than that of high-resolution scenario. The resolution also affects the detection accuracy in terms of the area computation. As shown in Figure 3.2 (c) and (d), the area of low-resolution point cloud data is smaller than that of high-resolution data. Once the pre-event roof model is reconstructed, the corresponding post-event building object extraction is implemented in the third stage. In the fourth stage, I compute the point-to-plane distance and the roof facet area ratio to indicate various types of damage patterns. And the damage status is determined at the last stage, which is based on the reconstructed model and computed features in the fourth stage. I will describe each stage in detail in the following sections.

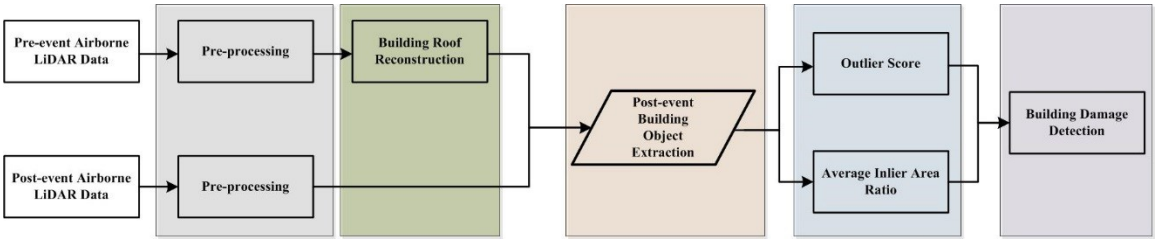


Figure 3.1. Overview of Proposed Framework

3.3.1 Pre-processing: Building Objects Identification

Given input airborne point cloud, pre-processing aims at the following two purposes, 1) identifying the building points from other objects, such as ground and vegetation, and 2) extract each building objects. The building objects extraction and ground points filtering have been an ongoing research topic in recent years, and numerous research effort have been made regarding to these topics such as (Zhang et al. 2003, Meng et al. 2009, Tseng et al. 2016, Zeng et al. 2016,

Zhang et al. 2016, Wei et al. 2017). These approaches, though perform well in identifying building objects from other objects, require further processing to cluster building segments. In Chapter 2, I propose a deep learning-based framework that is able to identify and extract building objects from a raw airborne LiDAR point cloud data. Compared with other approach, this framework combines the building objects detection and segments clustering procedures in the same framework, makes it easy-to-use for the applications when time is a serious concern. In this research, I use the deep learning-based approach to pre-process the building objects identification and clustering.

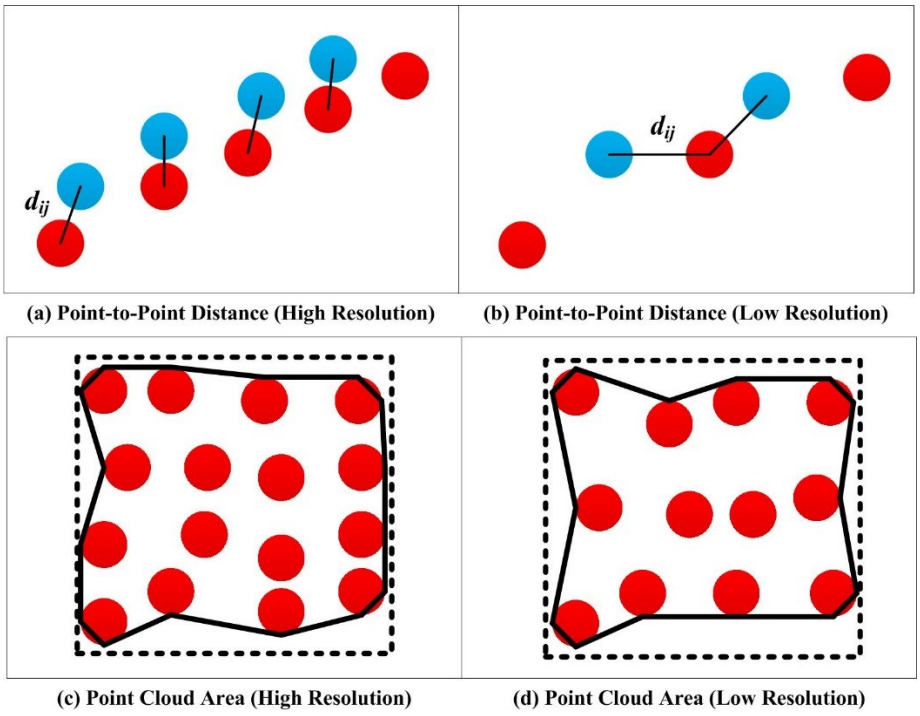


Figure 3.2. Two Issues associated with Detection Accuracy

3.3.2 Efficient Building Model Reconstruction

With individual building objects been extracted, I propose a novel and efficient approach to reconstruct the building models from multi-resolution airborne LiDAR point cloud data. Because of the purpose of rapid post-disaster damage assessment, efficiency is a critical concern to this stage. This approach is, therefore, developed to efficiently reconstruct the roof models regardless

the resolution of input point cloud data. The overview of the proposed model reconstruction is shown in Figure 3.3. The framework is consists of five major steps, and each step will be described in detail in in the following sections.

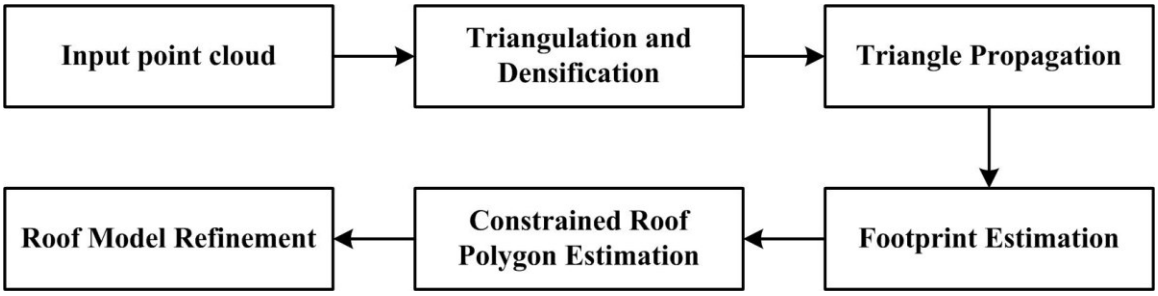


Figure 3.3. Overview of Efficient Building Model Reconstruction

Triangulation and Densification: Considering the various data resolution of different airborne point cloud, the roof model reconstruction (Wang et al. 2016, Cao et al. 2017, Jung et al. 2017) is easily affected by the low resolution of point cloud data. Instead of reconstructing the models on original data set with various point density, a geometry-preserving densification procedure is introduced to address the resolution issue. For an input point cloud, the triangle mesh is first constructed (Marton et al. 2009). As shown in Figure 3.4 (a) and (b), the mesh constructed on the sparse point cloud data have a issue demonstrated in Figure 3.4 (c). As shown, the red dots represent what is observed in the point cloud, and the red and blue dots represent what the real model should be. Due to the low resolution, a mesh is constructed via connecting vertex P_2 and P_3 (here the third vertex of the mesh is assumed to be outside the paper plane and therefore is omitted), which introduces error compared to the actual case. In order to avoid the densification on the wrong mesh triangle, only the valid triangle meshes are considered. A valid triangle mesh is defined as the triangle Δ_{123} shown in Figure 3.4 (d), whose normal vector is parallel to its edge-sharing triangles' normal with a tolerance δ. Mathematically, a valid triangle mesh could be formulated as

$$\Delta_i = \begin{cases} valid & \nexists(|n_i \cdot n_j| > \theta) \\ invalid & \exists(|n_i \cdot n_j| > \theta) \end{cases} \quad (3.1)$$

where n_i is the normal of Δ_i, and n_j is the normal of Δ_i's edge-sharing triangle mesh. And $\nexists(|n_i \cdot n_j| > \theta)$ means that for all the n_j, the angle between n_i and n_j is smaller than an angle threshold θ.

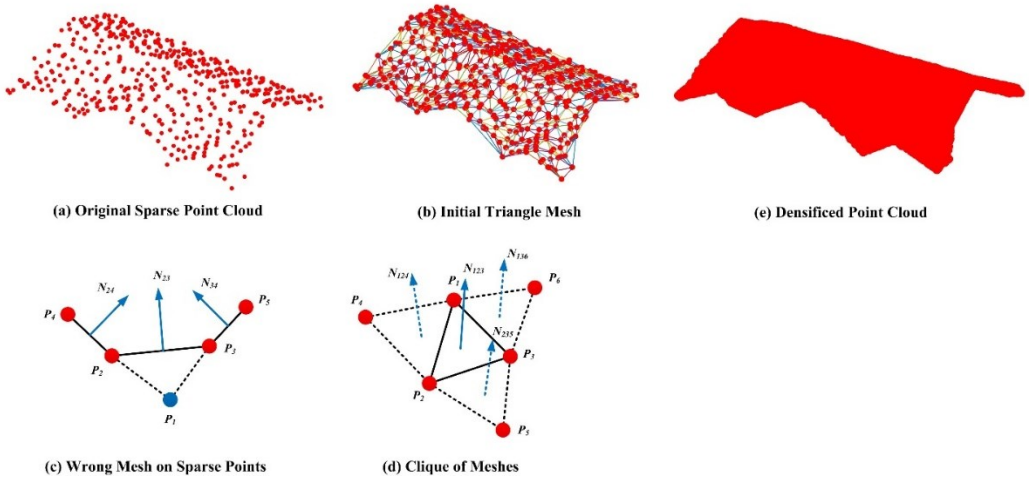


Figure 3.4. Point Cloud Triangulation and Densification

Triangle Propagation: After the densification, I first construct new triangle meshes. Then triangle propagation procedure is implemented to retrieve the roof facets. For an arbitrary triangle mesh Δ_i, I propagate its edge-sharing triangles Δ_j to the same roof facet if the angle between their normal is smaller than a threshold value. Started from an arbitrary triangle, this procedure is repeated until no more triangles could be propagated. And a new propagation procedure is initialized from an arbitrary triangle of the un-propagated meshes.

Footprint Estimation: Within the extraction roof facets, the next step is to estimate the building footprint polygon. Because the purpose of this research is focusing on rapid post-disaster damage assessment, high speed reconstruction of building model is considered as a priority issue. Sun et al. (2013) proposed a robust approach to estimate the rooftop orientation via rotating and fitting the minimum bounding box. In this research, I propose an efficient rapid building footprint estimation approach based on one assumption applies to common residential building objects: a residential building has two major axes, and the boundary lines of its footprint are parallel to one

principal axis, and perpendicular to another. Figure 3.5 (a) shows an aerial imagery[5] with multiple residential buildings. And Figure 3.5 (b) shows the roof layout of a gable roof configuration. It is observed from the figure that the boundary lines of the building footprint polygon is parallel one of its principal axis, and perpendicular to another. For example, line segment L_{AB} is parallel to principal axis 1 and perpendicular to principal axis 2. Since the principal axes of a residential building could be modeled by the direction of its roof ridges, this makes it straightforward to estimate the principal axes via intersecting its roof facets.

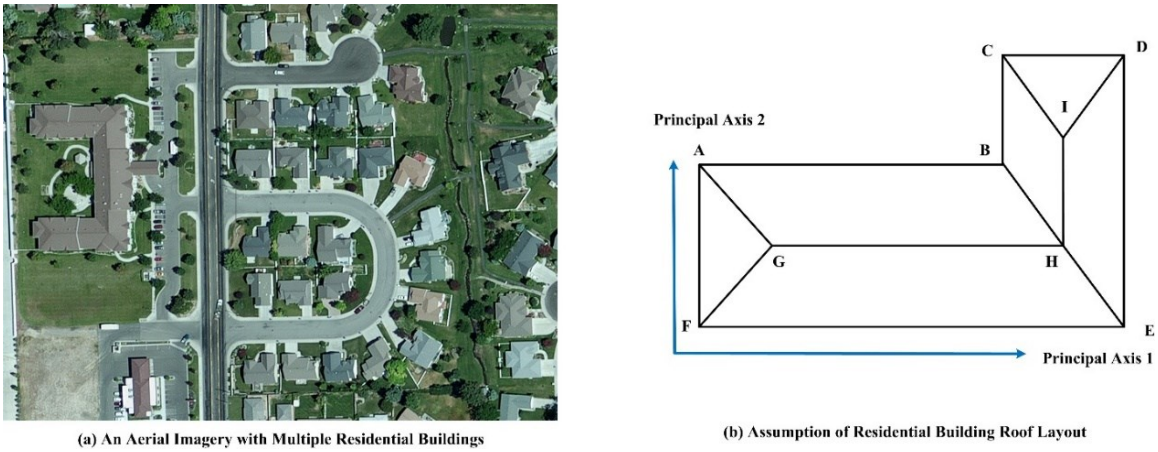


Figure 3.5. Assumption of Residential Building Roof Layout

I propose a novel footprint polygon estimation approach via shrinking a rectangle along the principal axes of the building. I denote the point data set of a building object as P, and the principal axes of the building as X^i, so that $\left|X^i \cdot X^j\right| = 0$. For X^i, I move a vertical line $\{L^i : L^i \perp X^i\}$ along X^i. At each location x_i along X^i, I compute the distance between each point P_i to L^i as d_i, and find the inliers with distance smaller than a threshold as $\{P_i \in P : d_i < \delta_d\}$. After that, I compute the maximum point-to-point distance along the line L^i as $\tilde{d}_i$. By swiping L^i along X^i with certain step length, I construct a 2-D signal where the value of each data represents $\tilde{d}_i$ (Figure 3.6 (a) black dot line). I define a clip point x_i^c along X^i as following: $\{ x_i^c \in X^i : \left|\tilde{d}_{i+\Delta} -$

[5] https://apollomapping.com/blog/update-on-digitalglobe-precision-aerial-imagery-program-usa-europe

$\tilde{d}_{i-\Delta}\big| > \delta_l\}$, where $\tilde{d}_{i-\Delta}$ is the maximum point-to-point distance at Δ step before x^i, and $\tilde{d}_{i+\Delta}$ is the maximum point-to-point distance at Δ after x^i, and δ_l denotes length threshold. I apply max-pooling and min-pooling operation to the original signal, and find the local maxima of the difference of max-pooling and min-pooling signal as the clip location (Figure 3.6 (b)). Once the clip points are estimated, I construct multiple rectangles by connecting the clip points. As shown in Figure 3.6 (c), this may introduce rectangles which do not overlap with building roof points, these rectangles are then removed to retrieve the final building footprint concave polygon as shown in Figure 3.6 (d).

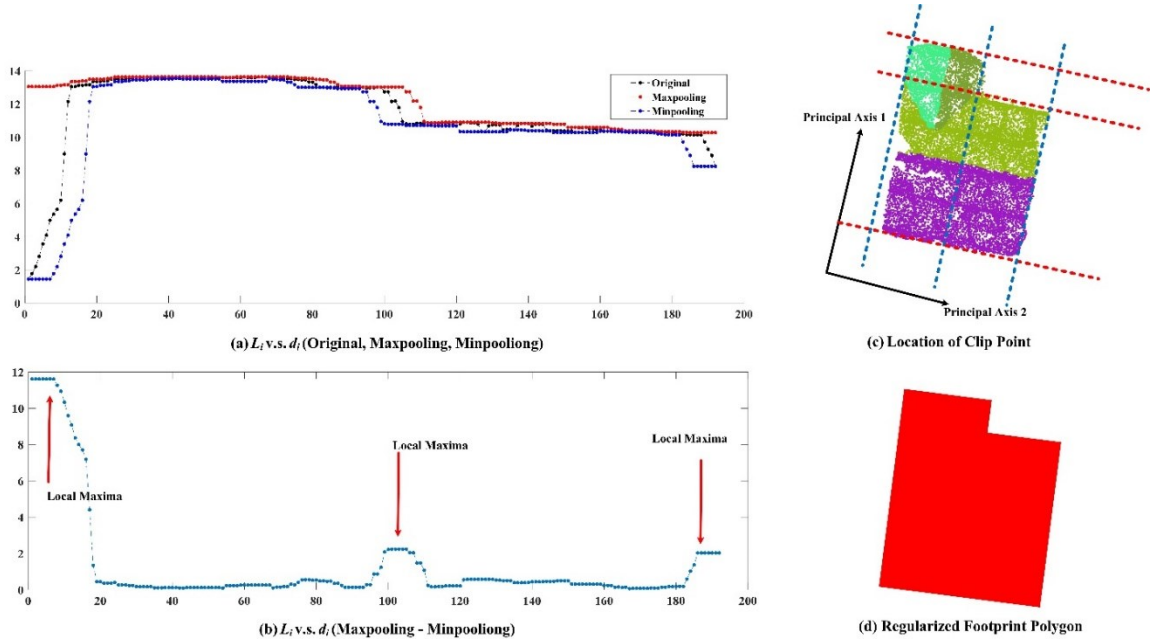


Figure 3.6. Building Footprint Polygon Estimation

Constrained Roof Polygon Estimation: After the estimation of building footprint polygon, I implement multi-rule constrained polygon fitting strategy to reconstruct the polygon of each roof facet. For simplicity, I fit the l(th) roof facet using quadrilateral polygon as $\mathcal{P}^l = \mathcal{P}(x)$, regardless the actual shape of the facet, where x is the vertex of the polygon, and the compact polygon will be obtained in the refinement stage. The facet reconstruction could be formulated as a constrained optimization problem, which could be mathematically expressed as:

$$\text{argmin}_x f(x) \qquad \text{subject to} \begin{cases} (x_i - x_j) \cdot (x_i - x_k) = 0 \\ (x_i - x_j) \cdot X^m = 0 \end{cases} \tag{3.2}$$

where the cost function is defined as: $f(x) = \sum_{i=1}^{N} |v_i \cdot n|$. In the cost function, $\{v_i = p_i - x_0 : p_i \in P^l\}$ is the vector from the ith point of the roof point cloud to the vertex of the polygon, and n is the unit normal vector of the polygon $\mathcal{P}^l$. The first constrain implies that any pair of vertex-sharing boundary lines of the polygon are orthogonal to each other, and the second constrain implies that the boundary line should be orthogonal to one of the principal axis of the building object. This optimization procedure will fit the polygon constrained by the principal axes of the building object. However, this optimization is not able to fit the polygon model with the optimal size, which means the fitted polygon could be either expanded or shrunk in size compared with the actual size. As shown in Figure 3.7, the fitted polygon is larger than the actual size of the corresponding roof facet (shown as pink points). This will be addressed in the refinement stage.

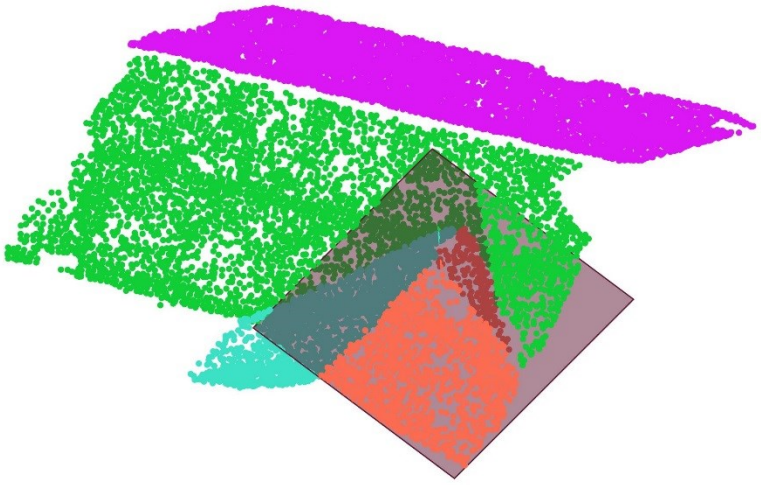


Figure 3.7. Roof Facet Polygon Estimation

Roof Model Refinement: The last stage of roof model reconstruction is refinement of the optimized polygons. For each fitted polygon $\mathcal{P}^l$, I first expand it along each principal direction by a certain value to ensure that the expanded polygon crosses either the building footprint boundary,

roof ridge, or other polygons. After the expansion, I implement polygon Boolean operation to clip the polygon $\mathcal{P}^i$ into multiple small polygons $\{\mathcal{P}_j^l : \cup \mathcal{P}_j^l = \mathcal{P}^l\}$ with either the footprint boundary, roof ridge, or intersected polygons. Due to the complex configuration of the building roof facets, the Boolean operation does not necessarily generate quadrilateral polygon anymore. Although the Boolean operation generates shape-compact polygons, it also generates polygons that do not have inlier points. To remove these polygons, I project the points $\{P^l\}$ on the clipped polygon $\mathcal{P}_j^l$ and find the points projected inside the polygon region as inlier points. Ideally, a polygon $\mathcal{P}_j^l$ is removed if no points projected inside the polygon region. However, this does not work very well in real scenario due to the densification procedure described in previous section when the data resolution of original point cloud is low. In order to robustly remove the invalid polygons, I compute the ratio between the alpha shape area of the inlier points and the area of polygon $\mathcal{P}_j^l$. If the ratio is larger than a threshold value δ_a, this polygon is considered as valid and will kept, otherwise, it is removed as invalid. In this research, the threshold value is set as $\delta_a = 0.1$.

The polygon refinement procedure is graphically illustrated in Figure 3.8. After the expansion (Figure 3.8 (a)) and Boolean operation (Figure 3.8 (b), (c)), the removal of invalid polygon is shown in Figure 3.8 (d), (e). As shown in the figure, the yellow dots represent the inlier points projected by the red dots. And the polygon enclosed by blue line segments represent the alpha shape of the inlier points. If the ratio between the area of projected alpha shape polygon and the clipped polygon does not satisfy the pre-defined criteria, the clipped polygon is removed. A typical example of invalid polygon is given as the red polygon in Figure 3.8 (e).

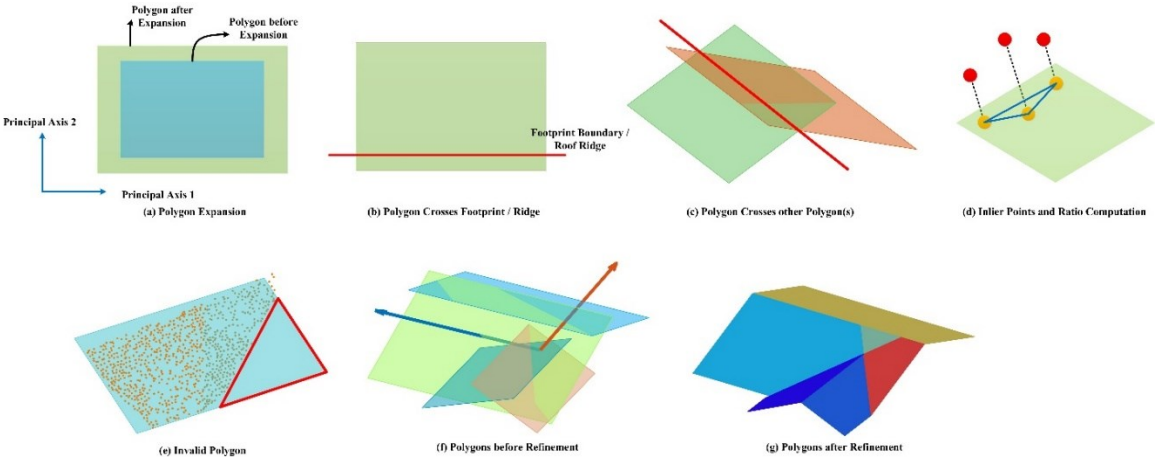


Figure 3.8. Polygon Refinement

Figure 3.9 shows the reconstruction of several typical building objects. The first row of the figure shows the original point cloud with low data resolution, and the second row shows the reconstructed roof models. As shown, the proposed approach is able to reconstruct the models of various type of building objects, such as the building in Figure 3.9 (b) with multiple pair of gable roofs, of the building in Figure 3.9 (e) with isolated parts of different elevation. I will show in the following section that these reconstructed models will be used to measure the building damage with various resolution post-event point cloud.

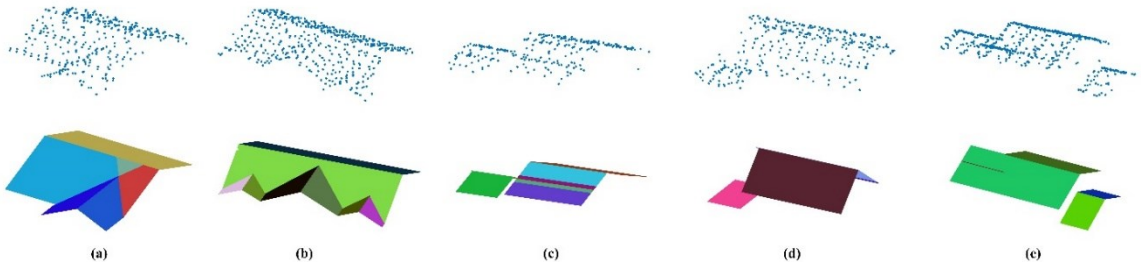


Figure 3.9. Results of Building Model Reconstruction

3.3.3 Post-event Building Objects Extraction

After the reconstruction of pre-event building model, the corresponding post-event building object is extracted in order to achieve the property-wise damage assessment. One solution to this is to detect the post-event building objects individually, and match the objects. This approach, however, is not robust in real case. One reason is derived from the fact that the post-event building could be severely damaged or even totally collapsed, and conventional building detection algorithms will not be able to robustly identify them. In addition, the time-sensitive post-event assessment does not necessarily collect high resolution point cloud data. The post-event airborne point cloud with very low resolution collected after Hurricane Sandy is a typical example (NOAA 2012). Figure 3.10 compares the point cloud resolution of multi-temporal dataset collected at the same region. For each building object, the resolution is represented by the average minimal point-to-point distance (m), therefore, the smaller the average distance, the larger resolution it is. It is shown that the pre-event data resolution is much larger than that of the post-event dataset.

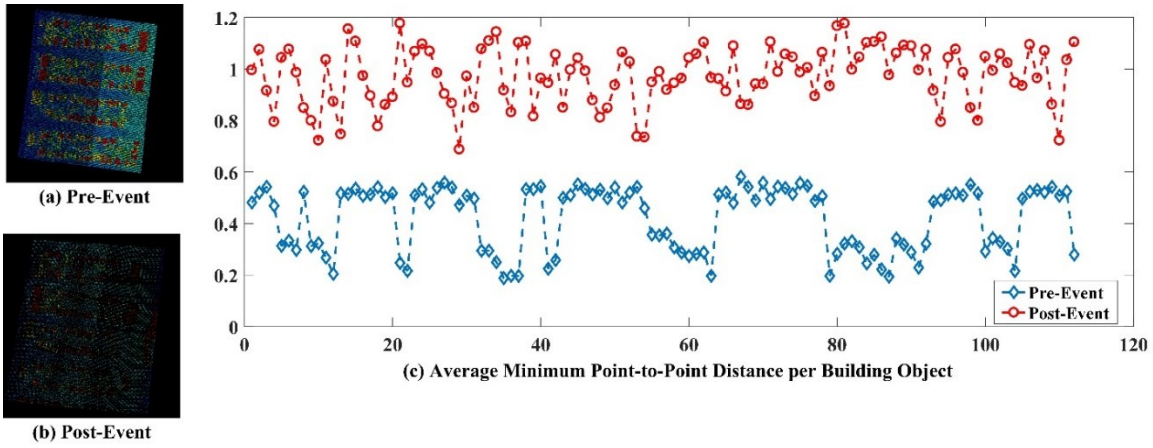


Figure 3.10. Point Cloud Resolution Comparison

To address this issue, I simply crop the post-event dataset using each pre-event building footprint polygons. Ideally, the post-event building object should have exactly the same layout and location of the corresponding pre-event building object. However, if a building is collapsed, rotated, or displaced due to natural hazard effects, the location and layout of the post-event

building object will vary from the pre-event object. Nevertheless, this approach is still recognized as easy and efficient to use for the purpose of property-wise damage detection for the reason that if the cropped point cloud does not match the pre-event model very well, the building is considered as damaged.

3.3.4 Feature Computation

In this section, I propose two features 1) outlier score, $\mathcal{S}$, of the point-to-plane distance, and 2) average inlier area ratio $\overline{\mathcal{R}_{\mathrm{A}}}$ for damage status identification. The outlier score is a robust statistical matric modified from the concept of median absolute deviation (Mosteller et al. 1977). In order to compute the outlier score, the point-to-plane distance d_i is first computed. Instead of subtracting the median of d_i from each d_i, I propose to subtract $\min(|d_i|)$ from each d_i as following

$$\mathcal{S} = \mathrm{median}(|d_i - \min(|d_i|)|) \tag{3.4}$$

Because I assume that for ideal case, the point-to-plane distance should as small as possible and the distribution of d_i is concentrated around zero. For damaged building, the distribution of d_i tends to be more spread and skewed towards large value. The histograms of the point-to-plane distribution of one typical intact building and damaged building are plotted in Figure 3.11 (b), it is noticed that the shape of histogram is distinguishable for various building conditions, and the corresponding outlier scores are comparatively far away from each other, thus this score is an efficient feature implying the building condition.

The average inlier area ratio $\bar{\mathcal{R}}$ is defined as

$$\bar{\mathcal{R}} = \sum_{k=1}^{K}(A_k/\mathbb{A}_k) \tag{3.5}$$

where $\mathbb{A}_k$ is the area of the kth pre-event roof facet polygon, and A_k is the area of the kth post-event roof facet point set (Figure 3.11 (c)). Although it is straightforward to compute the area of pre-event facet as the polygon area, the computation of the post-event point set area is sensitive to variation of data resolution. Conventional area computation relies on construction of alpha shape

or convex hull polygon, however, these approaches are sensitive to resolution variation. As shown in Figure 3.12, the low resolution of point cloud tends to decrease the area of polygon. To address this issue, I propose an efficient resolution-invariant point cloud area estimation approach. Instead of constructing the polygon on the point cloud, this approach estimates the equivalent square for each point. The procedure is illustrated in Figure 3.12 (e), (f). I first construct the triangulation on the point cloud, and remove the invalid triangle edges using MCMD_Z outlier detection algorithm (Nurunnabi et al. 2015) And the length of equivalent square is estimated as the median value of valid edge length. And the area of sparse point cloud is computed as $\beta \sum_i \tilde{r}_i$. In this equation, $\tilde{r}_i$ is the size of equivalent square, and $\beta = 1.2$ is a constant coefficient obtained via statistical experiment. As sown in Table 3.1, the approach is able to estimate the area of point cloud of various resolution. The third row of the table is the ratio of the estimated area to the actual polygon area. It is observed that for various resolution, the estimated area is around 1.2 time of the actual area, therefore, the constant coefficient is estimated as $\beta = 1.2$.

Table 3.1. Statistical Analysis of Resolution-Invariant Area Estimation

Resolution	100%	40%	20%	10%	2%
$\tilde{r}_i$	0.0154	0.0246	0.0344	0.049	0.11
$\sum_i \tilde{r}_i / \mathbb{A}$	1.1977	1.2041	1.1902	1.2023	1.2092

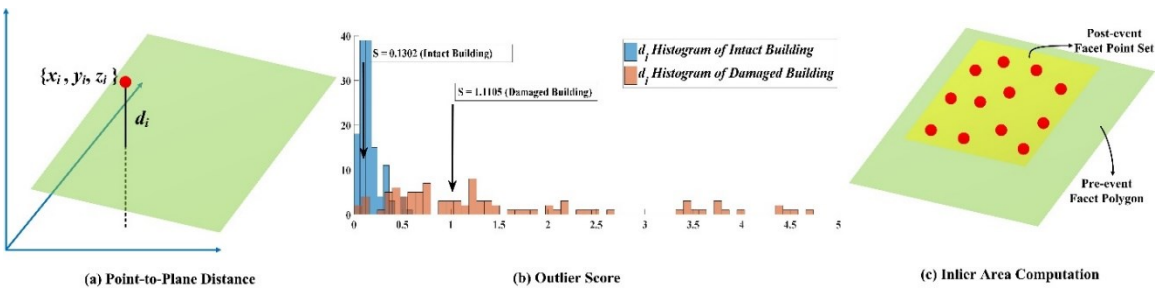


Figure 3.11. Definition of Three Features

In real case, not all the points could be counted as inlier points for roof area computation, I modify the computation of area of sparse point cloud as following:

$$A = \beta \sum_i \tilde{r}_i \mathcal{I}'(p_i, \tilde{p}) \mathcal{I}''(|d_i|, \epsilon_d) \qquad (3.6)$$

where $\mathcal{I}'(p_i, \tilde{p}) = 1$ if $p_i = \tilde{p}$, and $\mathcal{I}'(p_i, \tilde{p}) = 0$ otherwise, and $\tilde{p}$ indicates the point is inside the region of pre-event facet polygon. And the definition of another term is $\mathcal{I}''(|d_i|, \epsilon_d) = 1$ if $|d_i| < \epsilon_d$ and $\mathcal{I}''(|d_i|, \epsilon_d) = 0$ otherwise, where ϵ_d is the distance threshold value.

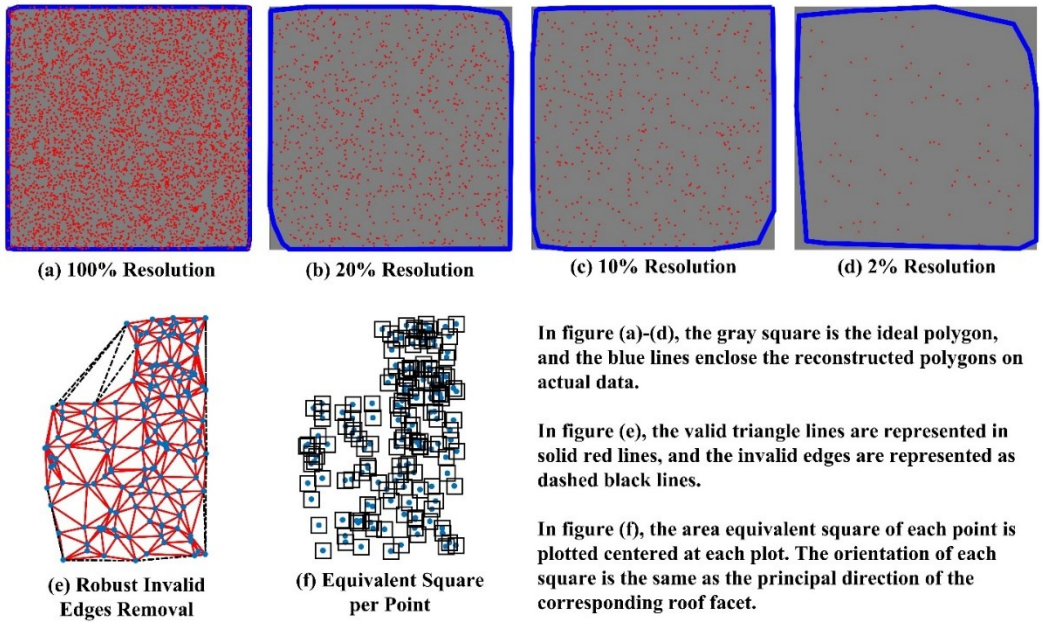


Figure 3.12. Resolution-Invariant Point Cloud Area Estimation

3.3.5 Damage Detection

The last stage of this framework is damage detection using the features described in previous section. I classify the building condition into the following categories, 1) Intact, 2) Partially Collapsed, and 3) Totally Collapsed. The concrete description of the rules associating each damage pattern to features is described as follow.

Intact: A building object is considered as intact if the following two criterions are satisfied simultaneously:

$$\{\mathcal{S} < \Delta_{\mathcal{S}} \; and \; \bar{\mathcal{R}} > \Delta_{\bar{\mathcal{R}}}\} \qquad (7)$$

where $\Delta_{\mathcal{S}}$, and $\Delta_{\bar{\mathcal{R}}}$ are two threshold values which are pre-defined by users. Intuitively, the first criteria implies that whether height change, collapsing, deformation, inclination, or rotation exists. If such scenarios exist, the distribution of d_i spreads, and as consequence, the outlier score raises significantly. If none of these scenarios exists, the post-event point cloud should fit the pre-event model, therefore, the distribution is highly concentrated and the outlier score drops. The second criteria implies that the area of post-event roof facets are close to that of pre-event roof facets.

Partial Collapsed: Building partial collapsing includes various damage patterns: 1) inclination, 2) displacement, 3) rotation. As shown in Figure 3.13 (b), (c), a building is considered as partially collapsed if one of the followings is satisfied:

$$\{\mathcal{S} > \Delta_{\mathcal{S}} \text{ or } 0 < \bar{\mathcal{R}} < \Delta_{\bar{\mathcal{R}}}\} \qquad (8)$$

The interpretation of the first criteria is straightforward, no matter what types of damage pattern exist, the post-event point cloud does not fit the pre-event model very closely, which leads to the raise of outlier score. I constrain the second criteria to be in between the range of $(0, \Delta_{\bar{\mathcal{R}}})$. On one hand, the upper boundary implies that the area of post-event roof facets are smaller than the pre-event facets, on the other hand, however, there are still part of the inlier post-event points (red points in Figure 3.13).

Totally Collapsed: I define the following two categories as totally collapsing: 1) height change, 2) wiped out. Specifically, the following criterions are satisfied if a building object is considered as totally collapsed:

$$\{\mathcal{S} > \Delta_{\mathcal{S}} \text{ and } \bar{\mathcal{R}} = 0\} \qquad (9)$$

As shown in Figure 3.13 (d), (e), for a totally collapsed building (height change or wiped out), it is assumed that no inlier points exist from post-event dataset. Therefore, the average inlier area ratio is 0 for this scenario.

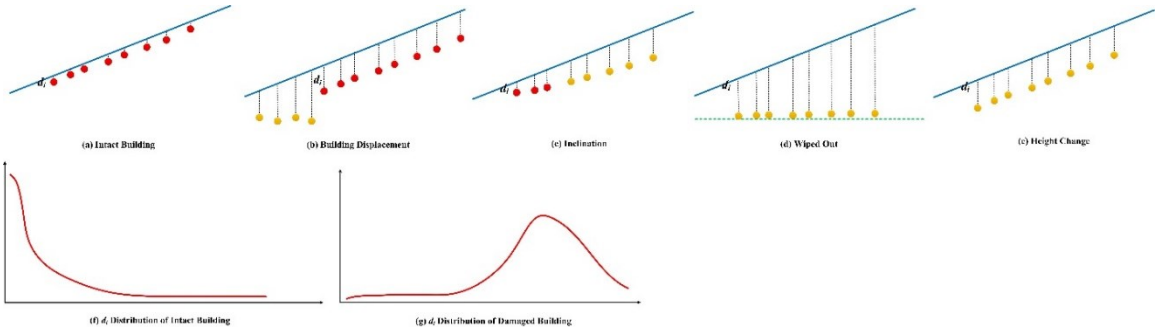


Figure 3.13. Damage Patterns

3.4 Experiment

3.4.1 Data Description and Pre-processing

I conduct an experiment on the data set collected at New Jersey shoreline area (Figure 3.14 (a)). The pre-event dataset was collected at 2010, and the post-event dataset was collected right after the occurrence of Hurricane Sandy. To be specific, I focus on the area where the impact was severe (Figure 3.14 (b), (c)). In order to validate the results of damage detection, two datasets are employed as ground truth. The first dataset is the mobile LiDAR point cloud collected at the same area after the particular natural hazard (Figure 3.14 (d)), and the visual interpretation approach is conducted on the mobile point cloud to estimate the building damage status. In the following parts, this dataset is denoted as MLS-2012 for simplicity and clarity. Another ground truth dataset is collected by Xian et al. (2015) and Hatzikyriakou et al. (2015) (Figure 3.14 (e)). During the survey, a very comprehensive evaluation was conducted which classifies the building damage conditions into first floor damage, second floor damage, third floor damage, roof damage, and overall damage. However, because for airborne LiDAR-based detection, only the rooftop is visible, the matric adopted in their research is not proper to be directly applied to the airborne point cloud. To address this, only the overall damage matric is used as for the validation. For simplicity, I denote their dataset as HS-2015.

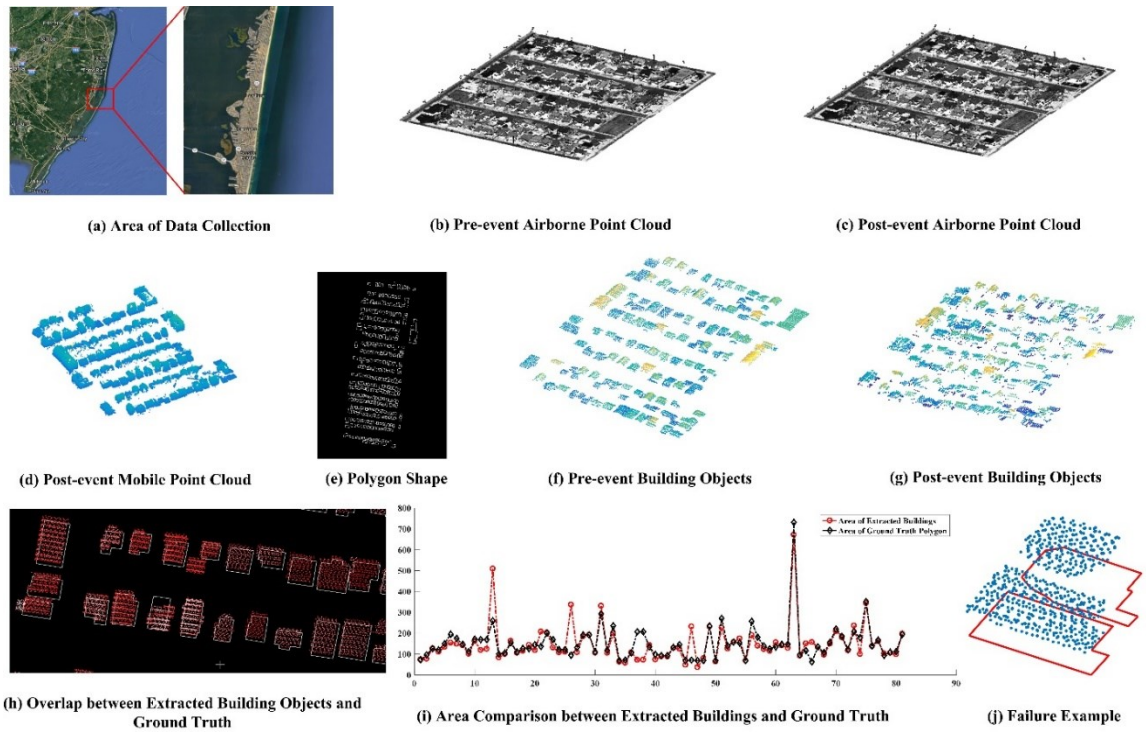


Figure 3.14. Data Description

The results of building objects extraction during the pre-processing step are shown in Figure 3.14 (f), (g). The pre-event building objects are extracted using the approach described in section 3.1, and the corresponding post-event building objects are extracted as described in section 3.3. To evaluate the accuracy of building extraction, the manually created building footprint polygons are used as shown in Figure 3.14 (h). It is observed that the extracted building objects overlap with the ground truth footprint polygon well. The quantitative analysis of the area of extracted building and the ground truth polygon is shown in Figure 3.14 (i). Although most of the pre-event building objects match the polygon shape accurately, there are still some buildings have much larger areas compared with that of the corresponding ground truth polygons. A typical failure example is shown in Figure 3.14 (j), because the buildings are adjacent to each other very closely, which is hard to be identified as two buildings from airborne LiDAR point cloud when the data resolution is close to the gap between two objects.

3.4.2 Pre-event Building Model Reconstruction

The parameters for building model reconstruction are listed in Table 3.2. The choose of δ_d and δ_a depends on the resolution of dataset, generally, the lower the resolution is, the larger the parameters are chosen.

Table 3.2. Parameters for Model Reconstruction

Parameter	θ	δ_d	δ_l	δ_a
Value	5°	0.25(m)	1(m)	0.1(m)

Figure 3.15 shows the results of building model reconstruction of some typical building configuration. The first row of the figure is the pre-event airborne LiDAR point cloud. The second row is the corresponding reconstructed models, and the third row shows the mobile LiDAR point cloud. To evaluate the performance of proposed efficient building model reconstruction approach, the ground truth polygon shapes are used for comparison. Figure 3.16 (a) visualizes the results of all the reconstructed models and the comparison with corresponding ground truth polygon. It is observed that the reconstructed polygons match the ground truth polygon well in terms of polygon size and shape. The quantitative analysis is carried out in comparing the area between reconstructed models and the ground truth models. As shown in Figure 3.16 (b), the area of reconstruction model is plotted in red color, and the corresponding ground truth is colored in black. As stated by the figure, the areas are close for most of the building objects, and the histogram of the area ratio suggests also suggests the consistent result (Figure 3.16 (c)).

