3D CLASSIFICATION OF POWER LINE SCENE USING AIRBORNE LIDAR DATA

HEUNGSIK KIM

A DISSERTATION SUBMITTED
TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

GRADUATE PROGRAM IN GEOMATICS ENGINEERING
YORK UNIVERSITY
TORTONTO, ONTARIO

MAY 2015

Abstract

Failure to adequately maintain vegetation within a power line corridor has been identified as a main cause of the August 14, 2003 electric power blackout. Such that, timely and accurate corridor mapping and monitoring are indispensible to mitigate such disaster. Moreover, airborne LiDAR (Light Detection And Ranging) has been recently introduced and widely utilized in industries and academies thanks to its potential to automate the data processing for scene analysis including power line corridor mapping. However, today's corridor mapping practice using LiDAR in industries still remains an expensive manual process that is not suitable for the large-scale, rapid commercial compilation of corridor maps. Additionally, in academies only few studies have developed algorithms capable of recognizing corridor objects in the power line scene, which are mostly based on 2-dimensional classification. Thus, the objective of this dissertation is to develop a 3-dimensional classification system which is able to automatically identify key objects in the power line corridor from large-scale LiDAR data. This dissertation introduces new features for power structures, especially for the electric pylon, and existing features which are derived through diverse piecewise (i.e., point, line and plane) feature extraction, and then constructs a classification model pool by building individual models according to the piecewise feature sets and diverse voltage training samples using Random Forests. Finally, this dissertation proposes a Multiple Classifier System (MCS) which provides an optimal committee of models from the model pool for classification of new incoming power line scene. The proposed MCS has been tested on a power line corridor where medium voltage transmission lines (115 kV and 230 kV) pass.

The classification results based on the MCS applied by optimally selecting the pre-built classification models according to the voltage type of the test corridor demonstrate a good accuracy (89.07%) and computationally effective time cost (approximately 4 hours/km) without additional training fees.

Acknowledgments

I would like to express the deepest appreciation to my supervisor, Dr. Gunho Sohn, who provided me with his excellent guidance and great encouragement. Without intellectual and financial supports from him, I would never have been able to accomplish my research and this dissertation. He is all the time full of the joy and enthusiasm for the research achievements. I hope that I could be as lively, enthusiastic, energetic, and confident as him at home and work.

I would like to extend my gratitude to the committee members, Dr. Costas Armenakis, Dr. Baoxin Hu, Dr. Regina Lee, Dr. Ruisheng Wang, and Dr. Minas Spetsakis, who offered many valuable advices and insightful discussions and suggestions on my research. Such great helps from them resulted in a considerable improvement over this dissertation.

I would like to thank all the faculty members and graduate students in ESSE at York University, especially Marcia Gaynor who made hard works and dedications as a graduate program assistant and collaborators in GeoICT Lab., Yoonseok, Bruce, and Solomon. I would like to be more deeply thankful to Connie, who spent her time to review the first draft of my dissertation.

I would also like to acknowledge Ontario Centres of Excellence (OCE), Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery, MITACS,

GeoDigital International Inc. (GDI), and Weka developers as important contributors to my study with respect to financial support and experimental data and software provision.

Finally, I am indebted to my family, who encouraged me and prayed for me throughout the time of my research. I am grateful to my father who trusted me and allowed me to the study abroad.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

ALS	Airborne Laser Scanning
CRF	Conditional Random Field
DTM	Digital Terrain Model
EM	Expectation Maximization
EMCS	Extended Multiple Classifier System
FI	Feature Importance
FMCS	Feature-based Multiple Classifier System
GEOBIA	GEographic Object-Based Image Analysis
GMM	Gaussian Mixture Model
GPS	Global Positioning System
IB	In Bag
IMU	Inertial Measurement Unit
kV	kilovolt
LiDAR	Light Detection And Ranging
MCS	Multiple Classifier System
MLC	Maximum Likelihood Classifier
MLS	Mobile Laser Scanning
MRF	Markov Random Field
OBF	Object Based Feature
OBPA	Object-Based Point cloud Analysis
OOB	Out Of Bag
PBF	Point Based Feature
PCA	Principle Component Analysis
PLS-CADD	Power Line System-Computer Aided Design and Drafting
QA/QC	Quality Assurance/Quality Control
RANSAC	RANdom SAmple Consensus
ROW	Right Of Way
SHOALS	Scanning Hydrographic Operational Airborne LiDAR Survey
SMCS	Scene-based Multiple Classifier System
SVM	Support Vector Machine
TLS	Terrestrial Laser Scanning
UAV	Unmanned Aerial Vehicle

1. Introduction

1.1. Motivations

Inadequate vegetation management within transmission line right-of-way (ROW), a segment of land used for the route of a transmission line, has been reported as a main cause of 2003 North America blackout (Final Report on the August 14, 2003 Blackout, 2004). This power outage led us for an awareness of importance of effective vegetation managements on and near the ROW. It affected an estimated 10 million people in Ontario and 40 million people in eight states in USA. Outage-related financial losses were estimated at $7 -10 billion USD. The Final report (Final Report on the August 14, 2003 Blackout, 2004) submitted to Congress stated that

"had all trees which contributed to the August 14th outage been adequately pruned or removed prior to the event, the blackout would likely not have occurred."……the vegetation and corridor management operations of the offending companies were within the range of current "average" industry standards…..we believe and strongly recommend that the industry "average" or standard needs to be substantially improved."

The failure of vegetation management was also a common factor in contributing 64 local outages in America since the 2003 Blackout (NERC annual report, 2013). The North American Electric Reliability Corporation (NERC) has designed transmission vegetation management compliance, FAC-003-1, to improve the reliability of the electric transmission systems by preventing outages from vegetation on ROW, minimizing

outages from vegetation adjacent to ROW, maintaining clearances between transmission lines and vegetation, and reporting vegetation related outages to regional transmission organizations (FAC-003-1, 2005).

Besides the standards, these vegetation related outages could easily have been mitigated by early fault diagnosis using remote sensing technology. However, current acquisition costs and the related manually intensive process of reviewing the remotely sensed data make such a service impractical and commercially unviable. To overcome this limitation an automated process is urgently required to determine the potential impact on power-line safety by observing changes in clearances to vegetation and other objects, operating temperature, the detection of new buildings or structures alongside or between towers and their associated line spans, erosion-induced terrain changes, and tree health and the detection of physical damage or deterioration of structures, wires or other assets. Traditionally such corridor analysis has relied on labour intensive manual approaches that entail manual inspection or the capture and inspection of video footage captured on site by ground personnel or during airborne patrols. Airborne LiDAR shows great potential for the cost effective capture of corridor information for mapping and inspection. This is because they can without the need for extensive ground control provide high dense point cloud of corridor objects with high density and three-dimensional information. In addition to their efficient and robust geo-referencing capability, the sensors provide a wide range of corridor information, including laser intensity, discrete laser echoes, and waveform data which provides an almost unlimited number of laser returns and thus results in a more detailed description of object structure. LiDAR having these informative

capacities has only very recently been introduced for use in corridor mapping applications and consequently research in this area is far from mature (Ussyshkin and Smith, 2007).

The scene analysis for the power-line change monitoring is a tedious task as it requires precise detection of all key corridor objects (i.e., power-lines, towers, insulators, splices, switches and other components as well as the terrain, buildings, trees, etc). Once data related to a line is established, a careful periodic comparison of the model to future datasets taken at different times under possibly varying conditions is necessary to detect changes. At present, the state-of-the-art technologies still utilize simple processing that works with a high level of manual interaction and supervision. Achieving a high level of automation in data processing chains is urgently required in order to meet the needs of the power-line industry. The complexity of corridor scene content and the huge data size currently exceeds the ability of current state-of–the-art machine vision systems or LiDAR processing systems to accurately detect corridor objects and safely recognize all possible threats to a power-line systems and human intervention and inspection is as such the primary methodology, which is costly, slow, tedious and expensive. Even partial automation of this task will greatly increase an analyst's productivity and enhance the reliability of the results.

Many utility companies have considerable amount of knowledge obtained through repetitive works such as classification, structure modeling, clearance report, etc. However, they do not utilize the gained knowledge for future works even though the work is being

done in the same area. Such sufficient knowledge contains assigned class labels for individual LiDAR points, transmission line model, structure position, structure type, and vegetation violation area, and corridor description (e.g., voltage type, topographic information, and so on) as well as raw LiDAR data. The knowledge can be categorized and analyzed to extract additional information such as features, statistics, and metrics for each category. This accumulated knowledge, recently referable as "Big Data" (MIKE 2.0), in terms of classification problem leads for more accurate and precise mapping of unknown data if it is possible to filter out appropriate knowledge according to very fundamental information of the data such as, for an example, voltage type, pylon type, and so on for power line corridor. Here, an issue on the "Big Data" is how to integrate and mine heterogeneous and complex information obtained from different sources, e.g., multiple data captured by different sensors at different time and multiple features extracted from different samples. A Multiple Classifier System (MCS), also referred as an ensemble system, potentially provides advantageous framework for reducing computational overhead and complexity (Crawford and Kim, 2009) by not directly handling a complex problem but decomposing it into sub-problems and combining individual solutions to them. Such consensus decision making of MCS has great strength in reducing the risk of leading to poor decisions (Dara, 2007).

In summary, this study is motivated by following:

- *Demands of power line mapping for effective vegetation management in ROW*
- *Potentials of airborne LiDAR emerging with cost-effectiveness*

- *Demands of automated processing*

- *Classification ensemble using already gained knowledge*

1.2. Objectives

Power line mapping today is an on-site surveying process that urgently requires the improved capability to quickly and accurately detect, classify and monitor objects within the corridor (Flood, 2011). These key corridor objects include terrain, vegetation, towers, power lines, buildings, roads and waterways. Traditionally such corridor scene analyses have relied on labour-intensive manual approaches that entail investigation of video footage captured on site. Recently, however, airborne LiDAR (Light Detection And Ranging) has attracted much attention for its potential to automate the complicated data processing tasks required for corridor scene analysis (Ituen and Sohn, 2010). This is because airborne LiDAR can rapidly provide highly dense and accurate three-dimensional (3D) information of corridor objects without the use of ground control. Having such high quality 3D information will facilitate difficult photogrammetric computer vision tasks, such as feature extraction, feature grouping and contextual analysis. Despite its potential, thus far not much research effort has been made employing airborne LiDAR data to explore classification methods of the corridor scenes. Hence, today's corridor mapping practice still remains an expensive manual process that is not suitable for the large-scale, rapid commercial compilation of corridor maps (Liang et al., 2011). Apart from the automation, another challenging issue of the power line corridor classification is to resolve difficulty in handling intra-class variation, particularly

electricity pylon exhibiting different shape, size, material (e.g., steel, wood, and concrete), and structure (e.g., pole and truss) depending on deploying region and supplying voltage. To address the indicated issue, this dissertation aims at developing an automated and knowledge-based classification method of power line corridor from LiDAR data. To reach the research aim, following achievements are required:

1. Development and investigation of diverse LiDAR features for the corridor objects (i.e., terrain, power line, electric pylon, building, vegetation, and remaining objects), each of which is able to distinguish one class object from others.

2. Examination of the potential of a supervised learning classifier, specifically Random Forests, in the classification of the corridor objects with airborne LiDAR data.

3. Production of multiple classifiers with considerations of diversity by training them using either different feature sets or different training samples.

4. Construction of an ensemble system which accommodates the built classifiers and provides an optimal committee of classifiers for unknown LiDAR data to be classified.

1.3. Contributions

Three most significant contributions of this dissertation to power line scene classification using LiDAR are enumerated. The first is automation in classification

which is able to classify large-scale corridors time-effectively. The second is providing a robust classification method which is highly flexible about variances within the same class object. The third is the use of knowledge-based classification which recycles already gained information from the past classification works as cues for future works. The aforementioned contributions are detailed as follows:

Automation in classification: LiDAR system can capture a large region, an entire corridor (i.e., station to station) for power line, and represent its scene with massive point cloud. For example, raw LiDAR data for a 500kV power line corridor (200km length and 150m width) near a forest area is recorded approximately 40GB (Giga Bytes) in a binary format and 0.2 billion points. Such a huge data can be collected in several days, but manually processing it is not achievable in time. To solve the limitation with manual processing, this dissertation designs and implements a prototype of classification ensemble system which achieves high rate of automation (larger than 90%) in feature extraction, classifier building, classifier optimization, classifier selection, and classifier fusion. Human hands are only required for converting outputs at the current stage into inputs at the next stage. Moreover, the system is able to classify 1km of power line corridor in approximately four hours, which is more time-efficient than manual classification (8 hours/km from our experience).

Robust classification: A critical problem with traditional supervised classification methods is building training samples for new data and producing reference data by manually classifying the selected training samples. This task will be repeated for new

data or data showing different characteristics from the training samples, i.e., a classifier built with 69 kV samples, which contains distribution lines and small pylons, cannot be applied to 500 kV data, which contains bundled transmission lines and complex shaped tall pylons. The suggested method in this dissertation constructs an ensemble system and obtains diverse classifiers, each of which is built with samples taken in a specific power line environment. This study categorizes power line corridors according to their carrying voltage. For classification of an unlabeled corridor the ensemble system makes most relevant classifiers to the corridor (i.e., classifiers trained using same voltage type of training sample as the corridor) greatly involve in the classification by assigning higher weight compared to other classifiers when classifiers are combined. For example, the weight value for the relevant classifiers is 1, while 0 for others. Without training new additional classifiers, the experimental results in chapter 5 present that the ensemble system considering characteristics of scene objects yields better classification accuracy (89.79%) than the best one (88.05%) among the single classifiers.

Knowledge-based classification: Considering repetitive working characteristics of power-line monitoring, it is important to support a knowledge-based mechanism for power line scene classification. However, current state-of-the-art solutions still remain as being static and deterministic decision-making process. Moreover, even though power utility companies have possessed considerable knowledge obtained from classification works carried out in the past for the monitoring, they do not recycle the gained knowledge for future classification jobs. The reuse of such knowledge will reduce a great amount of workload and financial burden for the companies. Engaging with this study,

the knowledge can be used to build a large database where a lot of classifiers live and they are able to discriminate the diversities in class, physical shape, size, site, voltage type, and so on. An ensemble system selects a committee of classifier by considering characteristics of scene objects in new incoming data without additional training and manual classification jobs and classifies the data. Due to a difficulty in obtaining company's assets and building the database (numbers of classifiers need to be trained), this study develops a proto type of the ensemble system and demonstrates a potential of practical use in power utility industries.

1.4. Overview of the Proposed Approach

This dissertation examined three different classification approaches to classify LiDAR data of power line corridors. However, first two methods (Chapter 3 and 4) are developed to construct a Multiple Classifier System (Chapter 5) which is suggested in this dissertation.

Classification using point-based features (Research Phase I): Traditional classification methods using airborne LiDAR mostly treat not individual points, but individual pixels which are generated by interpolating LiDAR points into a 2D grid space. These methods, called grid- or pixel-based classification, perform feature extraction and classification for each pixel, which assign a class label to all the membership points in the pixel. Thus, they are straightforward and low cost in terms of computation complexity. However, they are inappropriate for the power line corridor classification due to a limitation with discriminating multiple class objects in case present in the pixel which

frequently happens in power line corridors such as a vertical overlap of transmission line and terrain. This dissertation therefore suggests point-based classification that extracts features and assigns a class label for each single point. 21 LiDAR features are considered to characterize each of key corridor objects and supervised learning method, Random Forests, is employed for building classification model from the features. This study has improved classification performance through the classification model optimization including optimal training sample selection, balanced learning, relevant feature selection, and feature de-correlation.

Classification using object-based features (Research Phase II): In the aforementioned point-based classification method, the point features extracted from neighbors affect the class labeling. For such reason, classification errors are occasionally observed over some regions, where neighbors are not sufficiently gathered such as building roof edges, tree tops, etc. This error pattern, called "salt and pepper" effect in image classification problem, is frequently shown in the pixel-based classification method. To remove the pattern, Hay et al. (2005) introduced Geographic Object-Based Image Analysis (GEOBIA) which partitions an image into meaningful pixel groups and treats the groups as spatial units for classification. Likely, for point cloud Object-Based Point cloud Analysis (OBPA) has been introduced by Rutzinger et al. (2006). The OBPA concept is employed to lead for classification improvement over class objects having a representative spatial characteristic such as plane-likeness. This dissertation produces objects (groups of points) by transforming point cloud into two object domains, line and plane which are able to characterize line- (e.g., transmission line) and plane-like objects

(e.g., building roof) respectively, using a segmentation algorithm. Object features are then derived from each point group and interaction with its neighbors. Also, this approach utilizes Random Forests to build a classification model and performs same optimization procedures introduced in the point-based classification.

Classification using Multiple Classifier System (Research Phase III): This dissertation finally introduces MCS, also referred as ensemble classifier or classifier fusion, which combines a committee of pre-built classification models (or classifiers) for decision making. The strength of the MCS is that it mostly outperforms any of single models if high diversity between its membership models is maintained. For such a great advantage this study applies the MCS framework to multiple classification models early produced through point- and object-based feature extraction and classification, termed Feature-based MCS (FMCS). However, a problem with the FMCS is no guarantee of consistent results over power line corridors showing diverse characteristics for a class object, i.e., intra-class variations. As a solution, this dissertation builds multiple classifiers trained under different environment per carrying voltage: 69kV, 115kV, 230kV and 500kV. It is supposed that a certain class object, especially transmission line and electric pylon, in same voltage corridors presents similar spatial characteristics and this assumption is practically convincing due to their construction compliance restricted on voltage type. In the MCS, the multiple classifiers are optimally selected depending on the voltage type of new incoming corridor and they are combined for classification, termed Scene-based MCS (SMCS). This dissertation finally designs a hierarchical MCS composed of the FMCS and SMCS, termed Extended MCS (EMCS), which is the

suggested classification system through this dissertation. Figure 1-1 shows an overview of the EMCS (chapter 5), the classification model pool build-up from models produced by the point- (chapter 3) and object-based classification method (chapter 4), and how to classify an unknown scene using the model pool. The EMCS optimally selects 115 kV and 230 kV models based on prior information that the unknown scene is a medium voltage corridor where a range of 115 to 230 kV transmission lines pass. Then it combines outputs from the selected classification models to get classification results of the corridor.

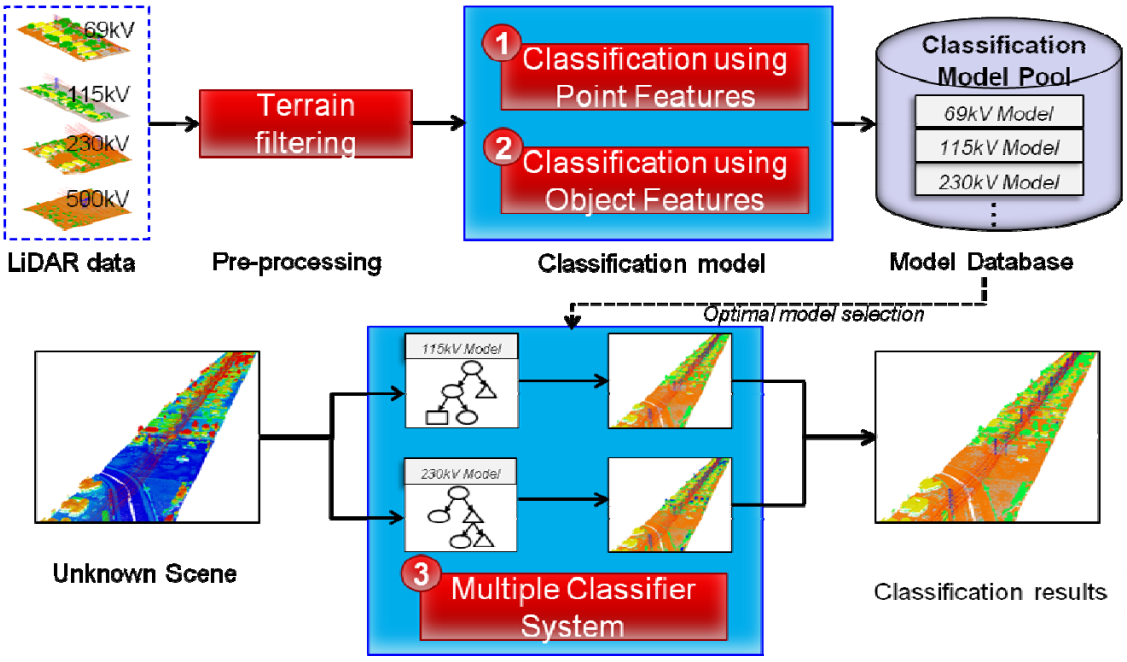


Figure 1-1. Overview of the proposed Multiple Classifier System

1.5. Thesis Outline

This dissertation is divided into six chapters:

Chapter 1 presents an introduction of this study including motivations, objectives, and contributions.

Chapter 2 addresses backgrounds to understand this thesis and provides a literature review to highlight a need of this study.

Chapter 3-5 introduce methodologies challenged and suggested in this study. Chapter 3 and 4 introduce two feature sets, point-based and object-based features, to separate power line corridor objects, each of which is derived through a specific piecewise analysis. Also, in those chapters a supervised learning algorithm, i.e., Random Forests, is introduced to build a classification model from each feature set. Chapter 5 introduces a Multiple Classifier System (MCS), which is a proposed classification system. The MCS combines the pre-built classification models generated from the features addressed in chapter 3 and 4 and the training samples per voltage type.

Chapter 6 summarizes the main conclusion of this dissertation and presents outlooks of future works for more improvements.

2. Literature Review

2.1. LiDAR System Introduction

LiDAR (**Li**ght **D**etection **A**nd **R**anging) or laser scanning system is a data collection method based on the laser mechanism, which leads to direct and cost-effective 3D measurement with high accuracy. The data collected from the LiDAR systems is commonly a point cloud representing the scanning targets such as a city, forest, coast, road, railway, tunnel, building façade, and so on. The point cloud is geo-referenced, i.e., positioned by x, y, and z values on a certain coordinate system, with the assistance from GPS (Global Positioning System) and IMU (Inertial Measurement Unit). Such geo-referenced data is utilized for the purpose of mapping, recognition, and classification. Depending on how the laser scanner is being mounted, the term of the LiDAR system is interchangeable with Airborne Laser Scanning (ALS) on an airplane or helicopter; Mobile Laser Scanning (MLS) on a vehicle or train; and Terrestrial Laser Scanning (TLS) system on a tripod or other stationary mount. Recently, Unmanned Aerial Vehicle (UAV), an aircraft without a human pilot aboard, is becoming the most promising platform for a laser scanner for economic reasons. However, the data processing techniques needed to produce a point cloud from raw data acquired by the UAV system are not concrete. The UAV system requires more carefulness to generate a point cloud complete on geometric quality because the UAV is more sensitive to the platform fluctuation and vibration than the ALS. Thus, the ALS system has more benefits in data quality, collection speed, and scanning coverage compared with other LiDAR systems.

Due to the advantages mentioned above, this study utilized LiDAR data acquired by the ALS system for the power line corridor classification.

2.1.1. **Airborne Laser Scanning (ALS) system**

Early day ALS system has been introduced in the 1970's by NASA to use it as a prototype for the practical sensor deployment on spaceships (McCormick, 2005). Since then, the ALS system evolved with the advancement of GPS and IMU technology. Moreover, these days it has been widely applied to various fields such as urban planning (Yu et al., 2010; Sohn and Dowman, 2007), forest inventory analysis (Coulston et al., 2012; Yu et al., 2011), coastal area mapping (Richter et al, 2011; Nayegandhi and Brock, 2009; Chust et al., 2008), corridor mapping (Jwa and Sohn, 2012; Yao et al., 2011; McLaughlin, 2006), DTM (Digital Terrain Model) generation (Lu et al., 2009; Sohn and Dowman, 2008) for its advantage of a precise and rapid data collection. The ALS system is typically composed of laser scanners, GPS and IMU. The leading manufacturers of the laser scanner are RIEGL (RIEGL), Optech (Optech), and Leica Geosystems (Leica) and their scanner models are LMS-Q series, ALTM, and ALS, respectively. The GPS and IMU are responsible for tracking and attitude measuring of an airplane respectively, where Applanix (Applanix) is a major provider of GPS/IMU system named POS AV. Also, there are several companies possessing airplanes integrated with the aforementioned laser scanner and GPS/IMU (i.e., built-in ALS systems): TopEye, Frugo, and TopoSys. And they operate the ALS system and provide LiDAR data for clients.

The LiDAR is an active sensor which sends a narrow and high energy ray from a laser transmitter to the scene and records the reflected energies. The basic principle of LiDAR data acquisition is based on the measurement of the laser ray direction and the ray travel distance. First, the laser beam fired from the transmitter on the airplane flying at a certain altitude reaches a target (e.g., ground). Secondly, it returns back to the receiver after the target reflection. Third, the respective transmission and reflection produce transmitted and received signals. As a result, the travel time (t_{travel}) of the laser ray can be derived from the time difference between the signals, and the round trip distance (2ρ) of the ray is calculated using the t_{travel} and laser speed (i.e., light speed), c. Finally, the range measurement (ρ) from the laser to the target is obtained as seen in Eq. 2-1.

$$\rho = \frac{c \times t_{travel}}{2} \tag{2-1}$$

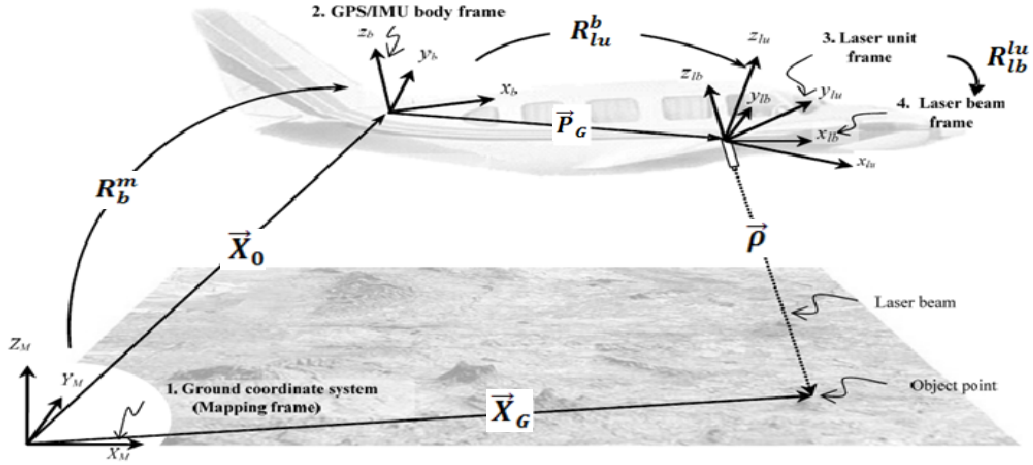


Figure 2-1. An overview of airborne LiDAR system and the basic principle for geo-location (Bang, 2010)

For more detailed description of the LiDAR geo-location as shown in Eq. 2-2. And Figure 2-1, the target coordinate ($\vec{X}_G$) is simply computed by the transformations between the four reference frames when the LiDAR systematic errors addressed in the following section are ignored: the mapping, GPS/IMU body, laser unit, and laser beam frame (Bang, 2010). The laser unit frame indicates a constant coordinate system of the laser body. While, the laser beam frame considers the variations of laser scan angle during scanning, so the coordinate system varies every laser transmit.

$$\vec{X}_G = \vec{X}_0 + R_b^m \cdot \vec{P}_G + R_b^m \cdot R_{lu}^b \cdot R_{lb}^{lu} \cdot \vec{\rho} \tag{2-2}$$

where,

$\vec{X}_0$	coordinates of GPS/IMU body on the mapping frame
$\vec{P}_G$	relative coordinates between the GPS/IMU body frame to the laser unit frame
$\vec{\rho}$	relative coordinates between the laser unit frame and the target on the mapping frame
R_b^m	a rotation matrix between the mapping and GPS/IMU body frame
R_{lu}^b	a rotation matrix between the GPS/IMU body and laser unit frame
R_{lb}^{lu}	a rotation matrix between the laser unit and laser beam frame

However, practically more variables corresponding to random and systematic error need to be introduced in the LiDAR equation (Habib et al., 2009). The random errors are caused by imprecision of the instrumental measurement, while the systematic errors occur due to biases in the mounting parameters of the system components. In order to improve data quality by reducing such errors, the ALS system requires calibrations before and after a flight. Two calibrations are typically performed: system calibration, which ensures that individual hardware (scanner, IMU, and GPS) meets its allowable specification and that determines relative offsets between them. Whereas data calibration,

which leads to geometric and radiometric correction by conducting Quality Assurance/Quality Control (QA/QC) and intensity calibration respectively. The QA/QC verifies data coverage and compares elevations in strip overlap areas for the spatial correspondence between the strips (Habib et al., 2008). Figure 2-2 illustrates the results of the strip calibration by using conjugate points in overlapping two strips (red and blue).

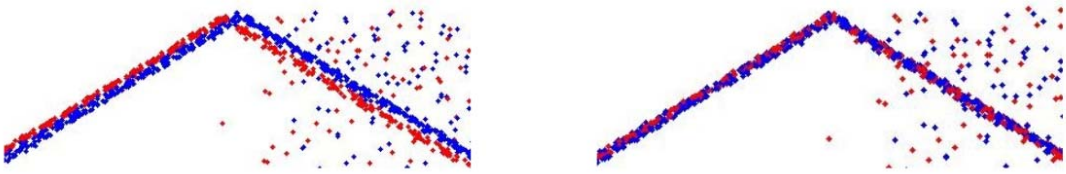


Figure 2-2. Relative misalignment between strips before and after strip adjustment

Such LiDAR data spatially corrected by the QA/QC and calibration is extensively used for many applications due to following advantages: (1) high accuracy (vertical accuracy of ±15cm and horizontal accuracy of ±25cm at 3000m altitude), (2) fast data acquisition and big coverage, (3) weather/light independence, (4) canopy penetration, and (5) GCP (Ground Control Point) independence. The next section describes applications utilizing the LiDAR.

2.1.2. **Applications**

As the demand of high accuracy and rapid data acquisition increases in a variety of surveying and mapping fields, traditional field survey on sites is no longer a solution because it does not meet both requirements. On the other hand, the ALS system provides 3D data with high accuracy, density, and cost-effectiveness. Over the past two decades,

the use of the ALS system has incrementally grown in terms of real-world application and recently, the LiDAR data is broadly applied for urban planning, forest mapping, coastal zone mapping, bathymetry mapping, and corridor mapping.

Urban planning

Amongst many LiDAR applications, urban mapping is the most typical application and classification of urban facilities based on their location and properties is a primary job in the urban mapping. Depending on the purpose and nature of the experiment, the facilities are classified into buildings (residential houses, commercial buildings, etc.), vegetation (tree, grass, etc.), road facilities (vehicles, roads, road sign, etc.) and so on. Such extracted mapping information could be integrated with other supplementary data such as digital map (Mason et al., 2007), optical image (Cheng et al., 2011; Lee et al., 2008), and road network (Yao et al., 2011) to increase information quality and complexity. For example, 3D building modeling approaches using a fusion of building boundary and height information from the respective digital maps and LiDAR tend to bring better results than ones using LiDAR alone (You and Lin, 2011). This fusion approach has been introduced in many studies where building roof models are derived from LiDAR and building boundary lines are extracted from optical images (Cheng et al., 2011; Lee et al., 2008; Rottensteiner et al., 2005). Alternatively, LiDAR change detection algorithm contributes to modifying and updating existing urban data (Hebel et al., 2013; Matikainen et al., 2010) and this automatic data update decreases the need of additional data acquisition and brings the cost reduction. Furthermore, 3D full analysis using LiDAR allows for 3D city modeling at a highly accurate rate. For vegetation (mostly

trees), two of the most popular applications are to classify vegetation types from raw LiDAR point cloud (Ko et al., 2013; Yao et al., 2012) and to separate single trees from the classified vegetation point cloud (Zhang et al., 2014; Yao and Wei, 2013). The individual trees are identified in terms of theirs species based on tree measures such as tree shape, height, crown size (Zhang et al., 2014) and then modeled in 3D depending on the tree inventories and species (Côté et al., 2009). This information is crucial for estimating tree biomass (Hecht et al., 2008) which can be used as a good estimation of carbon storage in a city (Raciti et al., 2014). On top of that, LiDAR data can also be useful for road extraction (Choi et al., 2008), vehicle detection (Yao and Stilla, 2011a), traffic monitoring & flow estimation (Yao et al., 2011), etc. However, recently MLS (Pu et al., 2011; Brenner, 2009) attracts more interest for those applications as compared with ALS since it produces much denser point cloud and has anticipated for better experiment results.

DTM generation

The generation of DTM (also referred as bare-earth surface) has been one of the most basic usage of LiDAR technology in the past and present. The DTM generation requires filtering out the ground (or terrain) from raw LiDAR data so that the above ground points can be separated from the ground points. Many terrain filtering algorithms have been developed over the past ten years. Examples are *simple filtering* which assigns a point with the lowest elevation in a local region to ground; *morphological filtering* which extends ground points if they are within a distance threshold (planimetric distance) to a seeded ground point (Vosselman, 2000); *recursive filtering* which recursively updates a

reference terrain surface by adding ground points obtained from the topological analysis between the previous reference surface and unattached points to the surface (Sohn and Dowman, 2008); *surface-based filtering* which removes above ground points from a surface model; which is initially constructed using all the points (Brovelli et al., 2004; Pfeifer et al.,2001) and *segment-based filtering* which identifies ground segments (points with a homogeneity are grouped as a segment) by comparing surface normal between already ground-assigned segments and others (Filin and Pfeifer, 2006; Sithole, 2005). To generate DTM, such classified ground points are converted into one of the formats, TIN (Triangulated Irregular Network), grid, mesh, and quad-tree. Regions occupied by the above ground points before the filtering have no ground point, so their elevations are inferred from the elevations of neighborhood for a smooth representation of the bare-earth surface. This process is called interpolation.

Vegetation mapping

In vegetation mapping, the fact that a laser shot can penetrate into the foliage brings an extra attention to this study area because LiDAR data can populate randomly scattering points over tree stems, branches, leaves, shrubs, and grass which are mostly invisible on optical images. Moreover, a single LiDAR pulse is capable of generating multiple returns over trees and the returned echo patterns are different from non-penetrable objects such as open ground and buildings. These characteristic contribute to the wide use of LiDAR for the vegetation mapping in both forested (Zhang et al., 2014; Ko et al., 2013) and urban areas (Yao and Wei, 2013; Hecht et al., 2008; Rutzinger et al., 2008). The identified tree points are used for individual tree detection; that is the

grouping of a set of points into a single tree (Zhang et al., 2014; Yao and Wei, 2013). As ecological applications, studies for dead tree detection (Kim et al., 2009) and tree habitat mapping (Hill et al., 2002; Hyde et al., 2005) have examined. In these applications LiDAR is most often fused with optical images. Since LiDAR data over a tree contains 3D point cloud of tree foliages, tree inventory metrics can be estimated such as Leaf Area Index (LAI), tree biomass, tree height, crown size, and diameter. Based on the retrieved or inferred information (tree species and tree inventory), 3D tree modeling studies have been performed (Côté et al., 2009). In addition to vegetation mapping, LiDAR data can be apply to the risk assessment and vegetation management such as the power line anomaly detection (Mills et al., 2010), forest fire modeling (Riano et al., 2003), and pest control (Coops et al., 2009).

Coastal zone mapping

In general, topographic ALS acquires no return over water bodies. This characteristic allows experiments such as coast line detection, which is deriving an adjacency between water body and land (Smeeckaert et al., 2013; Stockdon et al., 2002). Furthermore, remotely sensed LiDAR over inaccessible regions such as coastal cliffs and salt marshes is used to investigate an ecosystem in those regions (Kulawardhana et al., 2014). Spatial-temporal LiDAR can analyze changes in a coastal zone such as erosion caused by sea waves on coastal cliffs (Richter et al., 2013), coastal line change by sediments (Adam, 2006; Hilary et al. 2002), and sand volume variation by winds (Sallenger et al., 2003). A fusion with other sensory data, especially multi-spectral imagery, enables to conduct

habitat mapping for animals (Chust et al., 2008) and plants (Nayegandhi and Brock, 2009) which live near the shore.

Bathymetry mapping

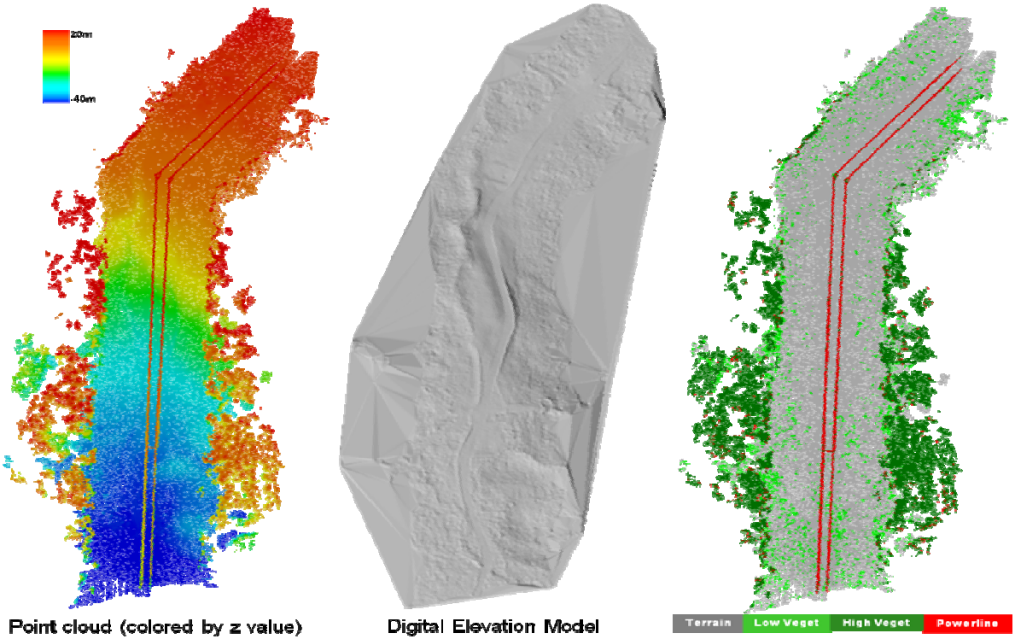
As mentioned, topographic ALS system cannot get returns from water surface and water bottom because the laser pulse cannot travel through the water column. To compensate this limit, Optech Inc. manufactured a new ALS system, SHOALS (Scanning Hydrographic Operational Airborne LiDAR Survey) to maximize the laser penetration in the water by using a laser scanner emitting a short green pulse. Two bathymetric LiDAR systems, SHOALS 1000 and 3000 have been developed and are able to receive laser returns from coastal water with the maximum of 50 m water depth. Typically, the bathymetric LiDAR records two returns from a laser pulse; the first return occurs from water surface (surface return) and the second return is reflected from seabed (benthic return). Such two returns are collected and utilized to compute water depths (Durand et al., 2008). The water depth obtained from the bathymetric LiDAR is combined with DTM from the topographic LiDAR to produce a seamless evaluation model in a coastal zone covering shore and sea (Gesch and Wilson, 2001). Another characteristic of the bathymetric LiDAR is that the received signal contains a lot of backscattering information from the particles in the water as high turbidity weakens the laser penetration. By considering such undesirable effect of the water turbidity the depth estimation can be improved. The bathymetric LiDAR is also employed for seabed habitat mapping (Chust et al, 2010) and benthic classification (Collin et al., 2012; Narayanan et al., 2011) by using different benthic return patterns depending on seabed targets. Apart from SHOALS,

other popular bathymetry systems includes LADS (Fugro LADS) and Hawk Eye (Leica AHAB) manufactured by Fugro and Leica respectively.

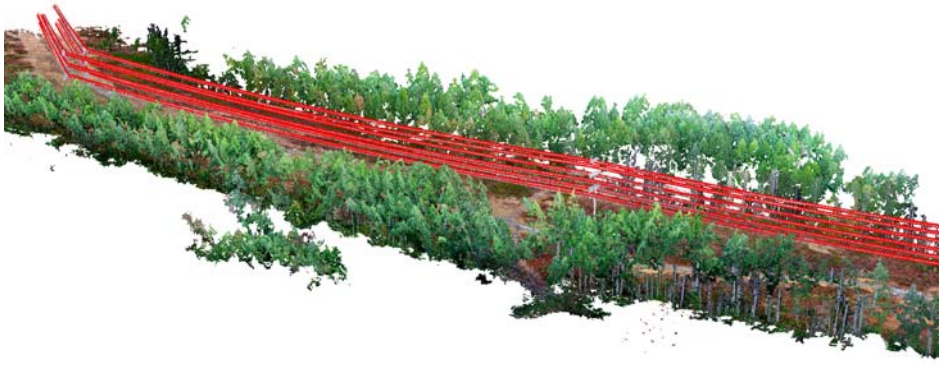
Corridor mapping

Corridor mapping is the mapping of a region that has a narrow width and long length such as the mapping of power line, river, and road. Therefore, ALS systems for corridor mapping (<300m) operate at lower altitude compared with ones for urban mapping (>1000m). Such low-altitude airborne LiDAR leads for a dense point cloud. The power line corridor mapping (Figure 2-3) is one of the most crucial applications in countries with a relatively large continent such as United States, Canada, India and China because a massive number of power line utilities are needed to be built across the continent and regularly monitoring such utilities are too expensive. The main objective of monitoring the power line infrastructures is to find in advance potential risks threatening the power delivery. For more accurate power line anomaly detection, 3D power line corridor classification (Kim and Sohn, 2013) and 3D transmission line modeling (Jwa and Sohn, 2012) have been studied from airborne LiDAR. Figure 2-3 shows results of classification (Figure 2-3(a)) and transmission line modeling (Figure 2-3(b)) from a corridor point cloud which is produced from optical images collected by a UAV, called photogrammetric point cloud. For another corridor mapping in river environment, the aforementioned bathymetric LiDAR is applied to estimate underwater bed elevations (Moretto et al., 2014). Furthermore, LiDAR data acquired for the same area on different dates is used for detecting changes on the river and its vicinity such as bank erosions (De Rose and Basher, 2011) and sediment transport amount estimation (Brasington et al.,

2003). Along with the river bathymetry, other general applications such as 3D road extraction (Choi et al., 2008) and road centerline localization (Cai and Rasdorf, 2008) are designed. Additionally, there are road related applications for traffic monitoring such as vehicle detection and motion estimation (Yao and Stilla, 2011a; Yao et al., 2011b). Typically, MLS is more utilized than ALS for road applications since it collects denser and more accurate point cloud.



(a) Corridor classification results



(b) Power line modeling results with colorized point cloud (converted from optical images captured by UAV)

Figure 2-3. LiDAR application for corridor mapping

2.2. Classification

The classification of LiDAR points into objects is known as a computer vision process for scene understanding. This process transforms remotely sensed data into information that can be utilized for application by properly assigning the class labels from a finite set of object classes to unknown observations. Supervised classification learning is an increasing trend in remote sensing community due to its robustness to given data. The literature review demonstrates two approaches in the supervised classification: generative and discriminative methods. Generative approach derives a posterior probability distribution over the data and class labels based on Bayesian theorem, whereas the discriminative approach directly models the data or estimates a decision boundary (Bishop, 2006) between the given data. For LiDAR point cloud classification, the discriminative approach such as Support Vector Machine (SVM) (Samadzadegan et al., 2010; Lodha et al., 2006; Foody and Mathur, 2004), ensemble methods (decision tree, boosting, bagging, and Random Forests) (Lodha et al., 2007a; Guo et al., 2011; Carlberg et al., 2009), and Conditional Random Fields (CRF) (Niemeyer et al., 2011) are popular methods due to the high speed, simple model and operational capability to handle large volume of data compared to the generative approach such as Bayesian classifier (Neuenschwander, 2009), Gaussian Mixture Model (GMM) (Lodha et al., 2007), and Marko Random Fields (MRF) (Zhang and Sohn, 2010) according to Niemeyer et al. (2014). On top of these methods, supervised classification can be applied with a graph-based representation to examine relevance on the class label (in MRF) or on the feature value (in CRF) with the information provided by the

neighboring values. This graph-based approach shows a higher classification quality (Lu et al., 2009). Another way for classification is to create a classification rule, called rule-based approach which generates a decision rule (a decision tree or a classifier) by recursively partitioning a given data and producing decision boundaries. Most traditional rule-based approaches tend to build a single classifier. However, some empirical comparative studies demonstrate that such single classifier works for solving some classes in learning problems but not all situations (Mitche et al., 1994). Recently emerging ensemble system such as bagging, boosting, and RF is able to solve this limitation by integrating decisions made by multiple classifiers which are built by using different algorithms and tend to be more flexible in diverse situations.