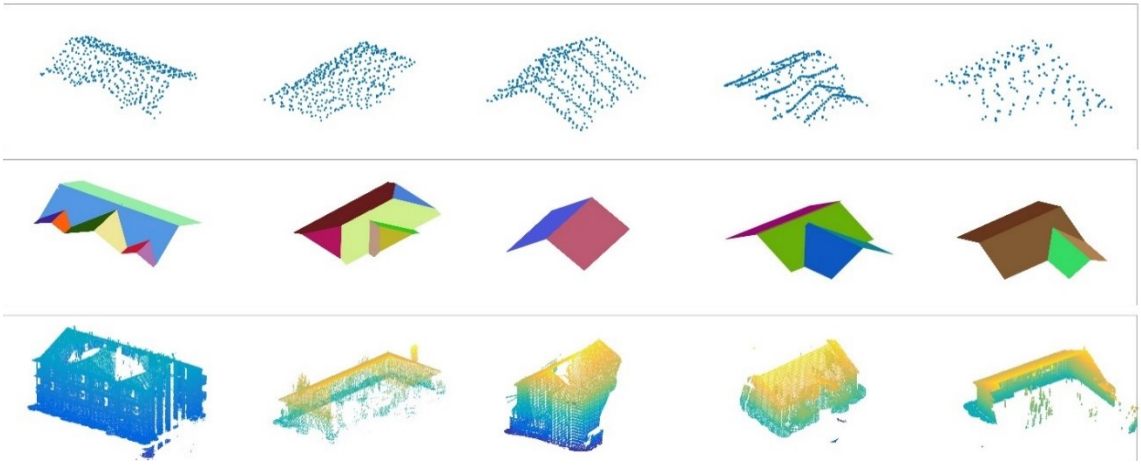


Figure 3.15. Results of Efficient Building Model Reconstruction

3.4.3 Feature Computation and Damage Status Detection

As described in section 3.3.4, two features are computed for each pair of pre- and post-event building objects. The results of features are shown in Figure 3.17. The result of outlier sore $\mathcal{S}$ shows distinguishable patterns, where most of the values are close to 0, and few of them jumps up significantly. The result of average inlier area ratio $\bar{\mathcal{R}}$ illustrates a more noisy pattern, comparatively. Shortly, I will argue that the average inlier area ratio is more sensitive to the data resolution compared with the outlier score. Although it is found in the following discussion that the outlier score is more robust and insensitive to data resolution in building damage detection, the area ratio is still a necessity feature in identifying partial collapsing scenario.

I first evaluate the result of proposed approach using the post-event mobile LiDAR point cloud as reference, and the corresponding building damage status is evaluated by the author manually based on the rules defined in section 3.3.5. The typical building status are illustrated in Figure 3.18. The intact buildings are visualized in blue, the partially collapsed buildings are colored in orange, and the totally collapsed buildings are identified in red. The comparison between the ground truth(MLS) and detected damage status the proposed approach suggests that the proposed approach is able to detect $8/12$ of the totally collapsed buildings correctly.

However, it is noticed that this approach tends to over-evaluate the building damage status. Another comparison between HX-2015 dataset and mine suggests that this approach is able to $11/14$ totally collapsed buildings correctly. Quantitatively, the confusion matrixes of each comparison are listed in Table 3.3. Each confusion matrix is constructed on binary bases, therefore, the off-diagonal elements indicate the error detection. Specifically, the error rate for confusion matrix I-NI i$\mathcal{E}_{I-NI} = 17.82\%$s, the error rate for P-NP is $\mathcal{E}_{P-NP} = 21.78\%$, the error rate for T-NT is $\mathcal{E}_{T-NT} = 3.96\%$, and the error rate for I-D is $\mathcal{E}_{I-D} = 24.75\%$, respectively. The error rate for partial collapse and intact are significantly higher than that of total collapse case, and this is because the actual intact buildings are recognized as partially collapsed, which will be analyzed shortly so that the data resolution does affect the building damage detection using airborne LiDAR point cloud. Although the error rate for the comparison between HX-2015 dataset and mine is also large, it is mainly because the standards of evaluation can hardly be unified. Therefore, the failure analysis will focus on the comparison between MLS-2012 dataset and ours.

Table 3.3. Confusion Matrixes

I-NI	Intact (MLS-2012)	Non-Intact (MLS-2012)
Intact (Ours)	69	0
Non-Intact (Ours)	18	14

P-NP	Partial Collapse (MLS-2012)	Non-Partial Collapse (MLS-2012)
Partial Collapse (Ours)	2	22
Non-Partial Collapse(Ours)	0	77

T-NT	Total Collapse(MLS-2012)	Non-Total Collapse(MLS-2012)

Total Collapse(Ours)	8	0
Non-Total Collapse(Ours)	4	89

I-D	Intact(HX-2015)	Damage(HX-2015)
Intact(Ours)	65	3
Damage(Ours)	22	11

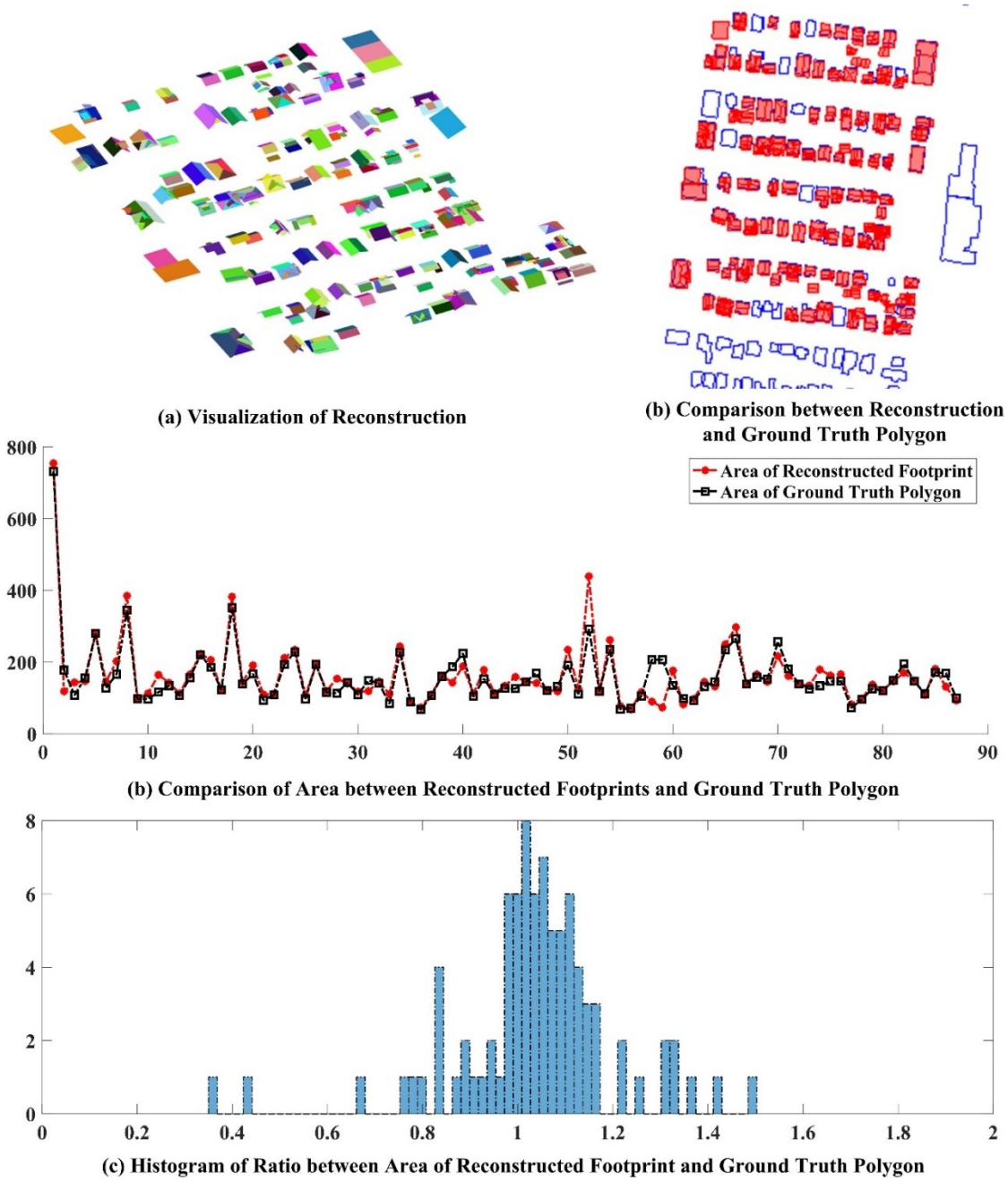


Figure 3.16. Comparison between Reconstruction and Ground Truth Polygon

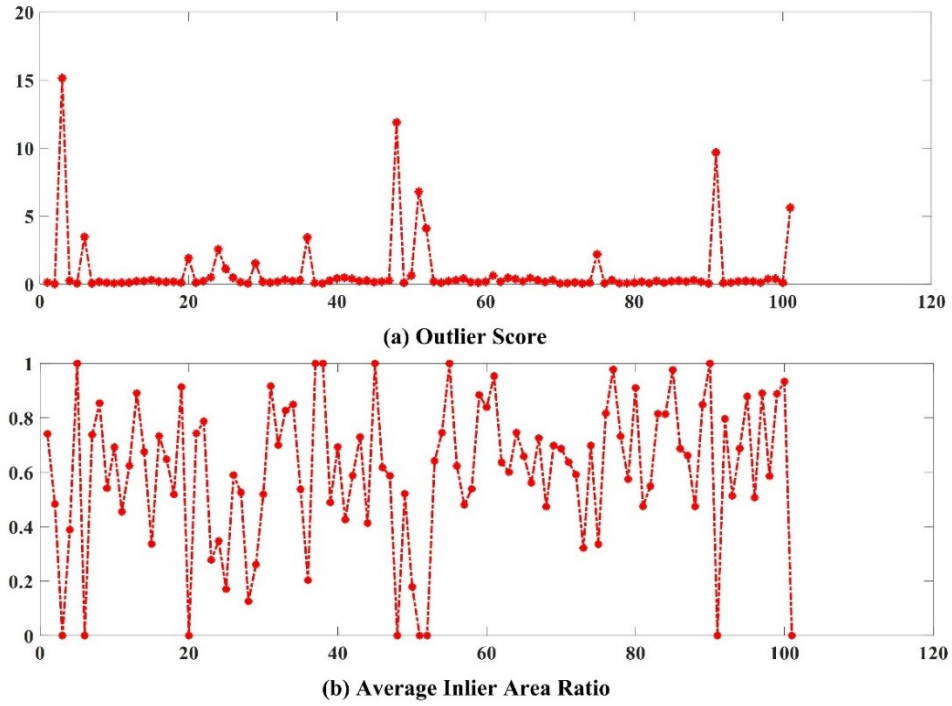
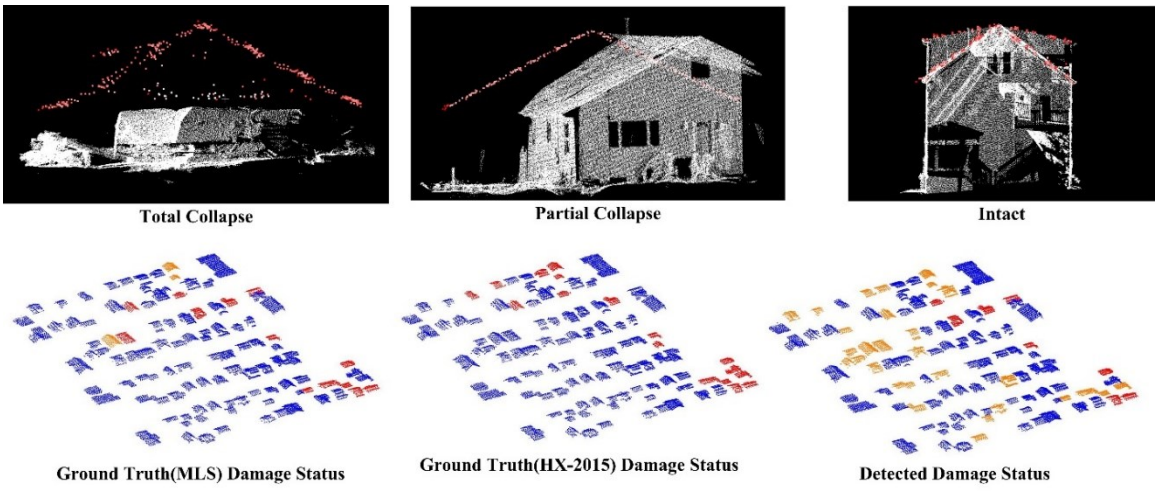


Figure 3.17. Results of Feature Computation



The first row shows three typical example of building status, including total collapsing, partial collapsing, and intact. The pre-event airborne LiDAR point cloud is plotted as pink points, and the post-event mobile LiDAR point cloud is visualized in gray-scale.

The second row shows the damage status of ground truth and detected result. The intact buildings are visualized in blue, the partial collapsed buildings are colored in orange, and the total collapsed buildings are represented in red.

Figure 3.18. Results of Damage Detection

3.4.4 Failure Analysis

In this section, I will closely look at some of the failure cases where intact buildings are recognized as partial collapse. Or totally collapsed buildings are recognized as intact. I categorize the miss-recognition into the following groups: 1) failure actually caused by proposed approach, 2) failure caused by building modification, 3) failure caused by low resolution of post-event data, and 4) failure caused by accuracy issue of post-event data. Figure 3.19 shows the four typical examples of each type of failure mode. As shown in Figure 3.19 (a), the building roof configuration is complex, and the densification of original airborne point cloud is unable to preserve the actual roof layout. As the result, the reconstructed roof model is inaccurate and misses part of the roof facets. Figure 3.19 (b) presents the case of building modification. The pre-event airborne point cloud is shown in pink, and the post-event mobile point cloud is shown in gray-scale. Clearly, the large outlier score and low inlier area ratio is caused by the modification of building configuration. Although the building is not damaged during the hazard, the proposed approach detects huge change and therefore, recognizes it as severe damage. The failure mode caused by low resolution of post-event data is shown in Figure 3.19 (c). The reconstruction of the building model correctly reflect the actual building configuration, while, the low resolution of post-event data (colored in orange) leads to significant drop of area ratio $\bar{\mathcal{R}}$. Lastly, the example of failure caused by post-event data accuracy issue is stated in Figure 3.19 (d). It is obvious that the reconstructed model matches the post-event mobile point cloud correctly, while the post-event airborne point cloud shifts upwards against the mobile data. This is because the registration error between scanning frame and navigation trajectory data. As a consequence, the building is detected as partially collapsed by this approach.

I conducted a thorough analysis on the failure cases and identified that among the error detection, five of them are caused by low resolution, three of them are caused by low resolution and accuracy issue. In addition, there are also two cases of building modification.

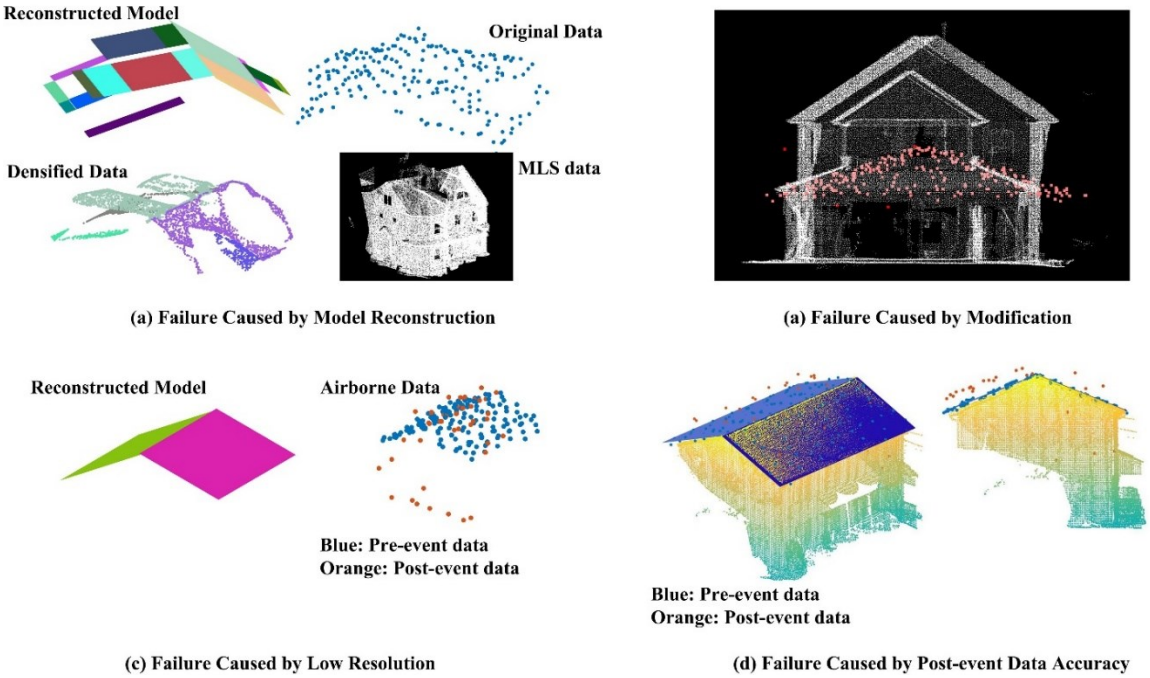


Figure 3.19. Failure Example

3.5 Discussion

In this research, I propose a novel framework to detect building damage using multi-temporal airborne LiDAR point cloud. To overcome the variation of data resolution, the pre-event point cloud is densified at first, and an efficient model reconstruction approach is introduced to estimate the building roof configuration. This is to address the issue of very low resolution of post-event. To detect the damage status, two features, outlier score and inlier area ratio, are estimated and three damage categories are identified based on these features. An experimental study was conducted on the data collected after the impact of Hurricane Sandy, and two reference datasets were utilized for ground truth. The result suggests that this approach is able to identify the totally collapsed building accurately regardless the variation of post-event data resolution. And the detection of partially collapsed building is relatively sensitive to the data resolution.

Chapter 4 : Automated Analysis Of Mobile Lidar Data For Component-Level Damage Assessment Of Building Structures During Large Coastal Storm Events

Rapid assessment of building damages due to natural disasters is a critical element in disaster management. Although airborne-based remote sensing techniques have been successfully applied in many post-disaster scenarios, automated building component-level damage assessment with terrestrial LiDAR data is still challenging to achieve due to lack of reliable segmentation methods for damaged buildings. In this research, a novel damage detection approach is proposed to realize automated building component level (wall, roof, balcony, column, handrail) damage assessment. Specifically, the proposed approach first conducts semantic segmentation of building point cloud data using a rule-based approach. The detected building components are then evaluated to determine if the components are damaged. I applied this method on a mobile LiDAR data set collected after Hurricane Sandy. The results demonstrate that the approach is capable of achieving 96% and 86% parsing accuracy for wall facades and roof facets, and obtain 82% and 78% of detection accuracy for damaged walls and roof facets.

4.1 Introduction

Severe weather events such as hurricanes and storm surge have been threatening places where economic and industrial activities are heavily concentrated across U.S and around the global. Damage assessment is critical for better understanding of the hazard impacts and damage condition. Traditional approaches rely on foot-on-ground survey, and the assessment is conducted one building after another often based on manual assessment (Chiu et al. 1999). While these approaches have the potential to collect very detailed damage data, they are slow, labor intensive, and error prone, which often lead to missed opportunities in collecting valuable data and gaining timely insights for building more resilient future communities. In the case of hurricane events which often occur cross states, it would take weeks or months for these manual methods to complete damage assessment for impacted areas. This could result in delay in obtaining baseline

data to inform how to restore to normalcy while preventing shortsighted decisions which can lead to repeated failures.

Damage detection in civil and infrastructure domain heavily relies on the use of sensors. Many in-depth research efforts have been carried out to detect the damage from input signal patterns obtained by close-range sensors (Cha et al. 2017, Lin et al. 2017, Rafiei et al. 2017). Although impressive performance is obtained, the damage detection of large-scale, especially community-scale, is another story as single close-range sensors cannot provide comprehensive observation for all the interesting objects. Therefore, advanced geospatial sensing technologies are playing an increasingly important role for decision support in disaster preparation, response, and recovery operations as they greatly expand this ability of collecting disaster data. The types of geospatial data that are useful for post-disaster damage assessment include photo imagery and LiDAR data. Infrared, hyperspectral, and radar imagery may also be useful since they complement characteristics of images captured in other portions of the spectrum. More recently, high-resolution spatial sensing instruments such as static/mobile LiDAR and 3D dense reconstruction from images have been increasingly used for damage mapping due to their apparent advantages in capturing finer-scale of damage patterns and in collecting damage data on vertical facades (Olsen et al. 2009, Olsen et al. 2012, Gong et al. 2014, Kashani et al. 2015, Zhou et al. 2015). While these high-resolution damage data sets promise better insights into damage mechanisms, interpretation and processing of these data sets in a timely fashion are also growing challenges facing damage assessment teams.

Use of computer vision methods to interpret remotely sensed disaster data is not a new topic. In particular, damage analysis tasks such as building detection, ground classification, and bridge detection from 3D point clouds and imagery are well studied (Sithole et al. 2006, Meng et al. 2010). However, most of these studies focus on imagery and LiDAR data captured from aerial-based platforms. A recent focus of computer vision based damage assessment has been placed on processing oblique imagery collected in disaster impacted areas with either manned or unmanned

aircrafts. For example, Vetrivel et al. investigated the application of high resolution oblique imagery data for building damage detection (2016, 2016). In their research, the texture features are extracted from the imagery to detect the building damages. One major limitation of imagery-based approach is that it cannot estimate the damage in actual size unless the imagery data is accurately geo-referenced. To overcome this, Tu et al. (2016) leveraged the fusion of oblique imagery and GIS data to improve the performance of damage detection. In this research, texture features of 2D imagery are employed to recognize the damage pattern. 3D GIS data are then combined to retrieve the damage size. However, a major constraint of this approach comes from the fact that the applicability is restricted by the data availability. In another direction, Verykokou et al. (2016) proposed to use UAV-based imagery to reconstruct dense 3-D model for damage assessment. Although very dense point cloud could be obtained from image-based reconstruction, the damage assessment is still conducted in global level, meaning that automated semantic damage detection is still a problem to be resolved.

Use of dense point cloud data reconstructed from oblique imagery is often complicated by the presence of reconstruction artifacts and lack of accuracy standards. The latter in particular challenges its use in evaluating damages to critical infrastructures. Automated damage assessment with static/mobile LiDAR data which often have clear accuracy standards and specifications is an intriguing option. Kashani et al. (2016) tested terrestrial LiDAR based approach to evaluate building damage due to tornado events. However, their approach mainly focused on rooftop failures. A general observation is that studies on interpreting building component-level damages with dense point cloud data are scarce. The reality is that while ones are capable of drastically reducing field time for collecting data about damage, debris, and waste with sensing instruments, the collected disaster data still have to go through painstakingly manual interpretation in office, delaying delivery of critical information to stakeholders.

To address this issue, I propose an automatic semantic building damage detection approach for post-disaster building structural damage evaluation with dense LiDAR point data. This

approach first semantically parses the building into segments, then the damage detection is conducted to extract the semantic structural damage information. This approach focuses on the parsing and detection of following structural components: wall, roof, handrail, and column. I then evaluate this approach on the dataset collected in the New Jersey Shoreline area after Hurricane Sandy. The experiment results show that the proposed approach is able to parse the component level building damage.

The contributions of this paper include: (1) the study provides a fully automated method to semantically parse dense point cloud data of residential buildings, regardless of whether they are damaged or not, into major building components including roof, exterior walls, columns, and handrails; (2) the method is designed to eliminate any data pre-processing steps, making it user friendly to practitioners; and (3) the damage detection is based on topological assumptions and constraints, and does not require any pre-event data as reference data. This makes the proposed method applicable even there is no pre-event data available.

The paper is organized as follow: The related research work regarding the semantic building parsing and 3D reconstruction are firstly reviewed. Then I introduce the methodology of the proposed approach in detail. Experiment is implemented to verify the approach. And finally conclusion of the research is drawn.

4.2 Related Work

As the point cloud processing applicable to post-hazard damage scenario, parsing the point cloud into semantically meaningful segments is a critical prior. A major approach used in semantic segmentation of point cloud data is plane extraction. RANSAC (Fischler et al. 1981) has been proven to be one of the most popular and robust algorithm since its emerge. Deschaud et al.(2010) proposed a region-growing based plane extraction algorithm. It starts from computation of normal and local scores, then the points are propagated according to the curvature. Based on these approaches, various algorithms are proposed in application of various scenarios such as(Schnabel et al. 2007, Tarsha-Kurdi et al. 2007, Subramaniam et al. 2014). Wang et al.(2013)

proposed a graph-cut based plane segmentation algorithm, the algorithm merges planes and cuts planes that minimizes the energy function. Nguyen et al.(2016) proposed to use scan line, plane objects are detected based on the scan profiles. A voting-based approach is introduced by (Feng et al. 2016). In their approach, point cloud is first over-segmented. And the co-planar segments are detected via their tensor-voting based algorithm. In many cases, the point cloud is corrupted by noise, to address the issue of plane extraction from noisy point cloud, Zhang et al.(2015) developed a linear subspace decomposition-based approach. In this research, the linear subspace of the noisy point cloud is retrieved through a sparse optimization procedure. Then an extended version of RANSAC, called MLESAC, is adopted to estimate the optimal model of shape primitives.

Once the geometric primitives such as planes are extracted, they are often parsed into high-level building elements based on rules and heuristics. Ripperda et al.(2009) introduced a system leveraging the structural information to parse the building into components. However, their approach is not fully automated, and, as a consequence, which is not an ideal tool in post-disaster scenario considering the time sensitivity. Becker et al.(2009) introduced a grammar-based building parsing framework. In their approach, façade segments are first extracted from the raw data in preprocessing phase, and a building grammar is used to parse the facades into semantic components. More complex grammar are introduced by (Pu et al. 2009, Toshev et al. 2010). Their approach leverages the patch features and topological relationship to achieve optimal global parsing results. Martínezet al.(2012) introduced an automated point cloud processing algorithm. Their approach assumes that the orientation of sensor is known. The point cloud is oriented and the planar objects are detected via finding the peaks of oriented profiled point cloud. Xiong et al.(2013) introduced a context-based parsing framework . Then a machine learning-based approach is used to assign the patches with semantic labels. Aijazi et al. (2013) proposed a voxel-based approach. In their study, a chain-link method was implemented to generate initial object candidates, and the geometric descriptors are computed from each linked voxels to segment the

point cloud into semantic meaningful objects. Vanegas et al.(2012) proposed an approach in reconstructing the urban scene from point cloud. Their approach does not require labeling, but it is based on the assumption of orthogonality of fundamental components. And therefore is not applicable to my scenario as in this case, the buildings might be damaged and might loss regular geometric relationship among building components. Monszpartet al.(2015) and Ochmann et al.(2016) proposed algorithms in reconstructing complex indoor scenes. Their approaches are able to reconstruct plane-based 3-D models from very complex point cloud dataset. Recently, Li et al. (2017) introduced a novel hierarchical approach to semantically segment urban environmental façade objects. In their approach, input point cloud is firstly converted into individual plane objects. Then a SVM-based approach, called BieS-ScSPM algorithm, is carried out to assign each individual plane objects semantic meaning. Lastly, an object regularization procedure is introduced to refine the parsing and segmentation result.

In addition, studies focusing on building reconstruction, modeling have been carried out in recent years. Truong-Hong et al. (2011) proposed fully automated approach to reconstruct geometric model of building façade from dense point cloud. Their approach first detect boundary points using Delaunay triangulation construction. Then the boundary line segments are fitted on the boundary points. Then voxel-based approach is leveraged to reconstruct the façade model. Instead of using Delaunay triangulation, Truong-Hong et al. (2013) proposed a novel approach that uses KNN searching to extract boundary points. A slicing based approach is proposed by Zolanvari et al. (2016). In this study, the façade boundary are extracted via horizontally and vertically slicing the point cloud. And the façade model could be reconstructed via fitting and adjusting the boundary points.

The aforementioned research effort focuses on processing point cloud data of intact building objects, which often have strong and well-understood topological cues. For example, a wall facet is always vertical to ground plane. In post-disaster scenarios, these patterns are not necessarily

preserved because the natural hazards generate much complex and rarely recognized damage patterns. For example, a vertical wall façade might be recognized as roof facet if severe inclination occurs. Also, a horizontal balcony floor might be recognized as roof if it inclines to certain degree because of the failure of supporting structures.

To deal with the segmentation of damaged buildings, Kashani et al. (2015) proposed a cluster-based approach using the point cloud with texture information to detect the roof damage. However, this approach is limited of application because it focused on the roof objects as a prior. In terms of the entire building objects, the various sizes of different building components will affect the performance of the k-means algorithm adopted in the study.

Therefore, this study proposes a novel approach aims at addressing the abovementioned issues. To the best of my knowledge, these is still no research attacking the problem of automatic parsing and detecting component level building damage using ground-based lidar point cloud.

4.3 Methodology

The proposed approach involves a data processing workflow that has five phrases (Figure 4.1). The first phase is plane segmentation. In this stage, I proposed to use a supervoxel propagation based plane segmentation to avoid the computing of point normal. The second phase is initial parsing. In this stage, the segments are parsed into semantic building components with a rule-based parser. A MRF based optimization approach is implemented in the third phase to optimize the parsing results. Based on optimized semantic segments, I parse the handrails and columns in the fourth phase. The last phase involves component-level damage detection and assessment. In the following, detailed description of the algorithms used in each phase is provided.

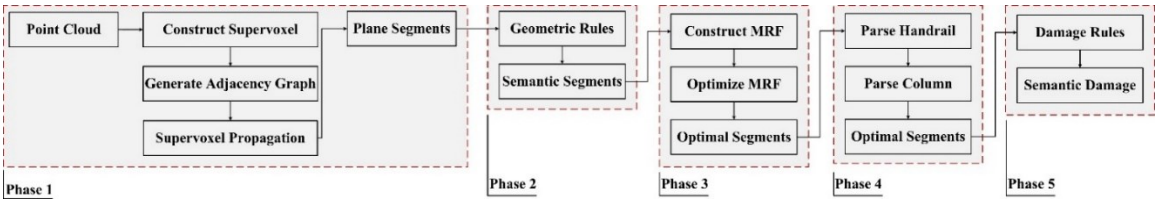


Figure 4.1. Workflow of Proposed Approach

4.3.1 Supervoxel Propagation-based Plane Segmentation

I use supervoxel propagation-based plane segmentation to conduct initial plane detection in this phase. Where the term 'supervoxel' refers a set of points that are locally connected, and share the similar normal direction. The reasons that I propose this approach rather than other commonly used algorithms such as RANSAC and Region-Growing are: 1) Region-Growing requires the computation of point normal and implementation of growing at a global scale, which exponentially increase the computational time with the increase of data size; 2) RANSAC fits a model when most points are considered as inliers. This might return invalid fitting result if not supervised. For example, it might fit an inclined plane if most points are considered as inliers, while in fact these points could belong to multiple models respectively. This is especially true for disaster scenes with full of clutters.

To address these issues, I propose to propagate supervoxels rather than individual points. I first construct supervoxel using the method proposed by (Papon et al. 2013). Due to the noise in point cloud and model complexity, it might return supervoxels which contain multiple planes (Figure 4.2 (a)). To ensure that each supervoxel only contains one plane, a RANSAC-based plane fitting is implemented to extract each plane models from the supervoxels. Because the points in a supervoxel are highly concentrated, RANSAC is not likely to fit models that merge together points belonging to different models in this approach. Because the RANSAC is applied on every supervoxel, I assume that each supervoxel contains at most two or three planes. During the implementation, when a plane is found and the inliers are removed, if the remained points is less than a predefined ratio (in this research, 1/4), I stop the plane search. Therefore, each supervoxel satisfies the following:

$$\{\mathcal{V}(\boldsymbol{p}_i) \mid \nexists(|(\boldsymbol{p}_i - \boldsymbol{p}_0) \cdot \boldsymbol{n}|/\|\boldsymbol{n}\|_2) > \delta\} \tag{4.1}$$

where $\boldsymbol{p}_i$ is a point in supervoxel, $\boldsymbol{p}_0$ is an arbitrary point on the fitted plane, $\boldsymbol{n}$ is the normal vector of the plane model, and δ is a positive constant value determines the inlier points. And $\mathcal{V}$ denotes a supervoxel after RANSAC plane fitting.

With the supervoxels constructed, I construct the adjacency graph (Figure 4.2 (c)) as: $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ denotes node in the graph, and $\mathcal{E}$ denotes the edge connecting two adjacent nodes. Two supervoxels $\mathcal{V}_i$ and $\mathcal{V}_j$ are adjacent to each other if there exist such a pair of points that the point-to-point distance is less than a pre-defined threshold. Mathematically, a pair of supervoxels are considered adjacent if the following equation is satisfied:

$$\{\exists(|\boldsymbol{p}_i - \boldsymbol{p}_j| < \varepsilon), \boldsymbol{p}_i \in \mathcal{V}_i, \boldsymbol{p}_j \in \mathcal{V}_j\} \tag{4.2}$$

I then propagate the supervoxels along the adjacency graph to extract the segments. The pseudo code is presented in Algorithm 4.1. Starting from an arbitrary supervoxel, I propagate its adjacent supervoxel if their normal vectors are parallel to each other: $(\mathcal{V}_i \parallel \mathcal{V}_j)$, here a pair or adjacent supervoxels are considered parallel if they satisfy: $(\boldsymbol{n}_i \cdot \boldsymbol{n}_j) > \theta$. The propagation is iteratively implemented until no more supervoxel can be propagated. Example results of supervoxel propagation based plane segmentation are shown in Figure 4.2 (d).

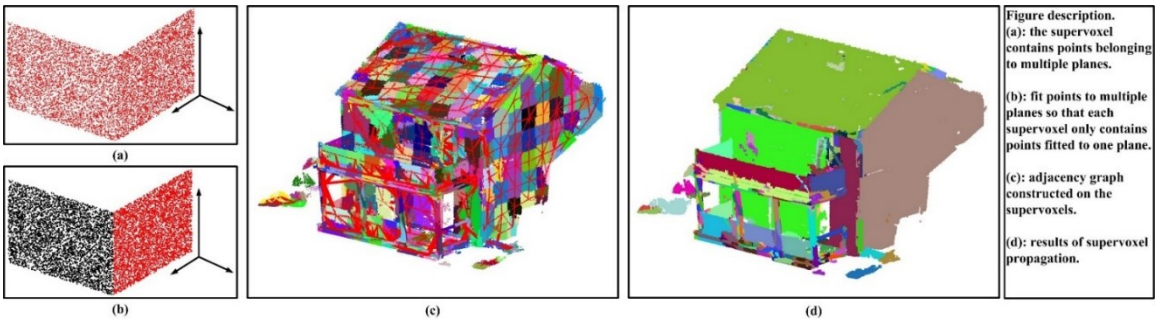


Figure 4.2. Plane Detection and Segmentation with Supervoxel Propagation

Algorithm 4.1. Pseudo Code of Supervoxel Propagation

```
{isVisited} = { };
{𝒱} = {𝒱_i, i = 1,2, … , n};
while ( {𝒱} is not empty )
  if ( 𝒱_i is not visited )
    {query} = { }
    {isVisited} ← 𝒱_i;
    {query} ← 𝒱_i;
    {𝒱} ← {𝒱} \ 𝒱_i;
    while ( {query} is updated )
      for (every supervoxel {𝒱_i ∈ query} , get its adjacent supervoxel 𝒱_j )
        if (𝒱_j is not visited and (𝒱_i ‖ 𝒱_j) )
          {isVisited} ← 𝒱_i;
          {query} ← 𝒱_j;
          {𝒱} ← {𝒱} \ 𝒱_i;
        end if
      end for
    end while
  end if
end while
```

4.3.2 Rule-based Parsing

Because the layout of building components always has geometrical patterns, a rule-based parsing algorithm is constructed in this phase to parse the segments into semantic building components. Algorithm 4.2 presents the pseudo code of the rule-based initial parsing. As listed in the table, the parsing can be conducted in three different ways: the first path is chosen when a segment is vertical, the second path is chosen when a segment is inclined, and the third path is chosen if a segment is horizontal. At the same time, the geometric relationships between different segments are defined as: $A \leftrightarrow B$ denotes A and B are adjacent to each other; $A \uparrow B$ denotes A is entirely above B; $A \downarrow B$ denotes A is entirely below B; $A \lhd B$ denotes A is partially or entirely inside the space of B; and A $\gtrsim$ B denotes the size of the minimal bounding box of A is equal or

greater than B. The notations of outputs are defined as: $\{W\}$ denotes wall segments; $\{R\}$ denotes rooftop segments; $\{G\}$ denotes ground segments; $\{F\}$ denotes fence segments; and $\{D\}$ denotes doorway/balcony segments. The following parameters are also defined in the criteria: θ_v - the threshold angle between the surface normal of the plane and the vertical direction; normal to vertical; θ_h - the threshold angle between the surface normal of the plane normal and the horizontal direction; δ_h - the height threshold; and δ_s - the threshold size.

Algorithm 4.2. Pseudo Code of Rule-based Parsing

```
Geometric Relationship Notation: A ↔ B, A ↑ B, A ↓ B, A ◅ B, A ≳ B;
Inputs: Plane segments {P} ← {P_i};
Outputs: {W} ← { }; {F} ← { }; {R} ← { }; {G} ← { }; {D} ← { };
Parameters: {V} = [0,0,1], θ_v, θ_h, δ_h, δ_s;
for ( {P_i ∈ P} )
   if ( |n_i · V| > θ_v )
      {path 1, see Algorithm 3}
   else if ( θ_v > |n_i · V| > θ_h )
      {path 2, see Algorithm 4}
   else
      {path 3, see Algorithm 5}
end for
```

The rules used in path 1 are presented in Algorithm 4.3. Starting from a vertical plane, I first propagate all the vertical planes that are adjacent to each other for the purpose of propagating vertical planes perpendicular to each other. Then I evaluate if there exist any inclined planes that are adjacent to any propagated vertical plane. If these included planes are above the height threshold, the propagated vertical planes are parsed as wall segments. If the inclined plane cannot be found, the propagated vertical planes are still parsed as wall segments as long as there are one or more planes whose minimal bounding boxes have sizes larger than the size threshold. Otherwise, the propagated vertical planes are parsed as fences. For the minimal bounding box of

a plane segment, I only use rectangle, triangle, parallelogram, and semi-circle as templates. The template that fits the points best will be selected as the minimal bounding box (Figure 4.6(a)). The visual illustration of the Algorithm 4.3 is shown in Figure 4.3.

Algorithm 4.3. Pseudo Code (Wall Segment Parsing)

```
1   if ( |n_i · V| > θ_v )
2     {C} = {P_i};
3     {C} ← {P_j}, where P_j and P_i are adjacent and satisfy {|n_j · V| > θ_v};
4     if ( {∃(P_k|θ_v > |n_k · V| > θ_h)} and {P_k ↔ P_l|P_l ∈ C} and {P_k ↑ δ_h} )
5       {W} ← {C};
6     else if ({∄(P_k|θ_v > |n_k · V| > θ_h)} and {∃(P_l ≳ δ_s} )
7       {W} ← {C};
8     else if ({∄(P_k|θ_v > |n_k · V| > θ_h)} and {∄(P_l ≳ δ_s} )
9       {F} ← {C};
10  end if
```

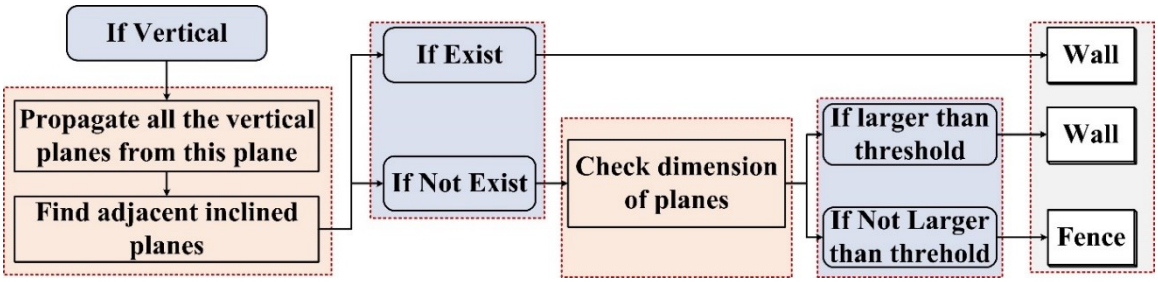


Figure 4.3. Workflow of Algorithm 3

Algorithm 4.4 lists the rules used In roof/ground segment parsing. If a plane segment, P_i, is inclined, I first construct the polyhedron from this inclined plane. To construct the polyhedron, the inclined plane is first projected onto the horizontal plane, and then the projected plane is extruded vertically in both negative and positive directions (Figure 4.6(b)). If there exists a vertical plane that is partially or entirely inside the polyhedron, and P_i is above the height threshold, it is parsed as a rooftop segment. If there exists a vertical plane partially or entirely inside the polyhedron, but P_i is below the height threshold δ_h, it is parsed as a ground segment. In

addition to finding the vertical planes, inclined planes are also checked. If the inclined planes are found that partially or entirely inside the polyhedron, the plane, P_i, is parsed as rooftop if it is above δ_h, otherwise, it is parsed as ground. The algorithmic workflow is presented in Figure 4.4.

Algorithm 4.4. Pseudo Code (Roof/Ground Segment Parsing)

```
1   if (θ_v > |n_i · V| > θ_h )
2     construct polyhedron 𝒫 from plane P_i;
3     for ( {P_j ∈ {P} : |n_j · V| > θ_v} )
4       if ( {∃(x_k | x_k ∈ P_j, x_k ∈ 𝒫)} and {P_i ↑ δ_h} )
5         {R} ← P_i;
6       else if ( {∃(x_k | x_k ∈ P_j, x_k ∈ 𝒫)} and {P_i ↓ δ_h} )
7         {G} ← P_i;
8       end if
9     end for
10    for ( {P_j ∈ {P} : (θ_v > |n_j · V| > θ_h)} )
11      if ( {∃(x_k | x_k ∈ P_j, x_k ∈ 𝒫)} and {P_i ↑ δ_h} )
12        {R} ← P_i;
13      else
14        {G} ← P_i;
15      end if
16    end for
17  end if
```

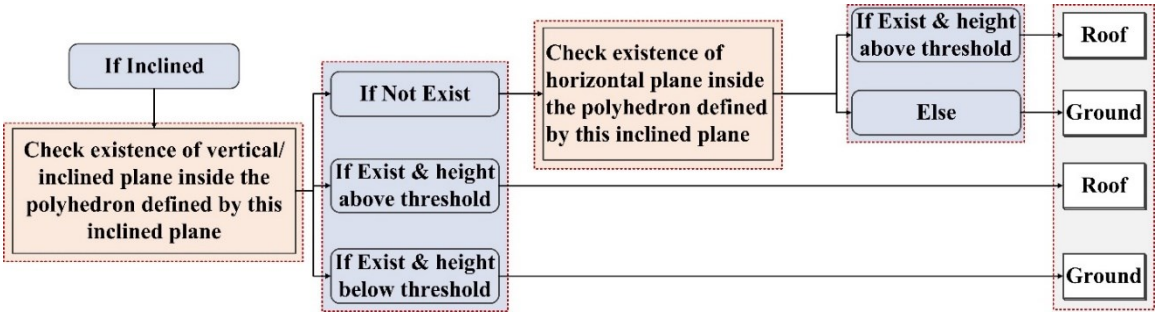


Figure 4.4. Workflow of Algorithm 4

The pseudo code of doorway/balcony segment parsing is presented in Algorithm 4.5 (Figure 4.5), which mainly parses horizontal plane segments into ground or doorway/balcony segments. For each horizontal plane, P_i, the polyhedron is constructed at first use the approach described in phase 2. If there exists vertical or inclined planes inside the polyhedron partially or entirely, the plane P_i is parsed as doorway/balcony segment if it is above the threshold height δ_h. Otherwise, it is parsed as ground.