Table 2-1. Comparison of grid-, point-, and object-based classification approach

Approach	Grid-based	Point-based	Object-based
Segment size	Points in a pixel	A single point	Points having similarity
Segmentation	Grid generation (Forlani et al., 2006)	-	Region growing, Clustering (Pu and Vosselman, 2006; Filin and Pfeifer, 2006)
Feature extraction	Statistics of membership points (Guo et al,., 2011)	Properties of each point Statistics of neighboring points (Niemeyer et al., 2011)	Statistics of membership points Relationships with neighboring segments (Lim and Suter, 2009)
Classification scale	A pixel	A point	A segment
Advantages	Easy segmentation	3D full analysis (Carlberg et al., 2006)	Fast and cost-effective (Yang and Dong, 2013)
Disadvantages	Unsuitable for multi-objects in a pixel	"salt and pepper" High cost in computation (Niemeyer et al., 2014)	Segmentation affects classification

Thus, all the classification algorithms require a feature set measuring characteristic properties of the object classes for each observation which provides basic input to differentiate one class from others or to group each class into a single cluster based on the similarity of the features (Guo et al., 2012; Samadzadegan et al., 2012). As seen in Table 2-1, typically for LiDAR point cloud classification, the grid-based, point-based and object-based approaches are common methods for extracting the features. The following sections describe three feature extraction methods for LiDAR classification.

2.2.1. **Grid-based approach**

Grid-based approach considers a 3D laser point cloud as a raster image, which is represented as a 2D array. In this method, the raw LiDAR point cloud is interpolated into a grid space and each grid (pixel) contains representative information such a mean height, number of returns, laser intensity, backscattering coefficient, width of reflected pulse, and so on. The classification approaches categorized in section 2.2 can be applied to individual pixels or groups of pixels. A classification rule for a specific object can be constructed from geometric and topological relations between regions resulted from segmentation over grid LiDAR data for detecting buildings (Forlani et al., 2006); classifying forest types (Antonarakis et al., 2008); detecting single trees (Zhang and Sohn, 2010; Lin et al., 2011); and predicting single tree attributes (Yu et al., 2011). Grid-based classification has often been used to fuse multiple laser echoes with multi-spectral information obtained from optical imagery, for urban classification purposes (Guo et al., 2011). A set of cues extracted from full-waveform LiDAR data is interpolated into an image grid to extract features for each pixel and to classify forest scenes using a Bayesian

pair-wise classifier (Neuenschwander, 2009). The utilization of grid data provides an advantage in easy implementation of applications based on comprehensive low-level computer vision algorithms, such as region growing or other segmentation methods, without significant alterations, and another advantage in cost-effective management of the huge data volume of a LiDAR point cloud. However, the grid-based classification implicitly assumes that a pixel represents only one class, although multiple scatters belonging to different classes are vertically distributed within the pixel. This may be a critical limitation to the classification of power line corridor scenes, where vertical overlaps exist between wire and terrain, vegetation and wire, wire and pylon, or vegetation and building.

2.2.2. **Point-based approach**

In contrast to the grid-based classification, the point-based method aims to extract a feature set and determine an object class for every single point. This requires a full investigation of individual laser points to label its corresponding object class. Each individual LiDAR point has geometric, e.g., height, and radiometric properties, e.g., intensity. In addition to these properties from individual points, some other features are examined through an interaction with neighbors. For instances, 2D point analysis per scan line helps one to extract a smooth surface by comparing z values of current, previous and next point to find an abrupt surface change (Axelsson, 1999). 3D point distribution from neighbors near a certain point provides salient features to identify a local geometry of the point such as line- and plane-likeness which are extractable using eigenvalues

(Verma et al., 2006) or Hough transform (Vosselman and Dijkman, 2001; Axelsson, 1999) or least squares method (Mitra et al., 2004).

Based on these features, different classification methods addressed in section 2.2 are applied for classifying individual points. For graph-based applications, Lu et al. (2009) constructed a graph comprising of individual LiDAR points and classified them into ground or non-ground points by applying three levels of local features (point, segment, and disc) to CRF. Niemeyer et al. (2011) also employed CRF to classify urban scenes; the authors created a graph where a node corresponds to a single LiDAR point and an edge is represented as a link between a current point and its cylindrical neighbourhood. Verma et al. (2006) used local planar properties extracted through principle component analysis (PCA) to detect building points on a graph. For rule-based approaches, Carlberg et al. (2009) developed a series of binary decision classifiers trained using RF, each of which can filter out a particular class from LiDAR points that are not labelled by the preceding classifiers. Lodha et al. examined many machine learning methods, Adaboost (Lodha et al., 2007a), Expectation-Maximization (EM) algorithm (Lodha et al., 2007), and SVM (Lodha et al, 2006) to classify LiDAR into urban key objects.

The greatest benefit of this point-based approach is the ability to generate multiple labels from a single transmitted laser pulse, for example, according to the number of scatters and the reflectance information interacted with each pulse, although the computational cost of such approach would be expensive. The drawback of point-based approach is occasionally assigning an incorrect label to a part of an object where

sufficient points are not available, such as building edges and tree tops. This misclassification is called “salt and pepper” (Chan et al., 2005). To solve this problem, many researchers employ the object-based classification which treats a group of points showing similar patterns as an object segment and assigns a class label to the segment.

2.2.3. **Object-based approach**

A conventional way of the object-based approach with LiDAR data (Yu et al., 2010; Antonarakis et al., 2008) is to apply object-based image analysis to a 2D grid created from LiDAR points. However, as addressed in section 2.2.1, data conversion from point cloud to grid could bring serious limitations to classification. To overcome these limitations, the object-based approach in this study describes a point set (or a segment) having homogeneities and considers it as a group to assign the same label to the set. In the object-based method, such segmentation is a pre-processing step before feature extraction and classification. Hence, many segmentation algorithms have been introduced so far and they can be categorized mainly into two groups: region growing (Sithole and Vosselman, 2004; Dold and Brenner, 2004; Pu and Vosselman, 2006) and clustering (Filin and Pfeifer, 2006). Region growing gathers neighboring points from a seeded point and expands a region containing the points according to a similarity criterion, e.g., curvedness (or surfaceness) (Dold and Brenner, 2004; Yang and Dong, 2013) and echo related features (Rutzinger et al., 2008). Unlike region growing which starts from seed points, clustering directly produces point clusters by considering proximity between points which are projected into a feature space. Some application studies employing the clustering method utilized various LiDAR features to represent each point in the feature

space such as height difference, distance, slope variance, vertical profile feature, and so on (Filin and Pfeifer, 2006; Biosca and Lerma, 2008; Lehtomaki et al., 2010).

As briefly addressed above, mostly segmentation algorithms take geometry property representing a point distribution in a local region. For example, the fundamental geometric properties such as linearity, planarity, and scatteredness are used to segment linear-like, planar-like, and scattered-like objects respectively. This idea is based on the fact that objects in man-made or natural environments can be generally represented by using those geometries (Biosca and Lerma, 2008). In practice, Hough transformation (Jwa et al., 2009) and linearity measure from eigenvalues (McLaughlin, 2006) are examined to describe a transmission line as multiple line segments. As well as planarity, slope and surface are used to partition terrain (Yao et al., 2009; Wang and Tseng, 2011; Yang and Dong, 2013), building roof (Filin and Pfeifer, 2006; Yao et al., 2009; Wang and Tseng, 2011) and building façade (Biosca and Lerma, 2008; Yang and Dong, 2013) into planar patches. For a complete scene representation piece-wisely, all of the aforementioned three features (linearity, planarity, and scatteredness) are employed (Lalonde et al., 2006; Lim and Suter, 2009; Bremer et al., 2013) to decompose a scene into a mixture of the three geometries. Besides the geometry properties, local height jump, echo information, and point density are also investigated to group terrain, vegetation, and pole-like object respectively.

Based on the results of such segmentation and feature extraction, diverse classification methods have been used: generative, discriminative, graph-based, and rule-

based approaches as addressed in the section 2.2. For example, Lalonde et al. (2006) took a generative method where saliency features, i.e., scatter, linear, and surface, are computed from the spatial point distribution in a local neighborhood and their distribution is estimated by GMM and EM. Then, a Bayesian classifier is built using the distribution models for the classification of MLS point cloud. Lim and Suter (2009) applied a discriminative CRF to classify point segments called super-voxels, which are the results of an over-segmentation of 3D point cloud. The size of each super-voxel is differently determined according to surface curvature and point density of membership points belonging to the super-voxel. As rule-based classification methods, Rutzinger et al. (2008) created a classification decision tree to identify vegetation from segments, each of which is derived from echo, full-waveform, and point density. Bremer et al. (2013) also generated rules to make point groups and to classify the groups based on eigen-related features (linear, planar, and volumetric). Additionally, the object-based method has been utilized to classify power line corridors containing transmission lines which are represented as linear features. This dissertation will give detailed descriptions of power line related studies in the next section.

Compared to the point-based approach, the object-based approach generally shows better classification quality for class objects, which are able to be individually described using a specific geometric feature, such as terrain (planar), building roof (planar), and power line (linear). The “salt and pepper” effect often observable in the point-based approach are minimized in the object-based approach. Moreover, the computational cost is not expensive compared with the point-based approach. However, there are several

limitations with this object-based method. One is that the segmentation results affect the classification quality. This is because that once the segmentation is completed, the object-based method assigns the same label to each segment although some membership points of the segment are not same class object as the other points. To solve this problem, some researchers applied the top-down approach utilizing a hierarchical class structure on multi-scale segmentation maps (Lamonaca et al., 2008). Another disadvantage is that for class objects with intra-class variations in shape, size or material, e.g., electric pylons with various shapes depending on voltage type, the object-based method produces lower classification accuracy than the point-based method.

2.3. Power-line Risk Management

A power line system is an interconnecting network of power facilities including power plant, transmission lines, and electric pylons. The most important requirement of such power line system is to safely and reliably deliver electric power to family and business. If not, a power transmission failure brings considerable economic loss and inconvenience as experienced from 2003 Blackout in North America. Such a large-scale disaster can be preventable in advance by detecting and removing potential risks in advance. One of the most potential risks for power lines is the contact of trees with transmission lines. For example, growing trees below the transmission lines and snow-laden and wind-blown trees approaching toward the transmission lines (Figure 2-4) are possible threats. Moreover, trees could come in contact with the transmission lines due to sagging and swaying of the lines caused by wind or ice load on the power lines (Figure 2-4). These risks can cause wide-spread power outages and/or fires. As a result, the typical

tasks for power line risk management projects is to manage vegetation within the corridor. This ensures the safety for the public and private properties as well as delivering a reliable electrical service. The power line risk management is also required to maintain an accessibility to the ROW, which indicates a corridor of land where electric transmission lines are located (Ituen and Sohn, 2010), for both emergency and routine maintenance of the power lines. ROW is commonly divided into "border zone", where potentially dangerous trees may live, and "wire zone", where transmission line, electric pylon, and a clear cut vegetation environment exist, as seen in Figure 2-4. Figure 2-5 illustrates the detailed zone partitions and the criteria on trees inhabiting in each partition. Trees unsatisfying the criteria are considered as dangerous trees that would touch power lines.

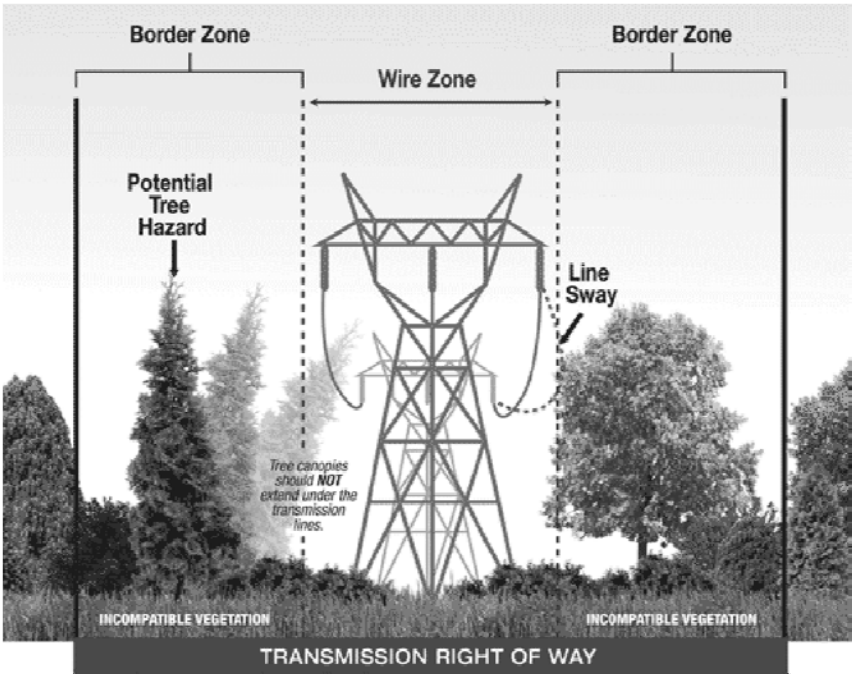


Figure 2-4. Compatible vegetation for transmission ROW (Rights Of Way)

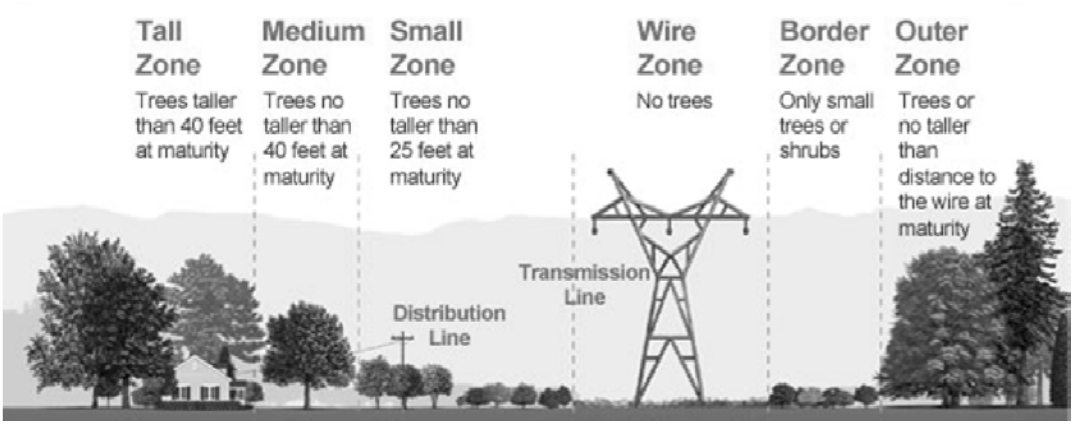


Figure 2-5. Detailed zone partitions and criteria for transmission ROW

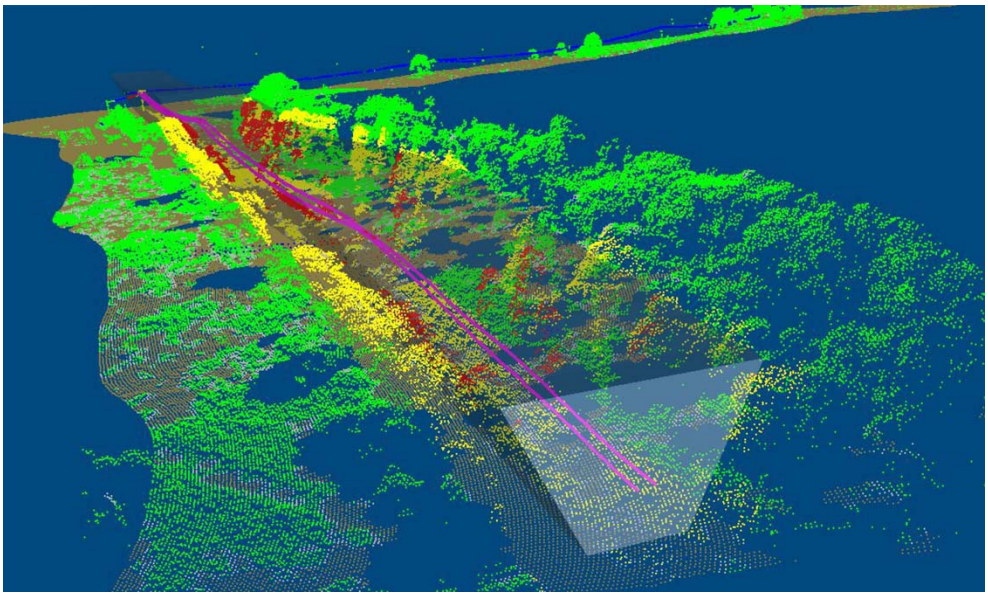


Figure 2-6. Vegetation clearance analysis results (VRMesh)

In the past, these potential trees that are posing threats were identified, reported, and eliminated by on site ground patrols requiring human powers. However, such human-centric method is limited with inaccessible area, small monitoring coverage, high labor cost, low processing speed, and low accuracy. For these limitations, recently remote

sensing technologies are employed, especially, airborne LiDAR which is a promising data source since it allows 3D mapping necessary to find horizontal and vertical encroachments (Figure 2-6). Several LiDAR-based mapping systems have been commercially developed and utilized for the corridor risk assessment. There are two commercial software widely used in power utility companies, Terrascan and PLS-CADD (Power Line Systems – Computer Aided Design and Drafting). TerraScan is a 3D LiDAR mapping software developed by TerraSolid Inc. (Terrascan), which runs as a plug-in on a CAD environment, i.e., Microstation (Microstation). Terrascan also has the ability to perform classification automatically (different filtering method for each class) and it is able to represent transmission lines as 3D vector models, i.e., catenary line models. However, this power line modeling process requires human-interactions; for instance, a user has to point out individual power lines at least once. Unlike Terrascan, PLS-CADD is a stand-alone program which is designed specifically as a mapping tool for the power line management. Distinct features of PLS-CADD are to generate realistic-looking models of the power structures including conductors, electric pylons, and insulators. Similarly to Terrascan, most of the tasks require manual intervention to localize the structures and to select proper models from a database where diverse model templates are stored.

To date, only few studies are able to overcome the limitation as commercial software where automating power line classification or modeling has been reported. Jwa and Sohn (2012) combined a constrained non-linear least squares adjustment with the model selection process for estimating the parameters of catenary curve for reconstructing

power line models from airborne LiDAR data. Melzer and Briese (2004) extracted power lines by applying iterative Hough transform (HT) to LiDAR data and grouping segmented line vectors through the Neural Gas Network. McLaughlin (2006) proposed a supervised knowledge-based classification method, where a learning model was established by applying Gaussian Mixture Model to eigenvalues computed using ellipsoid neighbourhoods from LiDAR data. The above studies have reported success, but are limited to only few object classes (wire, vegetation and terrain) and controlled environments where the scene contains a specific single type of power line that has little contact with vegetation. There are still increasing demands to advance classification algorithms to consider more diversified corridor instances including pylons, insulators and other power line attachments, and to make classifiers stable to variances within intra-object classes; for instance, a classification performance over power lines and pylons should not to be sensitive according to their voltage types.

2.4. Ensemble methods

Ensemble learning for classification is a machine learning method which combines the results of multiple classifiers built using different training data or different training algorithms. Unlike ordinary learning methods building a classifier from training data, ensemble methods construct a set of diverse classifiers and combine them. An ensemble is composed of a number of classifiers (or learners), called base learners. The potential of the ensemble exceeds the individual base learners. This indicates ensemble methods can boost weak learners that are slightly correlated with the true classification to strong learners which can make predictions highly correlated with the true classification (Zhou,

2012). Here, the base learners are referred to as the weak learners. The ensemble methods often make better predictions compared with any membership learners according to the empirical examination of Hansen and Salamon (1990) when the ensemble satisfies diversity between the base learners. The discussion of diversity has been addressed as a critical issue to make an improvement in the classification (Polikar, 2006). Chapter 5 will discuss ensemble diversity in further detail. Ensemble has been widely used for problem solving in a variety of research fields, especially in the pattern recognition, machine learning, and neural network community. Boosting (Schapire, 1990) and bagging (Breiman, 1996) are the first generation of ensemble method. Boosting combines weak learners to produce a strong learner and the strong learner is iteratively rebuilt by assigning a different weight to each weak learner. The weight for each weak learner is estimated depending on an error rate assessed by the corresponding weak learner in the previous iteration. Freund and Schapire (1997) implemented an algorithm based on such boosting procedure, AdaBoost. They first designed the AdaBoost.M1 which is able to solve a binary classification problem and upgraded it to the AdaBoost.M2 to handle multi-class classification. Unlike boosting that focuses on assigning different weights to weak learners, bagging (bootstrap aggregating) concentrates on generating diverse subsamples to create independent base learners. Given a training data, bagging employs bootstrap sampling (Efron and Tibshirani, 1993) to obtain subsets of the training data, each of which is generated by *sampling with replacement*. Each subset builds a base learner and the outputs of the base learners are aggregated for a final decision through *voting* for classification and *averaging* for regression.

Evolving from bagging, Random Forests (Breiman, 2001) was invented to maximize the diversity between base decision trees by injecting randomness, i.e., random sampling and random feature selection, into each decision trees. Random Forests has been recently studied for land-cover mapping from satellite imagery (Na et al., 2010; Rodriguez-Galiano et al., 2012; Waske and Braun, 2009), for the prediction of tree inventory (Coulston et al., 2012; Yu et al., 2011) from LiDAR, and for the classification of urban scenes from LiDAR and color images (Guo et al., 2011). Despite the successful ensemble applications, they make an assumption of stationary environment, i.e., prior class probability and the conditional distribution of objects in classes do not change while the ensemble operation (Jackowski, 2013). Thus, the results are promising under the condition if the testing data shows the same characteristics as training data. In practice, however, there exist different object types the training data does not contain. When the new object types are introduced to the training data, retraining is required for the entire dataset. To overcome these drawbacks, new ensemble methods that are capable of adapting to the non-stationary environments were developed such as incremental learning (Muhlbaier and Polikar, 2007; Elwell and Polikar, 2009) and evolutionary-adapted ensemble (Jackowski, 2013). The basic idea of these ensemble methods is to build a new classifier using a data sample where new object types appear and to insert the new classifier into the classifier pool. Then, a decision is made by the weighted fusion of the classifiers based on the discriminating strategies.

These aforementioned methods work for detecting new type of objects that are found in the new data set. However, they are not able to classify diverse objects intra-class

variations, i.e., variance in size, shape, material, and so on for an object class, which is a critical problem to be solved in scene classification, especially power line corridor scenes where different types of power lines could be found. In this dissertation, it is suggested that the ensemble system (or MCS) which accommodates a pool of classifiers, each of which can discriminate a specific type of object, e.g., 69kv electric pylon made by wood and standing in urban area, from others is the solution. Furthermore, the system produces an extensible committee of classifiers by adding, removing or replacing certain classifiers from the classifier pool depending on scene contents. Additionally this ensemble system is feasibly buildable due to a diverse number of data sources produced and processed in advance by users, industries, and governments, termed “big data” (MIKE2.0). From the big data, the system uses structured data where the classification has been done to generate and update the classifier pool. This dissertation follows the MCS framework and builds a proto-type of ensemble system which accommodates multiple classifiers trained under different corridor environments, i.e., different carrying voltages (chapter 5).

3. Classification Using LiDAR Point Features

As already addressed in chapter 2, traditional methods of LiDAR classification in remote sensing community treat 3D point cloud as raster imagery, called grid-based classification where LiDAR points are interpolated into a 2D grid space and each grid (pixel) contains representative information. A main drawback of this approach is class uncertainty in case that multiple objects are present in a pixel, which typically occurs in power line corridor scenes, e.g., vertically overlapped shrubs and transmission lines. This chapter introduces point-based feature extraction and classification, which derives features for each point and assigns a class label to individual points based on the extracted features, suitable for the power line corridor scenes. Section 3.1 addresses such features and their extraction methods. Section 3.2 introduces Random Forests, which shows high performance in classification thanks to its ensemble technique, to build a classification model using the features. Section 3.3 discusses supplementary tips to maximize the accuracy in supervised learning with given data: balanced learning, feature selection, feature de-correlation, and optimal training sample selection. Section 3.4 and 3.5 demonstrate experiment results including classifier optimization and comparative analysis on grid- vs. point-based approach in terms of classification accuracy and computation complexity. Finally, section 3.6 addresses a summary of this chapter. The research work of this chapter has been described in the publications: Kim and Sohn (2013) and Kim and Sohn (2010).

3.1. Point-based Features

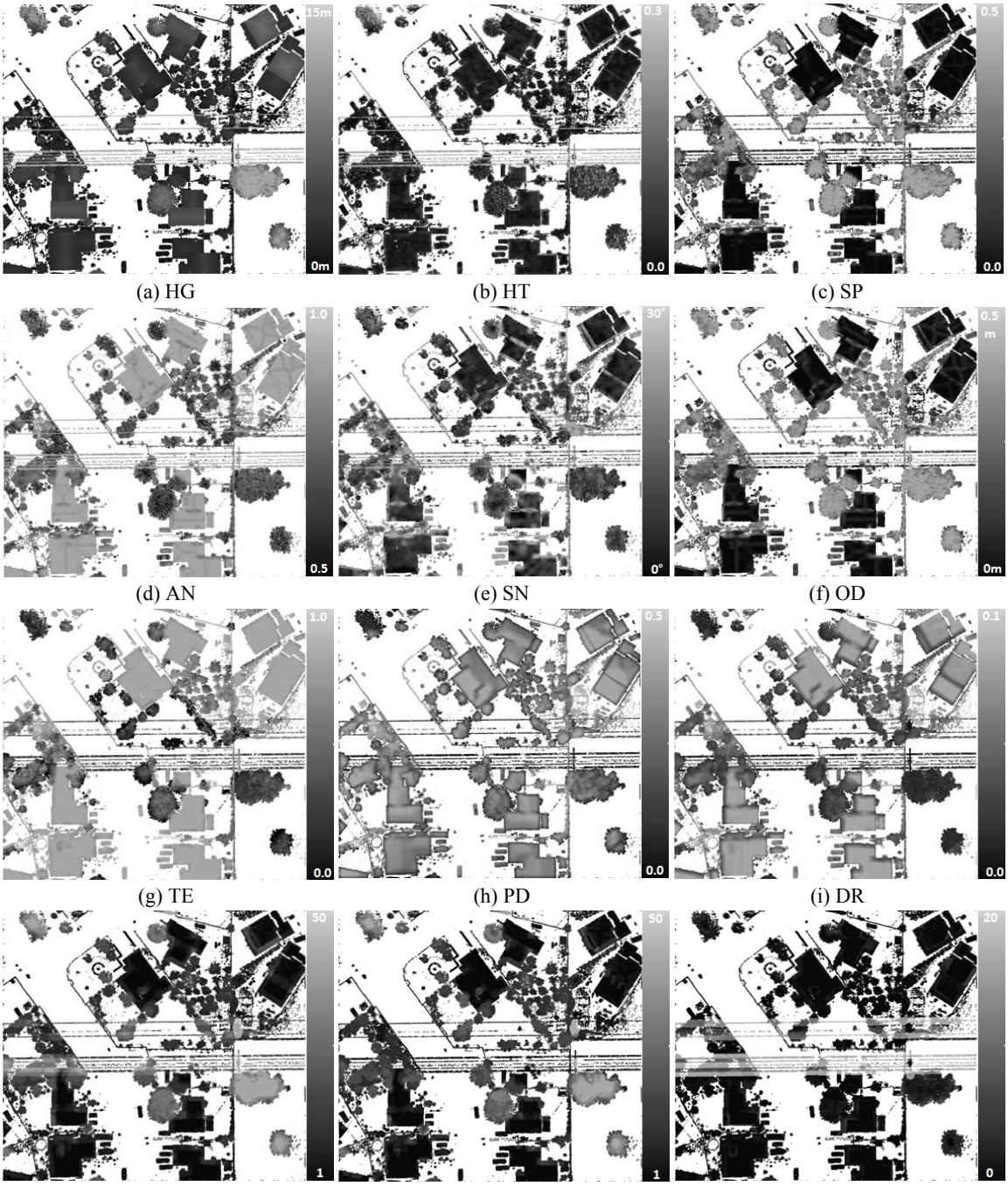
In general, the five targeted corridor objects are visually distinguishable from each other. This is because the objects are differently formed with respect to surface characteristics that determine depending on the degree of laser penetration, physical size and volume, surface roughness, structural patterns and so forth. These discernible characteristics can be seen by examining a spatial distribution of LiDAR points, not only horizontally but also vertically. Thus, 21 features for each LiDAR point were considered to characterize each object and computed by taking advantage of full 3D analysis.

3.1.1. **Feature Extraction Method**

As shown in Table 3-1, the features are categorized into 8 groups: *height*, *Hough Transformation*, *eigen-related*, *surface-related*, *convex hull*, *echo-related*, *density-related* and *vertical profile-related* feature. Depending on the feature group, a particular restricted space is used to collect neighbouring points for each point: a sphere with radius *r* is used for *Hough transform*, *eigen-related*, *surface-related*, *echo-related* features and *point density*, while a vertical cylinder with radius *r* and height *h* is used for analyzing the property of vertical distribution of points such as *vertical profile-related* features. Here, the radius *r* for the cylinder is constant, but the height *h* varies depending on the *z* value range from neighbouring points, i.e., the lower and upper bound of the cylinder are determined by the minimum and maximum values among z values of neighboring points. Only *density ratio* is calculated by using both sphere and cylinder. The sphere and cylinder for feature computation are denoted N_S and N_C respectively, and the number of points captured by N_S and N_C are n_s and n_c.

Table 3-1. 3D airborne LiDAR features

Category	Feature	Abbrev	Equation	Description	Reference
Height	*Height*	HG	$e_i - e_g$	Height of each point (e_i, elevation) from ground (e_g)	Lodha et al. (2007)
Hough Transforma-tion	*Hough Transform*	HT	$\frac{\sum_{p=1}^{n_m}\left(v_p + \sum_{q=1}^{8} v_{pq}\right)}{n_m \times n_S}$	Voting in Hough space to measure linear-likeness	Hough (1962)
Eigen	*Sphericity*	SP	λ_3 / λ_1	*Eigenvalues*,$\lambda_1 > \lambda_2 > \lambda_3$, a measure of spherical-likeness	Chehata et al. (2009)
	Linearity	LN	$(\lambda_1 - \lambda_2)/\lambda_1$	A measure of linear-likeness	
	Planarity	PL	$(\lambda_2 - \lambda_3)/\lambda_1$	A measure of planar-likeness	
	Anisotropy	AN	$(\lambda_1 - \lambda_3)/\lambda_1$	Asymmetric volume property of an object	
Surface	*Plane Slope*	PS	$\Delta\theta$	Angle difference between plane normal vector and z-axis	Rutzinger et al. (2008)
	Orthogonal Dist.	OD	-	Root mean square of perpendicular distances from each point to plane	
	Vertical Dist.	VD	-	Root mean square of vertical distances from z point values to corresponding z value on plane	
	Homo. of Surface Normal	SN	$\sum_{i=1}^{n_T}\left(\Delta\theta_i - \overline{\Delta\theta}\right)^2 / n_T$	Variance of *Plane Slope* ($\Delta\Theta$) of n_T triangles in a mesh surface	
Convex Hull	*Projection Area*	PA	$A/\left(\pi \cdot r^2\right)$	Bounding area of points projected on a horizontal plane (A)	Developed by us
	Bounding Volume	BV	$V/\left(4\pi \cdot r^3/3\right)$	Bounding volume of 3D convex-hull (V)	
Echo	*Vegetation Echo*	VE	$\left(n_{fr} + n_{ir}\right)/n_S$	Proportion of first (n_{fr}) and intermediate (n_{ir}) returns to all points (n_s)	Developed by us
	Building Echo	BE	n_{sr}/n_S	Proportion of single (n_{sr}) returns	
	Terrain Echo	TE	$\left(n_{sr} + n_{lr}\right)/n_S$	Proportion of single (n_{sr}) and last (n_{lr}) returns	
	Power-line Echo	PE	n_{fr}/n_S	Proportion of first (n_{fr}) returns	
Density	*Point Density*	PD	$3 \cdot n_S / 4\pi r^3$	Density of points within a sphere	Rutzinger et al. (2008)
	Density Ratio	DR	$3 \cdot n_S/\left(4r \cdot n_C\right) \approx n_S/n_C$	Ratio of point densities in a sphere and in a circle projected on a horizontal plane.	
Vertical profile	*On-segment*	OS	-	# of occupied bins	Developed by us
	Con. On-segment	COS	-	Maximum # of sequentially occupied bins	
	Con. Off-segment	CFS	-	Maximum # of sequentially empty bins	



(a) HG (b) HT (c) SP

(d) AN (e) SN (f) OD

(g) TE (h) PD (i) DR

(j) OS (k) COS (l) CFS

Figure 3-1. Important feature visualization

3.1.2. **Feature Descriptions**

Height from ground level

Height (HG) feature is an elevation measured from the terrain surface for each point above ground. The elevation of power lines is differently designed depending on the maximum voltage carried through them. In addition, the elevation can be considered as a useful classification variable to differentiate residential buildings from the other features such as pylon and vegetation. Similar to a method suggested by Lodha et al. (2007), HG is computed by measuring the vertical distance of each point from an underlying terrain surface model that is generated by applying a terrain filter proposed by Sohn and Dowman (2008) to LiDAR data.

Hough Transform

The geometry of a power line is formed as a catenary curve, but when it is projected onto a horizontal plane, the power line can be also represented with a set of line segments (Jwa and Sohn, 2012). Thus, analysing linear properties for each point provides important information to identify power line. The *Hough Transform* (HT) feature was designed to measure the likelihood that a LiDAR point belongs to a linear structure such as power lines or pylon structure. 2D Hough transform (Hough, 1962) was applied to 2D points obtained by projecting 3D points within N_S onto a horizontal plane. However, there are two limitations in applying the traditional Hough transform to power line LiDAR data. One is that 2D points corresponding to a cable often are not collinear due to systematic errors with airborne LiDAR. This may preclude the points from being mapped into a certain cell in the Hough grid, but they may be mapped into close cells to the cell with the

highest count. The other is that N_S might contain multiple cables such as bundled conductors which will produce n_m peaks (n_m>1). Therefore, for HT it is considered that n_m peaks (v_p is a vote count for each peak) and v_{pq}, vote count in the cells adjacent to each peak in the grid. In this way, HT measures total supports by the presence of linear features. In Table 3-1, HT feature is computed by summing of all the votes normalized by n_m and n_S, where n_m is heuristically determined relying on the site knowledge (assumed n_m=4 in this study).

Eigenvalue-related features

Analyzing eigenvectors and eigenvalues often provides useful information to classify objects in an image. For classifying an urban scene using LiDAR data, Chehata et al. (2009) defined four eigenvalue-related features including *Sphericity* (SP), *Linearity* (LN), *Planarity* (PL), and *Anisotropy* (AN). In this study, three eigenvectors are computed using all LiDAR points in N_S centred at a point. According to Chehata et al. (2009), SP is a measure of how spherical (round) an object is, while LN and PL are a measure of how linear or planar an object is respectively; AN is a measure of the directional anisotropic property of an underlying object. The equations of the eigenvalue-related features are described in Table 3-1. SP is useful for differentiating vegetations from the other objects, while the value of LN and PL would be higher for power lines and buildings respectively. AN helps to differentiate power lines and buildings from vegetations by showing an inequality of the scalars with three eigenvectors.

Surface-related features

Surface-related features were examined to characterize planar objects such as building rooftops. For each point, a plane surface was approximated over a set of points captured by N_S using the plane fitting algorithm with PCA. Then, *Plane Slope* (PS), *Orthogonal Distance* (OD), and *Vertical Distance* (VD) were computed on the basis the estimated plane. PS is the angular difference between the surface normal to the plane and the z-axis. OD and VD both represent surface residual, but they are different in a residual measurement way, orthogonal and vertical residual respectively. The last feature, called, *Homogeneity of Surface Normal* (SN), is defined as a measure of surface roughness of an object with respect to the similarity between normal vectors of n_T triangles approximating the surface of the object. A 3D triangular mesh surface is created from points in N_S. PS would be expected to show regular slope values over buildings, while arbitrary slopes over vegetation. Two surface roughness measures (OD and VD) are useful features to differentiate buildings with smooth surfaces from vegetation that shows high surface roughness (Rutzinger et al., 2008). However, OD and VD might not be suitable for measuring the surface roughness of a region where multiple planes intersect. In this case, SN is able to overcome this problem by measuring a local transit between the normal vectors of neighbouring surfaces at relatively finer scale compared to the one for computing OD and VD.

Convex hull-related features

Convex hull-related features are examined to measure the volumetric property of an object of interest captured by N_S for each point. Two features, *Projection Area* (PA) and

Bounding Volume (BV) are defined. PA is computed by applying a 2D convex hull algorithm to 2D points produced by a horizontal projection of the points in N_S and then normalizing the area of the convex polygon to the area of a circle with radius *r* of N_S. BV is computed by applying a 3D convex-hull algorithm directly to 3D points in N_S and then normalizing the volume of the generated polyhedron to the volume of N_S. In general, vegetation will show high values in both PA and BA as it usually has large volume, while a building will have high PA and low BV as its rooftop occupies a large space in 2D, but small in depth. The power lines will show small values in both PA and BA as they usually occupy small spaces in both horizontal and vertical space. However, a pylon might be difficult to characterize with respect to PA and BA as it has different shapes; pole-type towers would show low values in both PA and BA, while steel-frame towers would show high values.

Echo-related features

LiDAR is able to capture multiple echoes (returns) from a single laser shot. According to the number and the order of returns, the echo is classified into single return, first return, intermediate return, or last return. This echo information is a well-used feature to distinguish penetrable objects (e.g., tree and shrub) from rigid objects (e.g., building rooftops and terrain). For instance, most echoes from vegetation (*Vegetation Echo*, VE) are likely to be first returns or intermediate returns (Rutzinger et al., 2008). Single returns are predominant for building (*Building Echo*, BE), while both single and last returns are recorded for terrain (*Terrain Echo*, TE). Thus, the echo-related features for vegetation, building and terrain were designed by considering the aforementioned

returning patterns. In this study, another feature (*Power line Echo*, PE) is introduced for classifying power lines that usually present first echoes from power cables and other echoes from underlying features due to the relatively larger footprint of a laser beam compared to the diameter of a power cable. Depending on the pylon type, the echo patterns vary; the echo pattern of a steel-framed pylon would be similar to that of vegetation, while a pole-typed pylon may show a similar echo pattern to a building. Table 3-1 summarizes the echo-related features, where n_{sr}, n_{fr}, n_{ir}, and n_{lr} are respectively the number of points corresponding to single, first, intermediate, and last echoes captured within N_S.

Density-related features

The number of LiDAR points reflected from a unit surface area varies depending on the surface characteristic illuminated by the laser beam, which determines the degree of laser penetration of the surface. Two density-based features, *Point Density* (PD) and *Density Ratio* (DR), were investigated as classification features. PD is defined as the number of points within N_S divided by the volume of a sphere for Ns. In general, a higher PD would be usually obtained over a solid surface (e.g., building) than over a penetrable object (e.g., vegetation or power line). Also, a PD obtained over vegetation would be higher than one from power lines as vegetation usually has more scatters contained within a fixed volume size. DR is adopted as an additional feature that was initially proposed by Rutzinger et al. (2008) for identifying vegetation using airborne LiDAR data. DR is computed as a ratio of PD in 3D, which is the point density in a sphere, to PD in 2D, which is a point density in a circle created by projecting the sphere onto a

horizontal plane. DR is approximated as a ratio between point counts (n_S and n_C in Table 3-1) within N_S and N_C. As can be seen in Figure 3-1(h) and 3-1(i), PD shows different values over rooftops as it is affected by relative view angles between a laser scanner and the target's surface normal. However, DR is relatively less affected by this factor.

Vertical Profile-related features

Recently, there has been an increasing interest in detecting vertical objects (tree trunk, traffic light, lamp post, etc) from ground-based ranging imagery (Lehtomäki et al., 2010; Kim and Medioni, 2011; Rabbani and van den Heuvel, 2005). Although the proposed methods successfully demonstrated their performance, there are some limitations which hinder directly applying them to detecting vertical objects in a corridor scene. This is partly due to the diversity of shapes of vertical objects from simple (pole-typed) to complex (steel-framed) pylon and also partly due to a relatively insufficient point density compared to ground-based LiDAR data. Thus, new features called vertical profile-related features are devised to characterize the property of vertical distribution of LiDAR points reflected from corridor objects. A cylinder with radius r and height h is created by N_C of a point. Then, the cylinder is vertically divided by a fixed incremental height, Δh that produces a number of cylinder segments. A cylinder segment is marked as an occupied segment, called on-segment, if it contains more points than a pre-specified threshold. Otherwise, it is marked as an off-segment (Figure 3-2). The vertical profile-related features are computed by measuring three different counts; 1) the maximum number of on-segments that are continuously connected, called *Continuous On-Segment* (COS); 2) the maximum number of off-segments that are continuously connected, called

Continuous Off-Segment (CFS); and 3) the total number of *On-Segments* (OS). A pylon and vegetation would be expected to show high values in COS, while low in CFS. A vertically discontinuous object such as a power line and a building would show high value in CFS, while low in COS. The count of *On-Segment* (OS) is also taken into account to characterize the properties of a vertically structure object such as pylon.

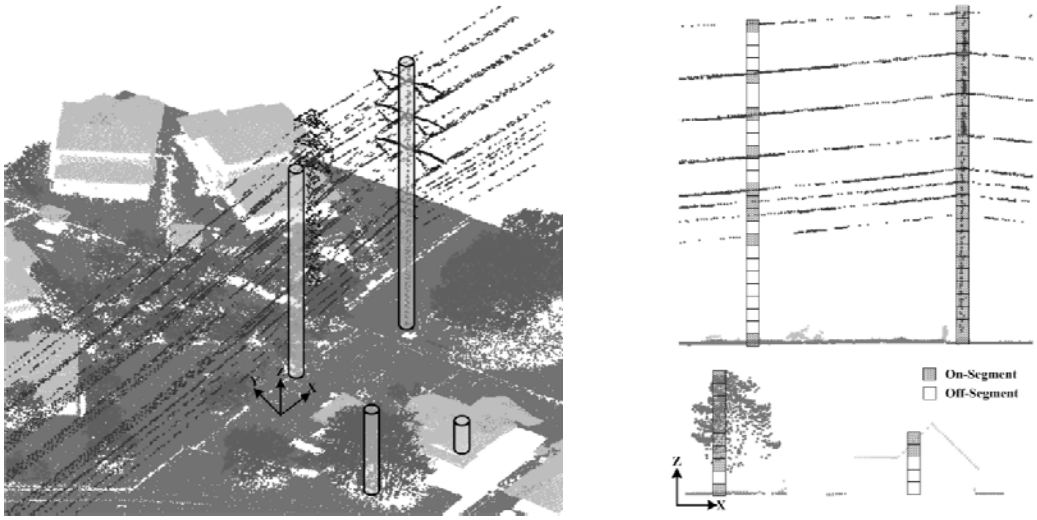


Figure 3-2. Vertical Profile feature; a cylindrical neighbourhood (left) and status (on or off) of vertically divided segments (right) for each class

3.2. Random Forests Classifier

Random Forests (Breiman, 2001) is an ensemble method which is evolved from bagging (Breiman, 1996). Using the same sampling strategy as bagging, it generates an ensemble of decision trees, each of which is built on a bootstrap sample resampled from the original data set. However, unlike bagging it infuses randomness to grow the trees differently by using a random feature subset at every tree node split (Breiman, 2001). A set of the grown trees is applied to given data for classification. The class of each instance is decided by the majority vote over all the trees. Random Forests requires two

parameter settings to train a training sample: T is the number of tress to grow and F is the number of features to select for each node split.

As a classification application tool, Random Forests has been recently studied for land-cover mapping from satellite imagery (Na et al., 2010; Rodriguez-Galiano et al., 2012; Waske and Braun, 2009), for the prediction of tree inventory (Coulston et al., 2012; Yu et al., 2011) from LiDAR, and for the classification of urban scenes from LiDAR and color images (Guo et al., 2011). In this study, Random Forests is treated as a base classifier to form an ensemble system. The ensemble system is composed of multiple classifiers, each of which is trained by Random Forests using different data sources.

3.2.1. **Basic Principle**

A Random Forests is an ensemble learner having T single descriptors respectively trained with T samples called bootstrap replicates. The t^{th} bootstrap sample ($\boldsymbol{S}_t$) is randomly drawn from training data ($\boldsymbol{S}$) with replacement. About 1/3 of $\boldsymbol{S}_t$ is excluded from the bootstrap sample and the remaining 2/3 of $\boldsymbol{S}_t$ is trained to generate a tree. The former is called "out-of-bag", OOB ($\boldsymbol{S}_{to}$) and the latter "in bag", IB ($\boldsymbol{S}_{ti}$). The OOB is an independent test sample used for testing the trees generated. A decision tree (h_t) is grown based on random variables ($\mathbf{x}$, $\hat{y}$) belonging to $\boldsymbol{S}_{ti}$. Note that $\mathbf{x}$ and $\hat{\mathrm{y}}$ indicate the vectors of input features and true label. While training, the best split subsets (or nodes) of an IB produce a decision tree. At each stage for the node splits, all the possible binary partitions of a current node using a random feature subset are considered, and then the one which leads to the greatest decrease in a node impurity is chosen as the best split for the node.