Algorithm 4.5. Pseudo Code (Doorway/Balcony Segment Parsing)

```
1   if (|n_i · V| < θ_h )
2       construct polyhedron 𝒫 from plane P_i;
3       for ( {P_j ϵ {P} : |n_j · V| > θ_v} )
4           if ( {∃(x_k | x_k ∈ P_j, x_k ∈ 𝒫)} and {P_i ↑ δ_h} )
5               {D} ← {P_i};
6           else
7               {G} ← P_i;
8           end if
9       end for
10      for ({P_j ϵ {P} : (θ_v > |n_j · V| > θ_h)} )
11          if ( {∃(x_k | x_k ∈ P_j, x_k ∈ 𝒫)} and {P_i ↑ δ_h} )
12              {D} ← {P_i};
13          else
14              {G} ← P_i;
15          end if
16      end for
17  end if
```

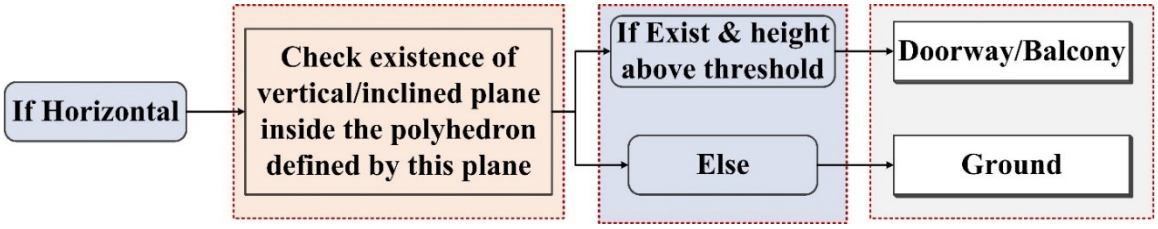


Figure 4.5. Workflow of Algorithm 5

4.3.3 Parsing Optimization

The supervoxel based propagation algorithm sometimes outputs wrong parsing results for some small-size supervoxels. As shown in Figure 4.6 (e), some parts of the walls are parsed as

ground incorrectly. This is because these supervoxels represent the connection of region between a building and the ground, and the computing of surface normal is affected by the low point density and irregular distribution of points.

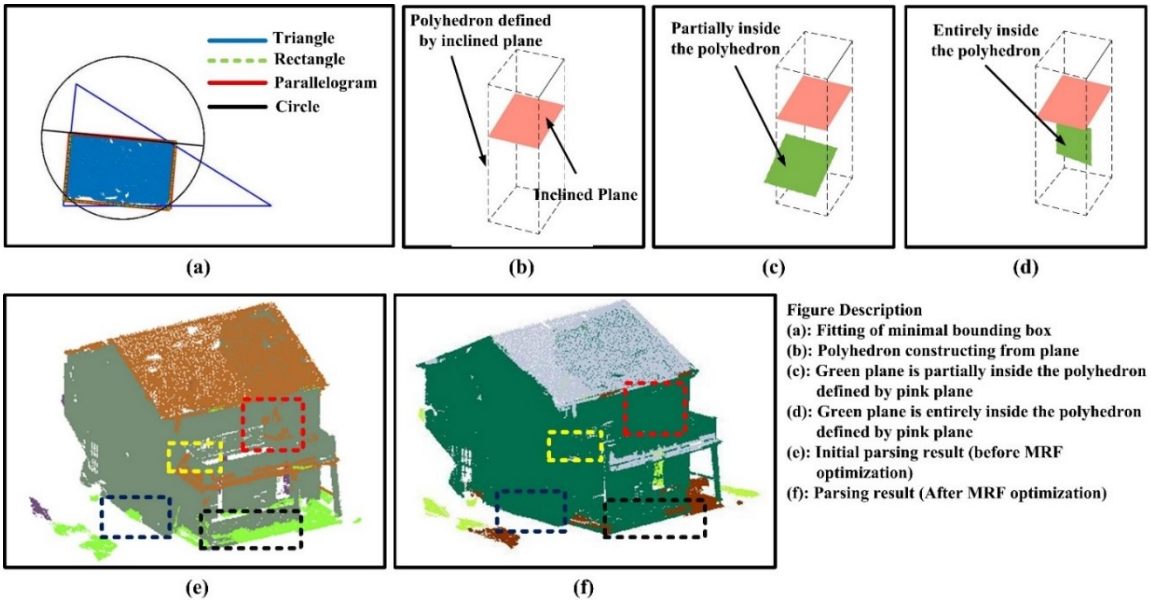


Figure 4.6. Rule-based Parsing

Denoising and smoothing the point cloud plays an important role in performance improvement. Although research about robust normal estimation and noise removal has shown promising results in point cloud segmentation and parsing (Sun et al. 2015, Wolff et al. 2016), these approaches tend to smooth the noise of the point cloud. While in this application, the noise might be the damaged components, and cannot be smoothed or removed. Alternative routine is to improve the parsing result according to local connectivity. MRF has been widely used in applications such as image denoising and segmentation(Lu et al. 2001, Diebel et al. 2005, Cao et al. 2011). The method constructs a MRF over the pixels or points, and denoises the 2D or 3D images by optimizing the energy function over the MRF. In this research, the initial parsing result is considered as noised 3D images denoted as $\hat{\boldsymbol{I}}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$, and the semantic building model with the optimal parsing result is denoted as $\boldsymbol{I}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$.

To retrieve the optimal semantic building model, I construct the MRF over the supervoxels as $\boldsymbol{\mathcal{G}}(\boldsymbol{\mathcal{V}}, \boldsymbol{\mathcal{E}})$, where $\boldsymbol{\mathcal{V}}$ represents the nodes of the MRF, and $\boldsymbol{\mathcal{E}}$ is the weighted edges connecting two adjacent nodes. To denoise $\hat{\boldsymbol{I}}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$, I construct the energy function on the MRF as follow:

$$\boldsymbol{E} = \boldsymbol{\sigma} \sum_{\boldsymbol{i}} \boldsymbol{\Psi_i} + \boldsymbol{\alpha} \sum_{\boldsymbol{i}} \sum_{\boldsymbol{j} \in \boldsymbol{\mathcal{N}(i)}} \boldsymbol{\Phi_{ij}} \tag{4.3}$$

In the equation, $\boldsymbol{\Psi_i}$ is the unary potential defined as:

$$\boldsymbol{\Psi_i} = \left(\frac{|V_i|_0}{|V_i|_0 + \mu}\right) \boldsymbol{\mathcal{I}}(y_i, \hat{y}_i) \tag{4.4}$$

and $\boldsymbol{\Phi_{ij}}$ is the pairwise potential function defined as:

$$\boldsymbol{\Phi_{ij}} = \left(1 - exp\left(-\frac{\mathcal{L}_{ij} \cdot (\theta_{ij})^C}{\beta}\right)\right)\left(\boldsymbol{\mathcal{I}}(\boldsymbol{y_i}, \boldsymbol{y_j})\right) \tag{4.5}$$

In equation (4.4), $|V_i|_0$ is the number of points in supervoxel V_i and μ is a predefined variable. In equation (4.5), $\mathcal{L}_{ij}$ is the length of connection boundary of supervoxel $\{V_i, V_j\}$, $\theta_{ij} = |\boldsymbol{n}_i \cdot \boldsymbol{n}_j|$ is the angle between normals of supervoxel $\{V_i, V_j\}$. $\boldsymbol{\mathcal{I}}(a, b)$ is an indicator function defined as $\boldsymbol{\mathcal{I}}(a, b) = 1$ if $a \neq b$, otherwise $\boldsymbol{\mathcal{I}}(a, b) = 0$. $\boldsymbol{y_i}$ is the label of supervoxel V_i and $\boldsymbol{\alpha}, \beta$ are two constant variables. In the pairwise potential function, consider the parallelism of the adjacent supervoxels is considered, which means that parallel supervoxels tend to have strong interactions with other. Another assumption is that the length of connection affects the pairwise interaction. Specifically, a strong interaction exists if two adjacent supervoxels share a long connection boundary, and a weak interaction exists if they are connected by a short connection boundary. In addition, I also assume the pairwise consistency, which states that adjacent supervoxels tend to have same semantic labeling. The assumptions are illustrated in Figure 4.7.

I implement a simulated annealing algorithm (Kato et al. 2012) to optimize the energy function. For each supervoxel, let the label y_i go through the entire label space and compute the corresponding energy value. If the energy value is accepted, I update the supervoxel with the corresponding label. This process is repeated until no more supervoxels are updated. The

optimization result is shown in Figure 4.6 (f), it is observed that the supervoxels that are originally parsed as ground are modified as wall segments correctly.

The comparison of parsing accuracy is made between the result with and without MRF optimization. And the number of correctly parsed points. The precision and recall in terms of the area of correctly parsed components (wall, roof, handrail, balcony), and the length of correctly parsed component (column) are used as evaluation metrics. The result is shown in Figure 4.8, it is observed that the parsing accuracy with MRF optimization is higher than that of without MRF optimization, in terms of metrics of number of points, area of components, and length of components. This indicates that MRF optimization does improve the parsing accuracy.

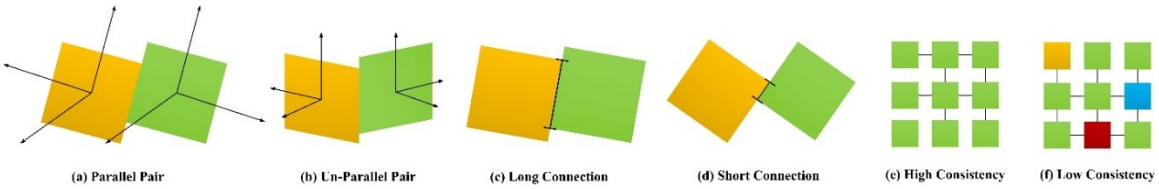


Figure 4.7. Assumption of MRF

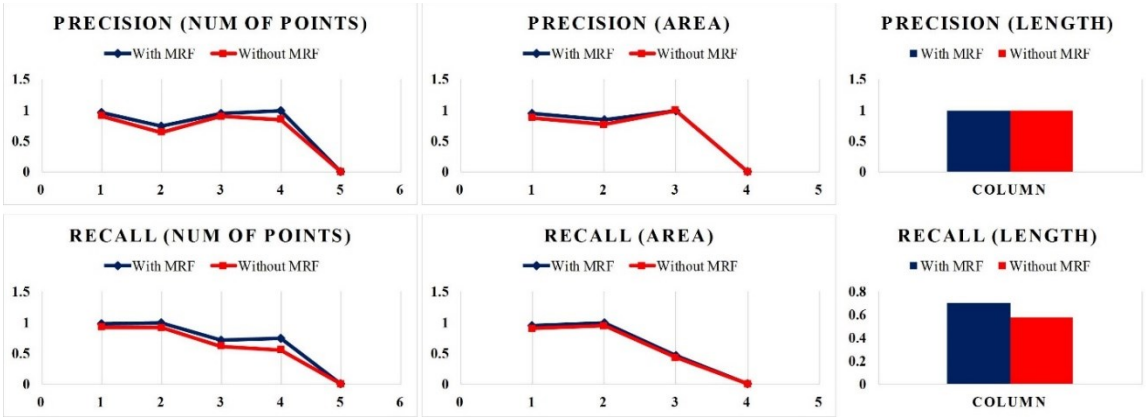


Figure 4.8. Comparison of Parsing Accuracy between with and without MRF

4.3.4 Handrails and Columns Parsing

The previous phases parse the segments into major building components. In this phase, the focus is on smaller building structural elements such as columns, post, and handrails. Detections of these components are beneficial for component-level building damage assessment as damage of columns indicate the damage of primary supporting structures, and handrail damage poses threats to safe inhabitance in the impacted homes.

When compared with wall and rooftop components, handrail structures have more complex structural patterns. If projected to 2-D space, a handrail structure could be identified from the diffraction grating-like pattern. Zolanvari et al. (2016) introduced a slicing based approach in windows extraction from point cloud, which inspires us in using point cloud slices to detect handrail structures.

Given a vertical plane $\mathcal{V}_i$, I first slice point cloud data horizontally, and for each horizontal slice, I slice it vertically. If a vertical slice overlaps with a short handrail column, it contains more points, otherwise, it contains less points. By counting the number of points in each vertical slice, I will be able to obtain a signal-like graph which shows the unique patterns of corresponding slices. Figure 4.9 (d) shows the signal pattern extracted from a handrail segment, and Figure 4.9 (h) shows the pattern of a wall segment. It is observed that the signal extracted from handrail segment shows an explicit periodicity with large amplitudes, while the signal extracted from wall segment shows some random periodicity with small amplitudes. Denote the signal as $\{\mathcal{S}(\ell_i)\}$, I detect the local maxima peaks as $\{\mathcal{S}^{max}(\ell_i)|\mathcal{S}(\ell_i) > \mathcal{S}(\ell_{i-1}), \mathcal{S}(\ell_i) > \mathcal{S}(\ell_{i+1})\}$, and local minima peaks as $\{\mathcal{S}^{min}(\ell_i)|\mathcal{S}(\ell_i) < \mathcal{S}(\ell_{i-1}), \mathcal{S}(\ell_i) < \mathcal{S}(\ell_{i+1})\}$. I compute the amplitude of a periodicity as $\mathcal{A}_i = min\big(\mathcal{S}^{max}(\ell_i) - \mathcal{S}^{min}(\ell_j), \mathcal{S}^{max}(\ell_i) - \mathcal{S}^{min}(\ell_k)\big)\}$, where ℓ_i, ℓ_j, ℓ_k are consecutive peak locations. I extract all the local maxima peaks at locations $\{\ell_i|\mathcal{A}_i > \delta\}$. For consecutive peaks, if $(\ell_i - \ell_{i-1}) < \epsilon$, the segment is recognized as handrails.

Similarly, I use a slicing-based approach (Wu et al. 2017) to detect building columns even if they have large inclination angles.

The point cloud data is first sliced horizontally, and the points are projected to a horizontal plane. Clustering techniques are applied to group the points into multiple segments according to the pair-wise distance. Then the area of each segment is computed. Because each slice is thin in z-direction, the inclination of objects could be neglected. Therefore, the cluster with area smaller than threshold δ_a, is considered as column candidate and represented by its centroid (Figure 4.10 (b)). When the candidates are detected, I construct the adjacency graph on the candidates, where the pair of centroids close to each other are connected. Because columns are linear structures, RANSAC fitting is then used to fit centroids to line models. For a line model, if its direction is close to vertical, the line model is considered as a column candidate. Then I first compute the length of the fitted line segment, and check whether the inlier points of this line model are adjacent to each other. If both are satisfied, the candidate is confirmed as a column object. Figure 4.10 (c) shows the bounding boxes of four detected columns. Although the columns are inclined due to disaster impact, this approach is still able to locate them.

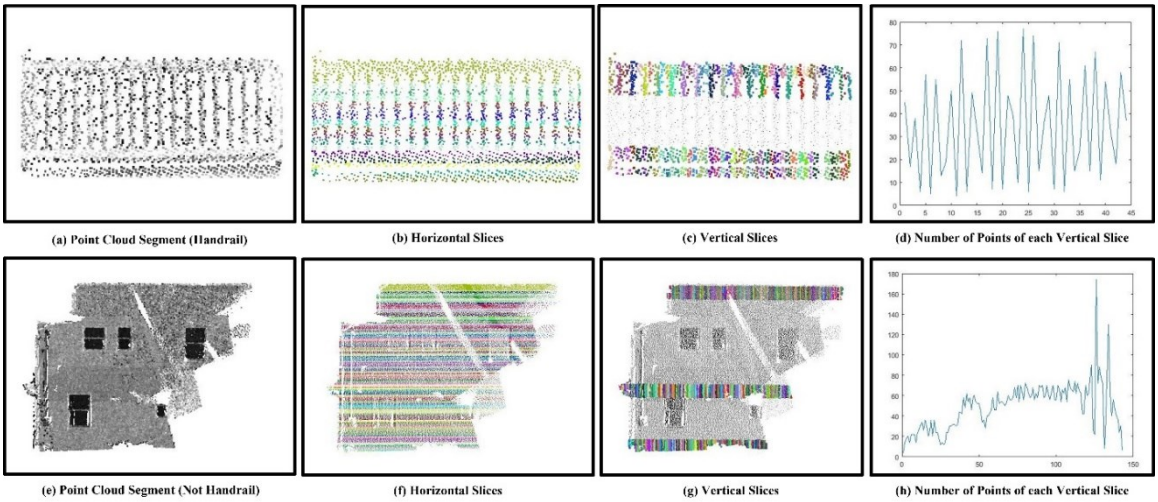


Figure Description: (b & f) Horizontal slices of the point cloud. Each slice is represented in different color. (c & g) Vertical slices of each horizontal slice. This figure shows the vertical slices of the topmost horizontal slice and the bottom-most horizontal slice. (d & h) Number of points of each vertical slice. For each horizontal slice, we count the number of points of each vertical slice, and generate one of this plot.

Figure 4.9. Handrail Extraction

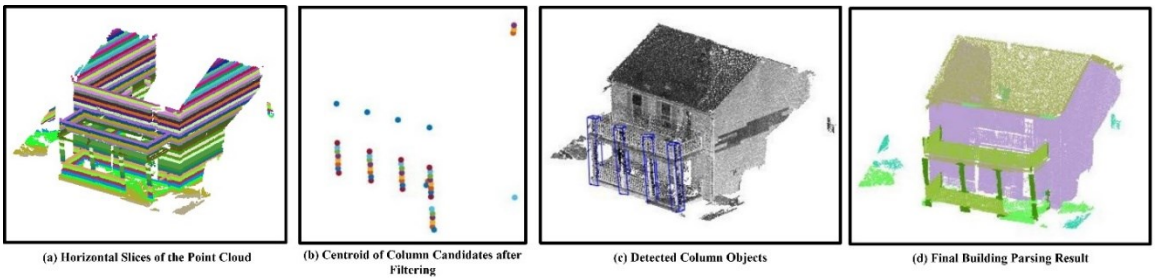


Figure Description: (a) Horizontal slices of the point cloud. Each slice at various elevation is represented is different color. (b) Each point in this figure represents one column candidate. These candidates are extracted according to the area of each cluster of a horizontal slice. (c) Bounding boxes of the detected columns. These columns are extracted via fitting line segment on the column candidates shown in (b). And a line segment is considered as valid if it is close to vertical and its length is longer than a threshold. (d) Final building parsing result. Components with different semantic meaning is represented in different color.

Figure 4.10. Column Extraction

4.3.5 Semantic Damage Detection

Within all the components have been parsed semantically, the final phase of the proposed approach is to characterize structural damage from each component. I start with wall damage detection as it is relatively more straightforward than others. I denote the wall, roof, column, and handrail segments as $\{P^{\mathcal{W}}, P^{\mathcal{R}}, P^{\mathcal{C}}, P^{\mathcal{H}}\}$ respectively. For a wall segment $\{P_i^{\mathcal{W}} \in P^{\mathcal{W}}\}$, if it is not plumb within a tolerance value, I deem it as damaged wall segments. If it is plumb, I find its adjacent wall planes $\{P_j^{\mathcal{W}} \in P^{\mathcal{W}}\}$, and compute the intersect line segments l_{ij}. If the intersect line is not plumb with a tolerance value, the pair of walls $\{P_i^{\mathcal{W}}, P_j^{\mathcal{W}}\}$ are determined as damaged. Otherwise, $P_i^{\mathcal{W}}$ is determined as intact. The pseudo code of wall damage detection is presented in Algorithm 4.6. The detection is based on the assumption that most of the residential building walls are rectangular and vertically built. Therefore, if detected out of plumb, the wall is very likely damaged. In addition, the check of intersect line direction is based on the observation that walls are rectangular in most cases. If the line direction is not plumb, it often indicates large deformation of wall segments.

Algorithm 4.6. Pseudo Code for Wall Damage Detection

```
Parameters: θ;
Outputs: damaged components {C^D} = { }, intact components {C^I} = { };
for ({P_i^W ∈ P^W}, compute its normal n_i )
  if ( |n_i · V| < π/2 − θ )
    {C^D} ← P_i^W;
  else
    find {P_j^W | P_j^W ↔ P_i^W}, and compute the intersect line l_ij = {x, n};
    If ( |n · V| < θ)
      {C^D} ← {P_i^W, P_j^W};
    else
      {C^I} = {P_i^W, P_j^W};
  end if
end for
```

Residential building roofs have multiple styles, including gable, hip and valley, mansard style, saltbox style, and so on (Figure 4.11). It can be observed that most of the common roof styles have the following features in common: 1) roofs intersect horizontally with at least one wall segment; and 2) if two roofs are rectangular or parallelogram, they intersect horizontally. Based on these observations, roof damage in this project is characterized in the following way: (1) for a roof segment $\{P_i^{\mathcal{R}} \in P^{\mathcal{R}}\}$, if there is an adjacent non-damaged rectangular wall, and the intersect line is horizontal, the roof segment is determined as intact; (2) If the intersect line is not horizontal, but there is an adjacent roof segment horizontally intersect with it, the roof segment is considered as intact as it is likely to be a gable roof style; (3) If a roof segment cannot fit into any of the previous two scenarios, the roof is determined as intact as long as it is adjacent to an intact wall segment and they are perpendicularly intersected; and (4)) If a roof segment cannot fit into any of the previous three scenarios, a roof segment is considered as damaged due to inclination. The pseudo code for the above logic is shown in Algorithm 4.7.

Algorithm 4.7. Pseudo Code for Roof Damage Detection

```
Parameters: θ;
Outputs: damaged components {C^D} = { }, intact components {C^I} = { };
for ({P_i^R ∈ P^R} )
  if ( ∃(P_j^W ↔ P_i^R), compute the intersect line l_ij = {x, n} )
    if ( |n · V| > π/2 − θ )
      {C^I} ← {P_i^R};
    else if (|n_i · n_j| > θ )
      {C^I} ← {P_i^R};
  else if ( ∃(P_j^R ↔ P_i^R), compute the intersect line l_ij = {x, n} )
    if (|n · V| > π/2 − θ )
      {C^I} ← {P_i^R};
  else
    {C^D} ← {P_i^R};
  end if
end for
```

Due to occlusion, noise, and variation in point cloud density, I only focus on the detection of roof structural damage including roof inclination and roof collapsing. Figure 4.11 shows a summary of definitions of wall damage and roof damage patterns. The definition is based on the assumption that the buildings are built in regular types, which indicates the walls are vertical if intact, and roofs are designed according to the general building roof styles.

Damage detection of other structural components, including columns, handrails and canopies are relatively straightforward. Detection of column damage is based on whether the primary direction of a column is plumb to a tolerance value. Detection of handrail damage is also based on its plumb, and detection of balcony damage is based on whether it is horizontal.

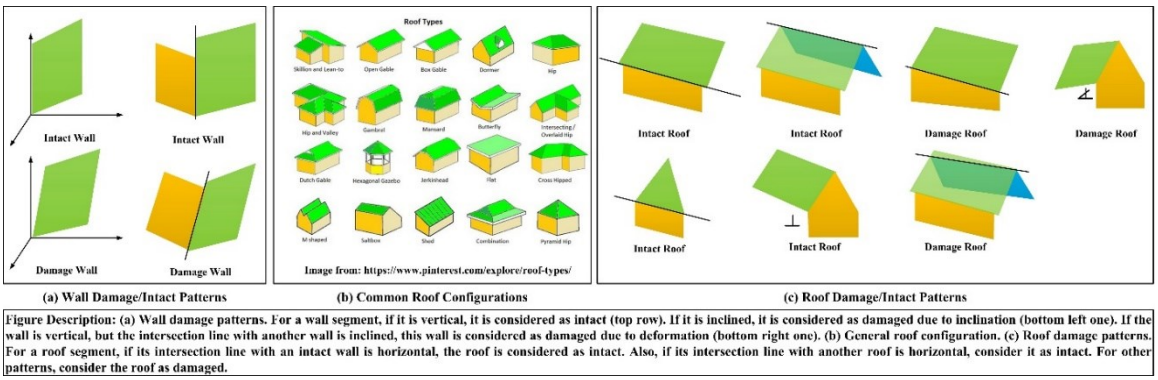


Figure 4.11. Roof and Wall Damage Patterns

4.4 Results and Discussion

The proposed approach was tested on a mobile lidar data set that was collected in severely impacted areas during Hurricane Sandy along the New Jersey coastal line. During the survey, a mobile LiDAR system integrated with a Leica ALS70 laser scanner, a navigation system, and four digital cameras, was employed to collect the data. The data set includes point cloud data of ten residential buildings spanning various damage categories including totally damaged, partially damaged, and intact buildings (Figure 4.14). The data set was chosen for validating the capability of the approach in parsing building components and identifying building damage in various scenarios.

The parameters I used in the experiment are listed in Table 4.2. The affects of different values of MRF parameters on parsing accuracy is evaluated. For each parameter, three different values are evaluated, and other parameters are fixed during the evaluation. Figure 4.12 shows the affects of different α value and σ value in parsing accuracy. As can been observed, the parsing accuracy of different α values and σ value does not change much, and the same trend is observed for other parameters. Therefore, the parameters in Table 4.2 are chosen. The parameters are selected via statistical analysis of hundreds of impacted buildings during Hurricane Sandy. Of particular note is that the angle thresholds for rule-based parsing phase are set to be relatively large values to parse not only intact buildings but also damaged buildings. Similarly, the height threshold for

parsing roof objects is set to be 0.5 meter to detect not only intact roofs but also damaged roofs. Although the top of an intact roof is obviously much higher than 0.5 meter, this setting facilitates the detection of damaged roofs even in situations they have collapsed to the ground. I implement the entire proposed framework in MATLAB (The MathWorks 2016) on Core i5-4570 CPU(3.20GHz) machine with 8.00GB RAM. Table 4.1 lists the running time of four buildings as well as the number of points of each building. It can be observed that with the increase of number of points, the running time of phase 1, phase 2, and phase 3 increases linearly. The trend of running time of phase 5 is not linearly proportional to the number of points, this is because some building does not contain handrails, columns, or balcony objects..

Table 4.1. Running time

Building ID	Num of Pts	Phase 1(sec)	Phase 2(sec)	Phase 3(sec)	Phase 4(sec)	Phase 5(sec)
a	263,235	76.42	12.74	99.62	283.61	472.39
b	738,906	169.41	69.73	255.69	371.26	613.48
c	465,111	1143.14	23.80	179.64	-	466.54
g	336,265	854.34	21.32	183.51	14.40	381.35

The test data were manually annotated to create ground truth (GT) for performance evaluation. As commonly used for accuracy evaluation, for a certain category of object, I 1) denote the true positive prediction (TP) as correctly recognized as a certain object, 2) denote false positive (FP) as that recognized as correct object, while the true condition is other, 3) denote false negative (FN) as that recognized as other object, while the true condition is the interested object, and 4) denote true negative (TN) as that recognized as other object, and the true condition is other object. Table 4.3 lists the accuracy of building component using the proposed approach the accuracy is evaluated by TP, FP, FN, and TN metrics.

Table 4.2. Parameters used in the Parsing and Damage Assessment Pipeline

Rule-based Parsing				
θ_v	θ_h	δ_s	δ_h	
80°	15°	2.5(m)	0.5(m)	
MRF Optimization				
α	β	μ	σ	C
1.0	1.0	1.0	0.3	5.0
Handrail Extraction		**Column Extraction**	**Damage Detection**	
δ	ϵ	δ_a	θ	
30	0.5(m)	$0.1m^2$	2°	

The results suggested that the proposed approach performs well in parsing wall and rooftop segments, regardless the conditions of building objects. The framework correctly detected all the visible wall segments for seven out of ten test buildings. The error in wall parsing occurs at building (d) and (h), where the handrails are misclassified as wall segment. This is because when the handrails are inclined towards the horizontal direction (Figure 4.15), the peak detection method fails to detect enough valid peaks from each horizontal slice. Overall, the approach is capable of recognizing and segmenting approximately 96% of the wall points. The proposed approach has similar performance in rooftop detection. For the ten buildings, the approach has a detection accuracy of 86%. The errors occur at building (d) and (i) because the damaged handrails at building (d) are misclassified as roofs and the wall protrusion at building (i) is misclassified as rooftop. In the former situation, the handrail is severely damaged with large inclination angle as 17.05°, which is larger than the angle threshold I defined. In the latter situation, the calculation of the normal of the wall protrusion is severely affected by point outliers. As a consequence, it is misinterpreted as roof objects. In addition, I use precision and recall to

evaluate the performance of the proposed approach in building semantic parsing. The precision is defined as $\text{Precision} = \text{TP}/(\text{TP} + \text{FP})$, and recall is defined as $\text{Recall} = \text{TP}/(\text{TP} + \text{FN})$. The calculation of precision and recall are shown in Figure 4.13. The same conclusion could be drawn from the precision and recall calculation. It can be observed that the accuracy of wall and roof parsing is comparatively higher, regardless the building size, configuration , and damage condition. The average precision for wall parsing in terms of number of points and area of component are 95.6% and 96.6%, respectively, and the average recall for wall parsing in terms of number of points and area of component are 92.3% and 86.2%, respectively. In terms of roof parsing, the average precision are 80.3% and 86.2%, respectively, and the average recall are 85.7% and 93.6%. For handrail parsing, the average of precision and recall drop to 70% and 50% approximately.

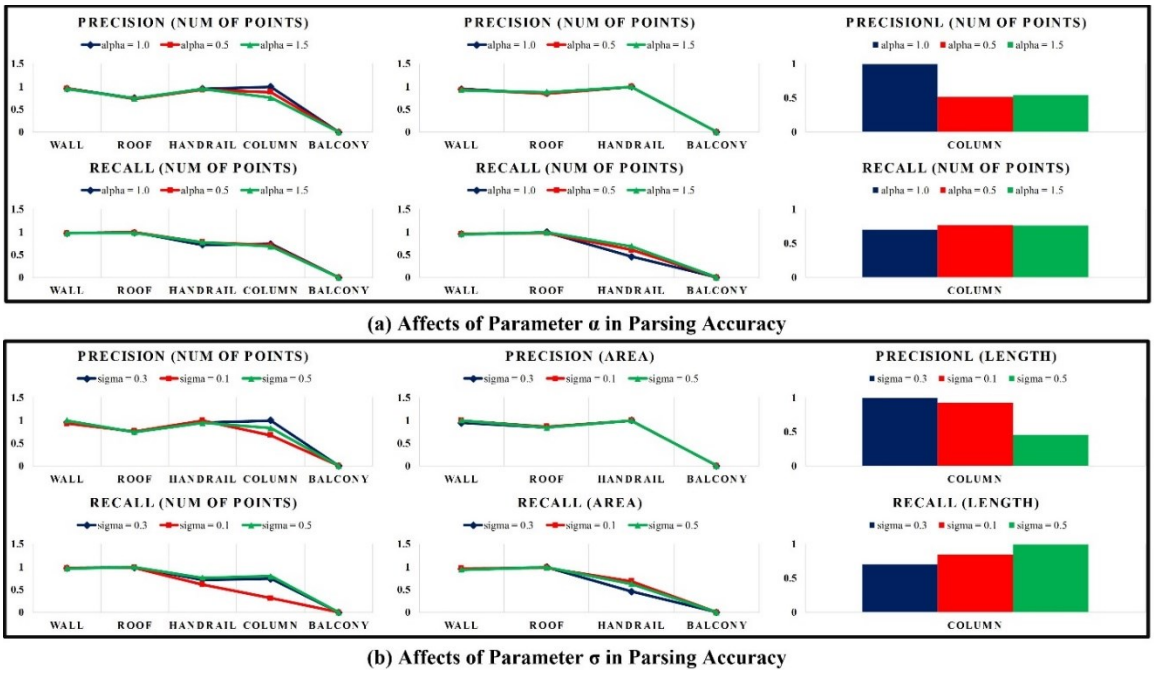


Figure 4.12. Evaluation of MRF Parameters ($\boldsymbol{\alpha}, \boldsymbol{\sigma}$)

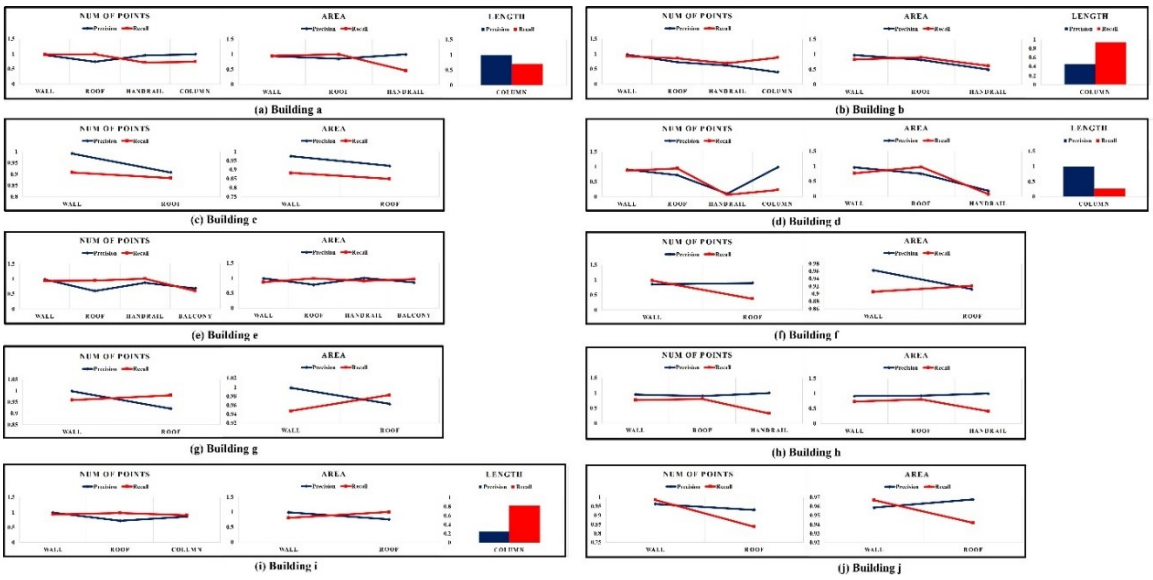


Figure 4.13. Precision and Recall Accuracy of Building Semantic Parsing

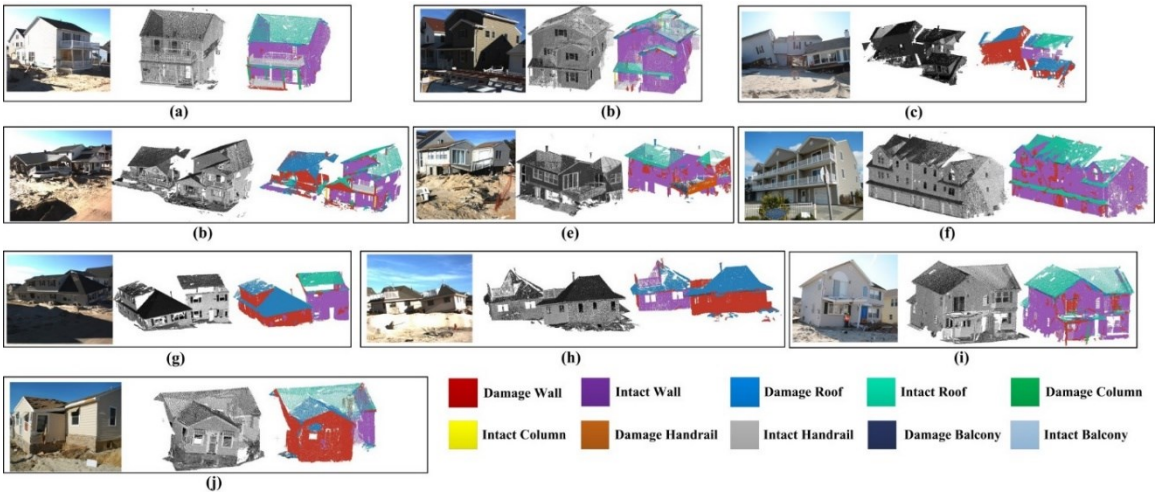


Figure 4.14. Damage Detection Results

The parsing of handrails and columns is a much challenging task. The following failure modes have been observed. If a handrail is severely inclined, it might be parsed as rooftops or ground objects. If it is vertical but deformed around the normal direction (Figure 4.15), the handrail extraction algorithm will fail to detect enough valid peaks. For the parsing failures involving column objects, one typical failure mode is caused by the lack of enough data points

(Figure 4.15). For a column to be detected, it needs to meet the following two criterions: (1) The column candidates (mentioned in section 3.4) that fit line model are adjacent to each other to ensure that the detected column object is a valid object; and (2) The column length is longer than a threshold value to ensure that the detect object is a valid column object. If a fitted line segment is split into multiple segments due to lack of points, and each segment is shorter than threshold, the line segment will be rejected. Because the lack of adequate number of points is the major reason of column detection failure, this happens when a building is partially occluded by debris, or the column is collapsed. The severe inclination of column objects also causes the parsing failure. This is typical when a building is severely damaged or collapsed.

I use the similar metric to evaluate the performance of the proposed approach in detecting damaged components. I first manually label out the damaged components as the ground truth. Because the approach assesses building damage at the component level, I label one building component as damaged if it is either inclined or deformed, regardless the severity of the damage. In addition, because the parsing is based on the propagation of supervoxels, the supervoxels at corners, eaves, and windowsills cannot be propagated as either wall or roof due to the lack of planarity. As a result, the proposed approach will detect many damages at these supervoxels. To address these detection errors, I omit the supervoxels that have points less than 500 if they are detected as damage.

In order to evaluate whether this system is able to detect structural damage correctly, I denote the actual damage as ground truth (GT). To evaluate the accuracy of damage detection, for a certain component, 1) denote true positive (TP) as it is recognized as damage, and the true condition is also damage, 2) denote false positive (FP) as it is recognized as damage, while the true condition is intact, 3) denote false negative (FN) as it is recognized as intact, while the true condition is damage, and 4) denote true negative (TN) as it is recognized as intact and the true condition is intact.

Table 4.4. It is observed that for some buildings such as (a), (b), (e) and (f), the FP in terms of wall points are much higher than that of TP. This is because there is no actual wall damage at these buildings, while the proposed approach finds small supervoxels, such as window, windowsills, eaves, and intersecting boundary of two planes, as damage. The frequent errors in damage detection are shown in Figure 4.16. For building (c), (d), (g), (h), and (j), the precision (defined as $\text{Precision} = \text{TP}/(\text{TP} + \text{FP})$) of wall damage detection is approximately 91.1% on average in terms of the number of points, and 84.2% on average in terms of area of components. In terms of recall (defined as $\text{Recall} = \text{TP}/(\text{TP} + \text{FN})$), the accuracy is 95.4% on average in terms of number of points, and 98.3% on average in terms of area of components. For these buildings, the damage detection accuracy in terms of rooftops is also very high. Specifically, the precision are 95.2% and 91.9% with respect to number of points and component area, respectively. And the recall are 99.6% and 99.5%, respectively. For building (e) and (h), the proposed approach detects 100% handrail damages in terms of number of points and component area, evaluated by precision and recall, respectively. For building (a), (d), and (i), the precision of column damage detection is 83.5% and 83.8% on average in terms of point number ad component length. And the recall accuracy is 100% for both metrics. These results indicate that the proposed approach performs well when the building has structural damage, while it tends to overestimate the structural damage if the building has little or no visible structural damage.

There are several critical threshold values used in this proposed method for building component parsing and damage detection (Table 4.2). An important question is the impact of these threshold values to the performance of the method. To answer this question, a sensitivity analysis is conducted. To reduce the computational intensity without the loss of generality, building (a) in Figure 4.14 is used as an example. Since the proposed method has eight parameters, the combination of parameter settings increases exponentially if these parameters are changed simultaneously. Therefore, for the sensitivity analysis of each parameter, I fix the rest

parameters as default to reduce the computation. The sensitivity analysis results are shown in Figure 4.17. It can be noted that the performance of parsing and damage detection for wall and roof components are insensitive to the choice of parameters. However, to ensure good performance in parsing and damage detection for column, handrails, and balconies, careful choices of parameters are of need. This is largely due to the size of these objects and the relative sparse point measurement on these objects.

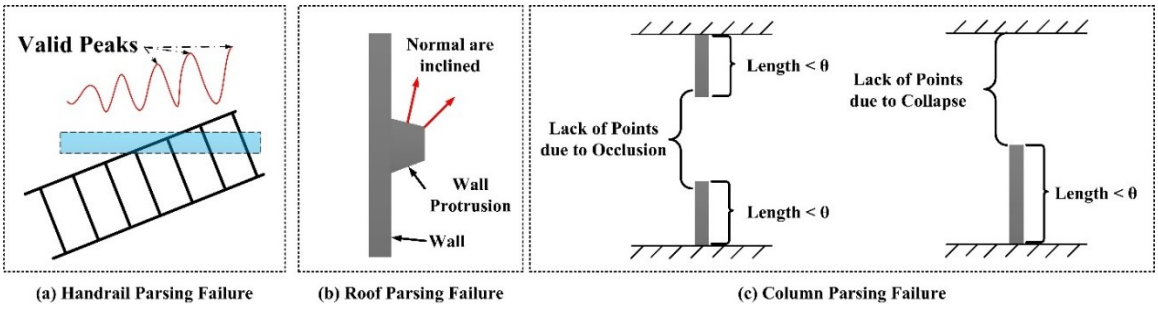
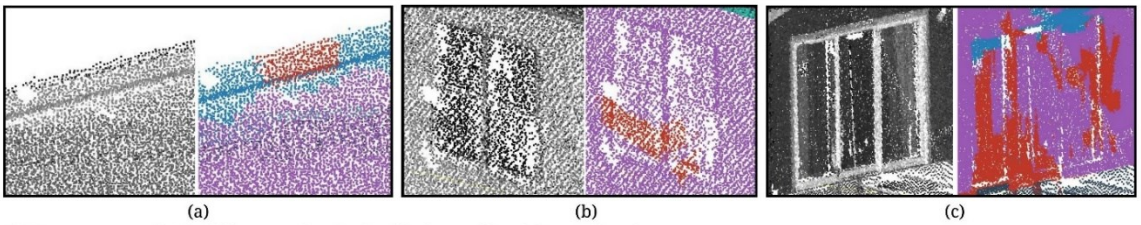


Figure 4.15. The Typical Failure Modes in Recognizing and Segmenting Handrail and Column Objects



(a) One supervoxel at building eave is mis-classified as wall and detected as damage;
(b) One supervoxel at building window is mis-classified as wall and detected as damage;
(c) Supervoxels at glass door is mis-classified as wall and detected as damage

Figure 4.16. The Typical Errors in Damage Detection

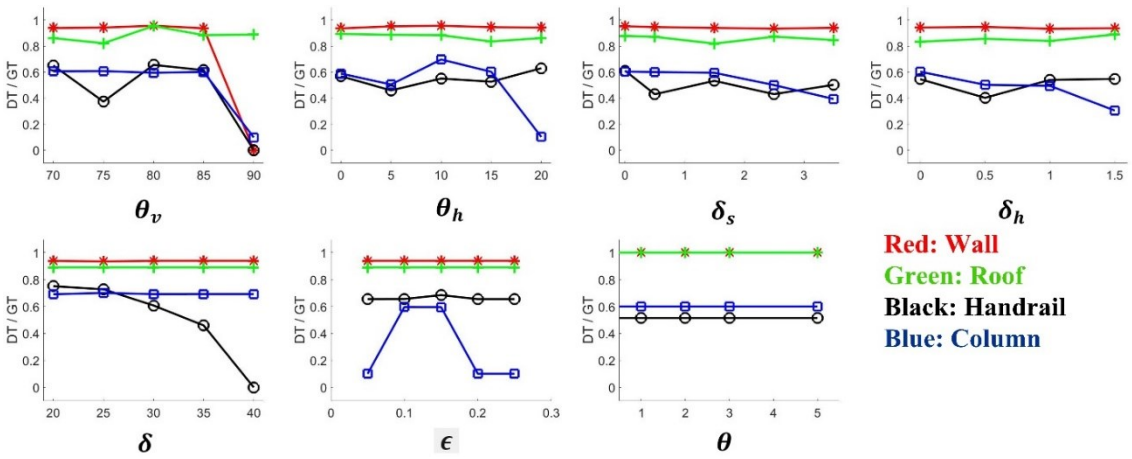


Figure 4.17. Sensitivity Analysis

4.5 Conclusion

In this research, an approach for fully automated building parsing and component-level damage assessment is proposed. The approach first parses a building into components with semantic meaning followed by assessment of damages at the building component level. Applications of this approach on a post-hurricane building damage data set show that the proposed approach performs well in both the parsing and detection phase. The proposed approach opens doors to automated detailed damage assessment of residential structures following major hurricane events. I envision that the proposed approach can be integrated into existing post-disaster assessment systems for quicker damage assessment and gaining more comprehensive understanding of the extent of hurricane damages in a shorter amount of time.

For future research, one major constraint of the proposed approach is the error of semantic parsing due to defects and complex topological relations of the data. Another constraint is due to the complex damage pattern. To address this, imagery data is proposed to be integrated to assist the robust and comprehensive automated semantic damage detection.