The impurity with a node is quantified as an impurity function which has a random variable on the proportion of instances belonging to possible classes in the node (Sutton, 2005). The impurity function is minimized when a node is completely pure, while it is maximized when the node includes the equal number of instances for all the possible classes. Such iterative partitioning to yield a tree continues until all the descendent nodes become terminal nodes which have the same class. *T* number of trees are independently grown in the same way using the *T* IBs. In the training stage, the Random Forests internally estimates the training quality of individual trees by testing them using their corresponding OOBs: strength of each individual tree, correlation between the trees, OOB error presenting trees' performance, importance of input features, and so on (Breiman, 2001). In particular, feature importance allows us to know what features are highly relevant to the classification accuracy. This is discussed in more detail in the next section. A logical workflow of Random Forests is depicted in Figure 3-3.

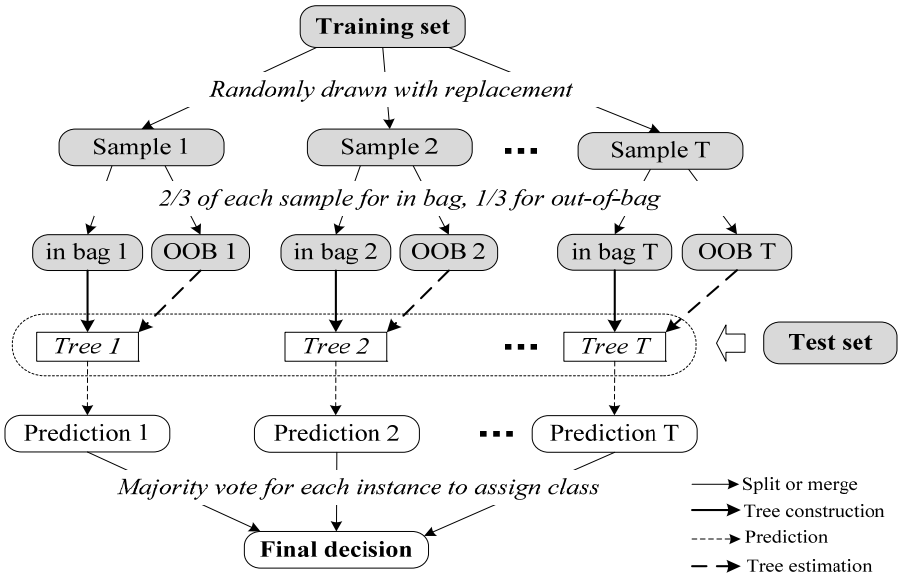


Figure 3-3. Logical architecture diagram of Random Forests

Random Forests commonly require two options to produce a classifier: T, the number of trees and F, the number of selected features. T is equal to the number of bootstrap samples and is a crucial factor to reach a good performance. As T increases, the generalization error drastically decreases, however it converges at a certain value of T (Breiman, 2001). Another variable F indicates the count of features randomly chosen from a given feature set of every node split for a single tree generation. This random feature selection allows each tree to maintain independence from the others, that is, the populated trees have a low correlation with each other. F is computed as the first integer less than log_2M+1, where M is the number of all input features (Breiman, 2001).

3.2.2. **Statistical Measure**

Feature Importance

Random Forests can sort the input features according to their contributions to overall classification performance. The contribution of a feature is quantified by the permutation accuracy in terms of the feature (Breiman, 2001; Guo et al., 2011). This measurement can be semantically described as a sensitivity of the performance on observation of a feature. To illustrate in more detail, for the importance ($FI_{m,t}$) of the m^{th} feature at a decision tree h_t, the Random Forests makes a duplicate of the OOB sample ($\boldsymbol{S}_{to}$) and randomly permutes the values of the target feature (the permuted values are within a range of the original value of the feature) over all the instances in the duplicate, called $\boldsymbol{S}_{m,to}$. The classifier built from the IB is applied to the respective $\boldsymbol{S}_{to}$ and $\boldsymbol{S}_{m,to}$. Finally, the Random Froests investigates an accuracy change between the two OOB samples. As the change is

bigger, the feature is more important. Such that, the importance of the mth feature, $FI_{m,t}$, is quantified as presented in the Eq. (3-1).

$$FI_{m,t} = \frac{1}{|\boldsymbol{S}_{to}|}\left[\sum_{(\mathbf{x}_i,\hat{y}_i)\in \boldsymbol{S}_{to}} \boldsymbol{I}(h_t(\mathbf{x}_i) = \hat{y}_i) \quad - \sum_{(\mathbf{x}_{i,m},\hat{y}_i)\in \boldsymbol{S}_{m,to}} \boldsymbol{I}(h_t(\mathbf{x}_{i,m}) = \hat{y}_i)\right] \quad (3\text{-}1)$$

where i is the index of an instance in $\boldsymbol{S}_{to}$ and $\boldsymbol{I}$ is an indicator function. The $h_t(\mathbf{x}_i)$ stands for the predicted class in Eq. (3-1). The importance of the m^{th} feature (FI_m) is finally averaged over all T trees as follows:

$$FI_m = \frac{1}{T}\sum_{t=1}^{T} FI_{m,t} \quad (3\text{-}2)$$

The importance score in Eq. (3-2) is converted into a percentage over all the M features as shown in Eq. (3-3).

$$FI_m(\%) = 100 \times FI_m \Big/ \sum_{i=1}^{M} FI_i \quad (3\text{-}3)$$

The estimated feature importance is used as a criterion for feature selection, which is a typical optimization problem in supervised learning, to find an optimal feature subset from a given feature set. This is because some of the given features may be uninformative and irrelevant to the classification, so they might contribute to over-fitting classification model. Therefore, an optimal classification model can be generated by selecting relevant features and using only them for the model training.

Prediction confidence

Random Forests is able to output many metrics while training a sample: feature importance, out-of-bag error (OOB error), and so on (Breiman, 2001). The metrics are often used to estimate the quality of the Random Forests classifier. Also, Random Forests can estimate a prediction confidence for each instance as an output produced in the classification stage. The prediction confidence is a measure of how confident the prediction for each class is. For each instance the confidence value (c_i) on a class (y_i) is simply computed as a percentage of the number of trees (T_i) whose predictions are y_i for the instance out of all the trees (Eq. 3-4). The prediction confidence has a value between [0, 1] and the sum over all the classes is 1. This confidence is crucial in the ensemble system as it is used as a key variable to combine the classifiers comprising the system. The confidence values coming from the classifiers are combined through various operators which are described in the section 4.4.3 and 5.1.3.

$$c_i = \frac{T_i}{T} \tag{3-4}$$

3.3. Classifier Optimization

3.3.1. **Balanced Learning**

Training classification models with an unbalanced sample, where the number of instances per class is seriously biased, is a potential problem in practical classification (Chen et al., 2004). Using unbalanced data, most supervised classification algorithms tend to learn toward the correct classification of the majority classes, rather than paying a special attention to the minority classes. An airborne LiDAR system for power utility

management typically collects a laser point cloud along a main centre line of power lines and covers tens of meters of a buffer area (approximately 50 m in our data) from the centre line. Coverage for wire (4.26%), pylon (0.81%) and low object (15.06%) in the buffer area is considerably smaller compared to that for other classes (vegetation, 46.46% and building, 33.41%). Here, a number in each bracket stands for the class proportion of the test site T00 in Figure 3-4. Such unbalanced data also affects a feature selection because the feature importance relies on the overall classification accuracy which might be biased to the majority classes. Consequently, features associated with the majority classes will be more important in their contributions to classification results. To solve such unbalanced problem, balancing our training data is applied by using a combination of under-sampling majority classes and over-sampling minority classes, which is introduced by Chawla et al. (2002). The sample size for each class is determined as a ratio of the total number of instances in the unbalanced data to the number of classes. In addition, all the samplings (under- and over-sampling) are done using the sampling method with replacement, where a sample is randomly drawn from a population and it is put back to the population for next sampling. Thus, the number of instances in the balanced data is finally equal for each class.

3.3.2. **Optimal Feature Subset**

Feature selection is to find an optimal subset from an original feature set which considerably affects classification results. It also has the advantage of the decrease in computational complexity and the increase in classification accuracy by discarding uninformative features. Many feature selection algorithms have been introduced for

supervised classification problem and they are categorized into one of three feature selectors: *filter*, *wrapper*, and *embedded approach* (Saeys et al., 2007). *filter* uses an evaluation function, e.g., feature relevance and mutual information between features, that relies only on properties of the data and that is independent on any particular classifier. While, in *wrapper* classifiers are involved in feature selection. A feature set minimizing classification error is chosen as relevant features. Unlike *wrapper* where algorithms for feature selection and classification are separated, in *embedded approach* classification model (classifier) contains the algorithms. The *embedded approach* has the advantage of interaction with the classification model as well as less computation cost. In this study most relevant feature selection is done by the *embedded approach*, i.e., Random Forests. Another effort for feature optimization is to remove correlation between features. A general feature decorrelation method is Principle Component Analysis, PCA (Rodriguez and Kuncheva, 2006).

3.4. Experiment Setups

3.4.1. **Study Area**

Our study area is a power line corridor of 9.5 km length (electrical substation to substation) in Sacramento, California, USA (Figure 3-4). The RIEGL's LMS-Q560 was mounted on a helicopter and scanned along the transmission corridor at an altitude of 300 m in August 2007. The LiDAR system acquired 24,929,992 points over the entire corridor with a point density of 25 to 30 points/m^2. An average laser footprint size was estimated as of 15 cm. This research selected 10 continuing spans (a span means the

region between two consecutive pylons, a length of 1.9 km) from the entire 65 spans in the corridor, which are large enough to evaluate a classification model using the selected spans and to anticipate its performance over all the entire spans. A reference classification map was produced using commercial software, called TerraScan, and in-house software by data processing expertise at GeoDigital International Inc. (GDI). Two days were taken to manually classify the 1.9 km length of the corridor (8 hours/day). The selected spans were subdivided into 16 subsets called TL08 to TL01, T00, and TR01 to TR07, each of which has an equal size of 125 m × 100 m (length × width). All the subsets were grouped into two categories depending on types of power lines and pylons contained in the scenes. Type I site is a subset which only has the transmission lines (TL08, TL07, TL05-TL01, T00, TR01, and TR02), while Type II site contains both transmission and distribution lines (TL06, TR03, TR04, TR05, TR06, and TR07). Class-dependent site characteristics are described as follows:

Power lines

The test corridor scene includes two different voltage types of transmission lines (115 kV and 230 kV) running parallel to each other, distribution lines (carrying below 110 kV) and pylons. The 230 kV transmission system is composed of 6 bundled cables and 8 single cables, while the 115 kV system is composed of 7 single cables. The transmission cables hang between 10 m to 40 m above ground. The distribution lines are present in some of the subsets and running parallel or perpendicular to the transmission lines. A tubal type and a steel lattice type of pylons support 230 kV and 115 kV transmission lines respectively, while the distribution lines are supported by simple and

small pole type pylons, which height is less than 10 m. All the subsets have pylons except three sites, that are TL07, TL03 and TR01 containing no pylon structures.

Building

The power lines in the test scene run through a residential area where different types of buildings exist: mostly residential houses, commercial buildings, storage houses, and sun-shield roofs built over parking lots. LiDAR points were captured by reflections from rooftops, but occasionally from building walls and chimneys. The overall height of the buildings is less than approximately 5 m. Most buildings are well detached from each other, but they make contact with trees at their edges on occasion.

Vegetation

The test site contains a mixture of deciduous and coniferous trees, where deciduous trees are more predominant than coniferous ones. The coniferous trees are tall, narrow, and columnar so that they look like a vertical structure such as pylons. The tallest tree is 15 m high and the largest one has a 20 m diameter of tree crown.

Low object

The low object class is defined as a class that include fences, vehicles, and grass. The fences are smaller than 2 m in height and are located between adjacent house gardens. Most of the vehicles are passenger cars, which have heights of 2 m or less. Large areas in TL08, TL07, and TL06 are covered by grass (Figure 3-4).

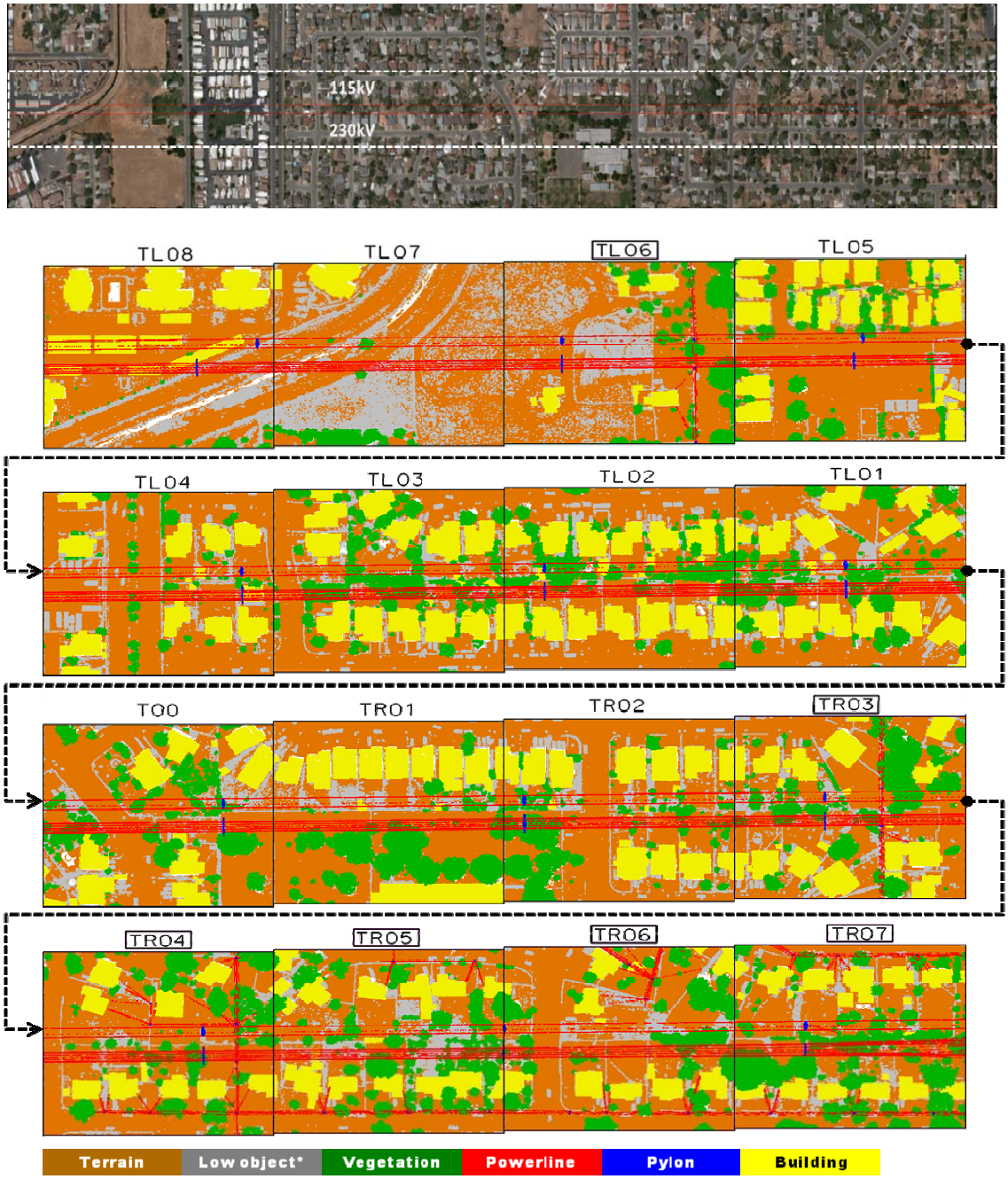


Figure 3-4. Aerial image (the first row) and LiDAR data (other rows) of the study area; LiDAR coverage (white line) and two voltage types of transmission lines (red lines) on the aerial image; the LiDAR is a reference data which is subdivided into 16 subsets including vegetation, wire, pylon, building, low object and ground; Type II sites (site name bounded by a black rectangle) and Type I sites (others).

Ground

In the corridor scene, the ground is the most predominant object. The ground surface is very flat except for a water region passing through the areas of TL07 and TL08 subsets (Figure 3-4).

3.4.2. **Experiment Environments**

Our classification approach is composed of two main processing procedures: feature extraction and Random Forests classification. An algorithm with C++ is developed for the extraction of the 21 features and *Weka* software (Weka 3.5) customized by Livingston (2005) is utilized for Random Forests. Testing is done on Windows 7 with Intel Core 2 Quad CPU and 8GB RAM.

3.4.3. **Accuracy Assessment**

In a similar way that Lodha et al. (2007) suggested, the performance of classifiers is evaluated by using two types of accuracies by comparing classification results and reference data: sample-weighted and class-weighted accuracy. The sample-weighted accuracy is the percentage of correctly classified points to entire numbers of points, while the class-weighted accuracy for a class is the percentage of points correctly labelled as the class to points having the same class label in the reference data. The sample-weighted accuracy indicates an overall classification performance regardless of the degree of predominance of each class in the entire scene. However, the class-weighted accuracy represents the classification performance of a classifier for each class. The averaged

class-weighted accuracy is a mean of the class-weighted accuracies over all the classes. A good classifier satisfies high accuracy in both measures.

3.5. Experimental Results

Classification performance analysis is conducted by the Random Forests trained with the proposed features using a corridor scene (16 subsets) shown in Figure 3-4. For extracting features shown in Table 3-1, two parameters of r and Δh for determining the size of neighbouring systems of sphere (N_S) and cylinder (N_C) were fixed as 1.5 m and 0.75 m respectively. For training the Random Forests, the number of trees T was set to 60. These parameter values are independently determined through an individual sensitivity analysis on each parameter. Another parameter for Random Forests, the number of features F, was differently set to 5 for 21 feature case (Section 3.5.1) and 3 for 7 feature case (Section 3.5.3, Section 3.5.4 and Section 3.5.5). The variable is decided based on the equation depending on the number of input features (Breiman, 2001).

3.5.1. **Balanced vs. Unbalanced Learning**

Random Forest classifier is built with balanced training sample and its performance is compared to the one obtained with unbalanced sample. The training sample was arbitrarily chosen as T00 in Figure 3-4. The balanced training sample was produced as discussed in Section 3.3.1 from the original T00 (unbalanced). Figure 3-5 presents the classification accuracies per class for two classifiers modelled with the balanced T00 and the unbalanced T00. The results showed that balanced learning produced more accurate classification results (97.95%) than unbalanced learning (96.62%). It is also found that

the balanced learning is more effective to avoid biases in classification due to unequal class distribution in the unbalanced sample. In the results, the balanced learning produced 4.4% higher accuracy over the minority classes (i.e., wire, pylon and low object) compared to the unbalanced learning. It is critical to reduce the classification errors in such key objects for conducting the power line safety analysis. Therefore, all the trainings henceforth are performed with balanced training samples.

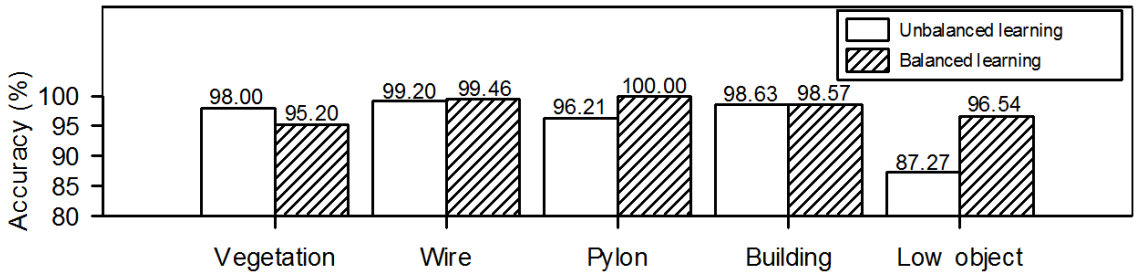


Figure 3-5. Class-weighted accuracies of unbalanced and balanced learning for site T00

3.5.2. **Point Feature Refinement**

The feature refinement process aims to reduce the excessive feature dimensionality by selecting the most suitable features containing relevant information of the targeted classes and linearly combining those features. The feature selection is done by computing the feature importance following the method proposed by Guo et al. (2011). Figure 3-6 shows the computed feature importance for classification results produced by learning the balanced T00: the higher the value, the more important the feature. Each feature is categorized into one of five feature groups (vegetation, wire, pylon, building and common) according to its relevance to the designated feature group. Note that the features for low object class are not specified as it comprises various objects and thus the

generalization of its characteristics is difficult. Instead, the low objects would be mostly characterized by **HG** (height from the ground), which is categorized in the common feature group. For finding the important features, the features which importance shows higher value than a threshold were selected. Among the selected features, a final feature set (f_{12}) was determined by selecting two most important features for each class and four as common features (black boxed features in Figure 3-6).

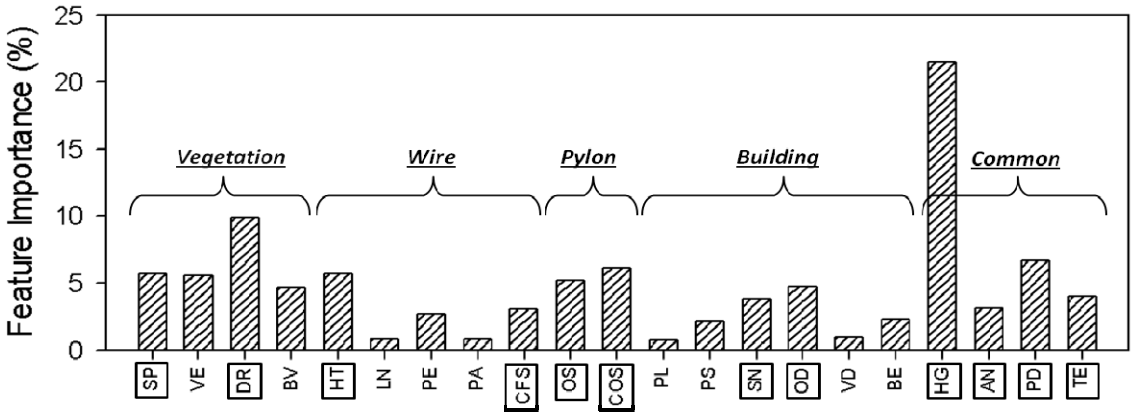
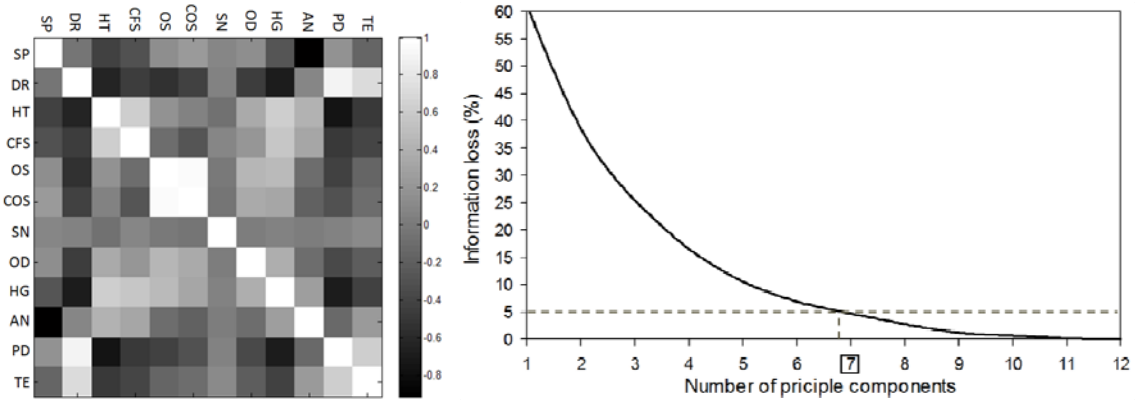


Figure 3-6. Feature importance and feature selection (black boxed features)



(a) Feature correlation map (1: positively correlated, 0: uncorrelated, -1: negatively correlated)

(b) Information loss rate on principle components

Figure 3-7. Feature generation using PCA (Principle Component Analysis)

Figure 3-7(a) shows a correlation of each pair of features in the selected feature set (f_{12}). As shown in the figure, several features including **SP-AN** and **DR-PD** are highly correlated each other. PCA (Principal Component Analysis) was employed to reduce such correlations between the feature pairs. First K principle components maintaining 95% of information were selected (K=7 is determined as shown in Figure 3-7(b)). Finally, 7-dimensional features (f_7) were produced. The same procedure for the feature selection was applied to all the test sites so that they have an identical feature set f_7.

3.5.3. **Sensitivity Analysis to Training Sites**

The classification results would be produced differently even over the same scene when the classifier is trained with different sample. For investigating its sensitivity to the training samples, four different classifiers (C_{T00}, C_{TR02}, C_{TL02}, and C_{TR03}) were used. Each classifier was modelled using Random Forests with the balanced T00, TR02, TL06, and TR03 respectively and with the refined feature subsets (f_7). This study selected T00 and TR02 representing Type I sites, which contains only transmission lines, while TL06 and TR03 for Type II containing both distribution and transmission lines.

Table 3-2 presents class-weighted and sample-weighted accuracies, which were computed by Type I (C_{T00} and C_{TR02}) and Type II classifiers (C_{TL06} and C_{TR03}). Table 3-2 suggests that selecting training sites affect the class-weighted classification accuracy. Type II classifier yielded better overall class-weighted accuracies than Type I classifier, while the opposite result can be observed in sample-weighted accuracy. In particular, compared with Type I classifier, Type II classifier resulted in more accurate classification

for wire, pylon, and low object. An ideal classifier is able to produce high classification performance in both sample-weighted and class-weighted accuracy. In this regard, C_{TR03} yielded the highest accuracies in both accuracies. Moreover, corridor scenes often contain different types of power lines and pylons. It is critical to correctly classify diverse objects with intra-class variations for power line mapping. C_{TR03} (Type II) showed relatively higher class-weighted accuracy in both object classes. This suggests that selecting a training site where many different objects within a certain class are present leads for an optimal classification model.

Table 3-2. Class-weighted and sample-weighted accuracies of Type I and Type II classifier

Classifier		Class-weighted accuracy (%)						Sample-weighted Accuracy (%)
		Vegetation	Wire	Pylon	Building	Low object	Average	
Type I classifier	C_{T00}	91.79	87.92	86.79	94.05	83.56	88.82	91.11
	C_{TR02}	91.56	90.92	73.18	94.90	86.97	87.51	91.92
Type II classifier	C_{TL06}	88.58	93.44	88.71	90.59	87.32	89.73	89.34
	C_{TR03}	90.20	93.10	85.49	92.92	88.64	90.07	91.04

3.5.4. **Classification and Results**

This study selected C_{TR03} as the optimal classifier showing the best classification performance. Figure 3-10 shows a classification map produced by applying C_{TR03} to all the 16 sites. Figure 3-8 provides the class-weighted accuracy for each site and Table 3-3 presents a confusion matrix that presents commission and omission errors per class produced by C_{TR03}. As shown in Figure 3-8, the proposed classification method achieved the class-weighted accuracy in the range of 84.40% to 98.48% for the classes. Its average accuracy was estimated as 90.78% with 3.84% standard deviation across the sites. For all the sites, wire shows the highest classification accuracy of 93.10% and similar accuracies

can be found for building (92.92%) and vegetation (90.20%). The pylon and low object class record less accuracy of 90%. Figure 3-9 shows classification maps over TR02 and TR06. TR02 (Type I site) obtained the highest class-weighted accuracy of 94.31% except the training site (TR03), while TR06 (Type II site) reported the worst class-weighted accuracy of 83.61%.

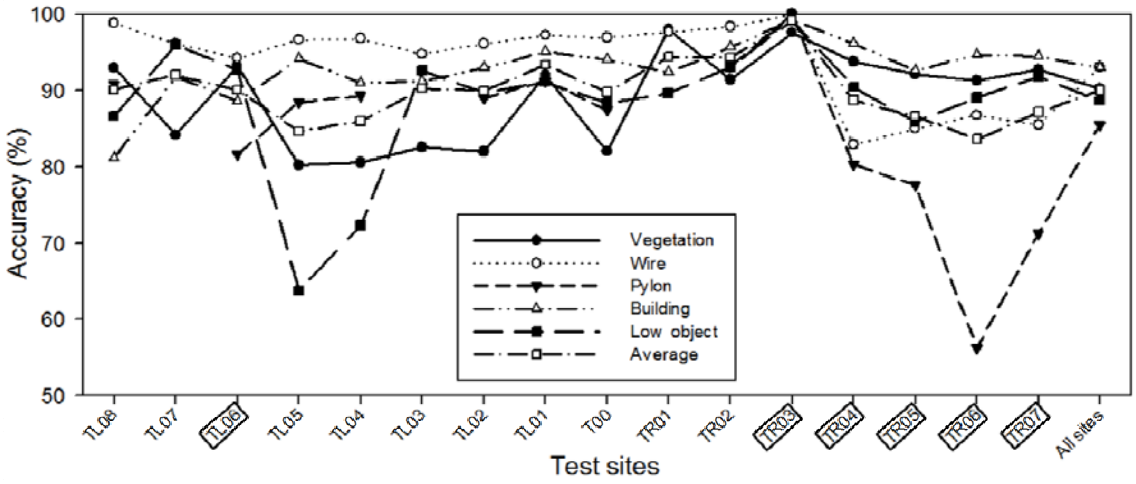
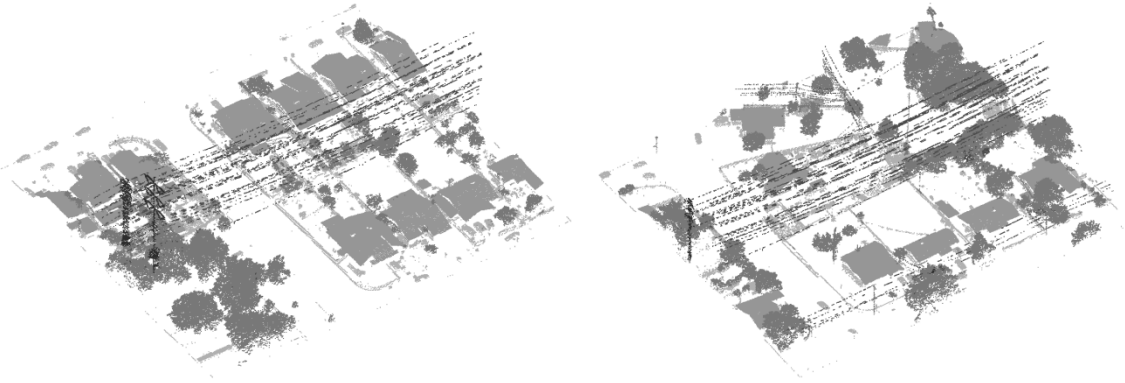


Figure 3-8. Class-weighted accuracies for each site: Type I sites and Type II sites (black boxed).



(a) TR02 (b) TR06

Figure 3-9. Classification results of Type I site (a) and Type II site (b); low object, building, vegetation, wire and pylon assigned in the order from light to dark gray

In Type II sites, the class-weighted accuracies for wire and pylon are relatively lower than those in Type I sites due to the misclassification of distribution lines and small pylons into vegetation. It is believed that such accuracy variance was caused due to high-degree of scene complexity of the test scenes, exhibiting large spatial overlaps between different classes and intra-class variations. This causes difficulties for the proposed features to differentiate one class from the others. For instance, it was observed that most of wire omission errors occurred over regions where the distribution wires pass closely over vegetations. This causes some confusion in vertical-related features to differentiate wire from vegetation or pylon. For pylon objects, most omission errors were observed from small pylons associated with distribution lines, while most commission errors appeared on vegetation, especially over treetops. For vegetation, it was discovered that deciduous trees were better classified than coniferous ones. This is because coniferous trees are narrow and columnar, so some of them were mislabelled as pylon. For building, low error rate was estimated in both omission and commission. As shown in Table 3-3, the omission error rate is higher than the commission one and the most omitted building points, especially building wall points, were misclassified into vegetation or low object. Other omission errors locally occurred around building edges where vegetation and buildings are partially overlapped. For low object, most of the omission errors were observed from container boxes (in TL05 and TL04), which have been misclassified into buildings because their surface properties are similar to building rooftops. Other misclassification errors occurred at fences located near vegetation, which have been mislabelled as vegetation. In addition, some grasses were misclassified into vegetation.

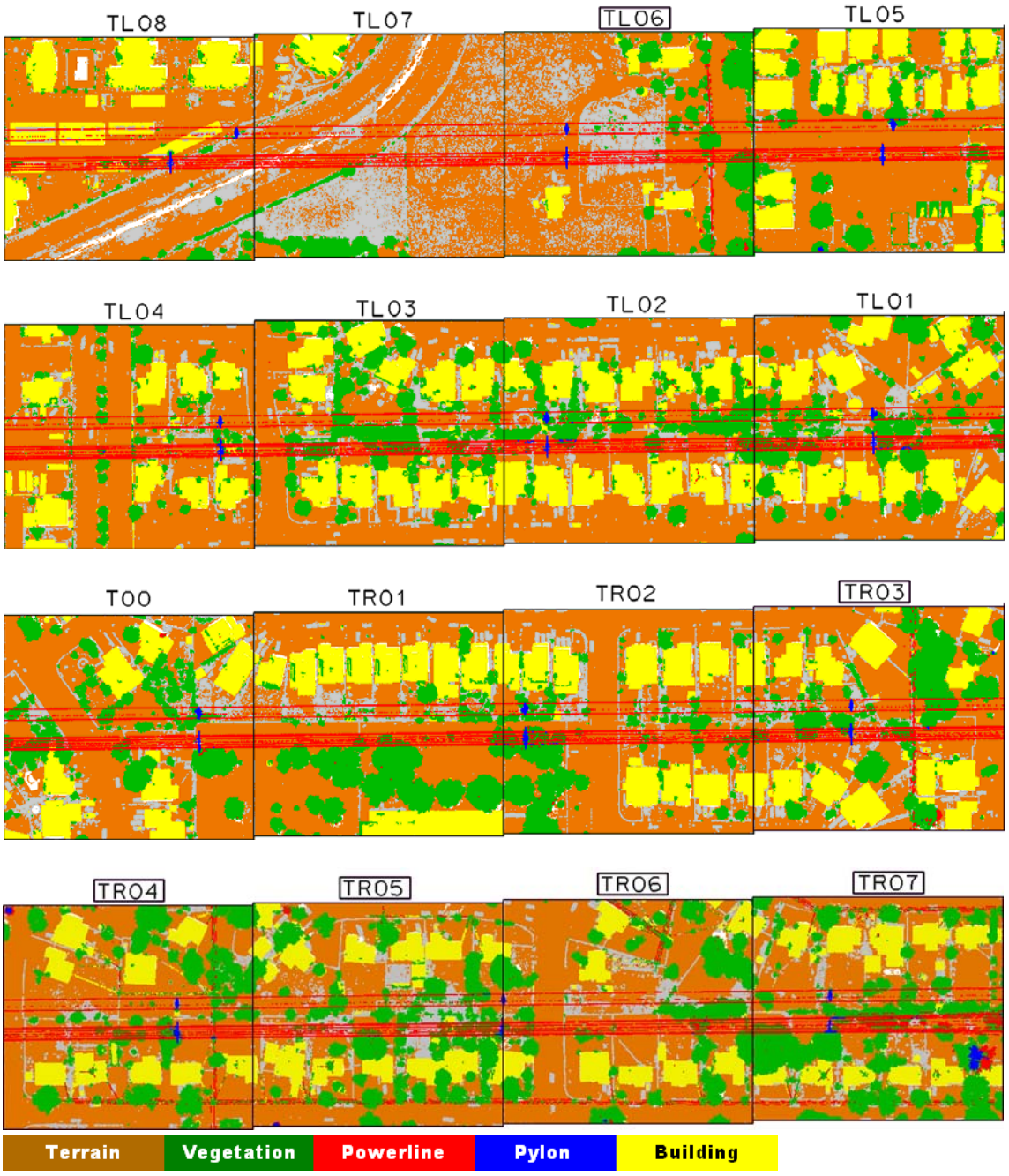


Figure 3-10. Classification map for all sites; vegetation (green), wire (red), pylon (blue), building (building), low object (gray) and ground (remainder); Type II sites (site name bounded by a black rectangle) and Type I sites (others).

So far, it is confirmed that Random Forests trained with proposed features can be considered as an excellent tool to produce high accurate classification map for the

purpose of power line risk management. However, this study also observed the commission and omission errors produced by the classifier which still requires manual editing to eliminate those errors. Random Forests assigns class labels to raw point of clouds, mainly relying on location information gathered from individual point with its neighbouring system. It does not consider certain contextual relations such as spatial arrangement between classes. A Markov Random Field as suggested by Lu et al. (2009) would be useful to further eliminate errors produced by Random Forests. In this study, different features were selected for each class, but their importance was measured for discriminating all the classes, not each class from the others. This would require a future investigation to explore new feature selection methods, especially class-dependent feature selection that can identify which features among entire features distinguish a particular class from the others.

Table 3-3. Confusion matrix across all sites (F=3, T=60)

Class		Predicted					Omission error (%)
		Vegetation	Wire	Pylon	Building	Low object	
Actual	Vegetation	**1,175,449**	9,204	2,171	17,323	99,042	9.80
	Wire	7,154	**131,628**	1,412	825	372	6.90
	Pylon	944	1,705	**16,386**	26	107	14.51
	Building	36,767	2,594	71	**1,005,384**	37,154	7.08
	Low object	30,336	338	50	22,374	**414,476**	11.36
Commission error (%)		6.01	9.51	18.44	3.88	24.80	

3.5.5. **Comparison of Point-based and Grid-based Feature Extraction**

As discussed in the previous section, the classification results shown in Table 3-3 was produced by Random Forests, which was trained with features that were computed by per-point investigation in 3D. In contrast to this, Guo et al. (2011) proposed a grid-

based approach where all the points contained in each grid cell were treated as one group, which has the same feature values and thus same class label.

This study has compared the performance of two Random Forests classifiers; point-based and grid-based classifiers. Both classifiers were trained on the same class-balanced TR03. To construct a point-based Random Forests, LiDAR points were projected into a grid space, where each cell size was set to 25 cm by 25 cm. A cylinder (N_C) with its radius of *r* was created centred at each grid cell for collecting its neighbouring points. To make a fair comparison, the same *r* value (1.5 m) for both classifiers was applied, with which the same features are shown in Table 3-1. Note that **DR** feature was excluded in this experiment because it is not applicable for a grid-based method. Following the same manner used for the point-based method (Section 3.5.2), the experiment selected the important feature set for the grid-based method of **{SP**, **VE**, **LN**, **PE**, **OS**, **COS**, **OD**, **VD**, **HG**, **AN**, **PD**, **TE**}. These features were reduced to five principle components by PCA. The grid-based classifier trained on the refined feature set classified each grid cell. Then, this study assigned the cell's label to all the points within the cell and finally used the labelled points to produce a confusion matrix (Table 3-4).

Table 3-4 summarizes classification result over all the test sites produced by the grid-based classifier. It was estimated that the grid-based classifier produced the sample-weighted accuracy of 86.18% and the class-weighted accuracy (an average of accuracies of all the classes) of 84.32%. Table 3-5 describes a confusion matrix of classification produced by the grid-based classifier subtracted from that of the point-based classifier.

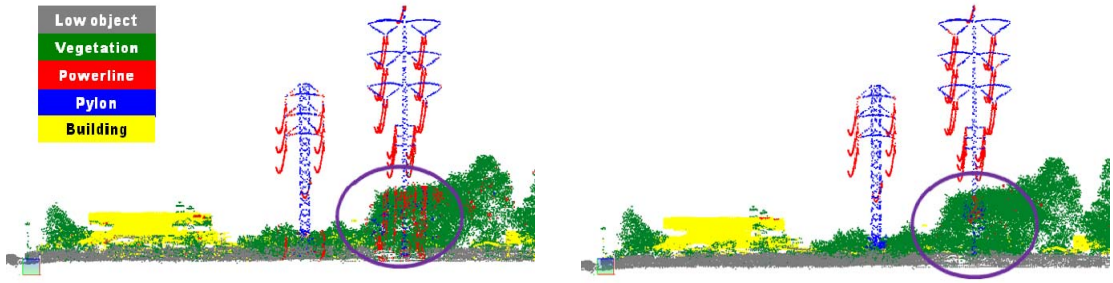
The positive value of diagonal element in the matrix means that the point-based classifier has more correct classification result than the grid-based classifier. The negative value of off-diagonal elements indicates that more misclassifications were caused by the grid-based classifier. The point-based classifier resulted in 4.86% and 5.74% higher than the grid-based classifier in the sample-weighted and class-weighted accuracy respectively. This study found that the point-based classifier is superior to the grid-based classifier across all the error assessments with respect to the omission and commission error rate. As can be seen in Figure 3-11, the grid-based classifier produced many classification errors over vegetation which is placed right under transmission lines, while the point-based classifier well classified the area.

Table 3-4. Confusion matrix of the grid-based classifier (F=3, T=60)

Class		Predicted					Omission error (%)
		Vegetation	Wire	Pylon	Building	Low object	
Actual	Vegetation	**1,133,569**	37,067	2,363	28,570	101,620	13.02
	Wire	8,758	**130,309**	1,035	520	823	7.87
	Pylon	1,004	3,111	**15,020**	3	30	21.64
	Building	64,158	6,370	82	**969,285**	42,075	10.41
	Low object	67,345	11,191	739	39,645	**348,654**	25.43
Commission error (%)		11.08	30.70	21.93	6.62	29.31	

Table 3-5. Subtracted confusion matrix of grid-based classifier from the point-based classifier

Class		Predicted					Omission error (%)
		Vegetation	Wire	Pylon	Building	Low object	
Actual	Vegetation	**41,880**	-27,863	-192	-11,247	-2,578	-3.21
	Wire	-1,604	**1,373**	377	305	-451	-0.97
	Pylon	-60	-1,406	**1,366**	23	77	-7.13
	Building	-27,391	-3,776	-11	**36,099**	-4,921	-3.34
	Low object	-37,009	-10,853	-689	-17,271	**65,822**	-14.08
Commission error (%)		-5.07	-21.19	-3.49	-2.75	-4.51	



(a) Grid-based method (b) Point-based Method

Figure 3-11. Classification results of two approaches

3.5.6. **Computational Complexity**

The computational times of two classification methods (point-based and grid-based classification), which were compared in Section 3.5.5, was estimated. Consequently, the point-based classifier requires 233 minutes per kilometre for computing the features and classifying the data, while the grid-based classifier took 181 minutes (52 minutes faster) as seen in Table 3-6. However, the grid-based method misclassified 219,215 points per kilometre, while the point-based misclassified 142,089 points. This suggests that the grid-based method requires additional time to manually re-classify 77,126 misclassified points compared to the point-based classifier. Moreover, implementing a parallel data processing with multiple computing systems will decrease the importance of time factor and consequently allow our approach to be applied to a rapid classification of power line scenes.

Table 3-6. Computational time of point-based and grid-based methods (minutes/km)

Method	Feature extraction								Random Forests		Total
	Height	Hough Trans.	Eigen	Surface	Convex Hull	Echo	Density	Vertical Profile	Training	Testing	
Point-based	9.61	175.69	0.55	8.06	16.67	0.06	0.02	0.66	4.05	4.02	232.93
Grid-based	6.78	137.68	0.44	6.72	10.42	0.04	0.01	0.21	4.37	4.02	180.89

3.6. Summary

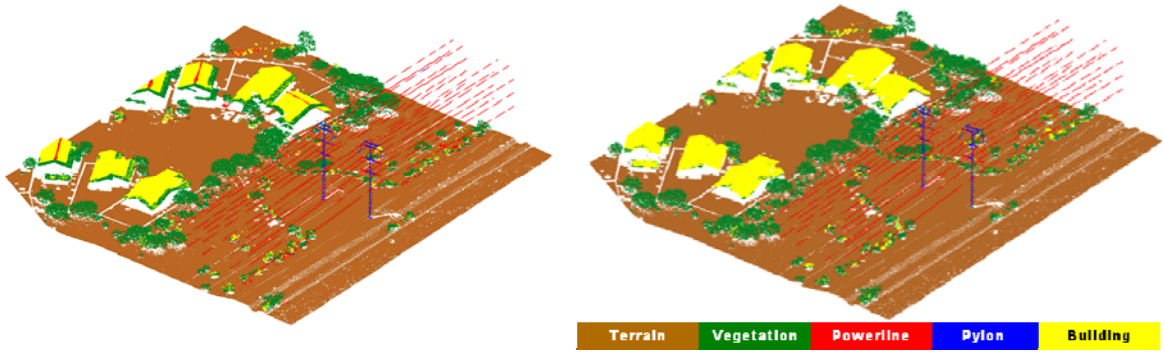
In this chapter, the potential of a supervised classification method, especially Random Forests, was investigated for classifying corridor scenes from airborne LiDAR data. The point-based method which extracts total 21 LiDAR point features to build Random Forest classifiers was applied. The experimental results suggested that it is important to train the classifier with class-balanced training samples, which produced 1.33% and 4.44% higher accuracies in respective measures compared to a classifier trained from class-unbalanced data. Additionally, the balanced learning resulted in almost equivalent accuracy across all the classes. Besides balancing training samples, optimal training sample selection, which chooses samples containing many different objects observed over test sites, led to more uniform and higher classification accuracy over the classes. In the comparison of point- and grid-based approach, the classification model trained from point-based features showed 4.86% and 5.74% higher in the respective sample-weighted and class-weighted accuracy than one from grid-based features. This result suggests that the point-based classifier is more suitable for discriminating vertical overlapping of multiple objects. All the procedures done in this chapter were taken approximately 4 hours to classify LiDAR data of 1.0 km power line corridor, which is much more efficient against manual classification. Even though the proposed supervised classifier has demonstrated its success in corridor scene classification, the classifier still produced misclassification errors, especially regions where sufficient neighboring points are not collected to exhibit their distinctive characteristic such as building roof ridges, hips, and eaves. Thus, future investigation is necessary to further rectify those errors by

introducing more features, called object-based features described in chapter 4, extracted with different perspectives.