Table 4.3. A Summary of Building Parsing Results

		Number of Points					Area (m^2)				Height (m)
		Wall	Roof	Handrail	Column	Balcony	Wall	Roof	Handrail	Balcony	Column
a	TP	151883	20664	22987	6247	0	134.9	42.6	10.0	0.0	7.5
	FP	5839	6986	1188	35	1500	8.3	7.7	0.0	2.8	0.0
	FN	3955	215	9132	2246	0	6.4	0.2	11.8	0.0	3.1
	TN	55652	189464	184022	208801	215829	59.3	155.9	183.8	200.0	54.7
b	TP	538182	70022	12870	1348	0	207.0	107.9	5.4	0.0	2.8
	FP	12078	26810	7808	2070	8390	5.6	25.4	5.6	10.8	3.4
	FN	39150	11962	5848	196	0	44.6	11.7	3.4	0.0	0.2
	TN	90168	570784	653052	675964	671188	119.1	224.4	341.4	342.3	133.6
c	TP	244334	99476	0	0	0	124.9	121.0	0.0	0.0	0.0
	FP	2244	10156	0	0	0	3.2	10.5	0.0	0.0	0.0
	FN	25043	13176	0	0	0	16.9	21.5	0.0	0.0	0.0
	TN	110408	259221	382029	382029	382029	140.1	130.9	274.4	274.4	114.0
d	TP	221896	103653	1440	5624	0	197.2	219.0	1.2	0.0	5.0
	FP	23561	40479	12623	132	3027	8.3	73.0	5.1	3.9	0.0

	FN	33489	7009	20135	19189	0	60.2	4.8	16.6	0.0	14.3
	TN	133489	261294	378237	387490	409408	246.1	212.2	458.2	477.9	184.6
e	TP	262205	21756	22436	0	6305	106.4	45.8	6.8	17.5	0.0
	FP	6827	15231	3350	0	3026	1.7	13.3	0.0	3.0	0.0
	FN	19557	1419	72	3044	4342	17.3	0.4	0.8	0.6	2.5
	TN	52547	302730	315278	338092	327463	71.3	133.8	173.9	161.1	67.2
f	TP	2134358	220226	0	0	0	457.9	241.0	0.0	0.0	0.0
	FP	378301	28219	0	0	16420	17.5	23.1	0.0	13.0	0.0
	FN	42952	379988	0	0	0	47.7	20.5	0.0	0.0	0.0
	TN	221913	2149091	2777524	2777524	2761104	241.5	471.0	729.5	716.0	248.8
g	TP	340113	158879	0	0	0	166.4	155.2	0.0	0.0	0.0
	FP	723	13760	0	0	3306	0.3	6.0	0.0	5.0	0.0
	FN	14677	3112	0	0	0	9.3	2.8	0.0	0.0	0.0
	TN	161268	341030	516781	516781	513475	157.0	168.9	327.7	323.3	118.7
h	TP	130825	66914	3013	0	0	85.2	78.2	4.6	0.0	0.0
	FP	6422	7543	6	0	48106	8.1	6.6	0.0	44.2	0.0
	FN	39336	16213	6528	0	0	32.8	19.7	6.7	0.0	0.0
	TN	86246	172159	253282	262829	214723	101.2	119.4	207.4	178.2	122.8

i	TP	183876	38403	0	4747	0	120.4	84.7	0.0	0.0	3.9
	FP	1990	15561	894	841	2832	0.9	28.4	0.7	2.0	12.4
	FN	14933	826	0	519	5840	30.2	0.5	0.0	7.3	0.9
	TN	48345	194354	248250	243037	240472	93.4	126.4	222.7	212.8	72.1
j	TP	135617	27900	0	0	0	95.4	66.8	0.0	0.0	0.0
	FP	5416	2142	0	0	16	4.1	2.2	0.0	0.0	0.0
	FN	2158	5416	0	0	0	3.3	4.1	0.0	0.0	0.0
	TN	27900	135633	171091	171091	171075	66.8	95.4	165.8	165.8	67.1

Table 4.4. A Summary of Damage Detection Results

		Number of Points					Area (m^2)				Height (m)
		Wall	Roof	Handrail	Column	Balcony	Wall	Roof	Handrail	Balcony	Column
a	TP	0	0	0	6247	0	0.00	0.00	0.00	0.00	7.52
	FP	2093	743	0	0	0	9.23	2.76	0.00	0.00	0.00
	FN	0	0	6408	0	0	0.00	0.00	0.00	0.00	0.00
	TN	149790	19921	16579	0	0	134.40	42.42	10.06	0.00	0.00
b	TP	0	0	0	0	0	0.00	0.00	0.00	0.00	0.00
	FP	54084	21274	1090	0	0	30.36	16.81	2.36	0.00	0.00
	FN	0	0	0	0	0	0.00	0.00	0.00	0.00	0.00
	TN	482422	50842	11780	1348	0	193.71	105.31	4.24	0.00	1.38
c	TP	134887	49476	0	0	0	90.91	75.83	0.00	0.00	0.00
	FP	2867	3115	0	0	0	1.25	4.50	0.00	0.00	0.00
	FN	38398	0	0	0	0	4.85	0.00	0.00	0.00	0.00
	TN	68182	46885	0	0	0	26.20	42.46	0.00	0.00	0.00
d	TP	62813	45870	0	1267	0	63.46	64.47	0.00	0.00	1.03
	FP	34079	5912	0	0	0	41.82	17.96	0.00	0.00	0.00

	FN	500	462	0	0	0	1.74	0.42	0.00	0.00	0.00
	TN	124307	51409	10933	4357	0	114.49	148.13	6.32	0.00	3.95
e	TP	910	0	22436	0	0	0.75	0.00	6.80	0.00	0.00
	FP	37922	984	0	0	0	41.65	4.39	0.00	0.00	0.00
	FN	360	0	0	0	0	0.00	0.00	0.00	0.00	0.00
	TN	223027	20772	0	0	0	99.22	45.44	0.00	0.00	0.00
f	TP	0	0	0	0	0	0.00	0.00	0.00	0.00	0.00
	FP	1331458	21054	0	0	0	132.37	10.40	0.00	0.00	0.00
	FN	0	0	0	0	0	0.00	0.00	0.00	0.00	0.00
	TN	802900	199172	0	0	0	428.31	235.01	0.00	0.00	0.00
g	TP	178120	136512	0	0	0	89.74	126.32	0.00	0.00	0.00
	FP	2507	557	0	0	0	7.86	0.92	0.00	0.00	0.00
	FN	19	640	0	0	0	0.41	2.07	0.00	0.00	0.00
	TN	202592	21170	0	0	0	101.27	31.20	0.00	0.00	0.00
h	TP	130533	66914	3013	0	0	84.81	78.29	4.64	0.00	0.00
	FP	0	0	0	0	0	0.00	0.00	0.00	0.00	0.00
	FN	292	0	0	0	0	0.41	0.00	0.00	0.00	0.00
	TN	0	0	0	0	0	0.00	0.00	0.00	0.00	0.00

i	TP	0	0	0	2590	0	0.00	0.00	0.00	0.00	2.33
	FP	15183	979	0	2527	0	15.56	2.08	0.00	0.00	2.21
	FN	0	0	0	0	0	0.00	0.00	0.00	0.00	0.00
	TN	175425	37424	0	0	0	120.85	84.46	0.00	0.00	0.00
j	TP	112363	14432	0	0	0	52.21	26.16	0.00	0.00	0.00
	FP	6567	923	0	0	0	22.41	3.63	0.00	0.00	0.00
	FN	6	60	0	0	0	0.00	0.00	0.00	0.00	0.00
	TN	31385	12485	0	0	0	34.57	39.28	0.00	0.00	0.00

Chapter 5 : Image-Based 3d Reconstruction For Post-Hurricane Residential Building Damage Assessment

Street-level storm damage photos are an essential type of data used in post-disaster damage assessment. However, the existing damage assessment approaches only leverage such storm damage photos in a two-dimensional context. With the recent rapid development in image-based 3D reconstruction, it is natural to question whether storm damage photos can be leveraged to create 3D virtual disaster sites for remote field work, and more importantly whether the 3D virtual disaster sites can assist damage assessors to perform accurate storm damage assessment. This research leveraged the storm damage photos collected by a foot-on-ground damage assessment team during Hurricane Sandy to explore the feasibility of using image-based 3D reconstruction for post-hurricane residential building damage assessment. Specifically, two commonly used SFM-based 3D image reconstruction pipelines are employed to reconstruct several impacted residential buildings from photos to evaluate their performances regarding key measurement needs in post-hurricane damage assessment. Damage data recorded by a mobile LiDAR system were used as the ground truth for performance evaluation. The study results suggest: (1) image-based 3D reconstruction can be used to reconstruct accurate 3D virtual disaster sites for individual buildings; (2) the reconstructed models in the form of point clouds can support damage assessors to conduct accurate building component-level damage assessment; and (3) image-based 3D reconstruction from street-level photos is limited in terms of supporting damage assessors to identify and measure geographic factors that have contributed to the failure of coastal structures.

5.1 Introduction

Recently, humankind has suffered tremendous loss as a result of several massive hurricane events (Lin et al. 2012). To develop better hurricane damage forecast models and understand the effectiveness of existing building construction practices and mitigation measures, rapid responses

are often launched after major disasters to collect empirical damage data (Friedland 2009, Massarra 2012). Rapid assessment of hurricane damage has traditionally been conducted through field reconnaissance deployments where damage information is captured and cataloged. The most commonly used field data collection methods are paper and pen, electronic, and video recording(Chiu et al. 1999, Crandell et al. 2005, Curtis et al. 2012). These methods are typically labor intensive and are difficult to be scaled up to cover large disaster areas without incurring significant cost. In addition, rapid assessment leaves little time for assessors to gather detailed field measurements. Therefore, there is inevitably a loss of valuable information for long-term research that could otherwise lead to improved understanding of severe wind and flood effects on the built environment. Virtual-reality based visualization is likely to play an increasingly important role in the rapid scientific response to future natural disasters because it permits large numbers of researchers to examine a site remotely and quickly, i.e., perform virtual field work (Cowgill et al. 2012), without incurring significant cost and diverting resources away from humanitarian efforts.

Remote sensing methods have been increasingly used for damage assessment purposes. Most of these approaches rely on data remotely sensed from air- or ground-based platforms. The types of remote sensing data that are useful for post-disaster damage assessment include photo imagery and LiDAR data. These data are primary spatial data sources used in many post-disaster studies (Masuoka et al. 2004, Ozisik et al. 2004, Li et al. 2008, Eguchi et al. 2010, Kashani et al. 2014, Kashani et al. 2014). Infrared, hyperspectral, and radar imagery may also be useful since they complement characteristics of images captured in other portions of the spectrum (Jha et al. 2008, Klemas 2009). It is possible to create virtual hurricane impacted sites with these remote sensed data. However, due to the data resolution limitation, the virtual sites created using these data can only support large-scale topographic investigation.

There are primarily two ways to address this limitation: (1) leveraging high resolution remote sensing instrument; and (2) leveraging algorithmic improvement to produce high resolution

spatial data. On the first path, static terrestrial LiDAR and mobile LiDAR are representative technologies that have been recently deployed for post-disaster assessment (Graettinger et al. 2012, Olsen et al. 2012, Yim et al. 2014). The high resolution LiDAR data captured in these studies have apparent potential for developing virtual disaster sites. However, these studies share a common limitation – they rely on ground-based platforms which limit their data collection window to a stage where ground access must be available. To address this limitation, unmanned airborne LiDAR is poised to be a breakthrough approach in the next few years (Lin et al. 2011). Ground-based LiDAR data collection can provide a higher resolution compared with the air-based data collection. However, air-based data collection method is expected to provide an effective routine for relatively larger scale mapping (Sturzenegger et al. 2007). On the second path, many studies have resorted to Structure From Motion (SFM)-based dense reconstruction methods, a computer vision technique reconstructing 3D high resolution point clouds based on a set of photographs taken from different angles(Cheng et al. 2011, Simões et al. 2012, Torok et al. 2013). Compared with LiDAR-based method which required ground access strictly, image-based 3D reconstruction methods do not suffer from this limitation that much. Images can be obtained using a digital camera as long as the safety of investigators is ensured.

SFM-based methods have apparent advantages over LiDAR-based methods largely due to the cost. In general, a LiDAR sensor is much more expensive than a digital camera. But the central concern is whether the SFM-based methods can provide data that have comparable accuracy, resolution, and coverage to terrestrial LiDAR sensors. While SFM-based methods have been tested for infrastructure modeling and evaluation in several ¬¬studies in indoor laboratory setting or outdoor small-scale sites (Jahanshahi et al. 2012, Jahanshahi et al. 2013, Lattanzi 2013), there is still a lack of field studies that investigated the utility of such methods in supporting damage assessment with large-scale real disaster data. To date, there has been little research effort focused on evaluating the utility of SFM-based methods for post-hurricane virtual field studies. Therefore, a central research question to be addressed by this study is: can SFM-based methods provide data

sufficient for developing virtual hurricane impacted sites that can be used by researchers and damage assessors to conduct accurate post-hurricane residential building damage assessment?

My approach to address this research question involves three components: (1) modeling the data requirement for field-based post-hurricane residential building damage assessment; (2) developing dense 3D point clouds of impacted residential buildings from a Post-Sandy photo collections; and (3) comparing the generated 3D point clouds with mobile LiDAR data of the same residential buildings in terms of supporting quantifying damage information as determined in the first component.

5.2 Related Work

3D reconstruction from images is a classic computer vision problem that has been studied extensively for a few decades. The early focus of the field is on recovering 3D geometry from images with known camera poses. A rising research interest in this field is running 3D reconstruction at large scales on images with unknown camera poses, in particular those images harvested from different sources on the Internet. Furukawa et al. (2010) described the basic technical ingredients for this type of research as: (1) matching algorithms for providing accurate correspondence; (2) SFM algorithms for estimating precise camera pose using the matched feature; and (3) multi-view-stereo (MVS) methods for taking images with pose as input and producing dense 3D point clouds. Indeed, these elements have become standard components in many types of open source or proprietary 3D reconstruction pipelines, such as Autodesk 123D Catch, VisualSFM, Photosynth, PhotoModeler, and openMVG.

In general, these pipelines start with detecting correspondence between images with local feature descriptors such as SIFT, Harris Corner, and SURF. During this step, certain consistency measures such as RANSAC are often introduced to reduce erroneous matches. SFM-based pose estimation is then conducted to estimate camera poses. Notable SFM methods include the mutli-frame SFM (Spetsakis et al. 1991, Tomasi et al. 1992, Szeliski et al. 1994) and the now widely used bundle adjustment (Triggs et al. 1999). Once the camera poses are solved, MVS methods are

launched for dense reconstruction. The most common MVS method is the Patch-based Mutli-View Stereo (PMVS)(Furukawa et al. 2010), although recently there are a few other variants such as SURE (Rothermel et al. 2012) — a Semi-Global Matching based method. It is beyond the scope of this paper to describe the technical details of various 3D construction methods. However, in the following, I provide a brief review of studies which are relevant to the focus of this study.

The accuracy of image-based 3D reconstruction methods has been under constant debate. Most studies suggested that SFM based 3D image reconstruction provides less accurate results than the aerial or ground based LiDAR scanning (Dandois et al. 2010, Westoby et al. 2012, Fritz et al. 2013, Mathews et al. 2013, Ouédraogo et al. 2014). However , Dandois et al. (2010), Leberl et al. (2010), and Fonstad et al. (2013) concluded that SFM-based 3D reconstruction is comparable if not more accurate than airborne LiDAR. One common characteristic of these studies is that they all used photos collected from an aerial platform. In contrast to using aerial photos, Fathi et al. (2013)) applied 3D reconstruction methods for as-built roof modeling with photos taken from the ground level. They reported that their approach can reach the accuracy of $\pm 1.6cm$ for simple roof and $\pm 2.4cm$ for complex ones. However, the accuracy can be increased to $\pm 1cm$ if multiple scanners are combined and registered. Comparing to terrestrial laser scanning, which often has millimeter range accuracy, it is safe to conclude that 3D reconstruction is still not as accurate as terrestrial laser scanning.

Recently, SFM-based 3D reconstruction has become a vibrant research area in Civil Engineering. Some of the intriguing applications include construction scene reconstruction and progress monitoring (Golparvar-Fard et al. 2011, 2012), infrastructure modeling (Fathi et al. 2011, Dai et al. 2012), digital fabrication (Fathi et al. 2013), bridge inspection (Jahanshahi et al. 2013, Lattanzi 2013), substation modeling (Simões et al. 2012), traffic sign modeling (Soheilian et al. 2013), pavement distress detection (Koch et al. 2012), post-disaster assessment (Golparvar-Fard et al. 2010), and underwater survey (Pizarro et al. 2004). The study described herein has a similar objective as the research work by (Torok et al. 2013). In their study, SFM-based 3D

reconstruction and a crack detection method were proposed as a system for detecting surface cracks on critical building elements. However, the evaluation of the system was conducted on simulated disaster scenes. In this research, I focus on evaluating the potential of SFM-based 3D reconstruction for supporting large-scale post-hurricane building damage assessment with a real hurricane damage data set. Instead of focusing on specific damage features such as cracks, I focus on the ability of 3D reconstruction to support the fundamental measurement needs in assessing post-hurricane damage. To my best knowledge, this is the first study that has systematically evaluated SFM-based 3D reconstruction methods for hurricane damage assessment. The findings of this research provide new knowledge base on how to reconstruct and evaluate hurricane damages with photos, which have the potential to significantly reduce field data collection efforts after a major hurricane event.

5.3 Data Requirement

Existing post-disaster damage assessment protocols can be divided into two major categories: protocols for rapid damage assessment and protocols for detailed damage assessment. The first type of protocols are designed mainly for gathering information that can be used to quickly determine whether a structure is safe to enter or inhabitable. Notable examples of protocols of this nature are the rapid assessment protocol as provided in the ATC45 manual – Safe Evaluation of Buildings after Windstorms and Floods (ATC 2014) and the manual from the International Search and Rescue Response System (INSAR). These protocols are designed with the overarching goal of rapidly collecting perishable data with the degree of accuracy and completeness of the data as principal tradeoffs (Massarra 2012) The limitation of a rapid assessment can be overcome by following with a detailed assessment.

The protocols used for detailed assessment are mainly concerned with information that is useful for estimating repair cost or other engineering analysis purposes. Since hurricane introduces both wind and flood damages, damage information resulted from both events has to be collected. There are several post-hurricane damage evaluation protocols that can be used for this

purpose. A notable one is the Department of Housing and Urban Development (HUD) protocol used during Hurricane Andrew. Recently, Massarra (2012) proposed a new hurricane damage assessment protocol, which appears to be the most comprehensive one among existing protocols.

In general, post-hurricane damage data collection concerns with three types of data: building structure data, environmental data, and hazard data. The building structure data focus on the characteristics of building assemblies and their damage extents. These data can be collected at the building or building component level. The environmental data are geospatial data sets that describe terrain conditions and geographic settings of the impacted sites. This type of data is often collected at the community level instead of at the scale of individual buildings. The hazard data concern atmosphere or oceanographic variables characterizing extreme weathers. In the context of a hurricane, they include, but are not limited to, Storm Track, 1-Minute Wind Speed, 3-Second Gust Wind Speed, Wind Duration, Wind Direction, Wave Speed, Significant Wave Height, and Storm Surge Elevation. Remote sensing-based methods, in particular those aerial-based methods, are mostly commonly used methods for gathering environmental and hazard data due to the nature of these data sets: large-scale and wide area. The merit of street-level photos is primarily for collecting building structure data. The purpose of this study is to evaluate whether ground-taken image-based 3D reconstruction can provide information about building structures, in particular those being quantitative and spatial in nature Table 5.1. I used a similar damage classification approach as used in (Massarra 2012).

Table 5.1. A Post-Hurricane Assessment Protocol with Integrated Wind & Flood Damage Information

Building Subassembly Data			
Roof Subassembly		**Structure Body Subassembly**	
Load Side	**Resistance Side**	**Load Side**	**Resistance Side**

Roof Geometry	Roof Geometry		Percentage of Glass
Roof Pitch	Roof Pitch		Area
Mean Roof Height	Mean Roof Height		
Foundation Subassembly		**Below First Floor Subassembly**	
Load Side	**Resistance Side**	**Load Side**	**Resistance Side**
Pile Row Orientation	Pile Row Orientation	Breakaway Wall	Breakaway Wall
Pile Diameter	Pile Diameter	Orientation	Orientation
General Building Attribute Data			
Number of Stories	Structure Area and	Lowest Floor	
	Perimeter	Height	
Wind Damage Description Data			
Roof	**Structure Body**	**Foundation**	**Below First Floor**
Roof Cover Damage	Wall Cladding Damage	Not Applicable	Not Applicable
Roof Deck Failure	Wall Sheathing Damage		
Roof Structure	Wall Structure Failure		
Failure	Opening Damage		
Flood Damage Description Data			
Roof	**Structure Body**	**Foundation**	**Below First Floor**
Not Applicable	Wall Cladding Damage	Scour of Slab or	Stairway Damage
	Wall Sheathing Damage	Pile	Breakaway Wall
	Wall Structure Failure	Lateral Movement	Damage
	Opening Damage	Foundation	
		Settlement	
		Racking of	
		Elevated Structure	

Joint Damage Description Data			
Roof	**Structure Body**	**Foundation**	**Below First Floor**
Not Applicable	Wall Cladding Damage Wall Sheathing Damage Wall Structure Failure Opening Damage Appurtenant Structure Damage	Not Applicable	Appurtenant Structure Damage

The specific focus of this study is on the fields requiring tedious field measurements. These measurements can be reasonably grouped into four types of measurements, being the length, area, angle, and volume measurements. The utility of 3D models created from photos would be highly related to these measurements. After all, it is of a great interest to damage assessors and researchers to know whether a virtual reality site built from a large collection of images would allow them to perform accurate virtual field work that are used to be difficult to perform in the actual field. Therefore, this study will focus on a representative set of these field works as shown in Table 5.2.

Table 5.2. Post-Hurricane Assessment Protocol Investigated in this Study

	Roof	**Structure body**
Detail assessment protocols	Roof pitch Roof area	Wall flatness Wall inclination angles Damage area Opening size

5.3.1 3D Reconstruction of Disaster Sites

Hurricane Damage Data Sets: Hurricane sandy was the most destructive hurricane in 2012. The winds and surge caused approximately $50 billion total damage to the residential buildings, infrastructure, and commerce. At least 650,000 homes were impacted, and 8.5 million people were left without power for up to one month (Blake et al. 2013). After Hurricane Sandy, a Rutgers team surveyed several hurricane impacted sites with both a mobile LiDAR scanning system and a traditional foot-on-ground survey approach. The covered areas include Staten Island and Rockaway in New York City, Ortley Beach, Seaside Heights and Mantoloking on the New Jersey Shoreline. In this paper, I used photo and mobile LiDAR data on two representative buildings sustained damage from Hurricane Sandy (Figure 5.1). A total of 55 photos and 33 photos were taken from different angles for building 1 and building 2 on ground, respectively. However, a faster alternative can be conducted using a mobile platform.

The LiDAR system used for data collection is an Optech LYNX Mobile Mapper M1 system. The LYNX system relies on two 500 kHz LiDAR sensors to collect a million points per second while maintaining survey grade quality precision. Without ground control points but with a mobile base station, the Optech system is capable of obtaining LiDAR data sufficient for feature extraction of planimetric and topographic features typically at an absolute accuracy of $+/-10cm$ @ 1 σ in good GPS coverage areas and an relative accuracies of $+/-5cm$ @ 1 σ anywhere within the project area. The mobile LiDAR point clouds for both buildings are used as ground truth information due to their high spatial accuracy. The purpose is to evaluate to what extent 3D image reconstruction can be used in post hurricane assessment.

Figure 5.1. Selected Residential Buildings for Post-Sandy Assessment. (a) Location of Two Selected Buildings; (b) Residential Building 1; (c) Residential Building 2

5.4 3D Reconstruction

Many commercial or open source implementations of 3D reconstruction are available and can be used for the proposed disaster site reconstruction. In this study, I chose the Comercial Autodesk 123D Catch software (Catch 2012) and SURE (Rothermel et al. 2012) as the tools to reconstruct the 3D disaster sites. Regarding these two programs, one is a commercial program; and the SURE is an open source implementation. More specifically, in the SURE-based approach, a set of images and the interior and exterior orientations are derived by VisualSFM (Wu 2011). Afterwards, epipolar images are generated, and the dense matching is conducted using a SGM algorithm (Rothermel et al. 2012). By using two approaches, I ensure the results are not biased towards one specific method. Because the knowledge of the absolute distances between camera poses is not available, the reconstructed point clouds are scale-free and they are properly scaled using reference lengths as collected onsite.

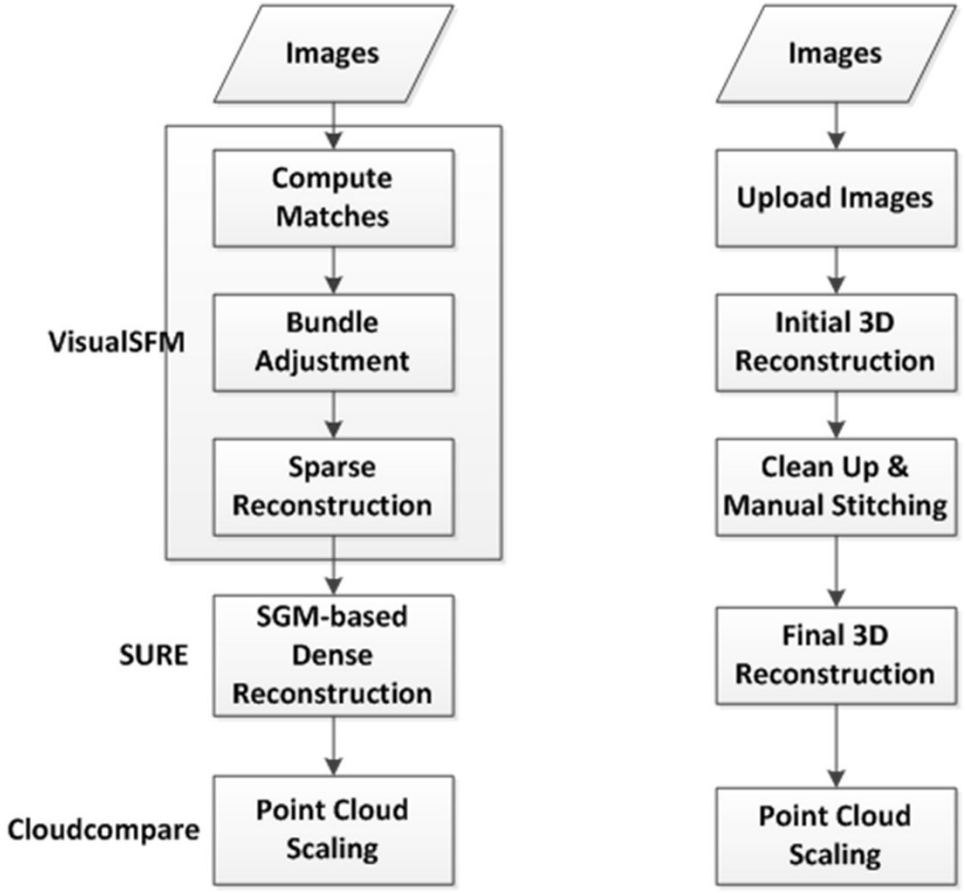


Figure 5.2. The Workflows used in SURE and 123D Catch-based 3D Reconstruction

5.5 Comparative Analysis

The reconstructed 3D point clouds of residential building 1 and building 2 are shown in Figure 5.3 and Figure 5.4. Each figure also includes the mobile LiDAR point cloud, which is used as the ground truth data. In the following, I compare these point clouds from a variety of perspectives including resolution, global accuracy and component-level accuracy.

5.5.1 Resolution and Details

In this section, the resolutions of point clouds from the mobile LiDAR, SURE, Autodesk 123D Catch are compared. To compute the point density, the Delaunay triangulation of this point set is computed. For each Delaunay triangle, the vertexes with triangle sizes larger than a user defined threshold are treated as outliers and removed. Afterward, the areas of each remaining

Delaunay triangles are calculated and the quotient of total number of points to the summation of each Delaunay triangles is taken as the density of this point cloud without outliers. Table 5.3 shows the comparison results. It can be noted that the amount of points generated by LiDAR and 123D Catch depends on the quality of laser scan and the quality of image collection, while the point cloud from SURE has much more points, and consequently it has the highest density. On the other hand, the point cloud from the 123D Catch has much sparser resolution.

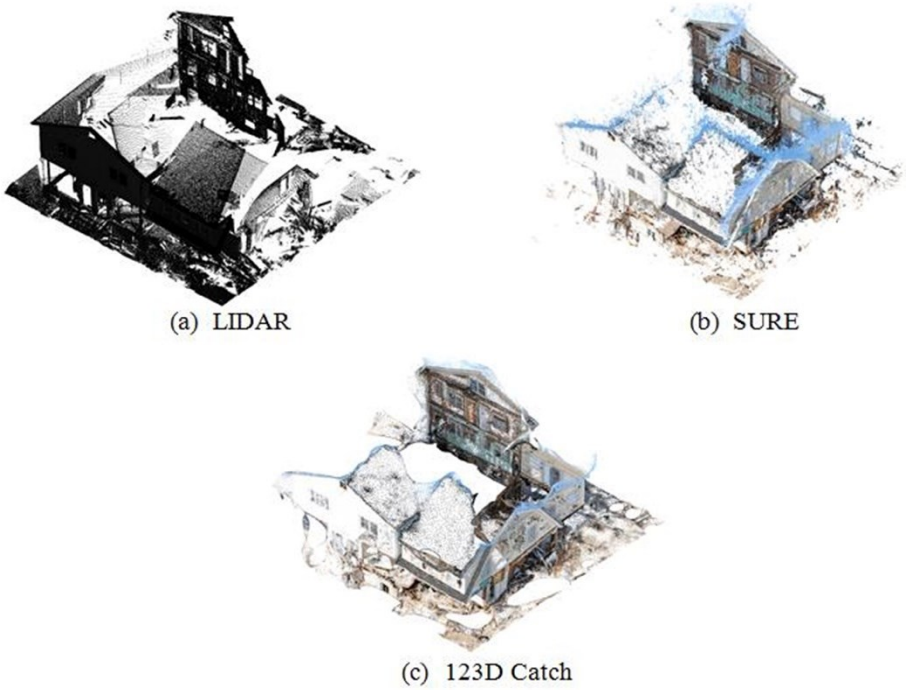


Figure 5.3. Point Clouds of Residential Building 1

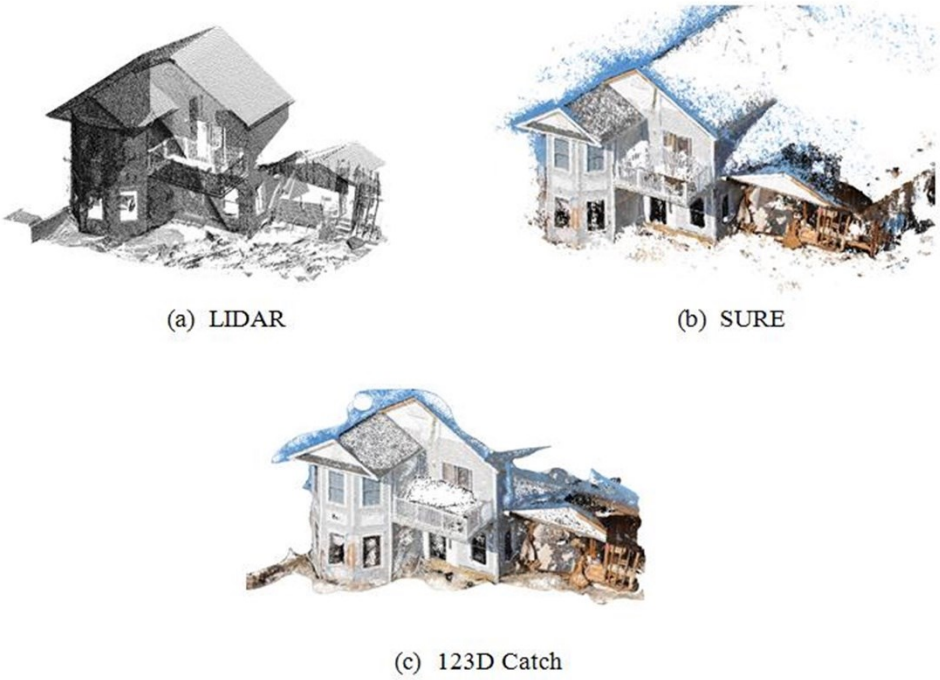


Figure 5.4. Point Clouds of Residential Building 2

Table 5.3. Resolution Comparison

	Residential Building 1		Residential Building 2	
	Amout of Points	Point Density(/m2)	Amout of Points	Point Density(/m2)
LiDAR	68,280,775	3,083	2,546,521	500
SURE	113,642,407	23,799	37,010,935	2,141
123D Catch	15,295,295	251	9,797,536	273

In terms of the destruction details that can be observed from these methods, I examined several exemplar cases. For example for wall cladding damage as shown in Figure 5.5, both the LiDAR and SURE point clouds can clearly display the local protuberance of the siding. In contrast, the 123D Catch point cloud tends to smooth the corners of the local protuberance through curving the linear legs of the corners.

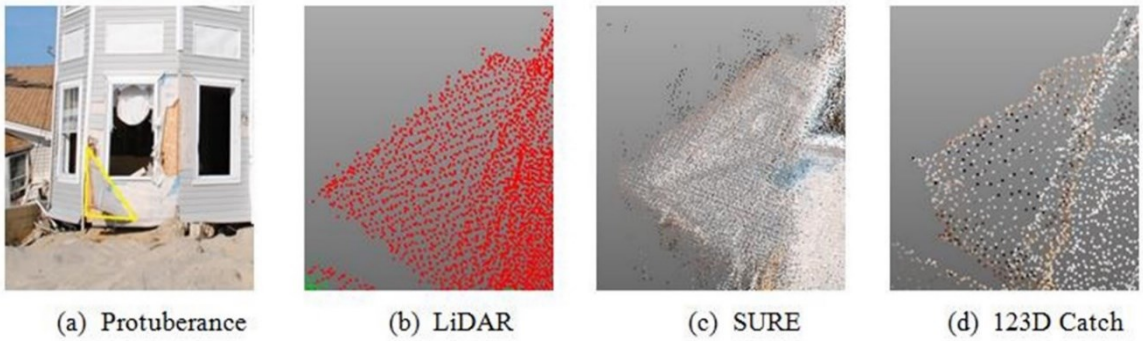


Figure 5.5. Visualizing Wall Cladding Damages in Point Clouds

In another example, I examined wall opening damages, in particular window damages. Since the laser beams do not reflect on glasses, windows with and without broken glasses cannot be differentiated based on the LiDAR point clouds. As shown in Figure 5.6, the top part of window in Figure 5.6 (a) is broken, which could be indicated from the color change in Figure 5.6 (c). However, the LiDAR point cloud returns nothing from both broken and intact part as shown in Figure 5.6 (b), and the low point density of Figure 5.6 (d) makes it hard to identify the status of window. The window in Figure 5.6 (e) is intact shown in digital image, and this is clearly presented through the uniformity of glass color in Figure 5.6 (g), but it is hard to tell whether this window is broken or not from the LiDAR point cloud shown in Figure 5.6 (f). For the third window illustrated in Figure 5.6 (i), LiDAR returns points from the lower part due to the objects behind the window, which differs from what can be seen in Figure 5.6 (k) and Figure 5.6 (l). Above all, LiDAR based opening damage assessment performs poorly compared with image-based approach, in which whether an opening is broken or not can be distinguished from the color.

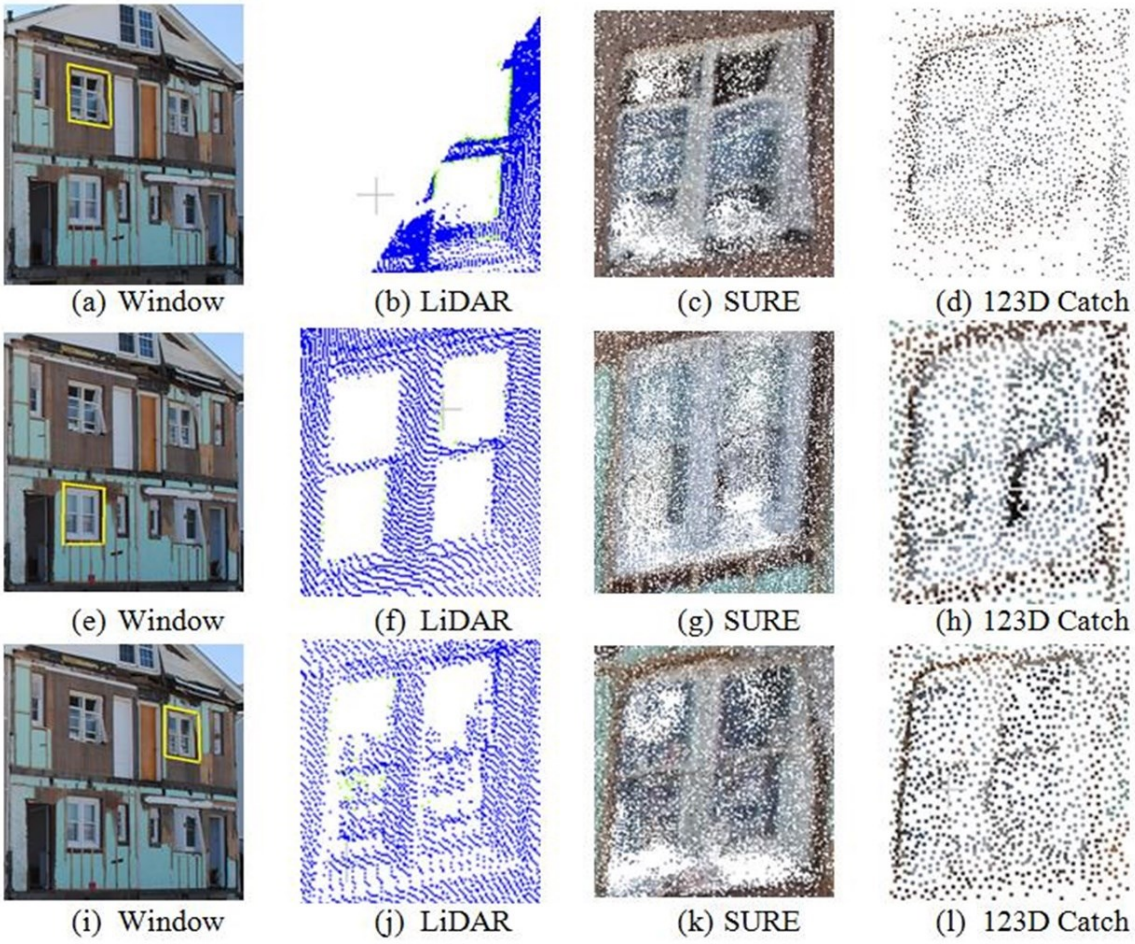


Figure 5.6. Visualizing Window Damages in Point Clouds

5.5.2 Global Accuracy

In this section, I examine the overall accuracy of the point clouds from image based 3D reconstruction methods by aligning them with the mobile LiDAR point clouds. The alignment process includes several steps: (1) aligning two point clouds roughly by picking several equivalent point pairs (at least three pairs); (2) an Iterative Closest Point (ICP)-based fine registration is performed to register the roughly aligned point clouds; and (3) performing Hausdorff distance computation to compute the distances between corresponding points in the aligned point clouds. Figure 5.7 and Figure 5.8 show the overall comparison results between the SFM based 3D reconstruction point clouds and the LiDAR point cloud. The results suggest that

the reconstruction has good spatial accuracy on flat objects such as walls, roofs, and balcony railings, but the accuracy suffers on objects such as eaves and openings. This is because: (1) for eaves, the reconstructed point clouds erroneously included a number of points representing the environment, such as the vegetation and sky along the edge of eaves. These points are also counted in computing the point-to-point distance, leading to the larger distance difference at the edges of eaves; and 2) for openings like windows, doors, and damaged areas, the reconstructed point clouds sometimes include points representing glasses and indoor components (e.g. partitions and ceilings) while the mobile LiDAR point clouds do not. This disparity increases the point-to-point distance dramatically. A statistics of the comparison results, including the median and mean of the differences, is also shown in Table 5.4. For the comparison between SURE algorithm and LiDAR, 90% of difference is less than 0.0669m for residential building 1 and 0.0307m for residential building 2, respectively. For comparison between 123D Catch and LiDAR, this value becomes to 0.2374m and 0.0412m regarding to residential building 1 and 2. This indicates that both image-based 3D reconstruction methods provide reliable accuracy regarding to post-disaster assessment for residential building scale objects.

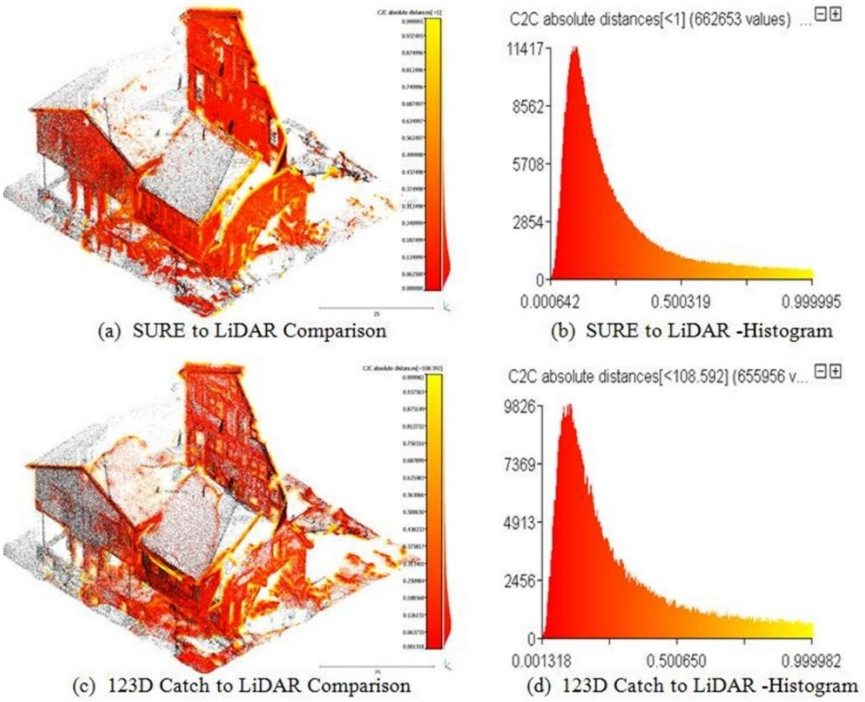


Figure 5.7. Overall Comparison of Residential Building 1

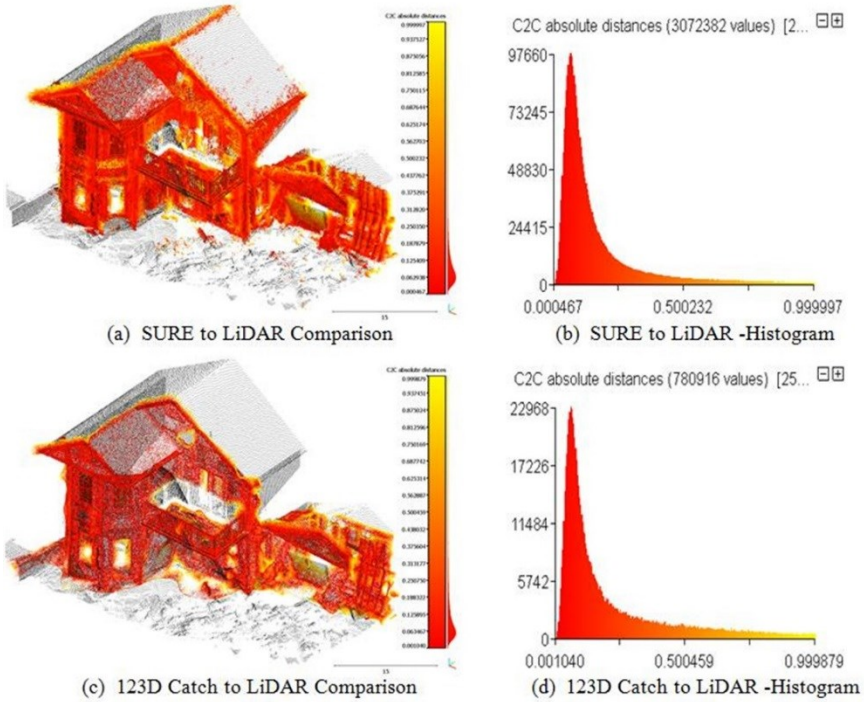


Figure 5.8. Overall Comparison of Residential Building 2

Table 5.4. Median and Mean of Comparison (m)

	Residential Building 1		Residential Building 2	
	Median	Mean	Median	Mean
SURE to LiDAR	0.15	0.24	0.04	0.08
123D Catch to LiDAR	0.15	0.23	0.05	0.12

5.5.3 Building Component-level Accuracy Evaluation

After gaining understanding on the overall accuracy of the reconstructed point clouds, the next step is evaluating the accuracy of the reconstructed point clouds in terms of supporting component level damage assessment. In this perspective, I focused on a variety of component level damage assessment tasks as shown in Figure 5.9. These damage assessment tasks involve mostly length, area, and angle measurements

Direct measurement of length, area, and angles on the point cloud is difficult. However, this can be alleviated once the point clouds are fitted to planar surfaces. Since the roofs and walls of a residential building are, in most cases, flat surfaces, the roofs and walls can be modeled by planes fitted to the 3D point clouds. The following measurements of roof area, opening size, inclination angle, and wall flatness are conducted based on the fitted planes. The equation of a plane can be expressed as:

$$x\cos\alpha + y\cos\beta + z\cos\gamma + p = 0 \qquad (5.1)$$

where $\cos\alpha$, $\cos\beta$ and $\cos\gamma$ are the direction cosine of the normal vector at point $\{x, y, z\}$ on the fitted plane, and $|p|$ is the distance from origin to the plane. Let d_i denote the distance from point *i* to this plane written as:

$$d_i = |x_i\cos\alpha + y_i\cos\beta + z_i\cos\gamma| \qquad (5.2)$$

In order to obtain the best fitted plane, $(\sum_{i=1}^{n} d_i^2 \rightarrow min$, with condition of $(\cos\alpha)^2 + (\cos\beta)^2 + (\cos\gamma)^2 = 1$ must to be satisfied. Therefore, the plane fitting problem becomes a

Lagrange extreme problem. The inclination angles of roofs and walls will be represented as three angles: the angle between x-axis and the normal vector, θ_x; the angle between y-axis and the normal vector, θ_y; and the angle between z-axis and the normal vector, θ_z.