4. Classification Using LiDAR Object Features

In chapter 3, a point-based approach is applied to classify power line corridor scenes, where LiDAR features are extracted for every single point and a class label is individually assigned to each point. The method is more appropriate for power line corridors compared with conventional grid-based approach. However, as the class label assignment depends on features derived from neighbors, it occasionally shows classification errors when insufficient neighbors are collected as seen in Figure 4-1(a). Similarly to GEOBIA (Hay et al., 2005) which partitions optical imagery into meaningful image objects (groups of pixels, also called image segments) and treats the objects as spatial units for classification, this chapter introduces an object-based approach for LiDAR data where a point cloud is decomposed into a set of point groups (termed as point segmentation in this dissertation), each group (or point segment) is formed by gathering points based on a specific homogeneity. Plane-likeness is mostly-utilized homogeneity in detecting surface-like objects such as terrain (Yao et al., 2009; Wang and Tseng, 2011; Yang and Dong, 2013) and building roof (Filin and Pfeifer, 2006; Yao et al., 2009; Wang and Tseng, 2011). In addition to the plane-likeness, this dissertation considers line-likeness in the point segmentation so that linear objects such as transmission line and pole-type pylon are highlighted. A detailed description of the point segmentation methods is addressed in section 4.1. Section 4.2 describes object-based features, i.e., linear and planar features, derived from each segment or contextual properties with its neighboring segments. Section 4.3 introduces an ensemble system which combines classification models built from the respective linear and planar features.

Section 4.4 evaluates the ensemble system and compares it with the point-based approach addressed in chapter 3. Finally, section 4.5 summarizes this chapter. The research work corresponding to this chapter has been addressed in Kim and Sohn (2010) and partly (Section 4.1 and 4.2) described in Sohn et al. (2012).



(a) Classification results using point features

(b) Classification results using point and object feature

Figure 4-1. Considerable classification improvement over building ridges and edges by introducing object-based features (Kim and Sohn, 2010)

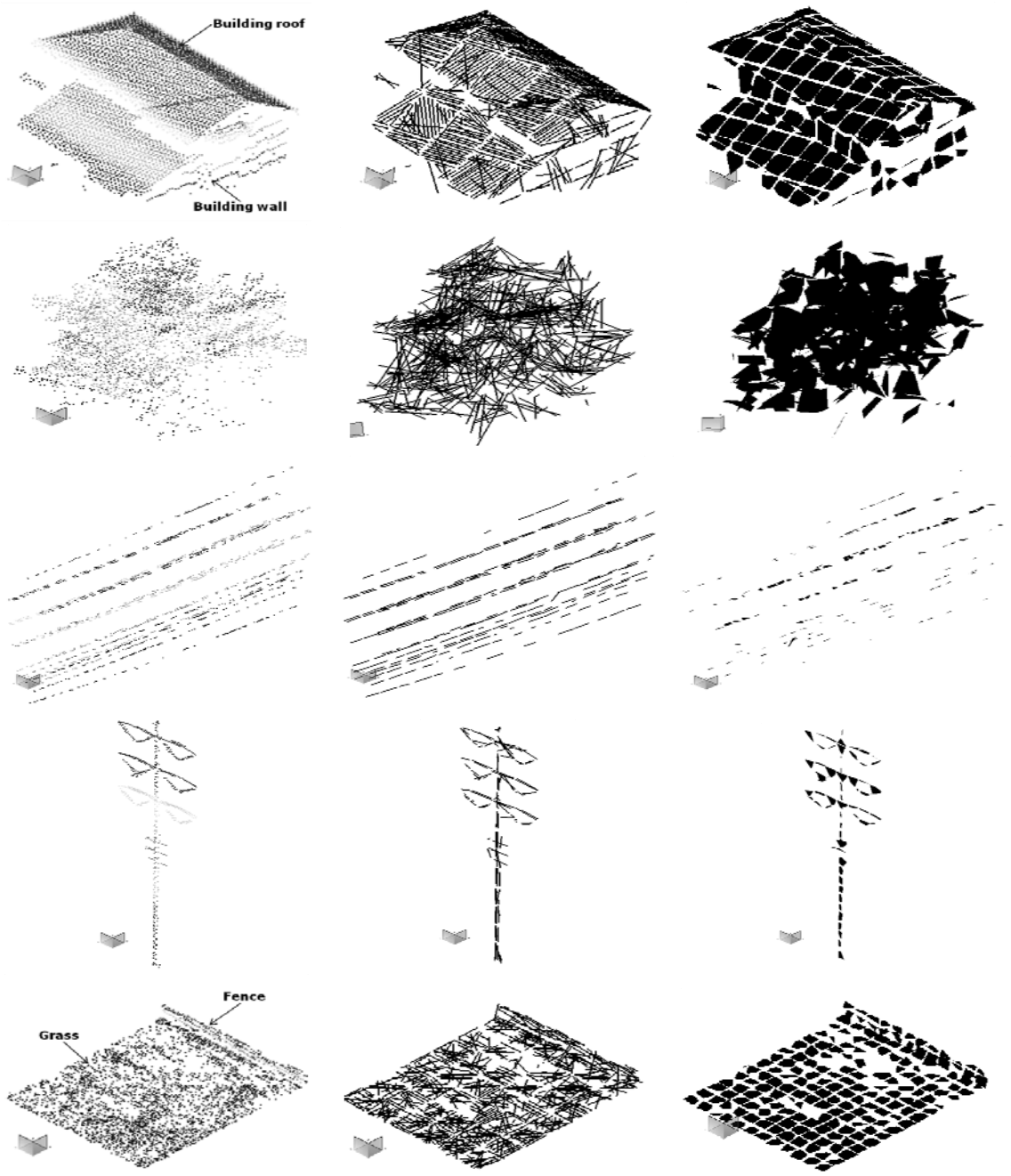
4.1. Pre-processing: Point Segmentation

Point segmentation is a common preprocessing procedure to object-based classification and OBPA. Two ways for segmenting 3D point cloud have been used to handle LiDAR data. The first one is the partition of given points into many clusters, each of which represents an object. For instance, unorganized points can be clustered into a mixture of inherent shape models, i.e., plane, sphere, cylinder, cone, and torus (Schnabel et al., 2007). The second method is the grouping of points according to a specific homogeneity attribute. Individual point is assigned to one of the groups as a membership. In general, surface feature is used to cluster more points over planar objects such as

terrain and building roofs than non-planar ones from Airborne LiDAR data (Sampath and Shan, 2010; Verma et al., 2006) and Terrestrial LiDAR data (Lim et al., 2009).

The segmentation method in this dissertation follows the second method which involves the gathering of meaningful point groups. In this chapter the point segmentation is performed twice to the LiDAR data independently: in the first point segmentation neighboring points with a similarity are grouped together by the measure of their linearity, while the second one is based on a planarity. The 3D line and plane models are approximated from point sets captured by a voxel space. This is because the object approximation is straightforward and computationally simple compared with point clustering from seed points. The membership points for each of the generated object models are considered as an object segment. Before extracting the object models, this chapter produces a 3D cubical voxel grid with the given LiDAR points. Denote that the ground points are excluded from our LiDAR data by an existing ground filter (Sohn and Dowman, 2008), that is, the voxel grid includes only above ground points. A sequential RANSAC (RANdom Sample Consensus) which repeats traditional RANSAC for outliers remained in the previous stage was applied to points within each voxel. The modeling fitting algorithm based on PCA is employed to produce line or plane models from inliers. Such iterative model approximation ends when it does not generate any model or n_0 models are already generated for a voxel. This chapter limits the maximum number of models to reduce the complexity in feature variable computation and set n_0=20 through visual inspections where few models (mostly 10 below models) were populated from power line and pylon (see the third and fourth row of Figure 4-2(b)) compared to models

from building, vegetation, and low object (see the first, second, and fifth row of Figure 4-2(c)).



(a) point cloud (b) line extraction (c) plane extraction

Figure 4-2. Results of line and plane extraction for each class object (building, tree, power line, pylon and low object from the top)

4.2. Object-based Features

The point cloud over a power-line corridor scene is applied into deriving geometric primitives (i.e., either line or plane in this study) through the point segmentation described in Section 4.1. Figure 4-2 presents the targeted corridor objects which are visualized as geometric primitives by the point segmentation. As shown in Figure 4-2(b), power line and pylon are visually well formed into regular line segments, especially individual power lines are modeled without any missing parts. On the other hand, building, tree, and low object (mostly grass) are unrecognizable and line segments over these objects have irregular line vectors compared to those over power lines and pylons. For the plane primitive depicted in Figure 4-2(c), relatively large and regular-shaped plane segments are produced over a building rooftop. Moreover, they have mostly equal surface slopes with their neighbors when they are not near roof edges where the surface slope varies. Unlike the building rooftop, plane segments coming from a tree are more irregular in terms of the size, shape, and slope. Power line and pylon mostly produce small sized plane segments, but for pylon, the segment size would be various depending on pylon type. A low object shows similar characteristics as the building rooftop. Based on these distinguishable characteristics between the power line corridor objects, this chapter designed object-based features extractable from the line and plane segments, termed linear features and planar features respectively. Linear features discriminate linear shaped objects such as power line and pylon from non-linear shaped objects such as building and vegetation, while planar features distinguish planar shaped objects like building rooftop from non-planar shaped objects such as power line, pylon, and

vegetation. Lastly, a low object is obviously characterized by its height rather than features from the geometric primitives because it includes a mixture of various objects, grass, fence, vehicle, and container box, so finding representative characteristics is not straightforward. Nevertheless, these object-based features help to rectify classification errors caused due to insufficient neighbors in the point-based classification. As seen in Figure 4-1(a), the areas near intersection edges (i.e., building ridges) over building roofs are misclassified into the wire or vegetation class, but they are mostly corrected in Figure 4-1(b) thanks to a contribution from the object-based features. The detailed description of the features is addressed in the following section.

4.2.1. **Feature Variables**

The object-based features (Table 4-1) are composed of unary and contextual features. The contextual features have been highlighted in grey colour in Table 4-1, and the others are unary features. A unary feature is an attribute representing shape, area, length, colour, etc. which is a single object segment (i.e., a line or plane segment), while a contextual feature indicates a spatial correlation between the segments and their neighbours. For example, neighbours (ls_i) for a certain line segment (ls) are line segments within voxels surrounding a voxel where ls lies. For the ls, the unary features are computed from the line segments (ls) and the contextual features are computed as averages of differences in feature values between line segment pairs (ls, ls_i). The linear features are stored in a 1D array vector (f_{L}). For a plane segment (ps), unary and contextual feature values (f_{P}) are derived in the same way. As another neighbouring system, this chapter regards segments as co-neighbours when they lie within a voxel column. This neighbouring system is used

to calculate contextual features in a vertical direction like Deviation of line vector (DLV) introduced in Table 3-1.

Table 4-1. Descriptions of linear and planar features

Feature		Abbrev	Equation	Description
Linear features	*Line height*	LH		Height from ground at the midpoint of a line segment (*ls*)
	Line slope	LS		Angle between the line vector of *ls* and the XY horizontal plane.
	Line residual	LR		Root mean square of perpendicular distances from membership points to *ls*.
	In-out cylinder	IOC	$\frac{n_o - n_i}{n_o}$	Ratio of point counts in two coaxial cylinders (inner and outer cylinder) with radius r_i and r_o (r_i>r_o) produced from an axis, *ls*. The number of point is n_i in the inner cylinder and n_o in the outer cylinder.
	Line count	LC		The number of line segments in a voxel. All the line segments within the voxel have the same value.
	Orientation variation	OV	$\sum_{i=1}^{n_l} \frac{\lvert\alpha - \alpha_i\rvert}{n_l}$	Variation of horizontal orientation angle difference between (ls,ls_i). α and α_i are the orientation angles for *ls* and ls_i respectively. $\lvert\alpha - \alpha_i\rvert = [0, \pi/2)$
	Collinearity	CL	$\sum_{i=1}^{n_l} \frac{\lvert\Delta\theta_l\rvert}{n_l}$	$\Delta\theta$l is an angle between (ls,ls_i). $\lvert\Delta\theta_l\rvert = [0, \pi/2)$
	Deviation of line vector	DLV	$\sqrt{\frac{\sum_{i=1}^{n_l} \Delta\theta_l^{\,2}}{n_l}}$	Standard deviation of $\Delta\theta_l$
	Coplanarity	CP	$\sum_{i=1}^{4} \frac{d_i}{4}$	Denote (p_1,p_2) is two endpoints for ls and (p_3,p_4) for ls_i. when a plane is produced using three points except p_i, d_i is a perpendicular distance from pi to the plane.
	Vertical space-occupying	VSO		Analyze the measure of space occupied by line segments present above and below ls.
	Horizontal space-occupying	HSO		Analyze the measure of space horizontally occupied by line segments from ls.
Planar features	*Plane height*	PH		Height from ground at the centroid of a plane segment (*ps*)
	Plane residual	PR		Root mean square of perpendicular distances from membership points to *ps*.
	Ground presence	GP		The number of ground points below ps.
	Plane area	PA		Area of ps.
	Plane count	PC		The number of plane segments in a voxel. All the plane segments within the voxel have the same value.
	Plane normal variation	PNV	$\sum_{i=1}^{n_p} \frac{\lvert\Delta\theta_p\rvert}{n_p}$	$\Delta\theta_p$ is Variation of plane normal between (ps,ps_i).
	Deviation of plane normal	DPN	$\sqrt{\frac{\sum_{i=1}^{n_p} \Delta\theta_p^{\,2}}{n_p}}$	Standard deviation of $\Delta\theta_p$

Linear features

From a line segment (*ls*) and membership points of *ls*, following unary features are computed: line height (LH), line slope (LS), line residual (LR), in-out cylinder (IOC), and line count (LC). LH is the elevation from ground surface to the midpoint of *ls*, and LH is a useful feature over all the classes, especially for objects that have a constant height such as grass and vehicle, i.e., the low object class. Line slope (LS) is an angle between the line vector of *ls* and the XY horizontal plane, and LS is expected to show slant slope values over power transmission conductors because pylons typically hold the conductors horizontally, while approximately 90 degrees over pole type pylons and 0 to 90 degrees slope values over frame type pylons composed of horizontal bracing, truss bar, and vertical steel beam. LS for buildings shows regular slope values (mostly flat or slant slopes) depending on the rooftop slopes, while arbitrary slopes over vegetation. Line residual (LR) is the root mean square of orthogonal distances from each membership point to *ls*. Linear objects such as power line and pylon would have low values in LR compared with non-linear objects. In this chapter it is anticipated that LR values exhibit a discrete distribution over all class objects (power line < pylon < building, vegetation, low object). In-out cylinder (IOC) is a ratio of point counts captured by two coaxial cylinders (inner of radius r_1=15cm and outer of r_2=50cm) extended from a *ls*. The length of the cylinders is the same as the line length of the *ls* and IOC directly checks if *ls* originates from a linear like object by investigating the presence of points in a gap between inner (C_i) and outer cylinder (C_o).

Denote the point count n_i in C_i and n_o in C_o, points in C_i are completely regarded as membership points used for approximating *ls*, hence there exist few points in the space ($\Delta C=C_o-C_i$) if *ls* is produced over a linear like object (i.e., power line or pole type pylon). Thus, power line and pole type pylon are expected to show low values in IOC due to their proximity clearance. On the other hand, ΔC for line segments generated over the building, vegetation, and low object contains some neighboring points because of their planar or volumetric characteristic. This leads to high values in IOC for them. As addressed in Section 4.1, the segmentation algorithm would generate either single or multiple line segments over a voxel depending on regularity in the spatial distribution of the points in the voxel. Line count (LC) is the number of the line segments populated within each voxel. As shown in Figure 4-1, linear like objects (i.e., power line and pylon) demonstrate relatively fewer number of line segments compared with non-linear like objects (i.e., building, vegetation, and low object).

In addition to unary features, this chapter examines contextual features whose values are computed from line segment pairs. Orientation variation (OV) indicates a mean variation of 2D orientation angles between pairs of line segments (ls, ls_i) which are projected onto a horizontal XY plane. Power lines typically run parallel with each other to maintain the clearance between them, so OV values for power lines might be close to 0 degree. Pylon would show arbitrary OV values because its line segments have random tangent values. Also, the other class objects are expected to have arbitrary OV values, but this chapter observed that building roofs mostly produce parallel line segments coming along each scan line. Collinearity (CL) indicates the mean of 3D angles between pairs of

lines segment and values range between 0 to 90 degrees. CL over power line has the smallest values which are close to 0 degree, while CL values over pylon vary depending on the pylon type and shape. A pole type pylon could have almost 0 degree around the pole, but larger values around intersecting areas between the pole and cross arms. Furthermore, CL values over a frame type pylon depend on the pattern of the steel bar (or truss) arrangement. For building the CL values are low due to the same reason addressed in the OV feature. The other class objects could have large CL values because of random line vectors of populated line segments.

Deviation of line vector (DLV) is the root mean square of 3D angles between the line vectors of line segment pairs (ls, ls_i) in a height direction. DLV values are small when the line vectors placed in a vertical direction are parallel, so power line and pylon are expected to be small in DLV. If line segments populated over building roofs are parallel because of the scanning direction patterns, DLV value is small. Otherwise, its value is large. For vegetation mostly large values might be derived because of random line segments over trees. Coplanarity (CP) investigates whether a pair of line segments lies on an infinite plane, which is able to distinguish line segments over planar objects (i.e., building roof) from ones over non-planar objects. A common way to measure it is to select three points out of the end points (two from each segment) from a line segment pair and to produce an infinite plane using the chosen points. An orthogonal distance from a remaining point to the plane is computed. This task repeats for each end point of the line segments, and CP is then an average value of the distances. Vertical space-occupying (VSO) is motivated from Continuous On-Segment (COS) described in the section 3.1.2 in

order to characterize vertical like objects (i.e., pylon) by investigating the number of continuous occupied voxels with line segments in a height direction. Horizontal space-occupying (HSO) tends to highlight objects horizontally occupying a large space such as building rooftops. When a voxel contains *ls*, HSO of the *ls* is the number of occupied voxels among 8 neighbours around the voxel in a horizontal direction. HSO shows a high value over building rooftop having relatively larger surface area and tree crown having larger volume compared with power lines and pylons.

Planar features

For a plane segment (ps), unary features, i.e., plane height (PH), plane residual (PR), ground presence (GP), plane area (PA), and plane count (PC), are measured. PH is a height value from ground to the centroid of ps. Similar to LH, PH is useful to discriminate over all class objects, especially in separating the low objects. PR is the root mean square of perpendicular distances from points to the ps. PR is expected to be helpful to recognize planar like objects such as building roofs and walls from objects having large volumes. However, linear like objects would be also highlighted because plan segments populated from a point group shown as a line are too much supported by the points which means that most points are involved in the plane model fitting. This brings almost zero value in PR. A feature to separate these objects (i.e., linear and planar like objects) is ground presence (GP), which investigates the presence of ground points under each plane segment. GP values changes according to the surface characteristics reflected by the laser beam, which determines the degree of laser penetration. The laser pulse which passes through the linear like objects produces multiple returns (including

reflections from ground surface) because their surface area is relatively smaller than the laser footprint size. Therefore, ground points are mostly observed under them. Vegetation (i.e., tree and shrub) is also a well-known penetrable object, so also ground points may be present. However, from building rooftops with a large rigid surface no laser penetration brings no ground points. Plane area (PA) indicates the area of the plane segment. For example, larger plane segments are populated over building rooftops because the rooftops occupy large spaces, while smaller plane segments are produced over power lines and pylons. PA values for vegetation vary because random plane segments in size are generated over trees. Similarly to LC, plane count (PC) is the number of plane segments produced in each voxel. Relatively high PC values are shown over vegetation as many plane segments are necessary to represent a volumetric object like a tree, while low PC values are expected from building rooftops and power lines as they can be modeled using a few plane segments.

As a contextual feature for a plane segment and its neighbours, plane normal variation (PNV) is a mean of angle differences between normal vectors of plane segment pairs (ps, ps_i). Plane segments formed over a building rooftop tend to have uniform normal vectors except for regions where the roof surface changes such as intersection edges and step edges. On the contrary, vegetation randomly produces plane segments with arbitrary normal vectors, so it is expected to show high PNV values. Like the DLV, deviation of plane normal (DPN) is a root mean square of angle difference between normal vectors of plane segment pairs lying on a voxel colume in a height direction. A zero DPN value is observed over building rooftop since a single plane segment would

typically come from it, which means a zero deviation for only one sample. Unlike building rooftop, vegetation produces many plane segments whose normal vectors are arbitrary in the vertical direction. Such that, vegetation shows higher DPN values than building rooftop.

Finally, this chapter summaries features introduced so far: 11 linear features, f_L={LH, LS, LR, IOC, LC, OV, CL, DLV, CP, VSO, HSO}, from a line segment and 7 planar features, f_P={PH, PR, GP, PA, PC, PNV, DPN}, from a plane segment.

4.2.2. **Unused Points**

The linear feature values f_L computed from a line segment are assigned to each of the membership points of the segment. 90.2% points across all the sites have been used to produce the line segments, that is, 10.8% points not belonging to any line segments do not have f_L. Note that this chapter does not treat ground points, so this statistic does not include the ground points. To assign f_L to each of the remaining unused points, this chapter measures the spatial proximity between the point and the line segments. Then, the f_L of the closest segment is assigned to the point. In the same way, this chapter assigns f_P to 9.4% points unused for the plane segmentation. In consequence, all the points have the object features, f_O={ f_L, f_P}.