Figure 5.9. Evaluated Aspects of Residential Building

Since the original LiDAR point cloud and SFM based reconstructed point clouds do not locate in the plane exactly, some of the points have the same direction to the planes as the normal vector of the fitted planes, and some have opposite direction to the planes. It is not reasonable to compute the areas and opening sizes based on the original point clouds. In this study, all the points are projected onto the fitted plane (Figure 5.10), all the red points represent the original point cloud, and the blue ones are projected points). Afterwards, the damage areas, opening sizes, etc., are computed based on the projected planes.

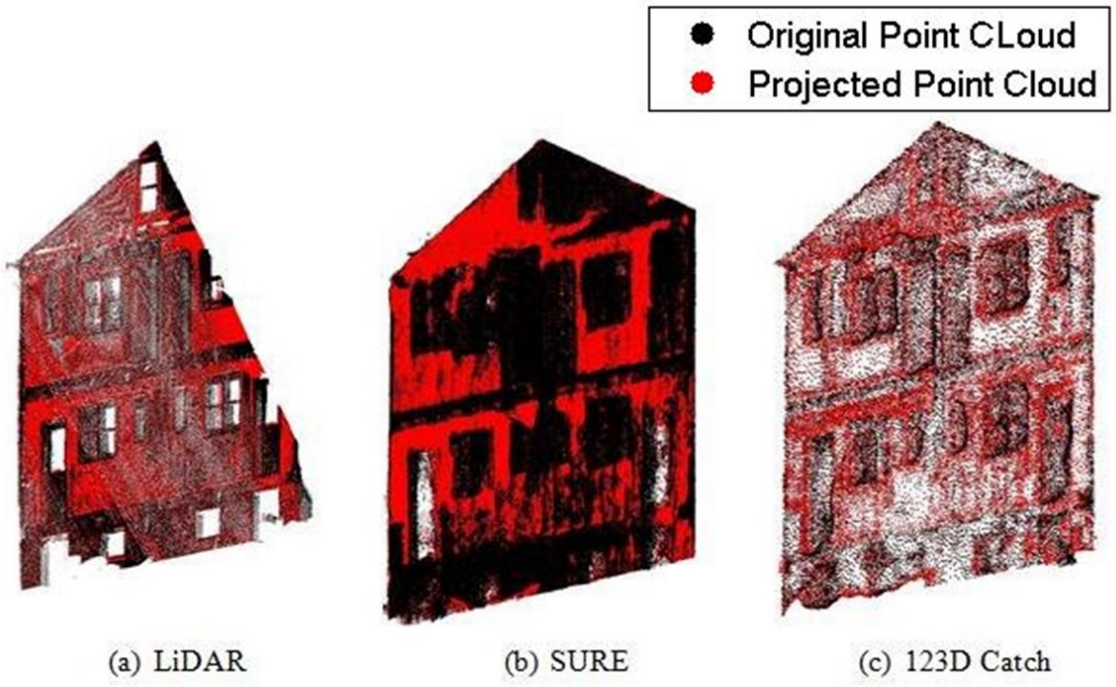


Figure 5.10. Projecting Point Clouds onto Planes

Length: Balcony Height / Floor Height / Roof Height: Balcony height, floor height, and roof height are important building attributes that will be collected on damaged housing structures. Figure 5.9 (a) shows the type of balcony height measurement performed using the point cloud data. As the balcony is slightly inclined, the heights from its four corners to the ground are measured, respectively. As mentioned before, the fitted surface is employed to represent the balcony point clouds because the original point clouds contain noises, which could affect the measurement accuracy. ∝-shape method proposed by (Shen et al. 2011) is employed to extract the boundary points and the corner points from a user segmented balcony and roof point set. The boundary points are detected using this approach. Afterward, a classification procedure is conducted to categorize the boundary points into different groups according to the edge they belong to based on the histogram distribution of their coordinates. A least-square approach is then carried out to predict the line of each edge and the corners are computed as the intersection of each pair of intersecting lines. Figure 5.11 (a) illustrates the extracted boundary points of one

selected roof from residential building 2 shown in Figure 5.9 (e). The grouped boundary points for each edge are shown in Figure 5.11 (b). The ground elevation is also imperative in the measurement. Due to the uneven of ground and sparsely located debris/ sand, however, the estimation of ground elevation is highly empirical and subjective. In this study, the elevation of the first floor, represented by the blue line in Figure 5.9 (a), is used instead of estimating the ground elevation. Table 5.5 shows the height measurement results using different point cloud data. The type of floor height measurements is shown in Figure 5.9 (b) and Figure 5.9 (c). The assessment of floor height is less subjective than the measurement of balcony height because the measurement of floor height, if not the first floor, does not rely on ground elevation estimation. The convex hull algorithm is, again, employed to detect the edge points. For both floors, edge AB and edge CD are detected separately, followed by the measurement of the edge length as the floor height. The roof height measurements are the perpendicular distance between the left and right edges of a roof (Figure 5.9 (d) and Figure 5.9 (e)). The floor height measurements, the roof height measurements, and their measuring errors (compared with mobile LiDAR data) (shown in the parenthesis) are summarized in Table 5.6.

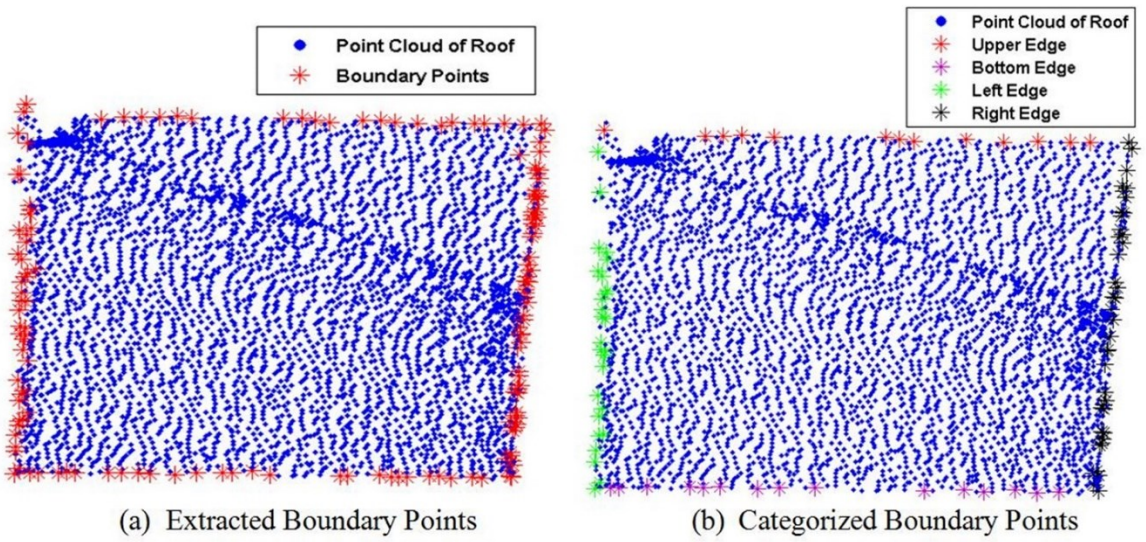


Figure 5.11. Boundary Points Extraction

Table 5.5. Balcony Height Measurements (m)

	Top-left Corner(A1-A2)	Top-right Corner(C1-C2)	Bottom-left Corner(B1-B2)	Bottom-right Corner(D1-D2)
LiDAR	2.59	2.84	2.60	2.86
SURE	2.65 (0.06)	2.76 (-0.08)	2.59 (-0.01)	2.71 (-0.16)
123D Catch	2.75 (0.17)	2.68 (-0.16)	2.69 (0.09)	2.61 (-0.25)

Table 5.6. Floor/ Roof Height Measurements (m)

		Residential Building 1		Residential Building 2	
		Left Edge(A-B)	Right Edge(C-D)	Left Edge(A-B)	Right Edge(C-D)
LiDAR	Floor Height	1.59	1.57	2.88	3.00
SURE		1.59 (0.00)	1.62 (0.05)	2.90 (0.02)	3.00 (0.00)
123D Catch		1.54 (-0.05)	1.62 (0.05)	3.02 (0.14)	3.12 (0.12)
LiDAR	Roof Height	2.58	2.61	1.49	1.47
SURE		2.64 (0.06)	2.92 (0.31)	1.61 (0.12)	1.37 (-0.10)
123D Catch		2.55 (-0.03)	2.67 (0.06)	1.53 (0.04)	1.36 (-0.11)

Table 5.7. Length Measurement Error Analysis

	Maximum (m)	Median (m)	Mean (m)	Variance (m)
SURE	0.630	0.000	-0.020	0.0238

123D Catch	0.300	-0.025	-0.019	0.0078

The errors in length measurement are summarized in Table 5.7. It can be noted that once the measurement involves estimation of the ground level, the accuracy generally suffers. Another observation is that due to missing data on the edge of roofs or on roof surface areas, it is generally difficult to obtain reliable results.

To evaluate the surface area measurement accuracy when the reconstructed point clouds are used, I performed measurements on wall opening and cladding damage areas. Figure 5.12 shows the measurement on a damaged door opening shown in Figure 5.9 (h). The measurement results are summarized in Table 8. The results suggest both approaches give good accuracy in measuring the area and opening sizes of the damaged door. The area computed by SURE is only 0.6% smaller than the area measured by LiDAR, and for 123D Catch, it is 0.9% smaller than LiDAR data. For each edge, the largest discrepancy between SURE and LiDAR is 5.6%, and for 123D Catch and LiDAR, it is 3.0%.

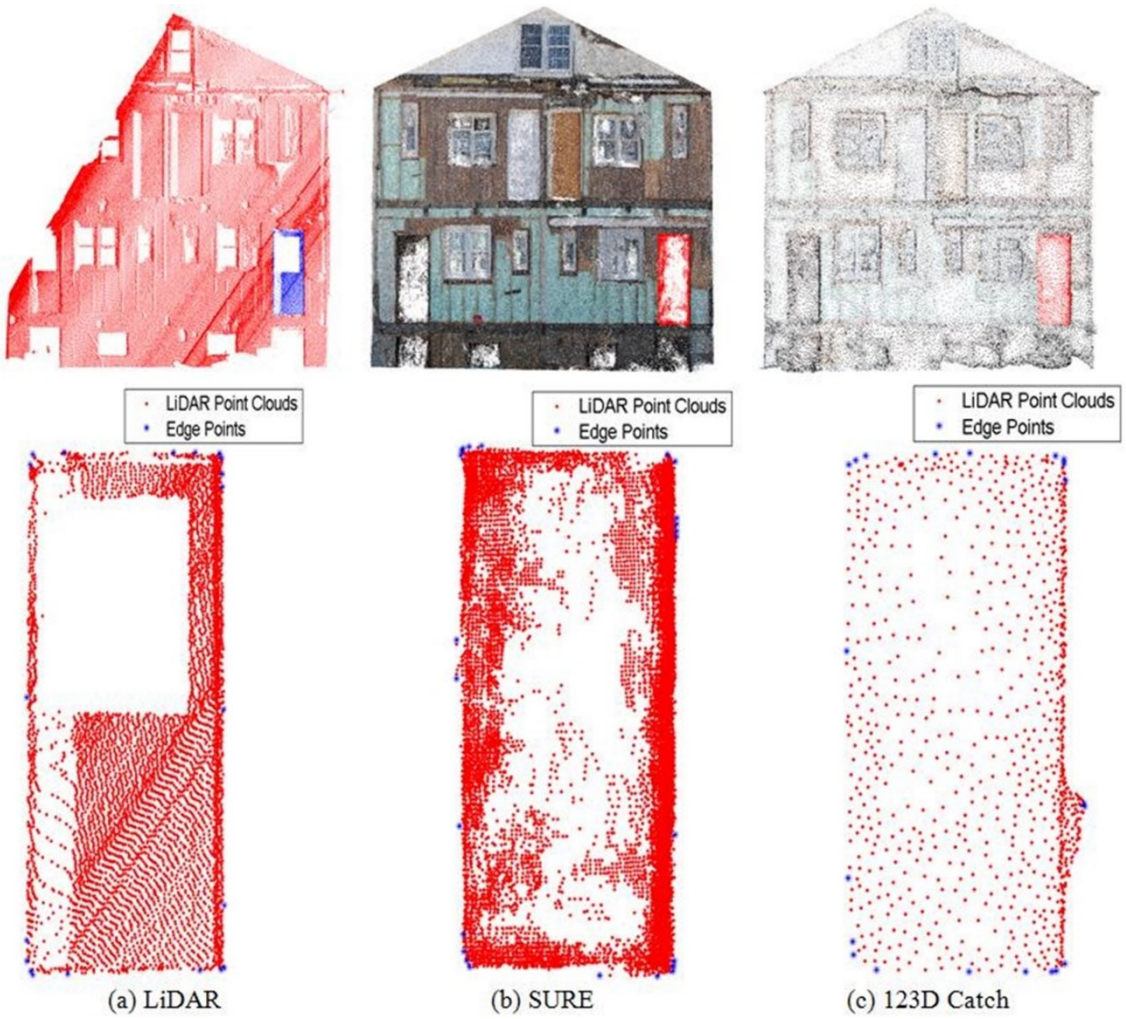


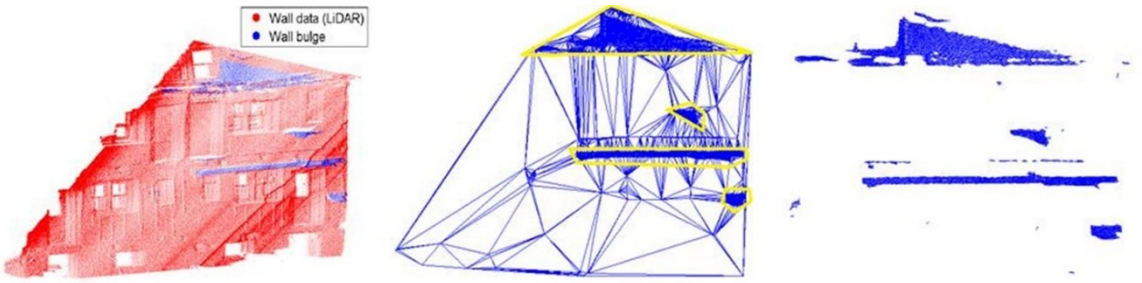
Figure 5.12. Wall Opening Size Measurements

Table 5.8. Area and Opening Sizes of Damaged Door

	Area(m^2)	Opening Sizes(m)			
		Upper Edge	Lower Edge	Left Edge	Right Edge
LiDAR	1.86	0.86	0.95	1.99	1.98
SURE	1.85	0.87	0.90	2.04	1.96
123D Catch	1.84	0.89	0.95	2.03	1.95

Other than wall opening size measurement, I also conducted measurement the area of lifted wall sidings by the storm surge and wind forces. In all the point clouds, the lifted sidings were

detected by recognizing points that were falling certain distance off the fitting planes. The blue areas in Figure 5.13 show these detected lifted sidings. The Delaunay algorithm in MATLAB, which maximize the minimum angle of all the triangles, is carried out to create meshes to represent the lifted sidings. The areas of these meshes are then computed for comparison (Table 5.9). The results suggest that the area measurement on the SURE point cloud has much higher accuracy than using the 123D Catch point cloud. More specifically, for residential building 1 (Figure 5.9 (f)), the area of damaged sidings given by SURE is 0.6% smaller than LiDAR scanning, and for residential building 2 (Figure 5.9 (g)), it is 5.5% smaller. However, the areas computed on the 123D Catch point cloud are 60.3% and 48.9% larger than the areas computing on the LiDAR data. This discrepancy occurs mainly because the 123D Catch tends to smooth the protuberance via curving the periphery. It tends, hence, to increase the area of damaged cladding.



(a) Original Point Cloud (b) Delaunay triangulation (c) Refined Delaunay Triangulation

Figure 5.13. Lifted Siding Area Detection and Measurement

Table 5.9. Areas of Damaged Wall Cladding

	Residential Building 1(m^2)	Residential Building 2(m^2)
LiDAR	2.39	0.10
SURE	2.37	0.10
123D Catch	3.83	0.15

Inclination Angle: Roof and Balcony Slope: Angle measurements, such as roof slope and structure inclination, are another important type of measurements performed during hurricane damage assessment. In this study, I performed measurement on balcony inclination (Figure 5.9 (c)) and roof pitches (Figure 5.9 (e)) using the point cloud data of Residential Building 2. The results are summarized in Table 5.10. It can be noted that the measurements on the SURE and 123D Catch point cloud yield accurate roof pitch angles and slightly less accurate balcony inclinations.

Table 5.10. Roof Pitch and Balcony Inclination of Residential Building 2

	Roof Pitch			Balcony Inclination		
	θ_x	θ_y	θ_z	θ_x	θ_y	θ_z
LiDAR	59.1°	90.8°	30.9°	89.7°	84.3°	5.7°
SURE	58.6°	90.7°	31.4°	89.1°	87.2°	3.0°
123D Catch	59.0°	90.9°	31.0°	89.2°	91.9°	2.0°

5.6 Discussion

The results of this research demonstrated that photos taken by ground-based assessment teams can be used to recreate damage sites in three dimensions. Compared to mobile LiDAR data, point clouds from image-based 3D reconstruction techniques have comparative resolution. This is not a surprise as close range images taken by today's digital cameras often have very high resolution. In terms of the accuracy of the derived point cloud data, I focused on the accuracy of length, area, and angle measurement. As examples, I focused on these measurements as demonstrated by measuring floor height, wall opening size, cladding damage area, roof pitch, and structure inclination on two damaged residential houses.

For length measurement, the study suggests that 10-centimeter accuracy is generally achievable except for certain situations where the edges of structures are erroneously

reconstructed. In terms of area measurement, point clouds reconstructed using the SURE approach can generally support area measurement at accuracy close to 6%. Nevertheless, the 123D catch approach tends to over-smooth the sharp edges of objects, leading to large area measurement errors. For roof pitch and structure inclination measurements, measurements on 3D reconstructed point clouds can yield fairly accurate results. It should be noted that I did not use any manual stitching methods to solve any ambiguity in the image matching process. Use of manual stitching method will certainly improve the reconstruction results.

One limitation of this study is that it assumes that the images are grouped by individual buildings. The assumption simplifies the reconstruction process as it is well known that buildings tend to have similar local features which often confuse SFM-based 3D reconstruction and lead to erroneous image matching. Despite of this limitation, it is a reasonable assumption that damage assessors usually keep a good log of pictures for different buildings. Another limitation of this study lies in how the photo data were collected. There is limited line of slight or visibility on roofing structures when a photo is taken from the ground. This has taken a toll on the 3D reconstruction as only a limited portion of the roof can be reconstructed. The rise of Unmanned Air Vehicles (UAV) will eliminate this limitation as photos taken from a UAV platform will be immune to this issue. It is expected that the combination of UAV and ground-based photo collection will provide high quality photo data for 3D reconstruction-based damage assessment.

5.7 Conclusion

This study evaluated to what extent SFM based 3D reconstruction methods can be employed in post-hurricane assessment. By leveraging image and LiDAR data sets collected during Hurricane Sandy, I reconstructed two damaged housing structures and compared them with highly accurate mobile LiDAR data to quantify the accuracy of the point clouds from 3D reconstruction in terms of length, area, and angle measurement. These measurements are part of hurricane damage assessment that was routinely performed on damaged residential buildings. It can be concluded that SFM based 3D reconstruction can adequately support hurricane damage

assessment needs for residential buildings. However, for critical infrastructures which rely on very accurate estimate (generally < 1 cm) of displacement, deflections, and other distresses, it requires more accurate methods than SFM-based 3D reconstruction. Despite of this limitation, as digital cameras are ubiquitous in mobile devices and the availability of more versatile sensing platforms such as UAVs, image-based 3D reconstruction will likely become main stream applications in post-disaster response. This study provided new knowledge base for using image-based 3D reconstruction for post-hurricane damage assessment.

Chapter 6 : Robust Alignment Of Multi-Sourced Imagery Data For Building Damage Assessment

Optical data have played a critical role in post-hazard damage assessment. Among them, LiDAR point cloud data and aerial/oblique imagery have been extensively used for damage assessment. While each of these types of data sets has unique strength in supporting damage assessment, they also have their limitations mostly due to their collection mechanisms. Alignment of these data sets into a consistent coordinate system such that precise pixel to point correspondence can be achieved opens doors to new opportunities for efficient damage assessment. However, due to the heterogeneousness in data resolution, data coverage, and data capturing angles, such alignment is a challenging task to achieve. In this study, I present a novel approach for robust alignment of multi-sourced street-level imagery with point cloud data to support essential building damage assessment tasks. In the proposed approach, geo-tagged images are first aligned with high-resolution LiDAR point cloud to retrieve the 3-D spatial information. This is achieved through a human-in-the-loop process, where human interaction is minimized to select several simple polygons. After the image is aligned with the point cloud, fine-scale building damage assessment can be conducted using the LiDAR infused street-level images. The proposed approach was applied on three damaged homes during Hurricane Sandy. For each building, I demonstrated the use of the processed images for assessing damaged areas and quantifying flood height. The experiment results show that the proposed approach can efficiently align arbitrary street-level images with respective point cloud data, and support quantitative damage assessment tasks, in particular those that cannot be conducted with LiDAR point cloud data alone.

6.1 Introduction

In recent decades, extreme events, especially hurricane and their associated storm surge, have caused tremendous damage to coastal communities. Immediately after the coastal storm events, post-disaster damage assessment is critical in many aspects, such as loss estimation and post-

event recovery. Traditional damage assessment approaches rely on foot-on-ground inspection of damaged properties and infrastructures (Chiu et al. 1999, Crandell et al. 2005, Hatzikyriakou et al. 2015, Xian et al. 2015). In many cases, disaster management agencies such as Federal Emergency Management Agency (FEMA) rely on public assistance from engineering firms to conduct these exhaustive damage surveys.

Recently, remote sensing technology has provided alternative solutions to post-hazard assessment. In general, remote sensing-based damage assessment could be categorized into 1) airborne LiDAR-based (He et al. 2016), 2) airborne imagery-based (Sui et al. 2014, Tu et al. 2016), 3) oblique imagery-based (Vetrivel et al. 2016, Vetrivel et al. 2016), and 4) mobile/static LiDAR-based (Gong et al. 2014, Kashani et al. 2014, Kashani et al. 2015). Compared with the traditional damage assessment approaches, these approaches have important advantages.

Airborne-based approaches, including LiDAR and imagery, are efficient means of capturing necessary data to support damage assessment. They can be used to conduct immediate assessment, and are less affected by the road accessibility issues. In addition, they can capture data from over-large geographic areas in a short amount of time. These factors often make the airborne-based approaches the first choice in post-disaster damage data collection. However, these approaches often limit the data collection to the areas which are visible from the aerial view, and are generally have issues with dense vegetation. Oblique imagery from manned airplanes can alleviate some of these limitations, but often have limited resolution due to high data collection attitude. Oblique imagery from Unmanned Aerial Vehicles (UAV) such as multi-rotatory copters can provide data at a much higher resolution. However, battery life is still a major constraint that limits the area of UAV data collection to targeted areas instead of large geographic regions.

Ground-based approaches, including static and mobile LiDAR, provide higher resolution and accurate data which are often coupled with co-registered street-level imagery. However, both mobile-based and static-based approach suffers from several issues including road accessibility

and limited field of view. For example, mobile LiDAR is often hosted on a vehicular platform such that data collection can only be conducted along the roads.

Volunteered geographic information (VGI), which in essence leverages crowd-sourced information, is a relatively new approach for damage assessment (Poser et al. 2010, Barrington et al. 2012, Xie et al. 2016). Among the types of crowd-sourced data, images are an important type of information as they are intuitive and information rich. Many types of VGI data are geotagged due to the ubiquitous use of mobile devices. These geotag make it feasible to rapidly localize VGI data. While these types of geo-localization work well for text type of data, they fall short of unleashing the true potential of crowd-sourced image data. In most cases, these images still need manual interpretation to extract information. Crowd-sourced image data, once aligned with and registered with other modes of visual damage data such as those remotely sensed from aerial and mobile platforms, can pave the way towards a data fusion framework for intelligent damage assessment with multi-modal visual damage data, such as flood height measurement (Schnebele 2013, Triglav-Čekada et al. 2013).

This study addresses the challenge of robust matching and aligning VGI images with high-resolution imagery and LiDAR data collected during ground assessment trips. The goal is to recover precise geometric information in VGI images with the assistance of 3D LiDAR data. By doing so, it also addresses a common limitation associated with remotely sensed data. Remotely sensed data are often limited in terms of field of view as the data collection paths are limited to the platform trajectories. More specific to this study, the registered VGI images can serve as additional high-resolution overlays on mobile LiDAR data to support quantification of hazard impacts. To this end, an algorithmic framework for fusing VGI images with mobile LiDAR data with a minimum human-in-the-loop actions is developed in this study. My method is capable of aligning arbitrary street-level images containing geo-location information with high-resolution LiDAR data to support various damage assessment task.

6.2 Related Work

In recent years, the fusion of imagery data and LiDAR point cloud data have been paid great attention in the field of robotics, mobile mapping, and emergency response. As the LiDAR sensor could be categorized into airborne and ground-based, the fusion of imagery data and LiDAR data can also be grouped accordingly.

6.2.1. Fusion in Robotics and Mapping Applications

Fusion of image and range sensors data has been widely applied in robotics, navigation, surveying, and mapping. There are two major categories of fusions: one is the fusion of aerial/oblique imagery and airborne LiDAR, and another is the fusion of ground-based imagery and LiDAR.

Fusion of aerial imagery with airborne laser system (ALS) has been excessively used in large-scale mapping and surveying. For example, Zhang et al. (2015) proposed a registration approach applied in forest survey and mapping. In their approach, the registration between UAV image and LiDAR data is achieved using inherent geometric constraint. For a pair of corresponding object, if the registration is correctly determined, the points should be back-projected within the object boundary in the image. Based on this assumption, their approach estimates the calibration parameters by optimizing the objective function defined by the ratio of points projected within the boundary to the total number of points. For aerial case, linear feature is widely used in research efforts and engineering application. Rönnholm et al. (2013) uses a linear feature-based approach to register the image with lidar data. In this study, the roof ridges are extracted from both lidar and image data. And the orientation parameters are estimated via solving least square adjustment subject to coplanarity condition. A similar approach is also presented by Liu et al. (2016). In their study, the roof ridges and building boundaries are extracted using a semi-automated approach. Using the initial exterior orientation (EO) parameters obtained by POS, these line objects are back-projected to images, and the corresponding line segments from the image are identified using semi-automated manner. Then the fine EO

parameters are obtained iteratively by solving a co-planarity function. More recently, Sheng et al. (2017) described a line vector-based registration framework. It starts with line detection and feature description computation. The exterior orientation parameters are obtained by iteratively solving a coplanarity equation and updating the parameters based on the line vector features.

Instead of registering the image with lidar frame by frame, Yang et al. (2015) and Abayowa et al. (2015) utilized the partial overlapping of aerial imagery to recover the relative image position. It is done by generating dense point cloud using Structure from Motion technique. After the photogrammetry derived point cloud is generated, the lidar point cloud to derived point cloud registration is implemented with ICP process. And the fine image to lidar registration is obtained through this process.

In addition to the abovementioned feature-based approaches, data-driven approaches have been also studied. For example, mutual information between the image gray-scale value and the reflectivity value of point cloud is used as statistical dependency measurement (Mastin et al. 2009, Parmehr et al. 2014). Therefore, the registration problem is converted to an optimization problem, where the objective of the problem is to find a set of camera-to-lidar parameters that maximizes the mutual information.

Fusion of ground-based camera and mobile laser system (MLS) has been applied excessively in many mobile mapping systems and robotic systems for mapping, navigation, and object recognition. Therefore, careful calibration of camera-range systems is critical. Many in-lab calibration systems have been proposed. For example, Geiger et al. (2012) proposed a fully automated approach to calibrate the camera to range system. Their system requires a bunch of checkerboards as target. By detecting the checkerboards in both sensors, transformation hypothesis is generated by randomly choosing plane associations. The fine parameters are then optimized using the best plane association. Similar idea of calibrating assisted by special targets is proposed by Alismail et al. (2012). In this system, a special target designed as a single circle with a marked center was used. It could benefit the estimation of the normal of its supporting

plane and location in both camera and range sensor. Then a point-to-plane optimization is carried out to recover the full camera calibration parameters. These approaches require auxiliary targets, and therefore cannot be applied to the in-situ application.

Instead of using designed target for registration, using features extracted from generally available objects, such as plane and line objects, for registration plays a more promising role. Liu et al. (2007) proposed a LiDAR-to-image registration approach that does not require any engineered targets. Their approach assumes the existence of at least two vanishing points. Their approach has six major steps. In the first step, all the possible pair of 3D and 2D lines are extracted. Then for each pair, the camera position is estimated and verified, and the potential pairs are stored for later use. If verifiable camera position is estimated, a weighted average representing the camera position is computed. Then the refined parameters are computed by finding the parameters that maximize the line overlapping. Tamas et al. (2013) proposed a target-less 2D-to-3D registration approach. Instead of measuring the reprojection error among correspondences, they convert the problem into integral problems. If correct registration parameters are obtained, the integral of projected points should close to the integral of image pixels of the same object after applying a non-linear function. Therefore, by using either one pair of corresponding plane, or two pairs of corresponding planes, the extrinsic parameters, or intrinsic and extrinsic parameters could be estimated. In addition to the registration between traditional digital image and range sensors, Cui et al. (2016) proposed a line-based registration framework that is able to register MLS point cloud data with panoramic image. In their approach, a spherical model is employed to describe the coordinate, and the line-based model is used to estimate the registration parameters.

6.2.2. Fusion in Post-disaster Response Application

Fusion of multiple sensors in emergency response applications has been paid great attention regarding aerial platforms, such as aerial/oblique imagery and ALS. Khoshelham et al. (2013) investigated the use of LiDAR and high-resolution satellite imagery in post-earthquake building

damage assessment. In the study, building extraction is first conducted, and then change detection is conducted on both imagery and LiDAR data to estimate the damage roughness index. Vetrivel et al. (2015) proposed a system that combines the point cloud and very-high-resolution oblique images. By fusing two data sources, their approach can classify the gaps presented in the image into true damage and artifacts of image acquisition. As a result, their approach can combine the advantages of both point cloud and imagery to improve the comprehensiveness and accuracy of building damage assessment. Recently, Vetrivel et al. (2017) proposed a general framework for damage detection. Their framework uses the combination and individual 2D image feature derived via CNN and 3D feature derived from point cloud. And an SVM classification procedure is conducted to detect the damage. The experiment shows promising performance of their approach. Although the fusion of imagery improves the level of detail in post-hazard assessment, the lack of ground-level perspective still hampers the comprehensive assessment procedure. For example, during storm surge events, building damage is likely to happen around the foundation, which is not visible in aerial imagery.

To address this, a more near-field data source, UAV-based image is adopted. Galarreta et al. (2015) applied UAV-based imagery to facilitate the assessment. As UAV could capture high-resolution images with large overlap, dense 3D point cloud could be generated from this image. The experiment shows that the multi-scale damage assessment could be conducted on the generated dense 3D point cloud. Zhou et al. (2015) used manual registration plus the iterative closest point (ICP) approach to accurately register the photogrammetry 3D model to the LiDAR point cloud. However, this process is tedious. Also, because any wrong correspondence may cause an unexpected error in registration, the correspondence picking has to be very concentrated, which makes it time-consuming.

These approaches, although proven successful in many general applications, have the following limitations in post-disaster scenarios. In terms of the fusion of aerial image to ALS data, the viewing perspective is downward or oblique with a great distance to the object, causing it

difficult to capture detailed condition about vertical facades. In terms of the fusion of MLS and their associated cameras, although detailed and high-resolution data can be captured, it is limited to capture data that are visible along the path of driving. Also, these approaches essentially capture LiDAR and image data at the same time. There are cases that the weather conditions during data collection are not favorable for taking images. Therefore, although high quality LiDAR data can be collected, the associated image data can be in poor quality. Last, but not the least, the disaster sites tend to be very dynamic as the time goes on. Comprehensive data collection would need multiple runs. Therefore, it is cost inefficient to collect multiple runs of data just to obtain high-quality image data. As an alternative solution, fusing images taken at different post-disaster phases with LiDAR data is a potential method to catch high-quality disaster data. However, current studies require either strong assumptions such as the existence of vanishing points or rely on extensive manual effort in annotating correspondences. These make them less ideal candidates for post-disaster applications.

To address these challenges and limitations, a novel efficient image to LiDAR registration approach was proposed in this study. In the proposed approach, LiDAR data are collected using an MLS equipped with multiple carefully calibrated cameras. The images taken by these fixed cameras are registered with the point cloud data through the classical stereo-pair approach. For a given street-level image, this system can efficiently register it with the point cloud via a couple of simple clicks and drags. The experiment shows that this approach can handle the cases that the images are taken at varied perspectives and lighting conditions. It is also shown that this approach can handle the case even the scene has changed.

6.3 Methodology

An overview of the proposed framework is presented in Figure 6.1. The framework has three major steps. The first step is the registration between the reference image and the point cloud model. Given an arbitrary street-level image, the second step is to retrieve the correspondence between this arbitrary image and the point cloud using either automated or semi-automated

approach. Within the correspondence been retrieved, the final step is the building damage assessment assisted by the registered street-level image. In this framework, the semi-automated registration approach is designed with minimum user interaction, which is designed for reducing the computational cost for the auxiliary matching extraction. Detailed descriptions of each step are given in the following sections.

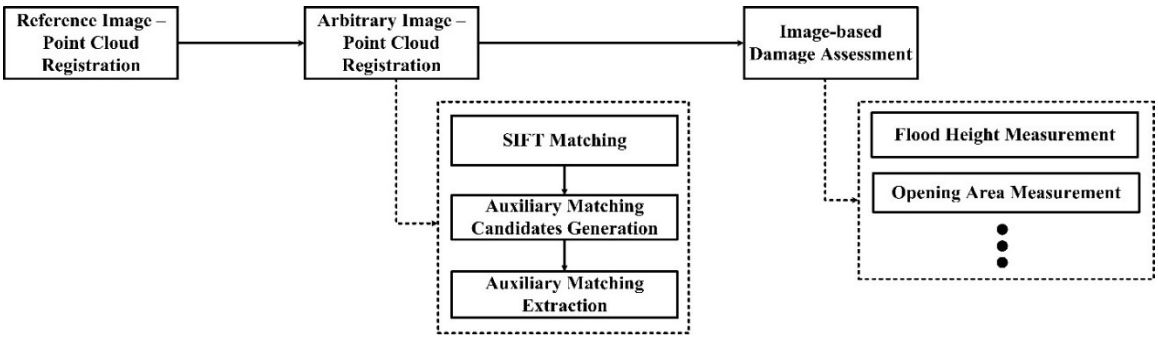


Figure 6.1. Overview of the Framework

6.3.1 Reference Image to Point Cloud Registration

In this research, I adopted a mobile LiDAR mapping system equipped with multiple rooftop cameras. The configuration of the data acquisition system is shown in Figure 6.2 (a). The system consists of a vehicle, a navigation system, a laser scanner, and a camera, where the relative geometric relationship between each pair of components remains fixed.

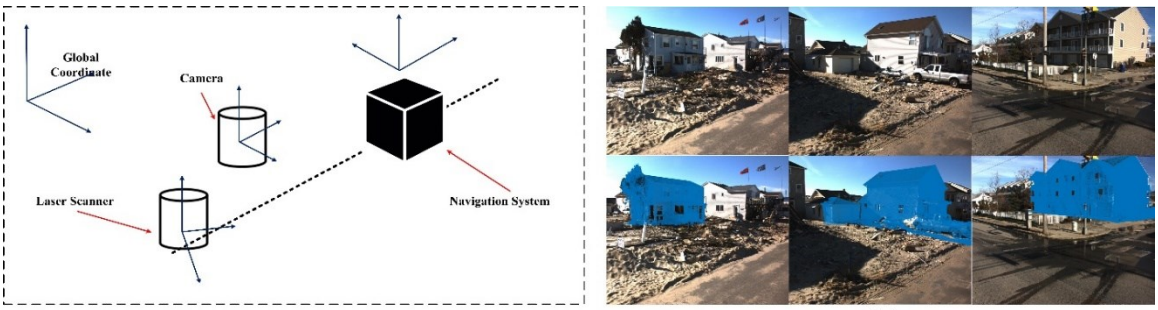


Figure 6.2. Data Acquisition System and Reference Image Registration

The camera calibration and the 3D-to-2D registration have been studied for decades, and very mature solution has been proposed by (Zhang 2000). The basic idea of 3D-to-2D registration is to solve the camera projection model of 11 parameters. The projection parameters could be decomposed from the camera projection matrix $\boldsymbol{\mathcal{M}} = \boldsymbol{A}[\boldsymbol{R}\ \boldsymbol{t}]$, where $\boldsymbol{A} = \begin{bmatrix} f_x & \gamma & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}$ is the intrinsic matrix of the camera model, $\boldsymbol{R} = \boldsymbol{R}_z(\theta_z) \cdot \boldsymbol{R}_y(\theta_y) \cdot \boldsymbol{R}_x(\theta_x)$ and $\boldsymbol{t} = [x, y, z]^T$ are the rotation matrix and translation vector, respectively.

For a pair of image and point cloud frame captured at the same timestamp, a set of corresponding points are manually extracted from the point cloud and image, respectively. Here, I denote these images taken by MLS-equipped camera as reference image, $\boldsymbol{I}^r$.Please be noted that these corresponding points need to be extracted from non-coplanar objects so that the full projection parameters could be recovered from single image. Because the relationship between the camera coordinate and the laser scanner coordinate is fixed throughout the entire mapping procedure, the projection matrix at one timestamp could be directly applied to other timestamps ideally. Due to synchronization error among different sensors, however, this projection matrix needs to be re-estimated every several timestamps, so that the system error will not be accumulated. Since the 3D-to-2D registration through correspondences is a standard solution, I will not talk about the detail because it is not the scope of this research. The readers are referred to (Quan et al. 1999, Triggs 1999, He 2012, Zheng et al. 2013, Ferraz et al. 2014) for state-of-the-art. Figure 6.2 (b) shows the back projection results of three point cloud datasets and the corresponding images captured at the synchronized timestamp. Because the objective of this research is to assess the building damage with the assistance of street-level imagery, the points of interested buildings are segmented out for simplicity. In Figure 6.2 (b), the upper row of images are the street-level images collected via the camera mounted on the mobile LiDAR system, and the lower row of the images are the overlay of back-projected points.

Once the projection between MLS-equipped camera and the point cloud is extracted, the correspondence between image and point cloud could be rapidly retrieved. Because imagery data provides more intuitive visualization, damage assessment procedure will benefit from the fusion of imagery and point cloud dataset. For example, damage assessors can easily identify wall opening damage or wall cladding failure from a street-level image. More important, it is possible to measure the watermark from the image, while it is significant harder, if not impossible, to do so from point cloud data. However, the MLS-equipped cameras suffer from the fact that the camera perspectives are restricted to the vehicle accessibility and what can be seen from the driving path. As a consequence, the image can only be taken from distance if the building is far away from the driving path. Also, the camera orientation is fixed so that the building object might be partially occluded.

Alternatively, there are a large number of street-level imagery captured in a crowd-sourced fashion or by ground survey personnel. Here I denote these image as target image, $\boldsymbol{I}^t$. Since those images are taken from arbitrary perspectives, these images tend to provide more detailed information in many cases. However, because these images are taken from unknown poses with unknown camera parameters, registration with LiDAR data is required for every single image in order to recover 3D information in the images. In the following section, I present an efficient image-to-point cloud registration approach. This approach does not require any manually selected correspondences, but only the reference image, $\boldsymbol{I}^r$, is used as a bridge.

6.3.2 Arbitrary Image to Point Cloud Registration

To register an arbitrary image, $\boldsymbol{I}^t$, to the point cloud, a Perspective-n-Point problem is required to be solved (Hartley et al. 2003), as shown in Figure 6.3 (a). For the simplicity of notation, I denote the PnP between $\boldsymbol{I}^r$ to the point cloud as $\mathbb{P}^r$, and the PnP between $\boldsymbol{I}^t$ to the point cloud as $\mathbb{P}^t$. However, this requires manual extraction of corresponding pairs of points, which is tedious and labor intensive. Because the correspondences $\mathbb{P}^r$ is known (single dot line in Figure 6.3 (b)),

instead of direct extraction of $\mathbb{P}^t$, I first extract the point-wise matching between $\boldsymbol{I^t}$ and $\boldsymbol{I^r}$(double dot line in Figure 6.3 (b)). Then this image-to-image correspondences could be transferred to retrieve $\mathbb{P}^t$. Therefore, this PnP problem is equivalent to image matching problem. However, I will show shortly that traditional pixel-wise feature matching approaches is not sufficiently robust for large perspective- and brightness-variant scenarios. In this section, therefore, a novel image-matching framework with minimum human-in-the-loop action is introduced to address the image matching with large perspective and brightness change.

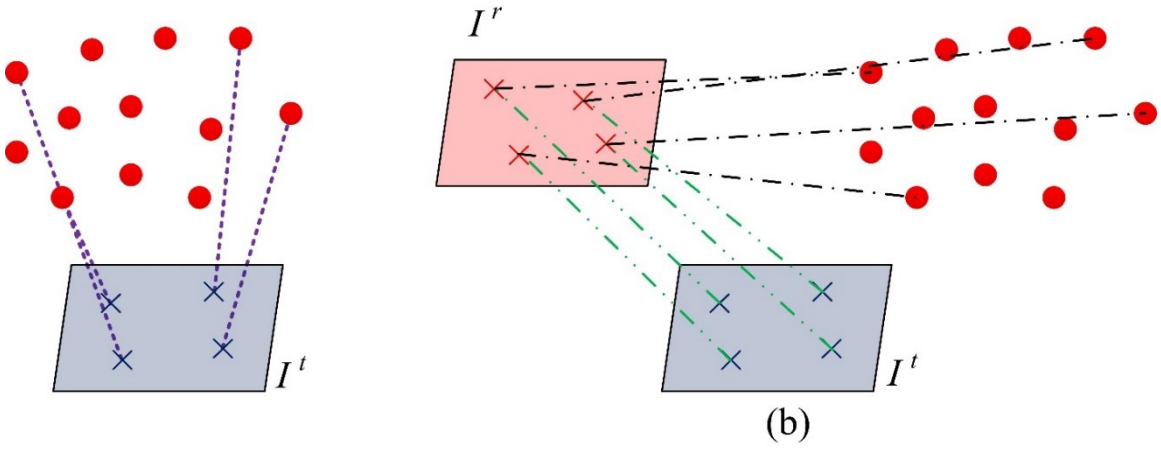


Figure 6.3. Abstraction of Image-to-PointCloud Registration

SIFT Matching: As the first step, the SIFT-based matching (Lowe 2004) is implemented to generate an initial matching. Because the arbitrary image $\boldsymbol{I^t}$ is likely taken from a different angle, time, and under different brightness condition. These, all, bring difficulty to the SIFT matching. Though taking image from various perspective, it is a rigid body transformation regarding to various camera locations. And as a consequence, for a corresponding point in $(\boldsymbol{I^r}, \boldsymbol{I^t})$, its surrounding pixels should have higher likelihood to match with each other (Bian et al. 2017). Unlike (Bian et al. 2017), I adopt a simple solution to extract dense SIFT matching candidates.

As shown in Figure 6.4, the proposed approach first detects SIFT descriptors $(\boldsymbol{P^r_{SIFT}}, \boldsymbol{P^t_{SIFT}})$ from both images, where $\boldsymbol{P^r_{SIFT}} \in \boldsymbol{I^r}$, and $\boldsymbol{P^t_{SIFT}} \in \boldsymbol{I^t}$. Each SIFT descriptor is a (1×128)

dimension vector. After this, the SIFT descriptors are matched by finding the mutual-unique match of each pair of $(\boldsymbol{P}_{SIFT}^{r}, \boldsymbol{P}_{SIFT}^{t})$. Specifically, for each descriptor $\boldsymbol{p}_i^r \in \boldsymbol{P}_{SIFT}^{r}$, I find its closest match of $\boldsymbol{p}_j^t \in \boldsymbol{P}_{SIFT}^{t}$. For the found descriptor p_j^t, if the closest match descriptors in $\boldsymbol{P}_{SIFT}^{r}$ is also p_i^r, I keep $(\boldsymbol{p}_i^r, \boldsymbol{p}_j^t)$ as a candidate of valid matching. After all the matches are founded, the invalid matches are removed via fitting homography matrix (Hartley et al. 2003). The valid matches after the homography matrix estimation are ususally sparse, especially when the images are taken with large perspective- and brightness-variance.

To generate dense SIFT candidates, for each pair of $(\boldsymbol{p}_i^r, \boldsymbol{p}_j^t)$, I detect the SIFT descriptors from their surrounding grid (G_i^r, G_j^t). Each grid is a $(n \times n)$ square centered at $\boldsymbol{p}_i$. Then I extract the matches of SIFT descriptors following the abovementioned approach. In this stage, I introduce another constrain other than the homography-based outlier removal. As shown in Figure 6.4 (b), for each set of matched descriptors, I estimate the homography matrix $\boldsymbol{\mathcal{H}}$. I assume that a valid homography will preserve the geometric relationship of pixels during transform. That means if a set of pixels are in clockwise sequence, then their affine transformed form will also be in clockwise sequence. If after affine transformation, a set of pixels located in clockwise sequence does not preserve the clockwise sequence, the homography is invalid. By introducing this auxiliary geometric constrain, I will be able to retrieve a set of correctly matched points denoted as $\bar{\mathbb{c}}$.

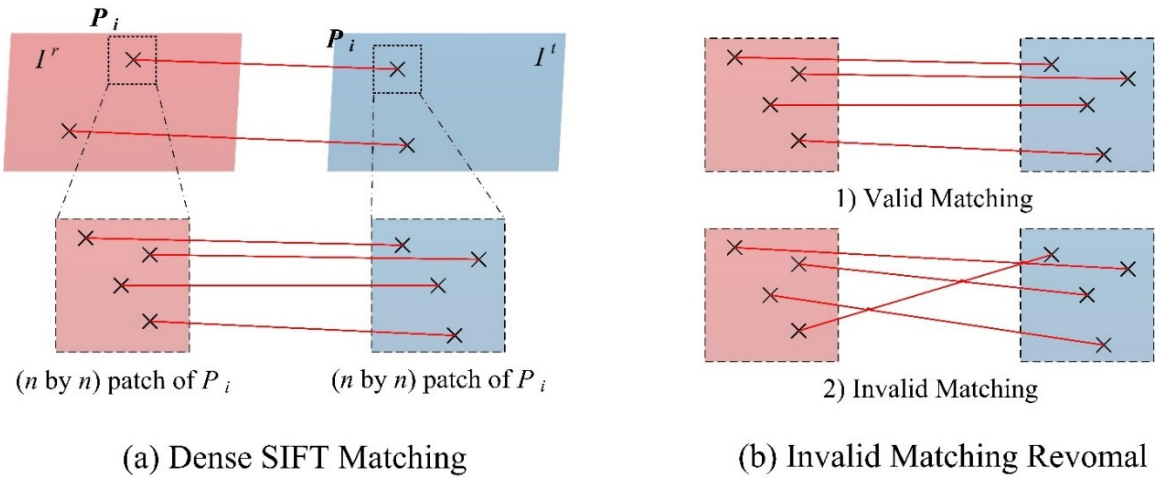


Figure 6.4. Dense SIFT Matching Generation

Auxiliary Matching Candidates Generation: Because SIFT matches are extracted by removing the outliers through estimation of homography matrix, the extracted matches are coplanar to a great extent. However, the traditional PnP problem needs a set of non-coplanar corresponding points to retrieve the camera parameters. Although a projection matrix can still be solved using an 8-points algorithm (Hartley et al. 2003), the solved projection parameters will not be sufficiently accurate to establish the precise corresponding $\mathbb{P}^t$. Although (Pandey et al. 2015) proposed a correspondence free approach to estimate the camera calibration parameters via maximizing the mutual information, the optimization is highly likely to stop at local minima due to the non-convexity of the problem. In addition, a bad initial solution will introduce great trouble to the optimization as well. In this research, therefore, a human-in-the-loop auxiliary matching generation approach is proposed to add more matching even with poor descriptor similarity.

To add auxiliary matching points, this solution is based on selecting a set of engineered auxiliary correspondences which maximizes the mutual information between the target image and the point cloud. In the proposed solution, the candidate matches are carefully generated so that a brute force searching strategy can be efficiently implemented to obtain an optimal matching. In

this research, the candidate matches are preferred to be corners of the building or points along edges. To extract these points, the image is first applied with L_0 smoothing (Xu et al. 2011) to remove the weak edges, such as the edge between wall panels (Figure 6.5 (b)). Because the edges of building structures show clear patterns, such as straight line, L-shape corner, etc. These local patterns of edge structures could be taken advantage in detecting real building boundaries or edges. Therefore, a sketch token plus random forest decision-making approach (Dollár et al. 2013) is implemented to compute the likelihood of pixel as building edges. This approach outputs the probability of each pixel assigned as building edges, the pixels along edges with large gradient magnitude have higher probability value, while the edges along edges with small gradient magnitude or on smooth region tend to have lower probability value (Figure 6.5 (c)). To improve the computational efficiency, a simple binarization is conducted, only the pixels with probability larger than a threshold value are set with value 1, and all other pixels are set to 0 (Figure 6.5 (d)). In this research, the threshold value is set to 0.3. Then the line segments are detected using Hough Transform (Duda et al. 1972, Matas et al. 2000, Fernandes et al. 2008). The Hough Transform-based line detection is able to extract sufficient number of pixel segments satisfy the line model. This will excessive line segments and, as a consequence, increase the computational cost. Therefore, only those line segments longer than a threshold value are selected (Figure 6.5 (e)). Then the auxiliary corner candidates are considered as the end vertices of each line segment. As shown in Figure 6.5 (f), most of the building corners have at least one detected auxiliary corners points, which indicates good candidates generation. As shown in Figure 6.6, a comparison of Harris (Harris et al. 1988) corners and mine is visually presented. As noticed, the Harris approach fails to detect the corners of building objects, such as windows, doors, etc. On the contrary, it extracts excessive corners at debris and sand dune region, which is considered as noise for the purpose of this research.

Auxiliary Matching Extraction: For a given pair of images $(\boldsymbol{I^r}, \boldsymbol{I^t})$, the auxiliary corner candidates are computed as $\boldsymbol{P_C^r} \in \boldsymbol{I^r}$, $\boldsymbol{P_C^t} \in \boldsymbol{I^t}$. To extract the correct match among $(\boldsymbol{P_C^r}, \boldsymbol{P_C^t})$, a

human-in-the-loop match detection approach is developed. As shown in Figure 6.7 (a), the proposed human-in-the-loop approach has five steps. The first step is the selection of region of interest ($\boldsymbol{R}^r, \boldsymbol{R}^t$) from the pair of image ($\boldsymbol{I}^r, \boldsymbol{I}^t$). In fact, this is the only step that human interaction is required for image registration. The SIFT-matching points are usually coplanar, therefore, the auxiliary matching is preferred to be detected from the regions that are not coplanar with them. In this stage, the users are suggested to crop out the region of building object roughly with a rectangle from ($\boldsymbol{I}^r, \boldsymbol{I}^t$) (Figure 6.7 (b)). After that, the selected rectangle is splited into multiple rectangle blocks ($R_i^r \in I^r, R_j^t \in I^t$), each covers partial of the building object. In the next step, the auxiliary corner points are assigned to each rectangle blocks they lying in.

For each corner point inside the RoI, the HoG feature (Dalal et al. 2005) is computed. Because there are numerous repeated patterns in a building object, a corner in one image might have multiple matches in another image. This leads to the estimation of many-to-many correspondence between corners in ($\boldsymbol{R}_i^r, \boldsymbol{R}_j^t, i = 1,2, \dots, j = 1,2, \dots$) via computing the square root of Euclidean distance (SRED) between two HoG feature vectors. Because a unique one-to-one matching is not robust in this stage, I propose a block-wise matching approach to narrow down the searching space and reduce the computational cost drastically. Specifically, for the corner points in each block $R_i^r \in I^r$, I match them to the corner points in every block $R_j^t \in I^t$, to reduce the number of matching, for the corner points in each pair of (R_i^r, R_j^t), I select the matchings with SRED smaller than a threshold value. For those pair (R_i^r, R_j^t) where no matching with SRED smaller than the threshold exist, I select the first 10 matchings with smallest SRED as the matching candidates of the pair of (R_i^r, R_j^t). As shown in Figure 6.7 (c), for each block in I^r, I match the corners inside the block with all the blocks in I^t. Denote a block-wise matching between R_i^r to all the $R_j^t \in I^t$ as $\mathcal{B}_i$, and the number of approximately matching points is $\mathcal{N}_i$, this would result N pairs of block-wise matching ($\mathcal{B}_i, \mathcal{N}_i$), where N is the number of blocks in I^r. Because the number of auxiliary matching required to robustly estimate the new projection is

unknown, I need to check all the possible combination of $\mathcal{B}_i$ and find the optimal matching. Specifically, I first check the case where only one auxiliary matching is needed, this would result $\sum_{i=1}^{N} \mathcal{N}_i$ possible matching. Then I check the case where two auxiliary matching are needed, this would result $\left(\sum_{i=1}^{\binom{N}{2}} \mathcal{N}_j \mathcal{N}_k, j = 1,2,..,N, k = 1,2,...,N, j \neq k\right)$ possible matchings, and so on. Because I split the building object into multiple rectangle blocks, I introduce a restriction that each block-wise matching $\mathcal{B}_i$ cannot have more than one valid auxiliary matching point. Therefore, by splitting the building object in I^r into N rectangle blocks, the total number of possible matching is computed as:

$$\sum_{i=1}^{N} \sum_{j=1}^{\binom{N}{i}} \{(\mathcal{N}_1)^{\mathcal{J}(1)} (\mathcal{N}_2)^{\mathcal{J}(2)} \dots (\mathcal{N}_N)^{\mathcal{J}(N)}\} \tag{6.1}$$

where $\binom{N}{i}$ is the number of i-combination, $\mathcal{N}_i$ is the number of approximate matching in $\mathcal{B}_i$, and $\mathcal{J}(k)$ is an indication function. It equals to one if k is in the *j*(th) i-combination, and equals to zero otherwise. By stacking all the possible approximate matching, I obtain the set of all possible matching, $\mathbb{C}$, detected using HoG feature, and which will be used in the following optimization step. As can be seen from equation (6.1), the value of all possible combination increases drastically, which will become infeasible if *N* is large. In this study, I choose *N* equals to 4.

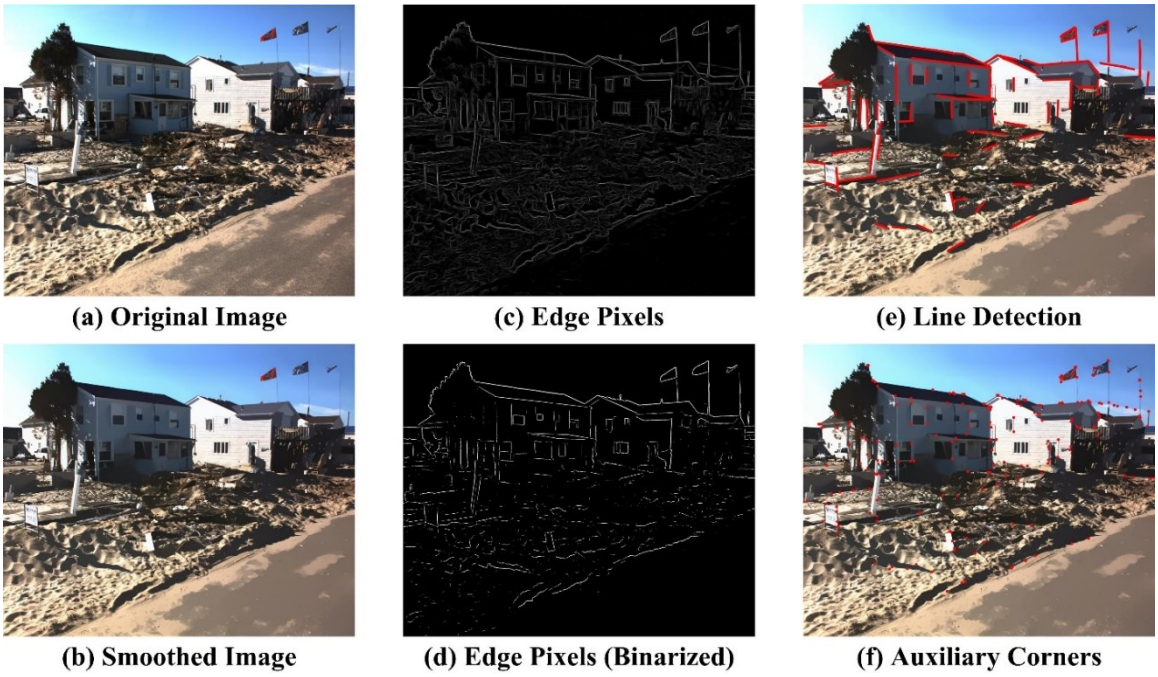


Figure 6.5. Auxiliary Corner Points Generation

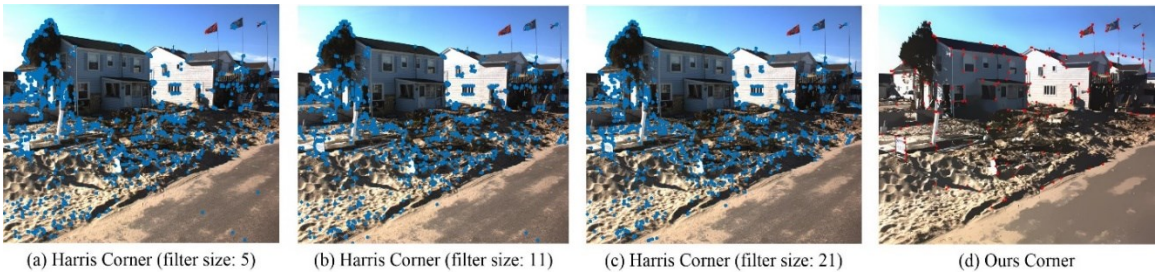


Figure 6.6. Comparison of Corner Detection between Harris and Mine

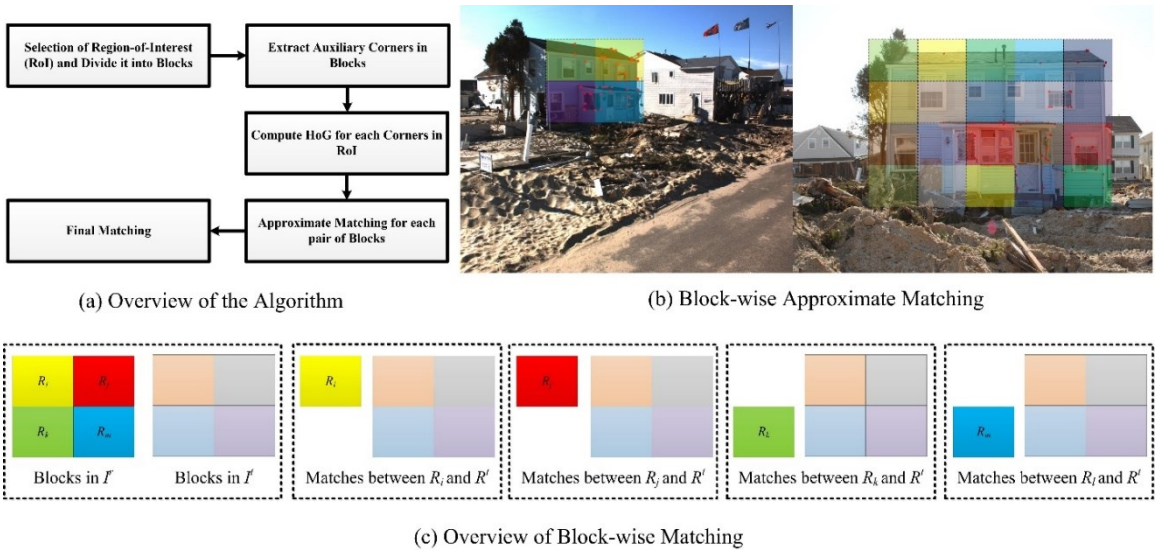


Figure 6.7. Auxiliary Matching Extraction

After the block-wise approximate matching is estimated, the final auxiliary match could be obtained by solving this optimization problem

$$\mathcal{F}(\tilde{\mathbb{c}}) = \max_{\mathbb{c}_i \in \mathbb{C}} MI(\boldsymbol{I}^t, \boldsymbol{I}^p) \tag{6.2}$$

In equation (6.2), $MI(\boldsymbol{I}^t, \boldsymbol{I}^p)$ is a standard mutual information function (Cover et al. 2012),

$$MI(X,Y) = H(X) + H(Y) - H(X,Y) \tag{6.3}$$

$$H(X) = -\sum_{x \in X} p(x)\, log\, p(x) \tag{6.4}$$

$$H(X,Y) = -\sum_{x \in X}\sum_{y \in Y} p(x,y)\, log\, p(x,y) \tag{6.5}$$

where $\boldsymbol{I}^t$ is the target image, and $\boldsymbol{I}^p$ is the image generated by projecting the point cloud to an image plane. $\mathbb{C}$ is the set of all possible matching detected using HoG feature, and $\tilde{\mathbb{c}}$ is the optimal matching. For this problem, $\boldsymbol{I}^t$ is constant, and $\boldsymbol{I}^p$ is a function of the point cloud $\boldsymbol{X}$ and the projection matrix $\boldsymbol{\mathcal{M}} = \boldsymbol{A}[\boldsymbol{R}\ \boldsymbol{t}]$. Because the point cloud is also given, for each $\mathbb{c}_i \in \mathbb{C}$, the only changed term is the projection matrix $\boldsymbol{\mathcal{M}}$. For each $\mathbb{c}_i \in \mathbb{C}$, I define $\mathbb{c}' = [\bar{\mathbb{c}}, \mathbb{c}_i]^T$ as the new matching. As the correspondence between $\boldsymbol{I}^r$ and point cloud $\boldsymbol{X}$ is already known, this correspondence $\mathbb{P}^r$ could be easily employed to transfer to the correspondences between target image and point cloud $\mathbb{P}^t$. Using Zhang's method (Zhang 2000), the projection matrix $\boldsymbol{\mathcal{M}}$ could

be estimated. Then the image $\boldsymbol{I}^p$ is generated via projecting the point cloud to image plane applying $\boldsymbol{\mathcal{M}}$. Then the mutual information of $(\boldsymbol{I}^t, \boldsymbol{I}^p)$ is computed. As shown in Algorithm 6.1, this process is repeated for all ($\mathbb{c}_i \in \mathbb{C}$), and the optimal $\tilde{\mathbb{c}}$ is solved by maximizing equation (6.2). Figure 6.8 shows the result of auxiliary matching extraction procedure. Figure 6.8 (a) presents the mutual information respect to each auxiliary match, and Figure 6.8 (b) shows the optimal matching by solving equation (6.1). As can be seen, the algorithm finds two auxiliary matching points. One match is obtained at the corner of the rooftop, and another match is obtained at the corner of the front door. Figure 6.8 (c) shows the generated images from point cloud with different projection matrix $\boldsymbol{\mathcal{M}}$ obtained from each possible block-wise matching combination.

Algorithm 6.1. Auxiliary Matching Extraction

Input: $\boldsymbol{I}^r, \boldsymbol{I}^t, \boldsymbol{X}, \left(\mathbb{P}_{ij}^r \in \mathbb{P}^r \middle| \mathbb{P}_{ij}^r : p_i^r \leftrightarrow X_j\right), (\mathbb{c}_i \in \mathbb{C} \,|\mathbb{c}_i : p_l^r \leftrightarrow p_k^t), (\bar{\mathbb{c}}_i \in \bar{\mathbb{c}} \,|\bar{\mathbb{c}}_i : p_i^r \leftrightarrow p_i^t), \widetilde{mi} = 0$

Output: $\tilde{\mathbb{c}}$

for $\bar{\mathbb{c}}_i$ in $\bar{\mathbb{c}}$

  $\mathbb{c}' \leftarrow [\bar{\mathbb{c}}, \mathbb{c}_i]^T, \mathbb{P}^t \leftarrow \left(X_i,\ p_i^t\right), \boldsymbol{\mathcal{M}} = f\left(X_i,\ p_i^t\right), \boldsymbol{I}^p = g(\boldsymbol{M}, \boldsymbol{X})$

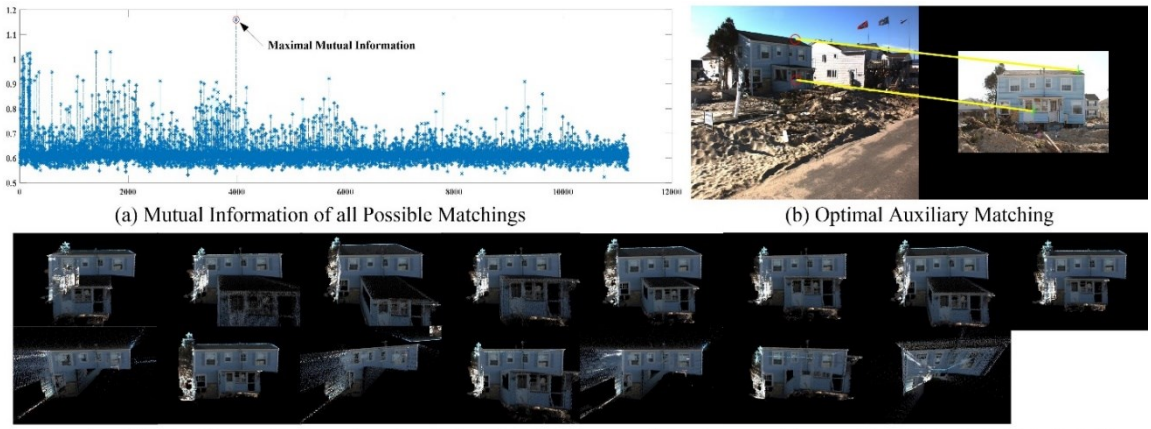
  $mi_i = MI(\boldsymbol{I}^t, \boldsymbol{I}^p)$

  if $\widetilde{mi} < mi_i$

    $\widetilde{mi} \leftarrow mi_i, \tilde{\mathbb{c}} \leftarrow \mathbb{c}_i$

  end if

end if



(a) Mutual Information of all Possible Matchings (b) Optimal Auxiliary Matching

(c) Results of Maximal Mutual Information of each Combination, from top-left to bottom-right are: $\{B_1\}$, $\{B_2\}$, $\{B_3\}$,$\{B_4\}$,$\{B_1\}$,$\{B_3, B_4\}$,$\{B_2, B_4\}$,$\{B_2, B_3\}$,$\{B_1, B_4\}$,$\{B_1, B_3\}$,$\{B_1, B_2\}$,$\{B_1, B_2, B_3\}$,$\{B_1, B_2, B_4\}$,$\{B_1, B_3, B_4\}$,$\{B_2, B_3, B_4\}$,$\{B_1, B_2, B_3, B_4\}$

Figure 6.8. Optimal Auxiliary Extraction

6.3.3 Image-based Damage Assessment

The building component level damage could be measured and evaluated using the assistance of street-level imagery. As shown in Figure 6.1, the street level image could help with the evaluation of building damage in many aspects. In this research, I focus on the assessment of building opening size measurement and flood height measurement. The following will describe each of them in detail.

Point Cloud Pre-processing: Before the image-based damage evaluation, the pre-processing is conducted to segment each planar component of the building. Plane segmentation is a major research area in point cloud processing, and the segmentation approaches could be categorized into RANSAC-based, Smoothness constraint-based, and Hough transform-based (Rabbani et al. 2006, Pu et al. 2009, Deschaud et al. 2010, Hackel et al. 2016, Li et al. 2017). Because the point cloud segmentation is not the scope of this research, the technical detail will not be discussed here. In this approach, the smoothness constraint-based approach is applied.

Flood Height Measurement: The measurement of flood height is critical for post-hurricane or post-storm surge hazard assessment. Because the point cloud cannot be acquired until the flood ebbs. Which makes it impossible to identify the watermark from LiDAR data as the point cloud

intensity won't change much. Luckily, the watermark could be identified from the high-resolution street-level image. Because the arbitrary image to point cloud registration has been described in the previous section, it is assumed that the correspondence between arbitrary image and point cloud, $\mathbb{P}^t$, is known for the following sections.

To identify watermark from point cloud, a human-in-the-loop process is developed. The users are encouraged to select a straight line via simply picking two points, $(\boldsymbol{p}', \boldsymbol{p}'')$, as the end vertices of the line segment. Therefore, the straight line is denoted as $\ell(\boldsymbol{p}', \boldsymbol{p}'')$ for notation simplicity. This straight line should be close to the watermark in the image. Within the end points selected, the pixels along the line segment are extracted as

$$\{\boldsymbol{p_i} \in \boldsymbol{I}^t \mid f(\boldsymbol{p}_i, \boldsymbol{p}', \boldsymbol{p}'') < 0, g(\boldsymbol{p}_i, \ell) < \delta, i = 1,2, \dots\} \tag{6.6}$$

In equation (6.6), $f(\boldsymbol{p}_i, \boldsymbol{p}', \boldsymbol{p}'') = (\boldsymbol{p}' - \boldsymbol{p}_i) \cdot (\boldsymbol{p}'' - \boldsymbol{p}_i)$ represents the angle of $\angle \boldsymbol{p}_i \boldsymbol{p}' \boldsymbol{p}''$. And $g(\boldsymbol{p}_i, \ell)$ is the orthogonal distance from $\boldsymbol{p}_i$ to $\ell(\boldsymbol{p}', \boldsymbol{p}'')$, which is expressed as $g(\boldsymbol{p}_i, \ell) = \|(\boldsymbol{p}_i - \boldsymbol{p}') \cdot \boldsymbol{n}\|_1 / \|\boldsymbol{n}\|_2$, where $\boldsymbol{n}$ is an arbitrary length vector satisfying $\boldsymbol{n} \cdot (\boldsymbol{p}' - \boldsymbol{p}'') = 0$. The use of threshold value δ is because the coordinate of $\boldsymbol{p}_i$ can only take positive integer, while its orthogonal projection on ℓ might be decimal. The visual explanation of equation (6.6) is shown in (Figure 6.9).

Once the pixels along $\ell(\boldsymbol{p}', \boldsymbol{p}'')$ are extracted, the corresponding points in the point cloud could be rapidly retrieved. Although the manual selecting watermark line may not be parallel to the actual watermark in the image, this could be handled in point cloud efficiently. Because the static flood level is always horizontal, the 3D watermark could be estimated via finding a horizontal line that best fits the points.

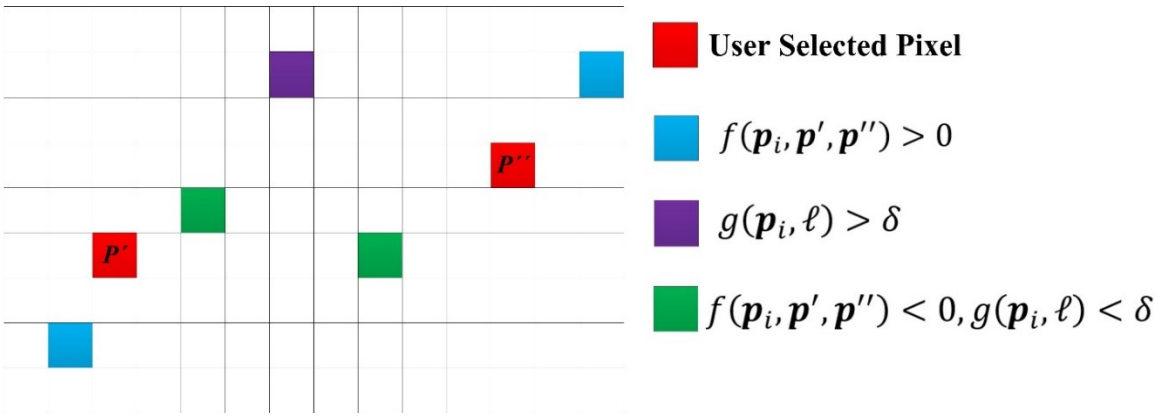


Figure 6.9. Extraction of Pixels along Two Pixels

Opening Size Measurement: A human-in-the-loop opening area measurement procedure is described in this section. As an opening is, equivalently, an empty space enclosed by a set of points distributed along its boundary. It is, therefore, critical to extract the boundary points of an opening area. To select the region-of-interest from the image, the approach presented in previous section (flood height measurement) is implemented. Because the boundary of an arbitrary opening could be approximately considered as a convex/concave hull, the sequence of the picked vertex is of importance. Although the users could select each vertex of the opening area in clockwise or counter-clockwise sequence, a concave polygon generation algorithm (Moreira et al. 2007) is implemented here. So that users could select vertex in any sequence. In this research, the users can specify opening in multiple planes at once, and the proposed approach is able to compute the size and area of opening in each plane. For each pair of sequentially consecutive vertex, the points along the line segment are extracted from the corresponding plane. After all line segments are visited, they are grouped according to their corresponding segment, and the evaluation is conducted on each group, respectively. If an opening area crosses multiple planes (polygon A⋯G in Figure 6.10 (a)), the line segment along the intersection of two non-coplanar planes is missed ($\ell(C, F)$in Figure 6.10 (b)), this is because the concave hull algorithm will only connect the points along the compact boundary of the polygon. To deal with this issue, the line

segment is generated by intersecting the two planes the opening area crosses, as the dot line in Figure 6.10 (c). After the missing line segment is estimated, there are two cases. The first case is that there is no points along the estimated line ℓ, and I can use the estimated straight line directly. Another case is that there are points along ℓ. For this case, the points along ℓ should be extracted.

In addition, if a line segment belongs to two plane ($\ell(B, C)$ in Figure 6.10), it is a shared boundary line, and it should be assigned to one of the adjacent planes. To do this, I sort the intersection points along this shared line segment from either top to bottom, or from bottom to top. For notation simplicity, denote the shared line as $\tilde{\ell}$, and all the line segments that intersect with $\tilde{\ell}$ as ℓ_i^k, where i denotes the index of line segment, and k denotes the index of plane they belong to, and the intersection point as $\boldsymbol{x}_i(\tilde{\ell}, \ell_i^k)$. If the sequence of sort is from top to bottom, I assign $\tilde{\ell}$ to plane k if $\boldsymbol{x}_i(\tilde{\ell}, \ell_i^k)$ is the topmost point. If the sequence is from bottom to top, assign $\tilde{\ell}$ to k if $\boldsymbol{x}_i(\tilde{\ell}, \ell_i^k)$ is the bottommost point.

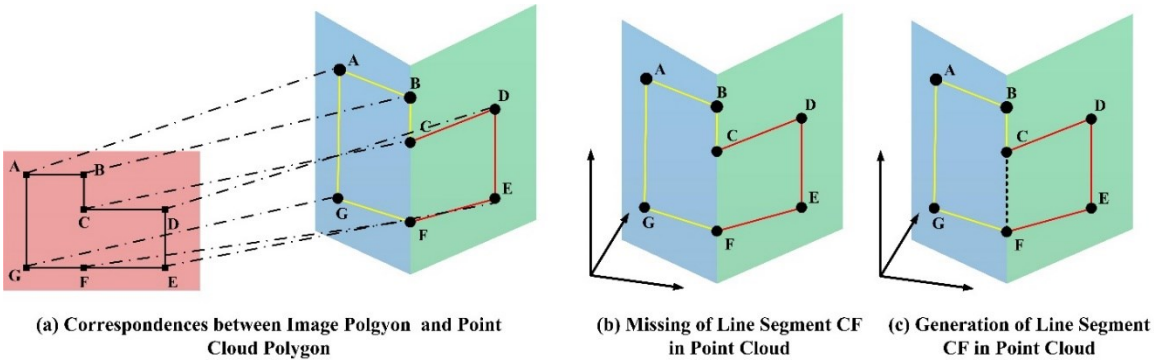


Figure 6.10. Concave Polygon in Point Cloud

For the polygon on each plane, the interior boundary points are extracted by alpha-shape algorithm (Edelsbrunner et al. 1994). Alpha-shape algorithm uses a circle (in 2D case) with radius α rolling around its interior hull, and extracts the points on its arc as the boundary points. The area of the opening region is computed as the area of concave polygon of enclosed by the interior boundary points (Figure 6.11). And the perimeter of the opening region is computed as the summation of the line segment of each pair of consecutively connected boundary points.

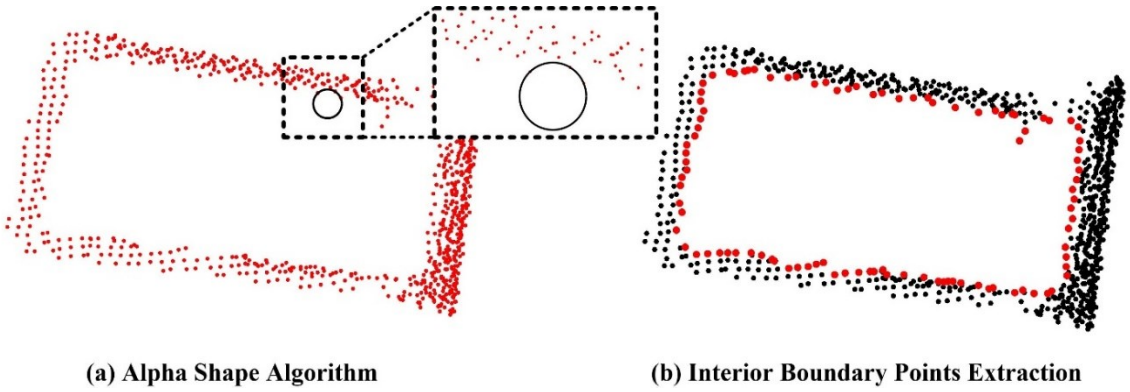


Figure 6.11. Interior Boundary Points Extraction

6.4 Experiment

6.4.1 Data Description

An experiment study is conducted on real damage data. The data was collected during Hurricane Sandy at shoreline area of New Jersey. A mobile LiDAR system equipped with a panoramic camera was used to capture the point cloud and image data. Because the system is carefully calibrated, the point cloud could be projected back to image accurately. In addition, an on-site survey teamed by Rutgers University, Princeton University, and University of Notre Dame was conducted shortly after the event. During the survey, geotagged high-resolution images were collected to cover more information as well. Figure 6.12 shows the data collected during hurricane Sandy. Figure 6.12 (a) is the overview of the mobile LiDAR data, Figure 6.12 (b) is the image captured by the panoramic camera equipped on the mobile LiDAR system of three particular buildings, and Figure 6.12 (c) is the images captured by on-site survey team at various perspectives. In this research, these three buildings are selected as examples as they represent three typical scenarios in post-hazard response.

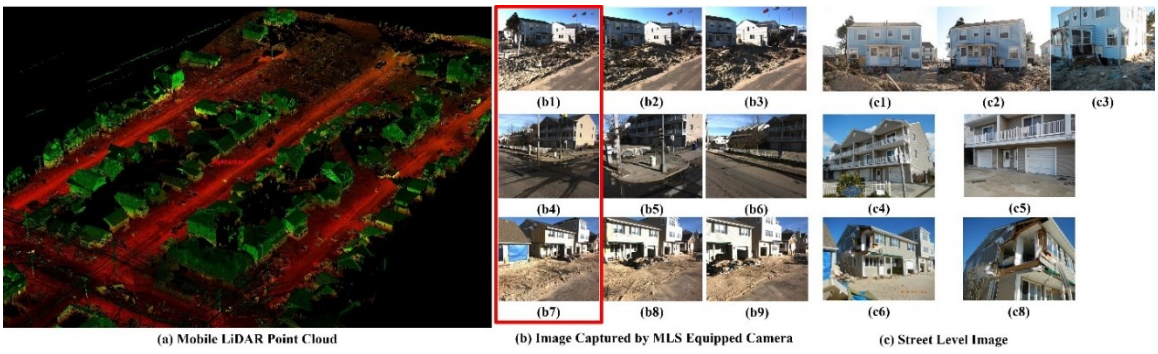


Figure 6.12. LiDAR and Imagery Data Collected during Hurricane Sandy

6.4.2 Human-in-the-Loop Image Registration

For each ground-level image captured by on-site survey team, the corresponding building is firstly identified via GPS information. For each set of $\boldsymbol{I}^r$ corresponding to the same building, the one that covers the building mostly is chosen. In this experiment, the chosen images $\boldsymbol{I}^r$ are bounded in the red rectangle as shown in Figure 6.12 (b). For the on-site collected images (Figure 6.12 (c1), (c3), (c5)), denote them as $\boldsymbol{I}_1^t$ for the convenience of notation. The registration could be efficiently conducted with the help of $\boldsymbol{I}^r$ respectively. The results of corresponding extraction are shown in Figure 6.13 (a)-(c). As can be noticed, the proposed framework described in Section 6.3.2 is able to retrieve dense corresponding using 1 or 2 pairs of auxiliary correspondences with human-in-the-loop assistance. However, for the images in Figure 6.12 (c2), (c3), (c5), and (c7), denote them as $\boldsymbol{I}_2^t$, the proposed approach fails to extract sufficient correspondences directly from $\boldsymbol{I}^r$. This is because the camera perspectives of $\boldsymbol{I}_2^t$ vary significantly from that of $\boldsymbol{I}^r$, after the constraint described in Section 3.2.1 is applied, only a limited pairs of correspondences are considered as valid. In addition, the detected correspondences are distributed closely (Figure 6.13 (d)), which is not a good correspondence for 2D-to-3D registration. In order to estimate all the camera parameters, more pairs of auxiliary correspondences need to be extracted, which increases the computational cost quadratically. In fact, the proposed approach is still able to handle this situation. If denote $\mathcal{G}(A, B) = A \leftrightarrow B$ as the correspondence between dataset A and B, where A, B

could be either image or point cloud, $\mathcal{G}(\boldsymbol{I}^r, \boldsymbol{I}_2^t)$ could be derived sequentially as $\{\mathcal{G}(\boldsymbol{I}^r, \boldsymbol{I}_1^t), \mathcal{G}(\boldsymbol{I}_1^t, \boldsymbol{I}_2^t)\}$. By defining this chain-like operation, dense correspondences could be extracted as shown in Figure 6.13 (e)-(h). However, a mismatching is identified in Figure 6.13 (g), this is because the image is occluded by vegetation, which will be discussed in detail shortly in section 6.4.4.

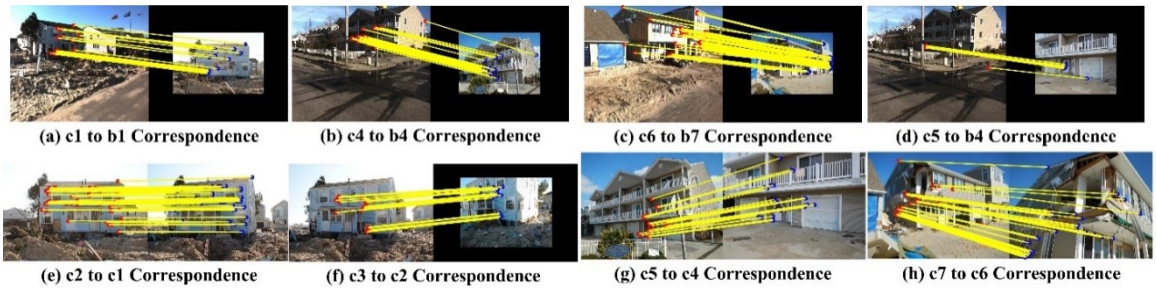


Figure 6.13. Image to Point Cloud Registration Result

The accuracy of registration is evaluated by back projecting the points to the image plane. If the registration is strictly correct, the 3D building model will be projected onto the image plane correctly, with every point projected on the one of the building pixels. Otherwise, partial 3D building point will be projected out of the building object in image. This two intuitive observations are adopted here as the accuracy evaluation criteria. For each building objects, the selected regions for the evaluation of registration accuracy are shown in Figure 6.14. In total, 15 region-of-interests are selected for evaluation. The evaluation is carried out according to the overlap ratio. The overlap ratio is computed as the area of concave polygon of projected points to the area of concave polygon of pixels inside RoI. To quantitatively analyze the accuracy, I define four terms to quantify the accuracy of back-projection as:

1) $\mathcal{O}_1 = \tilde{A}_{pt}/A_{pt}$,

2) $\mathcal{O}_2 = (A_{pt} - \tilde{A}_{pt})/A_{pt}$,

3) $\mathcal{O}_3 = \tilde{A}_{px}/A_{px}$, and

4) $\mathcal{O}_4 = (A_{px} - \tilde{A}_{px})/A_{px}$.

In these terms, A_{px} is the area of concave polygon of the image RoI, A_{pt} is the area of the point cloud RoI, and $\tilde{A}_{pt}$ is the area of point cloud that could be correctly projected to image RoI, and $\tilde{A}_{px}$ is the area of image RoI that has points projected on. The reason that different denominator (A_{pt}, A_{px}) are used for the evaluation is because the area of point cloud and area of image pixels are in difference space. The result of registration accuracy according to the ratio of overlap is shown in Table 6.1. It can be observed from the table that the overlap accuracy for rooftop regions is slightly lower than that of front façade, this is because the dense matching points are mainly extracted at front façade, while the matching at rooftop is extracted via auxiliary matching approach.

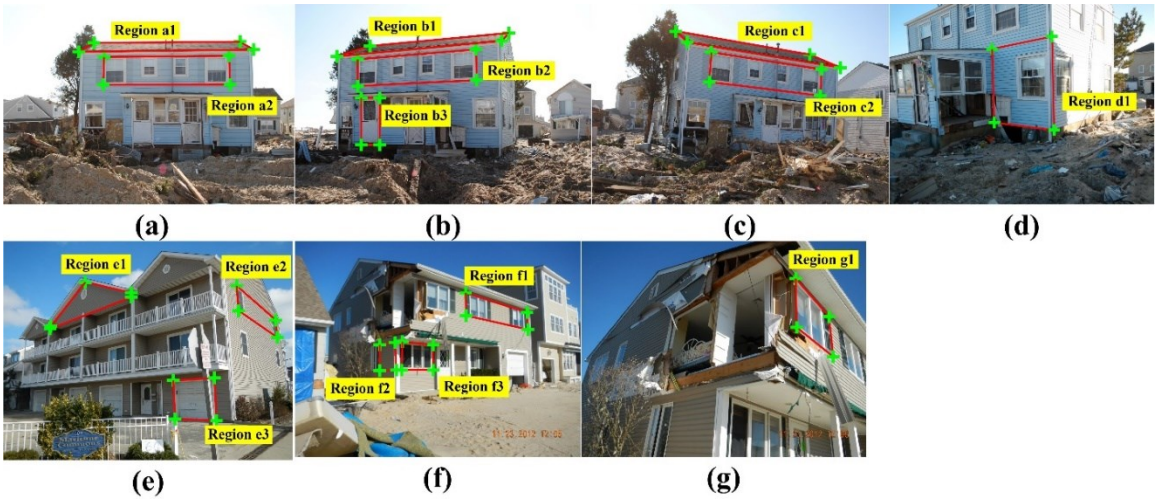


Figure 6.14. Region-of-Interest for Evaluation of Registration Accuracy

Table 6.1. Registration Accuracy Results (RoI)

Region ID	$\mathcal{O}_1$	$\mathcal{O}_2$	$\mathcal{O}_3$	$\mathcal{O}_4$	Region ID	$\mathcal{O}_1$	$\mathcal{O}_2$	$\mathcal{O}_3$	$\mathcal{O}_4$
a1	0.915	0.085	0.956	0.044	e1	0.961	0.039	0.981	0.019
a2	0.959	0.041	0.965	0.035	e2	0.993	0.007	0.975	0.025
b1	0.922	0.078	0.952	0.048	e3	0.738	0.262	0.922	0.078
b2	0.953	0.047	0.967	0.033	f1	0.981	0.019	0.997	0.003
b3	1.000	0.000	0.922	0.078	f2	0.933	0.067	0.917	0.083
c1	0.892	0.108	0.977	0.023	f3	0.888	0.112	0.543	0.457
c2	0.964	0.036	0.954	0.046	g1	1.000	0.000	0.974	0.026
d1	0.934	0.066	0.915	0.085					

In addition, I also evaluate the accuracy via measuring the overlap of line-of-interest (LoI) in point cloud and images. A pair of corresponding line segment is firstly identified in both point cloud and image. The projected line segment and the 2D line segment will be close to overlay if the registration is accurate. Based on this observation, 21 line segments are selected from both images (Figure 6.15) and LiDAR data for the purpose of evaluation. Denote each line segment via its end vertex and the orientation, I define the following three terms as:

1) $\mathcal{C}_1 = \|\boldsymbol{\mathcal{L}}_{pt}\|_2 / \|\boldsymbol{\mathcal{L}}_{px}\|_2$ is the length ratio between projected line and image line,

2) $\mathcal{C}_2 = (\boldsymbol{\mathcal{L}}_{pt} \cdot \boldsymbol{\mathcal{L}}_{px}) / \left(\|\boldsymbol{\mathcal{L}}_{pt}\|_2 \|\boldsymbol{\mathcal{L}}_{px}\|_2\right)$ is the angle of projected line and image line segment computed as the dot product of two vectors,

3) $\mathcal{C}_3 = \min\{x_{pt}, x_{px}\}$ is the minimum distance between the corresponding projected line vertex and image line vertex, and

4) $\mathcal{C}_4 = \max\{x_{pt}, x_{px}\}$ is the maximum distance between the corresponding projected line vertex and image line vertex, respectively.

In these four terms, $\boldsymbol{\mathcal{L}}_{pt}$ is the projected line segment, $\boldsymbol{\mathcal{L}}_{px}$ is the image line segment, x_{pt} is the vertex of projected line segment, and x_{px} is the vertex of image line segment. The unit of $\mathcal{C}_3$ and $\mathcal{C}_4$ is in pixel. The result of LoI overlap accuracy is presented in

Table *6.2*. It can be noticed that the length ratio and angle between projected line and image line is close to 1 for most of the LoI, and the distance between corresponding line vertex is approximately 10 pixels, considering the image size is larger than 1000 pixels, this re-projection error is around 1% of the image size. However, I noticed that these four terms go larger for d1, d2, and d3, this is because the registration of Figure 6.15 (d) to the point cloud is obtained via the chain operation $\{\mathcal{G}(\boldsymbol{I}^r, \boldsymbol{I}^a), \mathcal{G}(\boldsymbol{I}^a, \boldsymbol{I}^b), \mathcal{G}(\boldsymbol{I}^b, \boldsymbol{I}^d)\}$, where the registration error is accumulated three times.

Figure 6.15. Line-of-Interest for Evaluation of Registration Accuracy

Table 6.2. Registration Accuracy Results (LoI)

Line ID	$\mathcal{C}_1$	$\mathcal{C}_2$	$\mathcal{C}_3$	$\mathcal{C}_4$	Line ID	$\mathcal{C}_1$	$\mathcal{C}_2$	$\mathcal{C}_3$	$\mathcal{C}_4$
a1	1.007	1.000	2.785	9.041	d3	1.224	1.000	28.195	47.394
a2	1.100	1.000	4.236	4.588	e1	1.041	1.000	8.913	34.997
a3	0.964	1.000	5.767	10.793	e2	1.031	1.000	6.403	8.261
b1	1.010	1.000	2.193	8.505	e3	1.106	1.000	12.316	33.020
b2	1.108	0.999	3.458	6.806	e4	0.997	1.000	11.000	11.064
b3	0.961	1.000	8.378	12.981	f1	0.987	1.000	2.721	12.089
c1	0.995	1.000	0.538	6.947	f2	1.025	1.000	2.695	10.400
c2	1.068	0.999	7.837	10.725	f3	1.009	1.000	4.012	4.605
c3	0.960	0.999	9.359	13.361	g1	1.027	0.999	5.142	9.717
d1	1.055	0.981	4.581	66.410	g2	0.966	1.000	4.977	9.717
d2	0.830	0.996	13.118	27.203					

6.4.3 Building Damage Assessment

In this experiment, three damage condition assessments are conducted. For building Figure 6.16 (a), I estimate the flood height from the image, for building Figure 6.16 (b), I measure the area of the opening region at damage wall, and for building Figure 6.16 (c), I measure the area of the opening region at damage wall. The difference between damage at building Figure 6.16 (b) and Figure 6.16 (c) is, the opening area at building Figure 6.16 (c) was covered by temporary wood panel during the collection of LiDAR data, while it was exposed during on-site survey. Therefore, the image collected during ground survey could be used to measure the actual area of damage region. The extracted points at the height of watermarks are shown in Figure 6.16 (a2), and the estimated height of the watermark is given in Table 6.3. As can be seen, the height

difference between the two watermark is 2(cm). Table 6.4 presents the results of opening area estimation and the perimeter estimation. The experiment shows that the proposed approach makes the full advantage of multi-source data to evaluate the building condition. As can be seen, for building (b), the image collected by the MLS-equipped camera is far away from the building objects, therefore, is not an ideal data source for post-hazard respond personnel to revisit very detail damage information. While this could be solved via using street-level image collected close to the object. Another typical scenario is shown in Figure 6.16 (c) and Figure 6.16 (c1). Because the mobile LiDAR collection can be conducted one or two weeks after the hazard because of accessibility and entrance permit issue, some damaged area might have been temporarily covered by the homeowners. However, since street level image could be collected shortly after the event, which makes it a good complementary source to evaluate these damage objects.

Table 6.3. Result of Flood Height

	Watermark 1	Watermark 2
Flood Height (m)	3.25	3.23

Table 6.4. Result of Opening Area and Perimeter Estimation

	Building b1		Building c1	
	Opening 1	Opening 2	Opening 1	Opening 2
Area of Opening (m^2)	2.831	6.875	1.475	6.734
Perimeter of Opening (m)	0.816	3.747	6.657	10.602

Figure 6.16. Building Damage Assessment Results

6.4.4. Failure Analysis and Discussion

As stated in Section 4.2, a registration error is identified for building Figure 6.17 (a). Although dense matching points are extracted and the re-projection matches locally (Figure 6.17 (b)), the registration is not correct. Because the building has repeated patterns, the feature points in target image are matched to another patch of points in the reference image. As can be seen in Figure 6.17 (c), the garage door in target image is matched to the rightmost garage door in the reference image, while its correct matching is the garage door second to the left, as shown in Figure 6.17 (d). In addition to the repeated patterns that brings ambiguity, the failure is in part because the correct garage in reference image is occluded by bush. In either way, the experiment suggests that it has higher likelihood the registration will fail to match the correct correspondence if one of the images only covers local and repeated patterns.

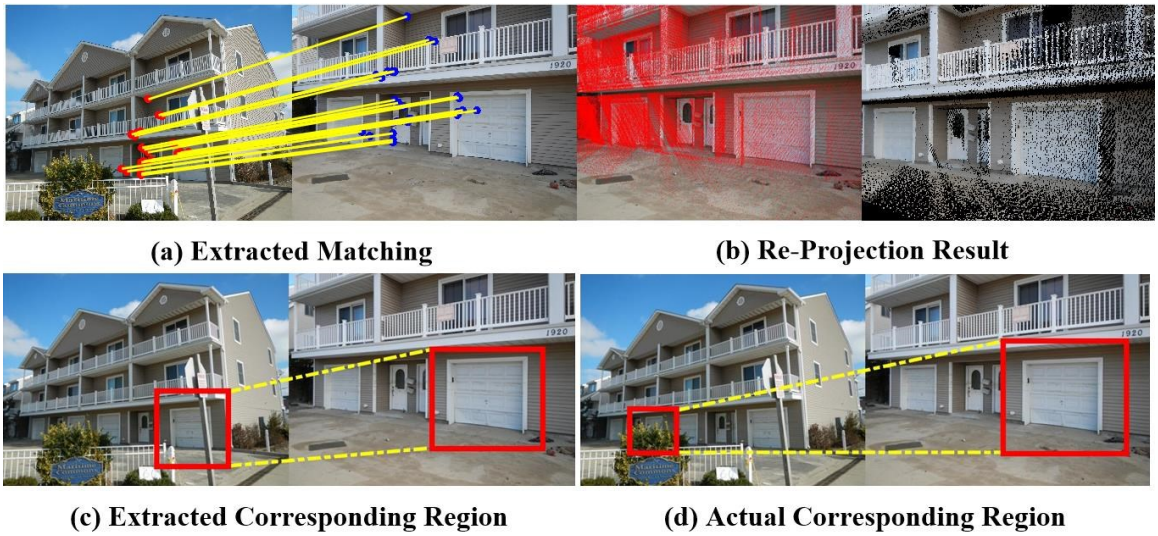


Figure 6.17. Failure Matching

6.5 Conclusion

In this research, a comprehensive framework of building damage assessment using the fusion of street-level imagery and point cloud with human-in-the-loop is proposed. The input of the framework is an arbitrary street level image and a set of registered point cloud and imagery data. Firstly, the arbitrary image is registered with the point cloud with minimum user interaction. Then the building damage condition could be assessed directly from the image. In this research, the flood height measurement and damage opening area estimation are presented in detail, and the experiment shows reliable performance. I believe the proposed framework could be extended to the assessment of other types of damage patterns.

: **Conclusions, Limitations and Future Work**The previous chapters described the proposed computer vision-based approaches for building damage assessment during major hurricane events. Validations of individual modules for processing various types of hurricane damage data are presented in each chapter. To understand how the integration of these varied approaches can be used to enhance or in some case transforms existing damage assessment approaches, this chapter focuses on a comprehensive evaluation of the proposed damage assessment approaches at the system level. More specifically, the study aimed at characterizing how the damage data including airborne/terrestrial lidar data and street-level imagery data can be gradually processed by the proposed approaches to extract damage information at different resolutions.

Computer vision based damage assessment can be achieved by two major approaches including change detection and anomaly identification. Change detection is to detect the changes of target objects between their pre- and post-event status. Anomaly identification is based on the fact that the structural components of an intact regular building object often follow certain topological relationships. If anomaly is detected on certain component, it is very likely that the component is damaged. At the same time, in order to recognize these anomalies, the semantic meaning of these structural components need to be extracted.

Figure 6.18 (a1 – a3) shows that in order to identify property-wise damage statuses (a3), building objects are identified and extracted from airborne lidar point cloud (a2). Figure 6.18 (b1-b3) shows for mobile lidar point cloud (b1), building objects are first extracted using the building footprint generated from airborne lidar data. Component-level damage assessment is conducted on each individual building object (b3). By identifying the damage status of structural components such as walls and roofs, fine-scale damage assessment can be accomplished.

In addition to these damage assessment approaches, this research also addresses the issue of incomplete data. For example, airborne LiDAR due to their data collection mechanism cannot acquire data on vertical facades of building objects. The incomplete data can also be caused by obstruction. During collection of mobile lidar data, for example, vehicles, vegetation, and debris

can obstruct the view of LiDAR scanners and cameras, leading to difficulties in differentiating gaps in data as openings or just missing data. In addition, disaster sites are dynamic and changing over time as post-disaster activities quickly unfold. As a consequence, damage data may be lost quickly over time. In addition, post-disaster data collection trips may be executed during unfavorable weather conditions, causing data quality issues. To address these issues, fusing data from different sources is a logical approach to enhance the accuracy of damage assessment. Based on this belief, Figure 6.18 (c1-c3, d1-d3) show that crowd-sourced imagery as an alternative data source are integrated into the proposed damage assessment workflow through a data fusion approach. More specifically, Figure 6.18 (c1 – c3) illustrates the use of image-based 3D reconstruction model for the assessment. If multiple images covering the same building objects are collected from varied perspectives, they can be used to reconstruct as-damaged 3D models. Fine-scale damage assessment can be conducted using the reconstructed model. Figure 6.18 (d1 – d3) shows by establishing the correspondence between lidar point cloud and high-resolution image, fine-scale damage condition such as flooding height can be efficiently and quantitatively assessed from images.

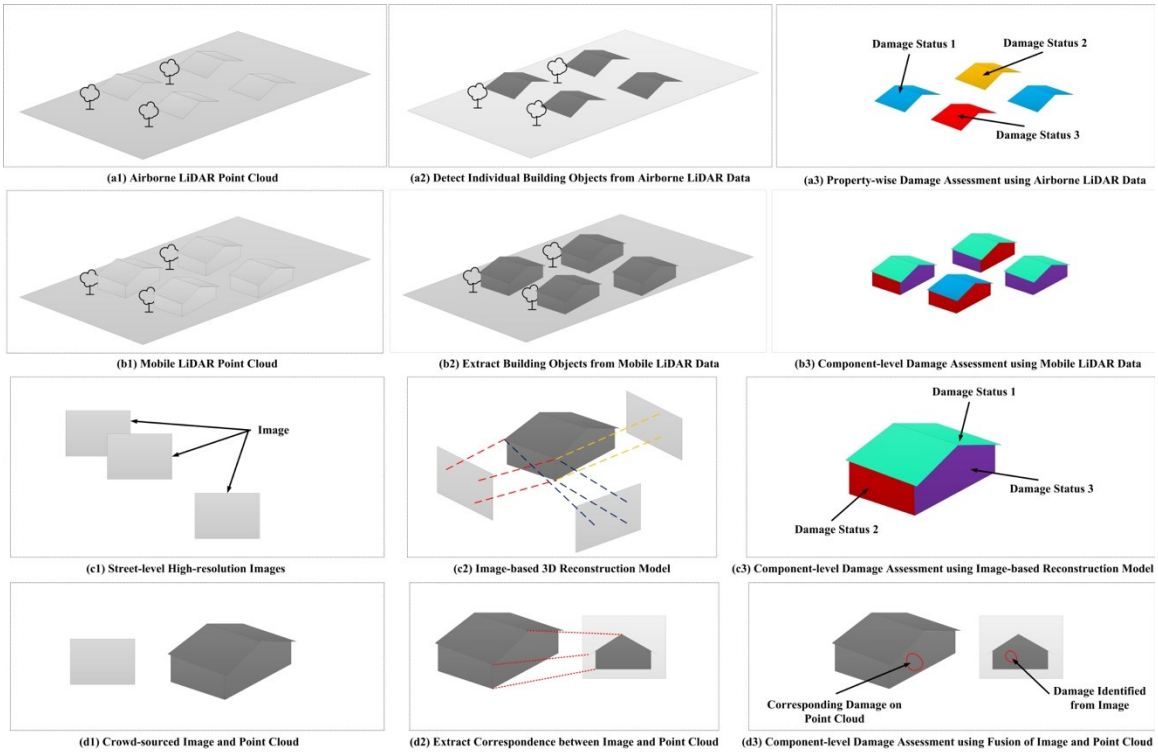


Figure 6.18. Synopsis of the Research Logic

7.1 A Synopsis of the Proposed Approach

7.1.1 Testing Data Sets

The data used in this chapter were collected at the Ocean County, New Jersey (Figure 6.19). In particular, the area enclosed by the 4th Ave to the 7th Ave and the Grand Central Ave to the Oceanfront was used to test the proposed approaches This area suffered devastating damages during Hurricane Sandy. The specific types of data used include:

1) Pre-event airborne LiDAR point cloud collected in 2010 prior to Hurricane Sandy;
2) Post-event airborne LiDAR point cloud collected in November, 2012, after Hurricane Sandy;
3) Post-event mobile LiDAR point cloud collected in 2012 after Hurricane Sandy, conveyed by ground survey team;
4) Post-event street-level images collected in 2012 after Hurricane Sandy, conveyed by ground survey team.

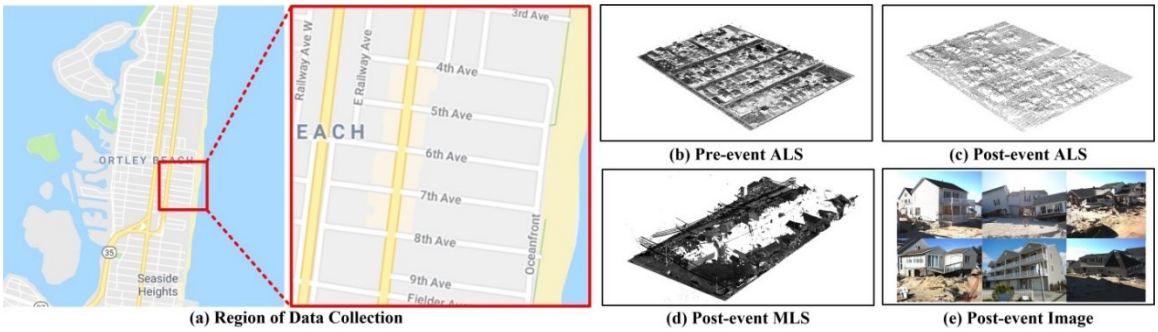


Figure 6.19. Validation Data

7.1.2 Damage Assessment Workflow

Building Detection from ALS

The proposed framework starts with the extraction of individual building objects from the airborne LiDAR data. A pre-trained convolutional neural network model was applied (Figure 6.20). In general, the proposed deep neural network approach is capable of reliably extracting building objects, in particular achieving higher precision and recall compared with other programs such as LAStools. Because of this, the proposed approach tends to identify more correct building objects, which in turn reduces manual effort in extracting building objects. Building object extraction is an important step in damage assessment. In the aftermath of Hurricane Maria, for example, crowd-sourced approaches are used for extracting building objects due to the vast number of damaged buildings and the lack of building footprint databases.

With the proposed approach, it took a computer with no GPUs about 20 seconds to inference a mini-batch of 64 images. For a city block with around 100 properties, it took a CPU-only machine approximately 1 hour to implement the building detection and extraction procedure because there were 12,352 bounding boxes randomly generated around 193 object proposals. In real-world applications, this number can be significantly reduced if GPU-equipped machines are used.

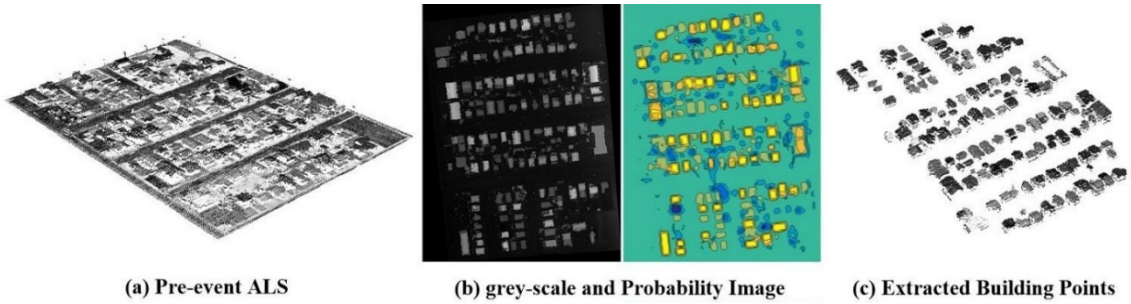


Figure 6.20. Results of Building Detection from ALS using CNN

Property-wise Damage Assessment

Within each building object extracted, the roof models were reconstructed in this step for each detected building to compute features related to damages. As can be seen in Figure 6.21, the extracted points of each building object are as shown in 7.3 (a), and the corresponding points in post-event dataset are shown in *Figure 6.21* (c). *Figure 6.21* (b) shows the reconstructed roof models for each individual buildings, and the corresponding property-wise damage status is presented in *Figure 6.21* (d). In this study, the property-wise damage status was categorized as

1) Intact: represented in blue;
2) Partially collapsed: represented in yellow; and
3) Totally collapsed: represented in red.

In this study, it took less than 1 minute to construct a roof model for a given building and assess the damage status. For the entire block as described in 7.2.1, it took approximately one hour to assess all the buildings in this block. To accelerate the damage assessment, a promising approach is to run the algorithms on multiple blocks simultaneously.

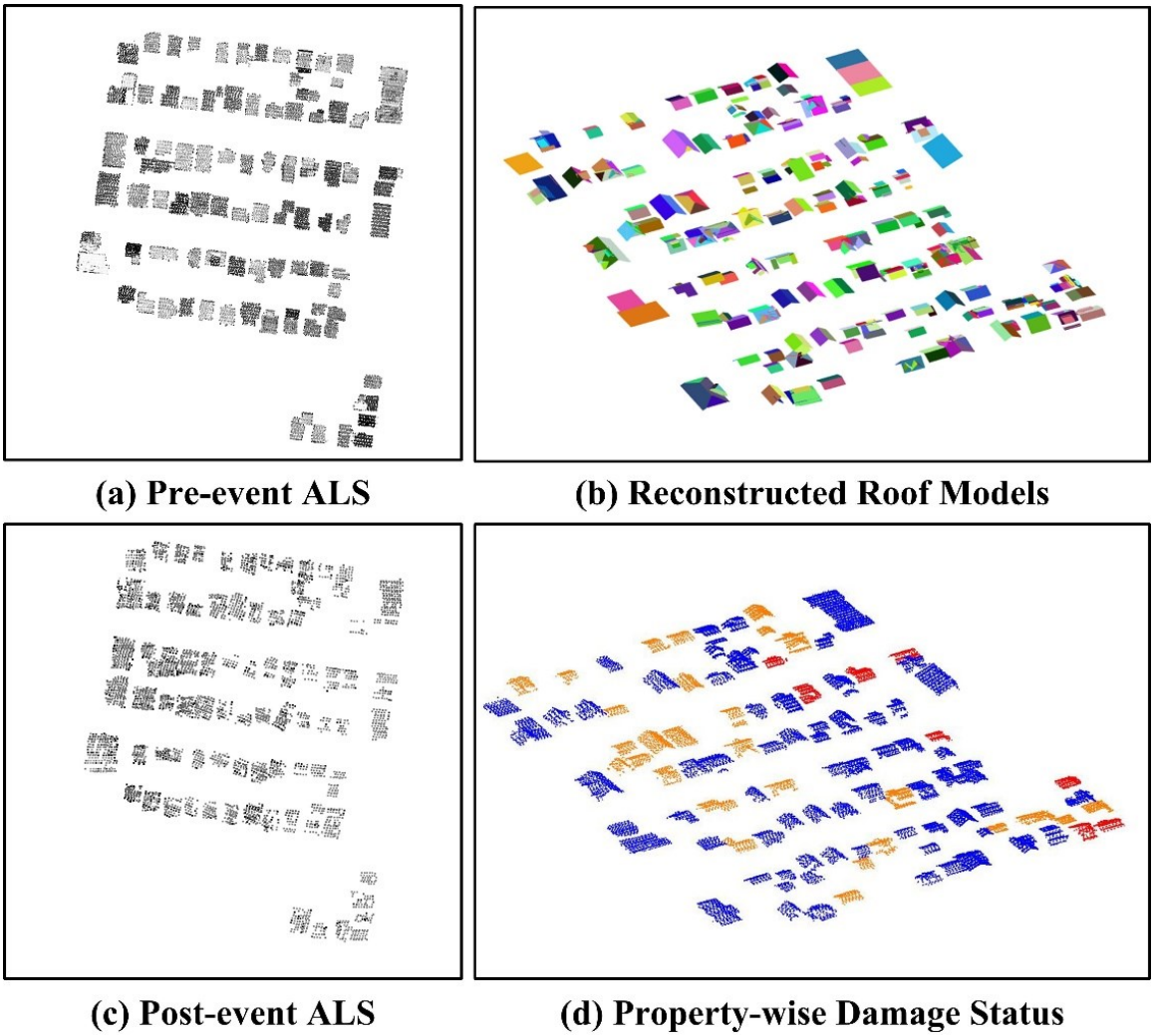


Figure 6.21. Results of Property-wise Damage Assessment

Component-level Damage Assessment

Component-level damage assessment provides detailed damage data for studying the performance of building practices. It is a natural extension of property-wise damage assessment. Within days after the occurrence of hurricane events, it is often possible to send ground assessment teams with various ground-based sensing systems such as cameras, terrestrial LiDAR, and mobile LiDAR. Foot-on-ground assessment with traditional photography approaches is the most common method used in detailed damage assessment. At the same time, it is also the most tedious approach. The proposed approach in this study leverages the increasingly available

terrestrial/mobile LiDAR data for component level damage assessment. However, such assessment is not done in-situ. With the already extracted building footprints and overall damage information from the previous stages, my approach is capable of efficiently extracting individual buildings from the high-resolution LiDAR data and conducting semantic parsing for component-level assessment. This approach has an advantage in terms of time efficiency as it is computationally efficient to process on sparse airborne LiDAR data than on dense mobile data.

Figure 6.22 shows the results of component-level damage detection. Figure 6.22 (a) shows the point cloud of each extracted building using the building footprint, and Figure 6.22 (b) shows the results of semantic parsing and building damage detection. In terms of computational cost, it is largely affected by the size of point cloud. Intuitively, if a building has more data points, it takes more time to process. Figure 6.23 shows the relationship between the computational cost with respect to the number of points per building of each phase. As can be seen, for phase 1, 2, 3, and 5, it follows a linear relationship to some extent. While the linear relationship does not hold for phase 4 because not all the buildings have exterior columns and handrails.

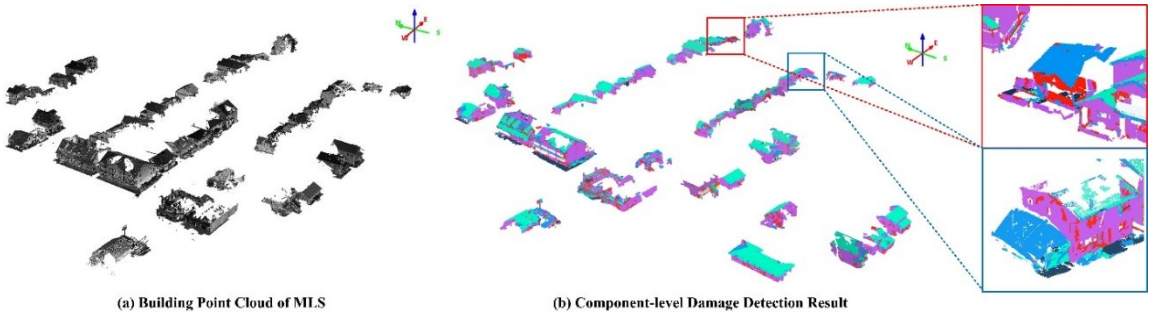


Figure 6.22. Results of Component-level Damage Assessment

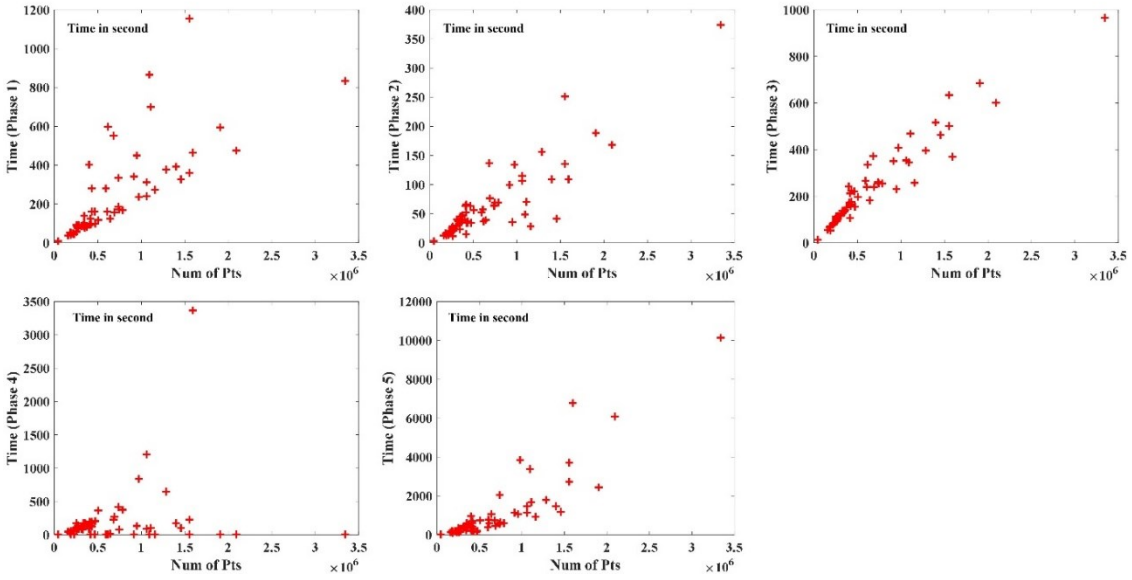


Figure 6.23. Relationship between Computational Cost and Data Size

Robust alignment of the multi-sourced imagery for damage assessment

The LiDAR-based damage assessment still has its own limitations. For example, one can hardly identify the watermark from mobile LiDAR data. In addition, because the collection of mobile LiDAR cannot be conducted without the permission of access, which normally would delay the survey. Consequently, the actual damage status might not be preserved because of the temporary repair of the house owners. For example in Figure 6.24, the watermark is hard to be visually identified from (a) and (a2), but with the assistance of (a1), it could be efficiently and quantitatively measured out. Also for the building in (b) and (b2), the large opening hole was covered by the homeowner prior to the collection of mobile LiDAR data. However, with the use of street level image collected before the temporary repair, the actual area of the opening damage was still successfully evaluated.

Figure 6.24. Results of Image Assisted Damage Assessment

7.1.3 Performance Evaluation

Time Efficiency

As shown in Figure 1.3, the proposed framework has five sequential phases. Table 6.5 summarized the detailed information of the data used in phase 1 to phase 3 of the proposed framework. Also the running time of each stage was summarized. Please be noted that in the table, the running time was summarized in the manner of time per building rather than the total time. Because the process of each individual phase could be either processed on single machine, or ported onto multiple machines for speeding up, it would be intuitive to estimate the running time given the number of buildings within a region of interest and with the number of machines available for process. Please also be noted that the summary of phase 4 and 5 were not included in this table. This is because for phase 4, the process was split into multiple procedures, and was also interacted with human operation such as image grouping, model alignment, etc. For phase 5, the running time was also affected by human-in-the-loop process, which makes less sense for summarizing the running time of entire procedure.

Table 6.5. Summary of Time-wise Efficiency

		Phase 1	Phase 2	Phase 3
Num. Points	Pre-event	338,994	58,446	-
	Post-event	27,829	5,280	26,273,442
Num. Buildings	Pre-event	86	86	-
	Post-event	86	86	36
Platform		Python 2.7 + Tensorflow 1.3.0 (CPU-only)	MATLAB 2016b	MATLAB 2016b
Time/per building		$\approx 20(\text{sec})$	$\approx 60(\text{sec})$	$\approx 22(\text{min})$

The same data used by this dissertation research was also evaluated in (Gong et al. 2014, Hatzikyriakou et al. 2015, Xian et al. 2015). Although remotely sensed data was used in their work, the property-wise and component-level damage assessment was still carried out with manual approach. For example, although mobile LiDAR data was collected and used in their research, the mobile data was not fully made use of. In Xian et al. (2015), the assessment was conducted manually, which was extremely time consuming. Although mobile LiDAR data was employed in Hatzikyriakou et al. (2015), it was mainly used for reconstructing the as-damaged model. The analysis was still carried out with a manual approach.

In Figure 6.25, an intuitive comparison among proposed approach and traditional foot-on-ground approach, and GXH (Gong et al. 2014, Hatzikyriakou et al. 2015, Xian et al. 2015) is presented. As can be seen, after the landing of Hurricane Sandy, both the proposed approach and

GSH started immediately with the availability of airborne data, while the foot-on-ground approach cannot start until the accessibility was permitted.

In terms of the time used in each task, foot-on-ground took longest time among all. Although GXH leveraged remotely sensed data, level of detail analysis still required manual work. On the contrary, the proposed approach largely took use of computer vision techniques to replace manual work. Because the remotely sensed data could be divided into blocks and be processed independently, the proposed approaches can be rapidly scaled up to process large-scale data sets.

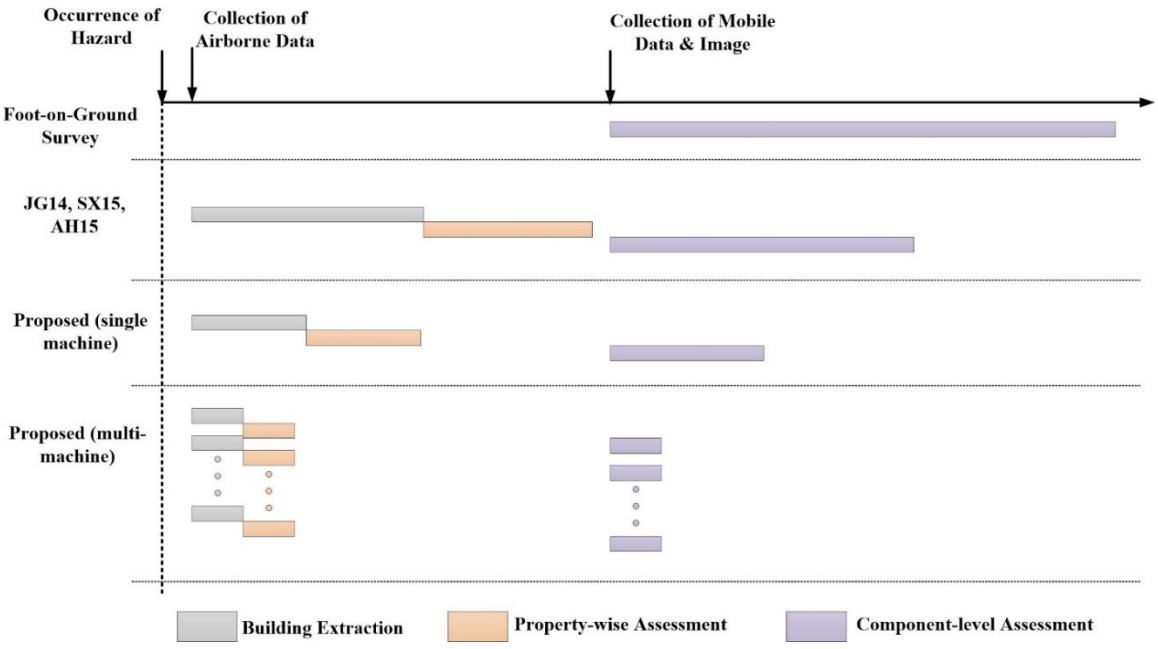


Figure 6.25. Comparison of Time-wise Efficiency

7.2 Conclusions, Limitations and Future Research

7.2.1 Conclusions

This dissertation research focused on the hurricane-induced building damage detection and assessment using remotely sensed data and computer vision techniques. After a major hurricane event occurs, airborne laser scanning is often immediately used for evaluating large topographical changes and for planning rapid response activities. After the road accessibility of the damaged region is granted by the authorities, ground survey is carried out. At this stage, mobile laser

scanning and terrestrial laser scanning are often used to acquire high resolution as-damaged data. Throughout the entire post-disaster response period, high resolution street level image data are often continuously collected by both response personnel and the general public. This dissertation formulates a cohesive and multi-scale damage assessment approach that considers the continuously evolving data sources and damage assessment needs during different phases of disaster management. The individual studies and their findings are concluded as the following.

1) At first, a deep convolutional neural network was developed to detect and extract each individual building object from the LiDAR point cloud. The network was trained using the dataset collected prior to the occurrence of the hazard, and an in-depth experimental study was carried out with the trained network and both pre- and post-event dataset. The result of experimental study suggested that the proposed convolutional neural network outperformed the open-source software, LAStools, particularly on post-event data with relatively low point cloud density. The result also suggested high precision and recall of the proposed approach in detecting building objects from airborne LiDAR point cloud data. Therefore, the proposed approach is shown as a generalizable framework of a preprocessing procedure of the post-disaster evaluation.
2) Within the building objects been identified and extracted, a multi-temporal data-based building damage detection approach is then developed. In this study, the building roof models are first reconstructed from pre-event data with the developed novel approach. Then the property-wise damage assessment procedure is carried out by computing the change detection between multi-temporal data. An experimental study was also carried out here to validate the performance of the proposed approach. And the result suggests that the proposed approach is able to identify the totally collapsed building accurately regardless the variation of post-event data resolution. And the detection of partially collapsed building is relatively sensitive to the data resolution. This would provide the

response personnel with the priority map of rescuing, evacuation, in-depth assessment, and recovery later on.

3) The next study of the dissertation focused on the in-depth building component-level assessment using post-event mobile laser scanning data only. Although mobile LiDAR has been used in post-disaster evaluation, it was mostly used for global damage perception and visualization. To my best knowledge, this dissertation is the first research effort that digs into this field. The experimental study suggested that using post-event mobile LiDAR only can benefit the building component-level damage detection and assessment in automated manner. This is encouraging for the engineering, insurance agency perspective. With the rapid development of laser scanning technology and cloud computing technologies, it is possible in the future that real time scanning and assessment would largely replace the foot-on-ground survey.
4) The fourth study focused on the use of high resolution imagery data for damage assessment. In this study, the 3-D as-damaged model of building was reproduced using structure from motion technique. Then the evaluated was conducted on the reconstructed model. The validation compares the evaluation results between reconstructed model and mobile LiDAR data. The results of comparison study suggested that image-based as-damaged model plays as an ideal alternative and complementary solution to LiDAR. This is particularly valuable when the point cloud model of the building object is in part missing.
5) The last study of this dissertation research focused on the fusion of street level image and LiDAR. In this study, a novel image-to-lidar alignment approach is proposed. It takes an arbitrary street level image as input, and the rigorous registration is estimated through a calibrated image with minimum human interaction. The experimental study revealed that it is potentially worth leveraging the volunteered geographic information image into post-disaster damage evaluation without much human work.

7.2.2 Limitations

The approaches developed in this dissertation have following limitations:

1) The proposed approaches cannot conduct indoor damage assessment. Because the approaches developed in this dissertation rely on remote sensing techniques, the authoritative geospatial data is collected from distance to the objects. The indoor data cannot be acquired by remote sensing approaches. Although crowd-sourced imagery data can be taken from both indoor and outdoor environment, it is still a challenging task to estimate the correspondence between indoor imagery and outdoor geospatial data.
2) Anomaly detection-based approach cannot deal with irregular building structures reliably. Because the proposed methodology adopts pre-defined damage patterns for decision making, it is practically impossible to take every possible building design configuration into consideration.

There is no unified damage assessment protocols. Although post-disaster assessment has been causing increasing attention in recent years, relevant research, including this work, use their own evaluation protocols. As a result, it is difficult to evaluate the damage condition by integrating multiple research efforts.

7.2.3 Future Research

This dissertation built a systematic framework using multi-sourced remotely sensed data in damage detection and evaluation. It also exposed the following aspects to be further investigated in the future:

1) Deep learning techniques have been continuously breaking through the state-of-the-arts in various domain. It is also fetching people's attention in terms of damage assessment. One of the prominent questions is that can deep learning technique benefit the building damage detection and assessment using LiDAR point cloud. It has two main branches,

one of them is detecting damaged buildings from airborne LiDAR data, and the other is detecting damaged buildings or even damaged building components from mobile LiDAR data. Because the detection of building objects from airborne LiDAR has already been accomplished in this dissertation, the first branch is equivalent to a classification problem using deep learning routine, here I call it object-wise classification. However, object-wise classification is not enough for the second branch. Because from the mobile LiDAR model, not only the property-wise damage status is retrievable, but also the component-wise status. Therefore, a point-wise damage status classification is recommended to be carried out in the future.

2) As stated in 1) of Sec 7.2.2, leveraging deep learning in damage detection is of high interest in my future research. Therefore, the significance of a comprehensive database cannot be overemphasized. A well organized and annotated database is vital but also difficult to create, as it is extremely labor intensive and ambiguous to systematically label the ground truth with damage status. Therefore, an annotation framework is prominent as the next stage research.
3) The fusion of image and LiDAR is always a worth exploring but challenging task. This is particularly true in terms of using VGI images and ground-based LiDAR. In this dissertation research, an efficient fusion framework was proposed, and I believe it leads me on the avenue to exploring more comprehensive solutions. For example, in this dissertation, the alignment of arbitrary image still needs a well calibrated image. It is worth exploring a generalized approach that does not require the bridge images. In addition, the region of damage in this dissertation is manually defined. It would be of great value if deep learning technique could be used to reduce the human interaction.

References:

Abadi, M., P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving and M. Isard (2016). TensorFlow: A system for large-scale machine learning. Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI), Savannah, Georgia, USA.

Abayowa, B. O., A. Yilmaz and R. C. Hardie (2015). "Automatic registration of optical aerial imagery to a LiDAR point cloud for generation of city models." ISPRS Journal of Photogrammetry and Remote Sensing **106**: 68-81.

Aijazi, A. K., P. Checchin and L. Trassoudaine (2013). "Segmentation based classification of 3D urban point clouds: A super-voxel based approach with evaluation." Remote Sensing **5**(4): 1624-1650.

Alexe, B., T. Deselaers and V. Ferrari (2012). "Measuring the objectness of image windows." IEEE transactions on pattern analysis and machine intelligence **34**(11): 2189-2202.

Alismail, H., L. D. Baker and B. Browning (2012). Automatic calibration of a range sensor and camera system. 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2012 Second International Conference on, IEEE.

ATC (2014). "ATC-45 Field manual: safety evaluation of buildings after windstorms and floods." Applied Technology Council.

Awrangjeb, M. (2015). "Effective generation and update of a building map database through automatic building change detection from LiDAR point cloud data." Remote Sensing **7**(10): 14119-14150.

Awrangjeb, M. and C. S. Fraser (2014). "An automatic and threshold-free performance evaluation system for building extraction techniques from airborne LiDAR data." IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **7**(10): 4184-4198.

Balz, T. and M. Liao (2010). "Building-damage detection using post-seismic high-resolution SAR satellite data." International Journal of Remote Sensing **31**(13): 3369-3391.

Barnes, C. F., H. Fritz and J. Yoo (2007). "Hurricane disaster assessments with image-driven data mining in high-resolution satellite imagery." IEEE Transactions on Geoscience and Remote Sensing **45**(6): 1631-1640.

Barrington, L., S. Ghosh, M. Greene, S. Har-Noy, J. Berger, S. Gill, A. Y.-M. Lin and C. Huyck (2012). "Crowdsourcing earthquake damage assessment using remote sensing imagery." Annals of Geophysics **54**(6).

Becker, S. and N. Haala (2009). Grammar supported facade reconstruction from mobile lidar mapping. ISPRS Workshop, CMRT09-City Models, Roads and Traffic.

Bian, J., W.-Y. Lin, Y. Matsushita, S.-K. Yeung, T.-D. Nguyen and M.-M. Cheng (2017). "GMS: Grid-based Motion Statistics for Fast, Ultra-robust Feature Correspondence." IEEE Conference on Computer Vision and Pattern Recognition.

Blake, E. S., T. B. Kimberlain, R. J. Berg, J. P. Cangialosi and J. L. Beven Ii (2013). "Tropical cyclone report: Hurricane sandy." National Hurricane Center **12**: 1-10.

Bloch, T., R. Sacks and O. Rabinovitch (2016). "Interior models of earthquake damaged buildings for search and rescue." Advanced Engineering Informatics **30**(1): 65-76.

Cao, R., Y. Zhang, X. Liu and Z. Zhao (2017). "3D building roof reconstruction from airborne LiDAR point clouds: a framework based on a spatial database." International Journal of Geographical Information Science **31**(7): 1359-1380.

Cao, Y., Y. Luo and S. Yang (2011). "Image denoising based on hierarchical Markov random field." Pattern Recognition Letters **32**(2): 368-374.

Carreira, J. and C. Sminchisescu (2012). "Cpmc: Automatic object segmentation using constrained parametric min-cuts." IEEE Transactions on Pattern Analysis and Machine Intelligence **34**(7): 1312-1328.

Catch, A. D. (2012).

Cha, Y.-J., W. Choi and O. Buyukozturk (2017). "Deep learning-based crack damage detection using convolutional neural network"." Computer-Aided Civil and Infrastructure Engineering **32**(3): 2013-2014.

Cha, Y. J., W. Choi and O. Büyüköztürk (2017). "Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks." Computer-Aided Civil and Infrastructure Engineering **32**(5): 361-378.

Chen, B., Z. Chen, L. Deng, Y. Duan and J. Zhou (2016). "Building change detection with RGB-D map generated from UAV images." Neurocomputing **208**: 350-364.

Chen, L.-C. and L.-J. Lin (2010). "Detection of building changes from aerial images and light detection and ranging (LIDAR) data." Journal of Applied Remote Sensing **4**(1): 041870.

Cheng, L., J. Gong, M. Li and Y. Liu (2011). "3D building model reconstruction from multi-view aerial imagery and lidar data." Photogrammetric Engineering & Remote Sensing **77**(2): 125-139.

Chiu, G. L. and S. J. Wadia-Fascetti (1999). "Assessment and quantification of hurricane induced damage to houses." Wind and Structures **2**(3): 133-150.

Claudino-Sales, V., P. Wang and M. H. Horwitz (2010). "Effect of Hurricane Ivan on coastal dunes of Santa Rosa Barrier Island, Florida: characterized on the basis of pre-and poststorm LIDAR surveys." Journal of Coastal Research: 470-484.

Clinch, A. S., E. R. Russ, R. C. Oliver, H. Mitasova and M. F. Overton (2012). "Hurricane Irene and the Pea Island Breach: Pre-storm site characterization and storm surge estimation using geospatial technologies." Shore & Beach **80**(2): 38-46.

Coulter, L. L., D. A. Stow, C. D. Lippitt and G. W. Fraley (2015). Repeat station imaging for rapid airborne change detection. Time-Sensitive Remote Sensing, Springer**:** 29-43.

Cover, T. M. and J. A. Thomas (2012). Elements of information theory, John Wiley & Sons.

Cowgill, E., T. S. Bernardin, M. E. Oskin, C. Bowles, M. B. Yıkılmaz, O. Kreylos, A. J. Elliott, S. Bishop, R. D. Gold and A. Morelan (2012). "Interactive terrain visualization enables virtual field work during rapid scientific response to the 2010 Haiti earthquake." Geosphere **8**(4): 787-804.

Crandell, J. H. and V. Kochkin (2005). Scientific damage assessment methodology and practical applications. Structures Congress 2005: Metropolis and Beyond.

Crooks, A., A. Croitoru, A. Stefanidis and J. Radzikowski (2013). "# Earthquake: Twitter as a distributed sensor system." Transactions in GIS **17**(1): 124-147.

Cui, T., S. Ji, J. Shan, J. Gong and K. Liu (2016). "Line-Based Registration of Panoramic Images and LiDAR Point Clouds for Mobile Mapping." Sensors **17**(1): 70.

Curtis, A. and J. W. Mills (2012). "Spatial video data collection in a post-disaster landscape: The Tuscaloosa Tornado of April 27th 2011." Applied Geography **32**(2): 393-400.

Dai, F., A. Rashidi, I. Brilakis and P. Vela (2012). "Comparison of image-based and time-of-flight-based technologies for three-dimensional reconstruction of infrastructure." Journal of construction engineering and management **139**(1): 69-79.

Dalal, N. and B. Triggs (2005). Histograms of oriented gradients for human detection. Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, IEEE.

Dandois, J. P. and E. C. Ellis (2010). "Remote sensing of vegetation structure using computer vision." Remote Sensing **2**(4): 1157-1176.

De Albuquerque, J. P., B. Herfort, A. Brenning and A. Zipf (2015). "A geographic approach for combining social media and authoritative data towards identifying useful information for disaster management." International Journal of Geographical Information Science **29**(4): 667-689.

Deschaud, J.-E. and F. Goulette (2010). A fast and accurate plane detection algorithm for large noisy point clouds using filtered normals and voxel growing. Proceedings of 3D Processing, Visualization and Transmission Conference (3DPVT2010).

Diebel, J. and S. Thrun (2005). An application of markov random fields to range sensing. NIPS.
Dollár, P. and C. L. Zitnick (2013). Structured forests for fast edge detection. Proceedings of the IEEE International Conference on Computer Vision.
Dorninger, P. and N. Pfeifer (2008). "A comprehensive automated 3D approach for building extraction, reconstruction, and regularization from airborne laser scanning point clouds." Sensors **8**(11): 7323-7343.
dos Santos Galvanin, E. A. and A. P. Dal Poz (2012). "Extraction of building roof contours from LiDAR data using a Markov-random-field-based approach." IEEE transactions on geoscience and remote sensing **50**(3): 981-987.
Du, S., Y. Zhang, R. Qin, Z. Yang, Z. Zou, Y. Tang and C. Fan (2016). "Building Change Detection Using Old Aerial Images and New LiDAR Data." Remote Sensing **8**(12): 1030.
Duchi, J., E. Hazan and Y. Singer (2011). "Adaptive subgradient methods for online learning and stochastic optimization." Journal of Machine Learning Research **12**(Jul): 2121-2159.
Duda, R. O. and P. E. Hart (1972). "Use of the Hough transformation to detect lines and curves in pictures." Communications of the ACM **15**(1): 11-15.
Edelsbrunner, H. and E. P. Mücke (1994). "Three-dimensional alpha shapes." ACM Transactions on Graphics (TOG) **13**(1): 43-72.
Eguchi, R. T., S. P. Gill, S. Ghosh, W. Svekla, B. J. Adams, G. Evans, J. Toro, K. Saito and R. Spence (2010). The January 12, 2010 Haiti earthquake: A comprehensive damage assessment using very high resolution areal imagery. 8th International Workshop on Remote Sensing for Disaster Management.
Elberink, S. O., M. Shoko, S. A. Fathi and M. Rutzinger (2012). "Detection of collapsed buildings by classifying segmented airborne laser scanner data." ISPRS–International Archives ofthe Photogrammetry, Remote Sensing and Spatial Information Sciences **38**(5): 307-312.
Ester, M., H.-P. Kriegel, J. Sander and X. Xu (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. Kdd.
Fan, H., W. Yao and Q. Fu (2014). "Segmentation of sloped roofs from airborne LiDAR point clouds using ridge-based hierarchical decomposition." Remote Sensing **6**(4): 3284-3301.
Fathi, H. and I. Brilakis (2011). "Automated sparse 3D point cloud generation of infrastructure using its distinctive visual features." Advanced Engineering Informatics **25**(4): 760-770.
Fathi, H. and I. Brilakis (2013). "A videogrammetric as-built data collection method for digital fabrication of sheet metal roof panels." Advanced Engineering Informatics **27**(4): 466-476.
FEMA (2016). "Damage Asseessment Operations Manual."
Feng, L. and Z. Lv (2016). "Plane surface detection and reconstruction using segment-based tensor voting." Journal of Visual Communication and Image Representation **40**: 831-837.
Fernandes, L. A. and M. M. Oliveira (2008). "Real-time line detection through an improved Hough transform voting scheme." Pattern recognition **41**(1): 299-314.
Ferraz, L., X. Binefa and F. Moreno-Noguer (2014). Very fast solution to the PnP problem with algebraic outlier rejection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.
Fischler, M. A. and R. C. Bolles (1981). "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography." Communications of the ACM **24**(6): 381-395.
Fonstad, M. A., J. T. Dietrich, B. C. Courville, J. L. Jensen and P. E. Carbonneau (2013). "Topographic structure from motion: a new development in photogrammetric measurement." Earth Surface Processes and Landforms **38**(4): 421-430.
Friedland, C. J. (2009). Residential building damage from hurricane storm surge: proposed methodologies to describe, assess and model building damage.

Fritz, A., T. Kattenborn and B. Koch (2013). "UAV-based photogrammetric point clouds—Tree stem mapping in open stands in comparison to terrestrial laser scanner point clouds." Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci **40**: 141-146.

Furukawa, Y. and J. Ponce (2010). "Accurate, dense, and robust multiview stereopsis." IEEE transactions on pattern analysis and machine intelligence **32**(8): 1362-1376.

Galarreta, J. F., N. Kerle and M. Gerke (2015). "UAV-based urban structural damage assessment using object-based image analysis and semantic reasoning." Natural Hazards and Earth System Sciences **15**(6): 1087.

Geiger, A., F. Moosmann, Ö. Car and B. Schuster (2012). Automatic camera and range sensor calibration using a single shot. Robotics and Automation (ICRA), 2012 IEEE International Conference on, IEEE.

Geiß, C., P. A. Pelizari, M. Marconcini, W. Sengara, M. Edwards, T. Lakes and H. Taubenböck (2015). "Estimation of seismic building structural types using multi-sensor remote sensing and machine learning techniques." ISPRS Journal of Photogrammetry and Remote Sensing **104**: 175-188.

Gesch, D. B. (2009). "Analysis of lidar elevation data for improved identification and delineation of lands vulnerable to sea-level rise." Journal of Coastal Research: 49-58.

Ghamisi, P., B. Höfle and X. X. Zhu (2016). "Hyperspectral and LiDAR Data Fusion Using Extinction Profiles and Deep Convolutional Neural Network." IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing.

Girshick, R. (2015). Fast r-cnn. Proceedings of the IEEE International Conference on Computer Vision.

Girshick, R., J. Donahue, T. Darrell and J. Malik (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition.

Golparvar-Fard, M., F. Pena-Mora and S. Savarese (2011). Monitoring changes of 3D building elements from unordered photo collections. Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on, IEEE.

Golparvar-Fard, M., F. Peña-Mora and S. Savarese (2012). "Automated progress monitoring using unordered daily construction photographs and IFC-based building information models." Journal of Computing in Civil Engineering **29**(1): 04014025.

Golparvar-Fard, M., J. Thomas, F. Peña-Mora and S. Savarese (2010). Remote assessment of pre- and post-disaster critical physical infrastructures using mobile workstation chariot and D4AR models. Proc., Int. Conf. on Computing in Civil and Building Engineering, ASCE, Reston, VA.

Gong, J. and A. Maher (2014). "Use of mobile lidar data to assess hurricane damage and visualize community vulnerability." Transportation Research Record: Journal of the Transportation Research Board(2459): 119-126.

Goodchild, M. F., P. Fu and P. Rich (2007). "Sharing geographic information: an assessment of the Geospatial One-Stop." Annals of the Association of American Geographers **97**(2): 250-266.

Graettinger, A. J., D. Grau, J. Van De Lindt and D. O. Prevatt (2012). GIS for the geo-referenced analysis and rapid dissemination of forensic evidence collected in the aftermath of the Tuscaloosa tornado. Construction Research Congress 2012: Construction Challenges in a Flat World.

Gurley, K., J. Davis, Robert H, S.-P. Ferrera, J. Burton, F. Masters, T. Reinhold and M. Abdullah (2006). Post 2004 Hurricane Field Survey--An Evaluation of the Relative Performance of the Standard Building Code and the Florida Building Code. Structures Congress 2006: Structural Engineering and Public Safety.

Hackel, T., J. D. Wegner and K. Schindler (2016). "Fast semantic segmentation of 3D point clouds with strongly varying density." ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Prague, Czech Republic **3**: 177-184.

Harris, C. and M. Stephens (1988). A combined corner and edge detector. Alvey vision conference, Manchester, UK.

Hartley, R. and A. Zisserman (2003). Multiple view geometry in computer vision, Cambridge university press.

Hatzikyriakou, A., N. Lin, J. Gong, S. Xian, X. Hu and A. Kennedy (2015). "Component-based vulnerability analysis for residential structures subjected to storm surge impact from Hurricane Sandy." Natural Hazards Review **17**(1): 05015005.

He, F. (2012). Co-registration Between Imagery and Point Cloud Acquired by MLS Platform, University of Twente Faculty of Geo-Information and Earth Observation (ITC).

He, K., X. Zhang, S. Ren and J. Sun (2016). Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

He, M., Q. Zhu, Z. Du, H. Hu, Y. Ding and M. Chen (2016). "A 3D shape descriptor based on contour clusters for damaged roof detection using airborne LiDAR point clouds." Remote Sensing **8**(3): 189.

Howe, J. (2008). Crowdsourcing: How the power of the crowd is driving the future of business, Random House.

Hu, X. and Y. Yuan (2016). "Deep-Learning-Based Classification for DTM Extraction from ALS Point Cloud." Remote Sensing **8**(9): 730.

Hua, C., J. Qi, H. Shang, W. Hu and J. Han (2016). "Detection of collapsed buildings with the aerial images captured from UAV." Science China Information Sciences **59**(3): 1-15.

Huang, H., C. Brenner and M. Sester (2013). "A generative statistical approach to automatic 3D building roof reconstruction from laser scanning data." ISPRS Journal of photogrammetry and remote sensing **79**: 29-43.

Huang, X., L. Zhang and T. Zhu (2014). "Building change detection from multitemporal high-resolution remotely sensed images based on a morphological building index." IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **7**(1): 105-115.

Huang, Z., G. Cheng, H. Wang, H. Li, L. Shi and C. Pan (2016). Building extraction from multi-source remote sensing images via deep deconvolution neural networks. Geoscience and Remote Sensing Symposium (IGARSS), 2016 IEEE International, IEEE.

Ioffe, S. and C. Szegedy (2015). "Batch normalization: Accelerating deep network training by reducing internal covariate shift." arXiv preprint arXiv:1502.03167.

Jahanshahi, M. R. and S. F. Masri (2012). "Adaptive vision-based crack detection using 3D scene reconstruction for condition assessment of structures." Automation in Construction **22**: 567-576.

Jahanshahi, M. R. and S. F. Masri (2013). "A new methodology for non-contact accurate crack width measurement through photogrammetry for automated structural safety evaluation." Smart materials and structures **22**(3): 035019.

Janalipour, M. and A. Mohammadzadeh (2016). "Building damage detection using object-based image analysis and ANFIS from high-resolution image (Case study: BAM earthquake, Iran)." IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **9**(5): 1937-1945.

Jha, M. N., J. Levy and Y. Gao (2008). "Advances in remote sensing for oil spill disaster management: state-of-the-art sensors technology for oil spill surveillance." Sensors **8**(1): 236-255.

Jung, J., Y. Jwa and G. Sohn (2017). "Implicit Regularization for Reconstructing 3D Building Rooftop Models Using Airborne LiDAR Data." Sensors **17**(3): 621.

Kahraman, F., M. Imamoglu and H. F. Ates (2016). "Disaster Damage Assessment of Buildings Using Adaptive Self-Similarity Descriptor." IEEE Geoscience and Remote Sensing Letters **13**(8): 1188-1192.

Karimzadeh, S. and M. Mastuoka (2017). "Building Damage Assessment Using Multisensor Dual-Polarized Synthetic Aperture Radar Data for the 2016 M 6.2 Amatrice Earthquake, Italy." Remote Sensing **9**(4): 330.

Kashani, A., M. Olsen and A. Graettinger (2015). Laser scanning intensity analysis for automated building wind damage detection. Computing in Civil Engineering 2015**:** 199-205.

Kashani, A. G., P. S. Crawford, S. K. Biswas, A. J. Graettinger and D. Grau (2014). "Automated tornado damage assessment and wind speed estimation based on terrestrial laser scanning." Journal of Computing in Civil Engineering **29**(3): 04014051.

Kashani, A. G., A. Graettinger and T. Dao (2014). 3D data collection and automated damage assessment for near real-time tornado loss estimation. Construction Research Congress 2014: Construction in a Global Network.

Kashani, A. G. and A. J. Graettinger (2015). "Cluster-based roof covering damage detection in ground-based lidar data." Automation in Construction **58**: 19-27.

Kashani, A. G., A. J. Graettinger and T. Dao (2016). "Lidar-Based Methodology to Evaluate Fragility Models for Tornado-Induced Roof Damage." Natural Hazards Review **17**(3): 04016006.

Kato, Z. and J. Zerubia (2012). "Markov random fields in image segmentation." Foundations and Trends® in Signal Processing **5**(1–2): 1-155.

Khoshelham, K., S. O. Elberink and S. Xu (2013). "Segment-based classification of damaged building roofs in aerial laser scanning data." IEEE Geoscience and Remote Sensing Letters **10**(5): 1258-1262.

Klemas, V. V. (2009). "The role of remote sensing in predicting and determining coastal storm impacts." Journal of Coastal Research: 1264-1275.

Koch, C., G. M. Jog and I. Brilakis (2012). "Automated pothole distress assessment using asphalt pavement video data." Journal of Computing in Civil Engineering **27**(4): 370-378.

Krizhevsky, A., I. Sutskever and G. E. Hinton (2012). Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems.

Lafarge, F. and C. Mallet (2011). Building large urban environments from unstructured point data. Computer Vision (ICCV), 2011 IEEE International Conference on, IEEE.

Lane, S. N., T. D. James, H. Pritchard and M. Saunders (2003). "Photogrammetric and laser altimetric reconstruction of water levels for extreme flood event analysis." The Photogrammetric Record **18**(104): 293-307.

Lattanzi, D. (2013). A computational framework for next-generation inspection imaging.

Leberl, F., A. Irschara, T. Pock, P. Meixner, M. Gruber, S. Scholz and A. Wiechert (2010). "Point clouds." Photogrammetric Engineering & Remote Sensing **76**(10): 1123-1134.

Li, M., L. Cheng, J. Gong, Y. Liu, Z. Chen, F. Li, G. Chen, D. Chen and X. Song (2008). "Post-earthquake assessment of building damage degree using LiDAR data and imagery." Science in China Series E: Technological Sciences **51**: 133-143.

Li, S., H. Tang, S. He, Y. Shu, T. Mao, J. Li and Z. Xu (2015). "Unsupervised detection of earthquake-triggered roof-holes from UAV images using joint color and shape features." IEEE Geoscience and Remote Sensing Letters **12**(9): 1823-1827.

Li, X., H. Guo, L. Zhang, X. Chen and L. Liang (2012). "A new approach to collapsed building extraction using RADARSAT-2 polarimetric SAR imagery." IEEE Geoscience and Remote Sensing Letters **9**(4): 677-681.

Li, Y., L. Li, D. Li, F. Yang and Y. Liu (2017). "A Density-Based Clustering Method for Urban Scene Mobile Laser Scanning Data Segmentation." Remote Sensing **9**(4): 331.

Li, Z., L. Zhang, P. T. Mathiopoulos, F. Liu, L. Zhang, S. Li and H. Liu (2017). "A hierarchical methodology for urban facade parsing from TLS point clouds." ISPRS Journal of Photogrammetry and Remote Sensing **123**: 75-93.

Lin, N., K. Emanuel, M. Oppenheimer and E. Vanmarcke (2012). "Physically-based assessment of hurricane surge threat under climate change."

Lin, Y., J. Hyyppa and A. Jaakkola (2011). "Mini-UAV-borne LIDAR for fine-scale mapping." IEEE Geoscience and Remote Sensing Letters **8**(3): 426-430.

Lin, Y. z., Z. h. Nie and H. w. Ma (2017). "Structural Damage Detection with Automatic Feature-Extraction through Deep Learning." Computer-Aided Civil and Infrastructure Engineering.

Liu, L. and I. Stamos (2007). A systematic approach for 2D-image to 3D-range registration in urban environments. Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on, IEEE.

Liu, S., X. Tong, J. Chen, X. Liu, W. Sun, H. Xie, P. Chen, Y. Jin and Z. Ye (2016). "A linear feature-based approach for the registration of unmanned aerial vehicle remotely-sensed images and airborne LiDAR data." Remote Sensing **8**(2): 82.

Liu, W., D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu and A. C. Berg (2016). SSD: Single shot multibox detector. European Conference on Computer Vision, Springer.

Lowe, D. G. (2004). "Distinctive image features from scale-invariant keypoints." International journal of computer vision **60**(2): 91-110.

Lu, Q. and T. Jiang (2001). "Pixon-based image denoising with Markov random fields." Pattern Recognition **34**(10): 2029-2039.

Lucieer, A., S. M. d. Jong and D. Turner (2014). "Mapping landslide displacements using Structure from Motion (SfM) and image correlation of multi-temporal UAV photography." Progress in Physical Geography **38**(1): 97-116.

Martínez, J., A. Soria-Medina, P. Arias and A. F. Buffara-Antunes (2012). "Automatic processing of terrestrial laser scanning data of building facades." Automation in Construction **22**: 298-305.

Marton, Z. C., R. B. Rusu and M. Beetz (2009). On fast surface reconstruction methods for large and noisy point clouds. Robotics and Automation, 2009. ICRA'09. IEEE International Conference on, IEEE.

Massarra, C. C. (2012). "Hurricane damage assessment process for residential buildings."

Mastin, A., J. Kepner and J. Fisher (2009). Automatic registration of LIDAR and optical images of urban scenes. Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, IEEE.

Masuoka, M. and F. Yamazaki (2004). Employment of lidar for disaster assessment. Proceedings of the 2nd 436 International Workshop on Remote Sensing for Post-Disaster Response. Newport Beach, Ca.

Matas, J., C. Galambos and J. Kittler (2000). "Robust detection of lines using the progressive probabilistic hough transform." Computer Vision and Image Understanding **78**(1): 119-137.

Mathews, A. J. and J. L. Jensen (2013). "Visualizing and quantifying vineyard canopy LAI using an unmanned aerial vehicle (UAV) collected high density structure from motion point cloud." Remote sensing **5**(5): 2164-2183.

McDougall, K. (2011). Using volunteered information to map the Queensland floods. Proceedings of the 2011 Surveying and Spatial Sciences Conference: Innovation in Action: Working Smarter (SSSC 2011), Surveying and Spatial Sciences Institute.

Meng, X., N. Currit and K. Zhao (2010). "Ground filtering algorithms for airborne LiDAR data: A review of critical issues." Remote Sensing **2**(3): 833-860.

Meng, X., L. Wang and N. Currit (2009). "Morphology-based building detection from airborne LIDAR data." Photogrammetric Engineering & Remote Sensing **75**(4): 437-442.

Monszpart, A., N. Mellado, G. J. Brostow and N. J. Mitra (2015). "RAPter: rebuilding man-made scenes with regular arrangements of planes." ACM Trans. Graph. **34**(4): 103.

Moreira, A. and M. Y. Santos (2007). "Concave hull: A k-nearest neighbours approach for the computation of the region occupied by a set of points."

Mosteller, F. and J. W. Tukey (1977). "Data analysis and regression: a second course in statistics." Addison-Wesley Series in Behavioral Science: Quantitative Methods.

Nex, F., E. Rupnik, I. Toschi and F. Remondino (2014). "Automated processing of high resolution airborne images for earthquake damage assessment." The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **40**(1): 315.

Nguyen, H., D. Belton and P. Helmholz (2016). "Scan profiles based method for segmentation and extraction of planar objects in mobile laser scanning point clouds." International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences-ISPRS Archives **41**: 351-358.

Niemeyer, J., F. Rottensteiner and U. Soergel (2014). "Contextual classification of lidar data and building object detection in urban areas." ISPRS journal of photogrammetry and remote sensing **87**: 152-165.

NOAA (2012). "2012 USGS EAARL-B Lidar: Post-Sandy, Barnegat Bay (NJ) Point Cloud files with Orthometric Vertical Datum NAVD88 using GEOID12A. ."

Nurunnabi, A., G. West and D. Belton (2015). "Outlier detection and robust normal-curvature estimation in mobile laser scanning 3D point cloud data." Pattern Recognition **48**(4): 1404-1419.

Ochmann, S., R. Vock, R. Wessel and R. Klein (2016). "Automatic reconstruction of parametric building models from indoor point clouds." Computers & Graphics **54**: 94-103.

Olah, C. (2017). "http://colah.github.io/."

Olsen, M. J., K. F. Cheung, Y. Yamazaki, S. Butcher, M. Garlock, S. Yim, S. McGarity, I. Robertson, L. Burgos and Y. L. Young (2012). "Damage assessment of the 2010 Chile earthquake and tsunami using terrestrial laser scanning." Earthquake Spectra **28**(S1): S179-S197.

Olsen, M. J., F. Kuester, B. J. Chang and T. C. Hutchinson (2009). "Terrestrial laser scanning-based structural damage assessment." Journal of Computing in Civil Engineering **24**(3): 264-272.

Ouédraogo, M. M., A. Degré, C. Debouche and J. Lisein (2014). "The evaluation of unmanned aerial system-based photogrammetry and terrestrial laser scanning to generate DEMs of agricultural watersheds." Geomorphology **214**: 339-355.

Ozisik, D. and N. Kerle (2004). Post-earthquake damage assessment using satellite and airborne data in the case of the 1999 Kocaeli earthquake, Turkey. Proc. of the XXth ISPRS congress: Geo-imagery bridging continents.

Pandey, G., J. R. McBride, S. Savarese and R. M. Eustice (2015). "Automatic extrinsic calibration of vision and lidar by maximizing mutual information." Journal of Field Robotics **32**(5): 696-722.

Pang, S., X. Hu, Z. Wang and Y. Lu (2014). "Object-based analysis of airborne LiDAR data for building change detection." Remote Sensing **6**(11): 10733-10749.

Papon, J., A. Abramov, M. Schoeler and F. Worgotter (2013). Voxel cloud connectivity segmentation-supervoxels for point clouds. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Parmehr, E. G., C. S. Fraser, C. Zhang and J. Leach (2014). "Automatic registration of optical imagery with 3D LiDAR data using statistical similarity." ISPRS Journal of Photogrammetry and Remote Sensing **88**: 28-40.

Pizarro, O., R. Eustice and H. Singh (2004). Large area 3D reconstructions from underwater surveys. OCEANS'04. MTTS/IEEE TECHNO-OCEAN'04, IEEE.

Poser, K. and D. Dransch (2010). "Volunteered geographic information for disaster management with application to rapid flood damage estimation." Geomatica **64**(1): 89-98.

Pu, S. and G. Vosselman (2009). "Knowledge based reconstruction of building models from terrestrial laser scanning data." ISPRS Journal of Photogrammetry and Remote Sensing **64**(6): 575-584.

Quan, L. and Z. Lan (1999). "Linear n-point camera pose determination." IEEE Transactions on pattern analysis and machine intelligence **21**(8): 774-780.

Rabbani, T., F. Van Den Heuvel and G. Vosselmann (2006). "Segmentation of point clouds using smoothness constraint." International archives of photogrammetry, remote sensing and spatial information sciences **36**(5): 248-253.

Rafiei, M. H. and H. Adeli (2017). "A novel machine learning-based algorithm to detect damage in high-rise building structures." The Structural Design of Tall and Special Buildings.

Rastiveis, H., F. Eslamizade and E. Hosseini-Zirdoo (2015). "Building Damage Assessment after Earthquake Using Post-Event LiDAR Data." The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **40**(1): 595.

Redmon, J., S. Divvala, R. Girshick and A. Farhadi (2016). You only look once: Unified, real-time object detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Rego, J. L. and C. Li (2010). "Storm surge propagation in Galveston Bay during hurricane Ike." Journal of Marine Systems **82**(4): 265-279.

Rehor, M., H. P. Bähr, F. Tarsha-Kurdi, T. Landes and P. Grussenmeyer (2008). "Contribution of two plane detection algorithms to recognition of intact and damaged buildings in lidar data." The Photogrammetric Record **23**(124): 441-456.

Ren, S., K. He, R. Girshick and J. Sun (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems.

Ripperda, N. and C. Brenner (2009). Application of a formal grammar to facade reconstruction in semiautomatic and automatic environments. Proc. of the 12th AGILE Conference on GIScience.

Rönnholm, P., M. Karjalainen, H. Kaartinen, K. Nurminen and J. Hyyppa (2013). "Relative orientation between a single frame image and LiDAR point cloud using linear features." Photogrammetric Journal of Finland **23**(2).

Rosser, J. F., D. Leibovici and M. Jackson (2017). "Rapid flood inundation mapping using social media, remote sensing and topographic data." Natural Hazards **87**(1): 103-120.

Rothermel, M., K. Wenzel, D. Fritsch and N. Haala (2012). Sure: Photogrammetric surface reconstruction from imagery. Proceedings LC3D Workshop, Berlin.

Schnabel, R., R. Wahl and R. Klein (2007). Efficient RANSAC for point-cloud shape detection. Computer graphics forum, Wiley Online Library.

Schnebele, E. (2013). "Improving remote sensing flood assessment using volunteered geographical data." Natural Hazards and Earth System Sciences **13**(3): 669.

Schnebele, E., G. Cervone and N. Waters (2013). "Road assessment after flood events using non-authoritative data." Nat. Hazards Earth Syst. Discuss **1**(4): 4155-4179.

Schnebele, E. and N. Waters (2014). "Road assessment after flood events using non-authoritative data." Natural Hazards and Earth System Sciences **14**(4): 1007.

Schweier, C. and M. Markus (2004). "Assessment of the search and rescue demand for individual buildings." Meirin Journal of Dental Engineering & Oral Health Science **4**(1): 16-21.

Schweier, C. and M. Markus (2006). "Classification of collapsed buildings for fast damage and loss assessment." Bulletin of earthquake engineering **4**(2): 177-192.

Sermanet, P., D. Eigen, X. Zhang, M. Mathieu, R. Fergus and Y. LeCun (2013). "Overfeat: Integrated recognition, localization and detection using convolutional networks." arXiv preprint arXiv:1312.6229.

Sharma, R. C., R. Tateishi, K. Hara, H. T. Nguyen, S. Gharechelou and L. V. Nguyen (2017). "Earthquake Damage Visualization (EDV) Technique for the Rapid Detection of Earthquake-Induced Damages Using SAR Data." Sensors **17**(2): 235.

Shen, W., J. Zhang and F. Yuan (2011). A new algorithm of building boundary extraction based on LIDAR data. Geoinformatics, 2011 19th International Conference on, IEEE.

Shen, Y., Z. Wang and L. Wu (2010). Extraction of building's geometric axis line from LiDAR data for disaster management. Geoscience and Remote Sensing Symposium (IGARSS), 2010 IEEE International, IEEE.

Sheng, Q., Q. Wang, X. Zhang, B. Wang, B. Zhang and Z. Zhang (2017). "Registration of Urban Aerial Image and LiDAR Based on Line Vectors." Applied Sciences **7**(10): 965.

Sherman, D. J., B. U. Hales, M. K. Potts, J. T. Ellis, H. Liu and C. Houser (2013). "Impacts of Hurricane Ike on the beaches of the Bolivar Peninsula, TX, USA." Geomorphology **199**: 62-81.

Sherrah, J. (2016). "Fully convolutional networks for dense semantic labelling of high-resolution aerial imagery." arXiv preprint arXiv:1606.02585.

Simões, F., M. Almeida, M. Pinheiro, R. Dos Anjos, A. Dos Santos, R. Roberto, V. Teichrieb, C. Suetsugo and A. Pelinson (2012). Challenges in 3d reconstruction from images for difficult large-scale objects: A study on the modeling of electrical substations. Virtual and Augmented Reality (SVR), 2012 14th Symposium on, IEEE.

Simonyan, K. and A. Zisserman (2014). "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556.

Sithole, G. and G. Vosselman (2006). "Bridge detection in airborne laser scanner data." ISPRS Journal of Photogrammetry and Remote Sensing **61**(1): 33-46.

Soheilian, B., N. Paparoditis and B. Vallet (2013). "Detection and 3D reconstruction of traffic signs from multiple view color images." ISPRS Journal of Photogrammetry and Remote Sensing **77**: 1-20.

Spetsakis, M. and J. Y. Aloimonos (1991). "A multi-frame approach to visual motion perception." International Journal of Computer Vision **6**(3): 245-255.

Sturzenegger, M., D. Stead, C. Froese, F. Moreno and M. Jaboyedoff (2007). Ground-based and airborne LiDAR for structural mapping of the Frank Slide. 1st Canada-US Rock Mechanics Symposium, American Rock Mechanics Association.

Subramaniam, N. A. and K. Ponto (2014). Hierarchical Plane Extraction (HPE): an efficient method for extraction of planes from large point cloud datasets. Proceedings of the 34th Annual Conference of the Association for Computer Aided Design in Architecture on, ACADIA.

Sui, H., J. Tu, Z. Song, G. Chen and Q. Li (2014). "A novel 3D building damage detection method using multiple overlapping UAV images." The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **40**(7): 173.

Sun, S. and C. Salvaggio (2013). "Aerial 3D building detection and modeling from airborne LiDAR point clouds." IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **6**(3): 1440-1449.

Sun, Y., S. Schaefer and W. Wang (2015). "Denoising point sets via L0 minimization." Computer Aided Geometric Design **35**: 2-15.

Susaki, J. (2015). "Region-based automatic mapping of tsunami-damaged buildings using multi-temporal aerial images." Natural Hazards **76**(1): 397-420.

Szegedy, C., W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich (2015). Going deeper with convolutions. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Szeliski, R. and S. B. Kang (1994). "Recovering 3D shape and motion from image streams using nonlinear least squares." Journal of Visual Communication and Image Representation **5**(1): 10-28.

Tamas, L. and Z. Kato (2013). Targetless calibration of a lidar-perspective camera pair. Proceedings of the IEEE International Conference on Computer Vision Workshops.

Tarsha-Kurdi, F., T. Landes and P. Grussenmeyer (2007). Hough-transform and extended ransac algorithms for automatic detection of 3d building roof planes from lidar data. Proceedings of the ISPRS Workshop on Laser Scanning.

The MathWorks, I. (2016). "MATLAB and Statistics Toolbox Release 2016b."

Tian, J., S. Cui and P. Reinartz (2014). "Building Change Detection Based on Satellite Stereo Imagery and Digital Surface Models." IEEE Trans. Geoscience and Remote Sensing **52**(1): 406-417.

Tomasi, C. and T. Kanade (1992). "Shape and motion from image streams under orthography: a factorization method." International Journal of Computer Vision **9**(2): 137-154.

Tong, X., Z. Hong, S. Liu, X. Zhang, H. Xie, Z. Li, S. Yang, W. Wang and F. Bao (2012). "Building-damage detection using pre-and post-seismic high-resolution satellite stereo imagery: a case study of the May 2008 Wenchuan earthquake." ISPRS Journal of Photogrammetry and Remote Sensing **68**: 13-27.

Tong, X., X. Lin, T. Feng, H. Xie, S. Liu, Z. Hong and P. Chen (2013). "Use of shadows for detection of earthquake-induced collapsed buildings in high-resolution satellite imagery." ISPRS journal of photogrammetry and remote sensing **79**: 53-67.

Torok, M. M., M. Golparvar-Fard and K. B. Kochersberger (2013). "Image-based automated 3D crack detection for post-disaster building assessment." Journal of Computing in Civil Engineering **28**(5): A4014004.

Toshev, A., P. Mordohai and B. Taskar (2010). Detecting and parsing architecture at city scale from range data. Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE.

Triggs, B. (1999). Camera pose and calibration from 4 or 5 known 3d points. Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on, IEEE.

Triggs, B., P. F. McLauchlan, R. I. Hartley and A. W. Fitzgibbon (1999). Bundle adjustment—a modern synthesis. International workshop on vision algorithms, Springer.

Triglav-Čekada, M. and D. Radovan (2013). "Using volunteered geographical information to map the November 2012 floods in Slovenia." Natural Hazards and Earth System Sciences **13**(11): 2753-2762.

Trinder, J. and M. Salah (2011). Airborne LiDAR as a tool for disaster monitoring and management. Proceedings of the GeoInformation for Disaster Management Conference (Gi4DM 2011).

Trinder, J. and M. Salah (2012). "Aerial images and LiDAR data fusion for disaster change detection." ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences **1**(4): 227-232.

Truong-Hong, L., D. F. Laefer, T. Hinks and H. Carr (2011). "Flying voxel method with Delaunay triangulation criterion for façade/feature detection for computation." Journal of Computing in Civil Engineering **26**(6): 691-707.

Truong-Hong, L., D. F. Laefer, T. Hinks and H. Carr (2013). "Combining an angle criterion with voxelization and the flying voxel method in reconstructing building models from LiDAR data." Computer-Aided Civil and Infrastructure Engineering **28**(2): 112-129.

Tseng, Y.-H. and H.-C. Hung (2016). "EXTRACTION OF BUILDING BOUNDARY LINES FROM AIRBORNE LIDAR POINT CLOUDS." International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences **40**.

Tu, J., H. Sui, W. Feng and Z. Song (2016). "Automatic Building Damage Detection Method Using High-Resolution Remote Sensing Images and 3D GIS Model." ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences: 43-50.

Uijlings, J. R., K. E. Van De Sande, T. Gevers and A. W. Smeulders (2013). "Selective search for object recognition." International journal of computer vision **104**(2): 154-171.

Vakalopoulou, M., K. Karantzalos, N. Komodakis and N. Paragios (2015). Building detection in very high resolution multispectral data with deep learning features. Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International, IEEE.

Vanegas, C. A., D. G. Aliaga and B. Benes (2012). "Automatic extraction of Manhattan-world building masses from 3D laser range scans." IEEE transactions on visualization and computer graphics **18**(10): 1627-1637.

Verykokou, S., A. Doulamis, G. Athanasiou, C. Ioannidis and A. Amditis (2016). UAV-based 3D modelling of disaster scenes for urban search and rescue. Imaging Systems and Techniques (IST), 2016 IEEE International Conference on, IEEE.

Vetrivel, A., D. Duarte, F. Nex, M. Gerke, N. Kerle and G. Vosselman (2016). "Potential of Multi-Temporal Oblique Airborne Imagery for Structural Damage Assessment." ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences: 355-362.

Vetrivel, A., M. Gerke, N. Kerle, F. Nex and G. Vosselman (2017). "Disaster damage detection through synergistic use of deep learning and 3D point cloud features derived from very high resolution oblique aerial images, and multiple-kernel-learning." ISPRS Journal of Photogrammetry and Remote Sensing.

Vetrivel, A., M. Gerke, N. Kerle and G. Vosselman (2015). "Identification of damage in buildings based on gaps in 3D point clouds from very high resolution oblique airborne images." ISPRS journal of photogrammetry and remote sensing **105**: 61-78.

Vetrivel, A., M. Gerke, N. Kerle and G. Vosselman (2016). "Identification of structurally damaged areas in airborne oblique images using a visual-Bag-of-Words approach." Remote Sensing **8**(3): 231.

Vijayaraj, V., E. A. Bright and B. L. Bhaduri (2008). Rapid damage assessment from high resolution imagery. Geoscience and Remote Sensing Symposium, 2008. IGARSS 2008. IEEE International, IEEE.

Vögtle, T. and E. Steinle (2004). "Detection and recognition of changes in building geometry derived from multitemporal laserscanning data." International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences **35**(B2): 428-433.

Wang, J. and J. Shan (2009). Segmentation of LiDAR point clouds for building extraction. American Society for Photogramm. Remote Sens. Annual Conference, Baltimore, MD.

Wang, O., S. K. Lodha and D. P. Helmbold (2006). A bayesian approach to building footprint extraction from aerial lidar data. 3D Data Processing, Visualization, and Transmission, Third International Symposium on, IEEE.

Wang, T., L. Chen and Q. Chen (2013). A graph-based plane segmentation approach for noisy point clouds. Control and Decision Conference (CCDC), 2013 25th Chinese, IEEE.

Wang, Y., L. Cheng, Y. Chen, Y. Wu and M. Li (2016). "Building Point Detection from Vehicle-Borne LiDAR Data Based on Voxel Group and Horizontal Hollow Analysis." Remote Sensing **8**(5): 419.

Wei, L., B. Yang, J. Jiang, G. Cao and M. Wu (2017). "Vegetation filtering algorithm for UAV-borne lidar point clouds: a case study in the middle-lower Yangtze River riparian zone." International Journal of Remote Sensing **38**(8-10): 2991-3002.

Westoby, M., J. Brasington, N. Glasser, M. Hambrey and J. Reynolds (2012). "'Structure-from-Motion'photogrammetry: A low-cost, effective tool for geoscience applications." Geomorphology **179**: 300-314.

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

Woolard, J. W. and J. D. Colby (2002). "Spatial characterization, resolution, and volumetric change of coastal dunes using airborne LIDAR: Cape Hatteras, North Carolina." Geomorphology **48**(1): 269-287.

Wu, B., B. Yu, Q. Wu, S. Yao, F. Zhao, W. Mao and J. Wu (2017). "A Graph-Based Approach for 3D Building Model Reconstruction from Airborne LiDAR Point Clouds." Remote Sensing **9**(1): 92.

Wu, C. (2011). "VisualSFM: A visual structure from motion system."

Wu, F., C. Wen, Y. Guo, J. Wang, Y. Yu, C. Wang and J. Li (2017). "Rapid Localization and Extraction of Street Light Poles in Mobile LiDAR Point Clouds: A Supervoxel-Based Approach." IEEE Transactions on Intelligent Transportation Systems **18**(2): 292-305.

Xian, S., N. Lin and A. Hatzikyriakou (2015). "Storm surge damage to residential areas: a quantitative analysis for Hurricane Sandy in comparison with FEMA flood map." Natural Hazards **79**(3): 1867-1888.

Xie, S., J. Duan, S. Liu, Q. Dai, W. Liu, Y. Ma, R. Guo and C. Ma (2016). "Crowdsourcing rapid assessment of collapsed buildings early after the earthquake based on aerial remote sensing image: A case study of yushu earthquake." Remote Sensing **8**(9): 759.

Xiong, X., A. Adan, B. Akinci and D. Huber (2013). "Automatic creation of semantically rich 3D building models from laser scanner data." Automation in Construction **31**: 325-337.

Xu, H., L. Cheng, M. Li, Y. Chen and L. Zhong (2015). "Using octrees to detect changes to buildings and trees in the urban environment from airborne LiDAR data." Remote Sensing **7**(8): 9682-9704.

Xu, L., C. Lu, Y. Xu and J. Jia (2011). Image smoothing via L 0 gradient minimization. ACM Transactions on Graphics (TOG), ACM.

Yang, B. and C. Chen (2015). "Automatic registration of UAV-borne sequent images and LiDAR data." ISPRS Journal of Photogrammetry and Remote Sensing **101**: 262-274.

Ye, X., M. Liu, J. Wang, Q. Qin, H. Ren, J. Wang and J. Hui (2017). "Building-Based Damage Detection From Postquake Image Using Multiple-Feature Analysis." IEEE Geoscience and Remote Sensing Letters **14**(4): 499-503.

Yeum, C. M. (2016). Computer vision-based structural assessment exploiting large volumes of images, Purdue University.

Yim, S. C., M. J. Olsen, K. F. Cheung and M. Azadbakht (2014). "Tsunami modeling, fluid load simulation, and validation using geospatial field data." Journal of Structural Engineering **140**(8): A4014012.

Yonglin, S., W. Lixin and W. Zhi (2010). Identification of inclined buildings from aerial LIDAR data for disaster management. Geoinformatics, 2010 18th International Conference on, IEEE.

Yuan, J. (2016). "Automatic Building Extraction in Aerial Scenes Using Convolutional Networks." arXiv preprint arXiv:1602.06564.

Zeng, Z., J. Wan and H. Liu (2016). "An entropy-based filtering approach for airborne laser scanning data." Infrared Physics & Technology **75**: 87-92.

Zhang, G., P. A. Vela, P. Karasev and I. Brilakis (2015). "A Sparsity-Inducing Optimization-Based Algorithm for Planar Patches Extraction from Noisy Point-Cloud Data." Computer-Aided Civil and Infrastructure Engineering **30**(2): 85-102.

Zhang, K., S.-C. Chen, D. Whitman, M.-L. Shyu, J. Yan and C. Zhang (2003). "A progressive morphological filter for removing nonground measurements from airborne LIDAR data." IEEE transactions on geoscience and remote sensing **41**(4): 872-882.

Zhang, K., D. Whitman, S. Leatherman and W. Robertson (2005). "Quantification of beach changes caused by Hurricane Floyd along Florida's Atlantic coast using airborne laser surveys." Journal of Coastal Research: 123-134.

Zhang, W., J. Qi, P. Wan, H. Wang, D. Xie, X. Wang and G. Yan (2016). "An Easy-to-Use Airborne LiDAR Data Filtering Method Based on Cloth Simulation." Remote Sensing **8**(6): 501.

Zhang, W., J. Zhao, M. Chen, Y. Chen, K. Yan, L. Li, J. Qi, X. Wang, J. Luo and Q. Chu (2015). "Registration of optical imagery and LiDAR data using an inherent geometrical constraint." Optics express **23**(6): 7694-7702.

Zhang, Z. (2000). "A flexible new technique for camera calibration." IEEE Transactions on pattern analysis and machine intelligence **22**(11): 1330-1334.

Zheng, Y., S. Sugimoto and M. Okutomi (2013). "Aspnp: An accurate and scalable solution to the perspective-n-point problem." IEICE TRANSACTIONS on Information and Systems **96**(7): 1525-1535.

Zhou, Q.-Y. and U. Neumann (2013). "Complete residential urban area reconstruction from dense aerial LiDAR point clouds." Graphical Models **75**(3): 118-125.

Zhou, Z., J. Gong and M. Guo (2015). "Image-based 3D reconstruction for posthurricane residential building damage assessment." Journal of Computing in Civil Engineering **30**(2): 04015015.

Zhou, Z., J. Gong, A. Roda and K. Farrag (2016). "Multiresolution Change Analysis Framework for Postdisaster Assessment of Natural Gas Pipeline Risk." Transportation Research Record: Journal of the Transportation Research Board(2595): 29-39.

Zolanvari, S. I. and D. F. Laefer (2016). "Slicing Method for curved façade and window extraction from point clouds." ISPRS Journal of Photogrammetry and Remote Sensing **119**: 334-346.