4.3. Ensemble system

An ensemble system is a committee of experts (i.e., classifiers), each of which is supposed to make a independent decision individually and the system decreases a risk to reach to a bad decision by combining the opinions from the experts even if the decision

of each one may not be the best. Many classification problems are always faced with the decision making and problem solving and it is believed that combining several decisions yield better classification quality. Ensemble methods utilize the same strategy which combines results from multiple classifiers for a final decision. According to a literature review, the ensemble methods are mostly superior to a single classifier (Kittler et al., 1998; Dietterich, 2000). In general, the ensemble methods are categorized into two groups, the first one selects the best expert amongst given an ensemble of classifiers and the selected classifier is used for decision making. The second one uses all the given classifiers by combining their results. In conventional ensemble methods, local decisions from individual classifiers are combined in parallel (parallel structure) or sequential (sequential structure). The sequential structure is mainly employed when each of the classifiers has a different role such that a classifier is able to identify a specific class, i.e., a binary class problem, while the parallel structure is used when all the classifiers work for all the classes, i.e., a multi-class problem. Random Forests used in this study follows the parallel structure. In such ensemble systems especially built with the parallel structure, what and how to combine is a critical issue. The following section will discuss this issue in further detail.

4.3.1. **General combining methods**

The combining methods are divided into two groups according to combining source and combining rule. The combining sources are typically data, feature, and classifier and they are termed data-level, feature-level, and classifier-level fusion respectively. The data-level fusion produces a new data by fusing different kinds of raw data, which is

different from original data. As an example, a high resolution color image can be generated by combining a panchromatic image and a multispectral image. The feature-level fusion combines the respective feature sets extracted from different sensory data. The individual feature sets can be produced from a data by extracting them under different circumstances. Data analysis using features from LiDAR and optical images is one example of the feature-level fusion. Finally, the classifier-level fusion is to combine outcomes of classifiers which are built with different training samples or different feature sets. In general, the classifier outcome is either a predicted class label or prediction confidence per class (described in the section 3.2.2). The classifier-level fusion has various combination rules depending on the classifier outcome. When the outcome is the predicted class label, typically majority voting is employed as well as weighted-majority voting, behavior knowledge space (BKS), and borda count. For the class-per prediction confidence, algebraic combining rule (average, weighted average, trimmed mean, minimum, maximum, median, product, generalized mean rule and so on), decision template and Dempster-Shafer based combination can be applied (Polikar, 2006). However, there is no best combination rule because it only depends on the particular problem. Therefore, finding the best combining method suitable for our ensemble system is necessary. A detailed description on the combining rule is addressed in the section 5.1.3.

4.3.2. **Ensemble system design**

The previous section describes the general combining methods and this section will discuss the ensemble design applied for this study. A summary of the design can be found

in Figure 4-3, as shown in the figure, an ensemble system has a parallel structure by combining two classifiers built with the feature set, f_L and f_P addressed in the section 4.2.1. The f_L and f_P are extracted by segmentation and feature extraction from a training sample where terrain has been filtered out. Then, Random Forests builds classifiers (C_L and C_P) using the respective f_L and f_P. The classifiers are applied to the corresponding feature sets of test samples, which are extracted from the same feature extraction method. Finally, prediction confidence values are combined based on the average rule for each class. A class label with highest confidence value is assigned. The average rule is typically used when the same kind of classification model (i.e., Random Forests) is applied in an ensemble system.

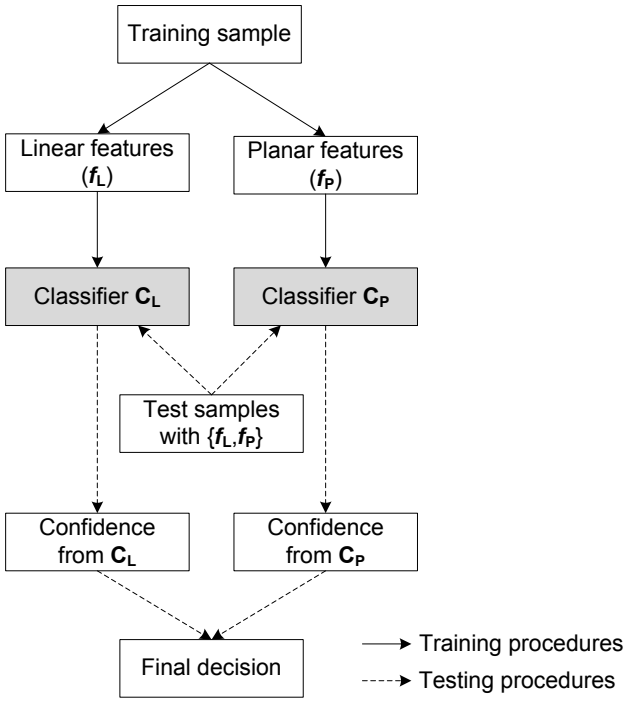


Figure 4-3. Classification workflow using an ensemble system composed of multiple classifiers (linear and planar classifier)

4.4. Experimental Results and Discussion

4.4.1. **Experiment Setup**

In this chapter, some of the classification model optimization methods addressed in the section 3.5.1 and 3.5.2 are taken, which are the balanced learning and feature de-correlation. The feature selection is not done in this chapter due to a difficulty in categorizing the features according to the class relevance. Instead, for treating features efficiently, this chapter rescales all features using a bipolar sigmoid function so that their values are in range of [-1, +1]. In order to maintain original values for the height related features (i.e., LH and PH), the actual value of the height is radically used as a classification variable to separate objects with different size rather than its rescaled value. In addition to the feature rescaling, this section conduct a Principal Component Analysis (PCA) to eliminate correlations between the features. According to (Kim and Sohn, 2013), this section has chosen K principal components to maintain 95% of information from an original feature set. The PCA sets K=9 and K=6 for f_L and f_P respectively.

Ideally, training samples require more diverse objects representing intra-class variations to produce optimal classification models (Kim and Sohn, 2013). Hence, in this section, the same training sample is being used, TR03 is optimally selected in the preceding study since it contains both distribution and transmission lines. TR03 has unbalanced class distribution which is categorized into majority (building and vegetation) and minority classes (power line, pylon, and low object). As training from such

unbalanced sample causes biases in classification, this section will balance TR03 by under-sampling the majority classes and over-sampling minority classes (Chawala et al., 2002; Kim and Sohn, 2013). Thus, this chapter produce two balanced TR03 samples, one is $bTR03_L$ having f_L and another is $bTR03_P$ having f_P.

4.4.2. **Linear features vs. Planar features**

This chapter has built two Random Forests classifiers, C_L and C_P, which are trained by $bTR03_L$ and $bTR03_P$ respectively. The C_L is expected to discriminate linear-like objects, while the C_P is expected to identify planar-like objects. The aim of this experiment is to investigate the relevance of the examined object features (f_O) to each class. Table 4-2 presents the averages of the class-weighted accuracies and the sample-weighted accuracies for the classifiers C_L and C_P when the classifiers are applied to the entire sub sites (TL08 to TL01, T00, and TR01 to TR07). C_L which leads to 4.2% higher in the class-weighted accuracy than C_P, and is a better classifier over all the classes, whereas the C_P has 1.48% better accuracy over all the points regardless of their class labels compared to the C_L. Figure 4-4 depicts the class-weighted accuracies of the C_L and C_P to represent the class relevance of the linear features and the planar features (i.e., f_L and f_P). As expected, linear like objects such as power line and pylon have been better classified by the C_L rather than the C_P (5.44% higher for power line and 19.67% for pylon). However, the accuracies of the two classifiers for pylon are relatively lower compared to ones for other class objects due to its various types, from pole typed to steel framed. Building, planar like object, shows better classification accuracy (3.77% higher) using the C_P than the C_L. For vegetation, two classifiers yield similar accuracies as it is

categorized into neither linear like object nor planar like. However, since low object includes a mixture of various types of objects, i.e., grass (neither linear nor planar), fence (linear), vehicle (planar), and container box (planar), obtaining principle features accommodating such diverse objects is difficult. After all, both C_L and C_P produce less than 70% accuracy for low object. In this section, it is concluded that C_L is able to classify linear like objects, while C_P can identify planar like objects.

Table 4-2. Classification accuracies for each individual classifier (C_L and C_P)

Classifier	Average of class-weighted accuracy (%)	Sample-weighted accuracy (%)
C_L	81.02	86.48
C_P	76.82	87.96
C_L - C_P	+4.20	-1.48

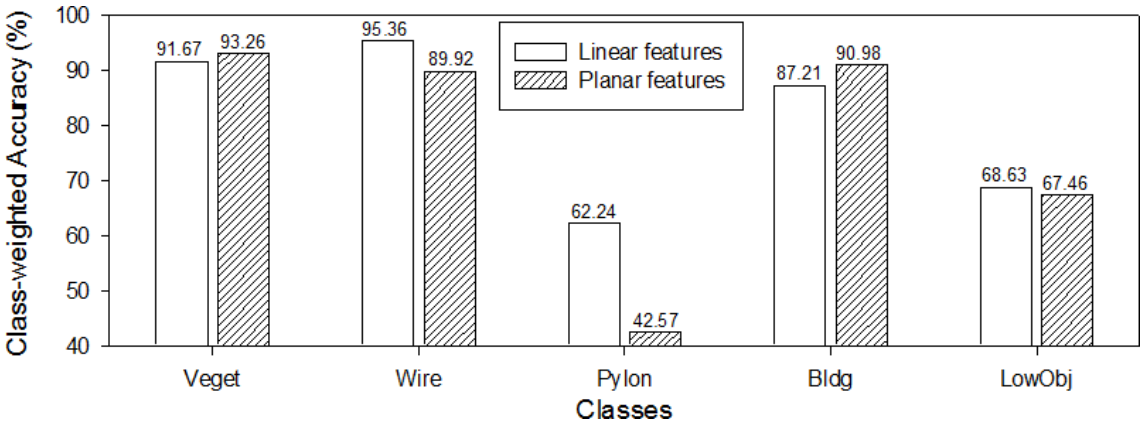


Figure 4-4. Class relevant accuracy comparison of classifier C_L and C_P

4.4.3. **Combining method selection**

As mentioned in the section 2.4, the selecting combining methods in an ensemble system is a crucial task as it directly affects the classification performance. This experiment finds the optimal methods to fuse information from two feature sets (i.e., f_L and f_P). In this chapter, the sensitivity analysis on two categories for the combining

methods (i.e., combining source and combining rule) is performed. Table 4-3 shows the accuracy assessments of two fusion methods with respect to combining source: feature-level and classifier-level fusion. For the feature-level fusion, Random Forests trained a classifier, C_A from $bTR03_A$ with f_O produced by combing f_L and f_P. Then, C_A was applied to the entire sub sites. While, the classifier-level fusion is a combination of results from two classifiers, C_L and C_P built in the section 4.3 over all the sub sites. The variable used for combining two classifiers is the prediction confidence described in the section 3.2.2, which is an outcome of Random Forests. The respective confidence values from C_L and C_P were combined by the average rule that is typically used in the parallel ensemble system. In Table 4-3, the feature-level fusion (83.82%) is superior to the classifier-level fusion (82.04%) for the class-weighted accuracy. The opposite result (88.99% for the feature-level fusion and 89.92% for the classifier-level fusion) is produced on the sample-weighted accuracy. Thus, the two methods are evaluated that their performances on the given feature sets are approximately equal. However, on the processing time the classifier-level fusion is much better than the feature-level fusion. This is because the feature-level fusion leads to more complicated classification system as the number of the combined features becomes larger, while the classifier-level fusion is more time efficient by building classifiers from small separated feature sets. Consequently, this section select the classifier-level fusion which is able to effectively handle a vast amount of data by splitting the data into smaller subsets, building classifiers using the subsets, and combining the outcomes from the classifiers to derive a final decision. This section also investigated the combining rules (i.e., average, maximum, and multiplication) in order to

find the optimal rule (Table 4-4). It is found that the multiplication rule is the best rule which demonstrates the highest performance in both class-weighted and sample weighted accuracy. Therefore, it is the best to multiply two confidence values corresponding to the class y_i, c_i^{L} from $\mathrm{C_L}$ and c_i^{P} from $\mathrm{C_P}$, and assign a class label (y*) having the maximum value of the multiplication to the instance (Eq. 4-1).

$$\mathrm{y}^* = \arg\max_{\mathrm{y}_i}\left(c_i^{\mathrm{L}} \times c_i^{\mathrm{P}}\right) \tag{4-1}$$

Table 4-3. Classification performance comparison of fusion methods

Fusion level	Feature-level fusion	Classifier-level fusion
Average of class-weighted accuracy (%)	83.82	82.04
Sample-weighted accuracy (%)	88.99	89.92
Processing time (minutes/km)	8.23	**4.70**

Table 4-4. Combination operator per classification accuracies

Operator	Average	Maximum	Multiplication
Average of class-weighted accuracy (%)	81.85	81.41	**82.04**
Sample-weighted accuracy (%)	89.81	89.55	**89.92**

4.4.4. **Classification results**

This chapter built a classification ensemble system composed of $\mathrm{C_L}$ and $\mathrm{C_P}$, which are trained with the balanced $\mathrm{TR03_L}$ and balanced $\mathrm{TR03_P}$. Then this section makes predictions for all the 16 sub sites by applying the Eq. 3-4 to the prediction confidences from the two classifiers. Table 4-5 shows a confusion matrix where omission and commission error rates for each class have been estimated. Figure 4-5 demonstrates a classification map produced by the ensemble system for all the sub sites and the

classification accuracies per class for each site are plotted in Figure 4-6. As presented in the figure, our classification system provides 74.89% to 99.17% in the averages of class-weighted accuracies over all the sites. As well, it presents uniform and high accuracies across all the sites for the wire, building, and vegetation classes, while it shows relatively low accuracies (below 50%) in some sites for the pylon and low object classes. It achieves the highest classification accuracy for the wire class (95.85%), the second highest (94.29%) for the vegetation class, and the third (91.99%) for the building classes. The low object and pylon classes show the classification accuracies of 72.52% and 55.52%, respectively. The large accuracy discrepancy between these two class groups is due to an inherent difficulty in describing the low object and pylon classes as either a linear or planar primitive. Additionally, it is caused by the spatial overlap with other classes which leads for a point segment produced over multiple classes and the intra-class variation which hinders to extract salient characteristics of the class. In practice, the low object (e.g., fences), wire (e.g., distribution wires) and pylon classes touching trees are occasionally misclassified as vegetation. As the low object and pylon classes include various types of objects within the class, they are difficult to be characterized by the suggested features. For such a reason (i.e., spatial overlap), most omission errors of the wire class are observed over distribution lines passing through Type II sites and they have been mostly misclassified as vegetation. Our classification system tends to classify horizontal linear structures as the wire class. Hence, some of pylon parts looking horizontally linear such as cross arms arranged perpendicularly to power lines are misclassified, which are omission errors for the pylon class. Nevertheless, the wire class

demonstrates the lowest omission and commission error rates amongst the targeted classes. Other omission errors for the pylon class are discovered over bodies of steel frame pylons where the extracted line segments have similar patterns as ones from trees, so our classification system assigns the vegetation class to those. As well, most pylons for distribution lines result in omission errors due to not enough line segments extracted over them to recognize them as the pylon class. Mostly, deciduous trees are well classified by our classification system, while coniferous trees are occasionally mislabelled as the pylon class because of their narrow and columnar shape. For the building class, omission errors mainly occurred from the rooftops of commercial buildings in TL08 and TL07, which are not treated by the classifiers due to their absence in the training sample (TR03). The commercial buildings have smaller roof patch and more slanting roof slope compared to residential buildings. Here, a roof patch indicates a polygonal roof part with an identical slope. Such discriminating characteristic makes the errors appeared near intersection regions with adjacent roof patches or with building walls, so those are mislabelled as vegetation. For the low object class, considerable amount of omission errors occurred in TL08 and TL07 where most grasses are misclassified as the building class. The small grasses (less than 30 cm) are widely distributed; hence the plane segments extracted over them have small surface roughness and smoothness like ones from the building class. For the same reason, the misclassification as the building class appears on container box rooftops, car tops, and car bonnets.

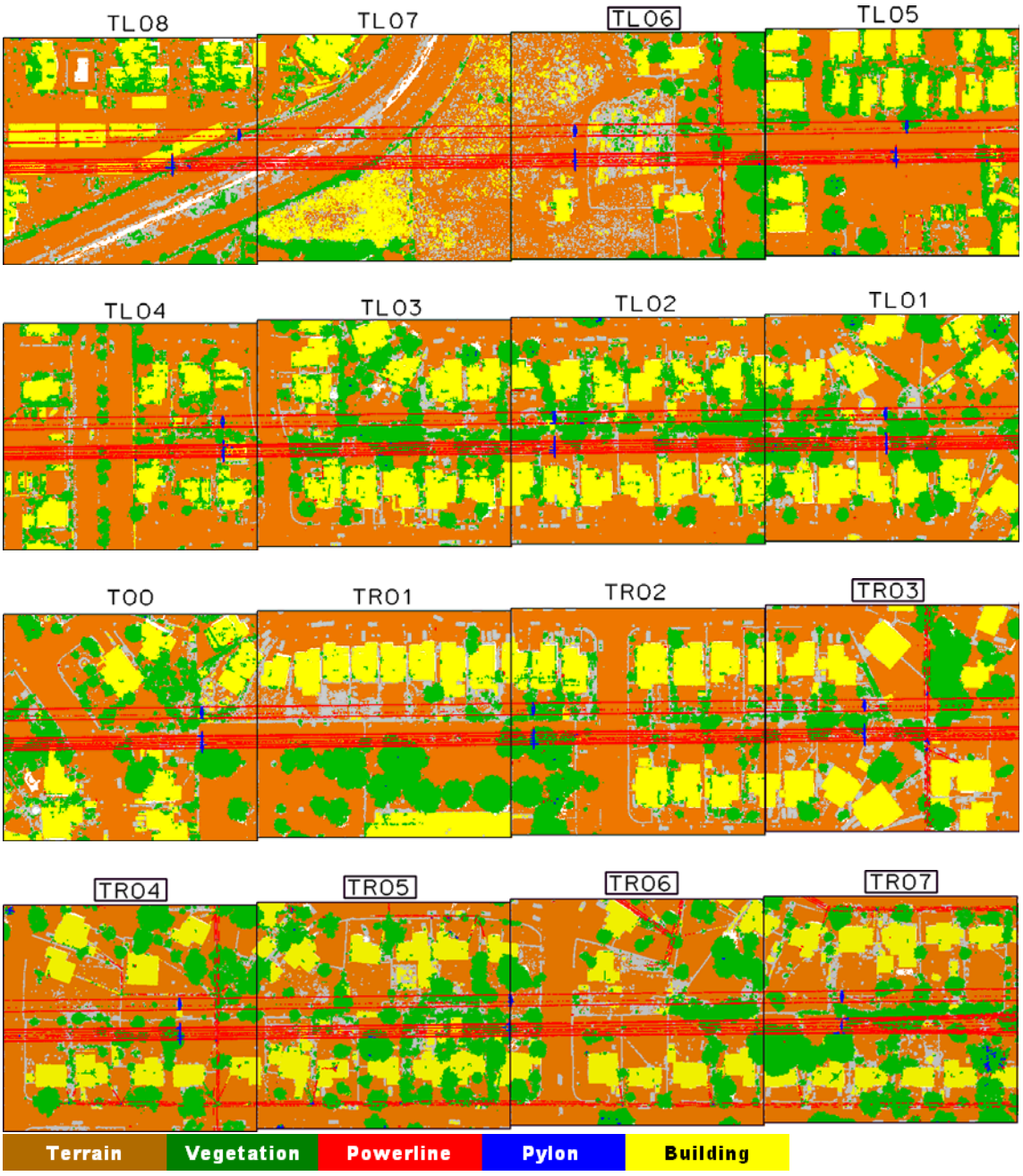


Figure 4-5. Classification map for all sites; vegetation (green), wire (red), pylon (blue), building (building), low object (gray) and ground (remainder)

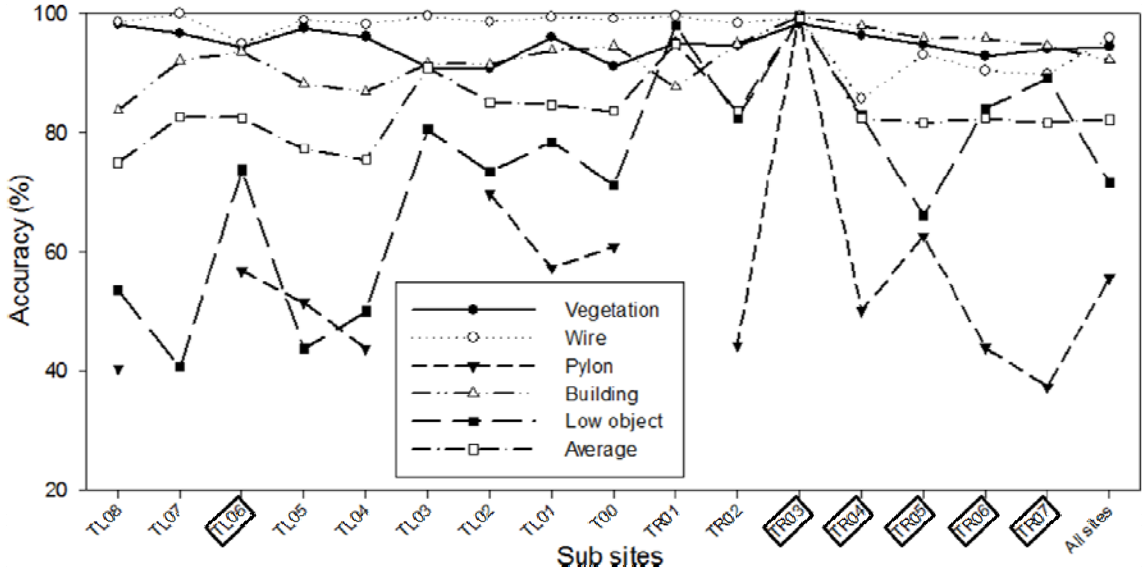


Figure 4-6. Class-weighted accuracies for each site

Table 4-5. Confusion matrix

Class		Predicted					Omission error (%)
		Vegetation	Wire	Pylon	Building	Low object	
Actual	Vegetation	**1,228,831**	489	459	8,133	65,277	5.71
	Wire	3,284	**135,573**	684	804	1,100	4.15
	Pylon	6,202	1,939	**10,643**	70	314	44.48
	Building	55,669	428	14	**995,341**	30,518	8.01
	Low object	73,926	511	47	53,998	**339,092**	27.48
Commission error (%)		10.17	2.42	10.16	5.95	22.28	

4.4.5. **Comparison with classification using point-based features**

As mentioned in the section 2.2.2 and 2.2.3, two approaches are typically used for extracting features for classification from 3D point cloud; the first is the extracting point-based (PBF) and the second is the object-based feature (OBF). To investigate the relevance between the power line scene classification and the two feature sets, this chapter has compared two classification models which are built using the PBF and OBF respectively. As addressed in Kim and Sohn (2013), for each point the PBF has been

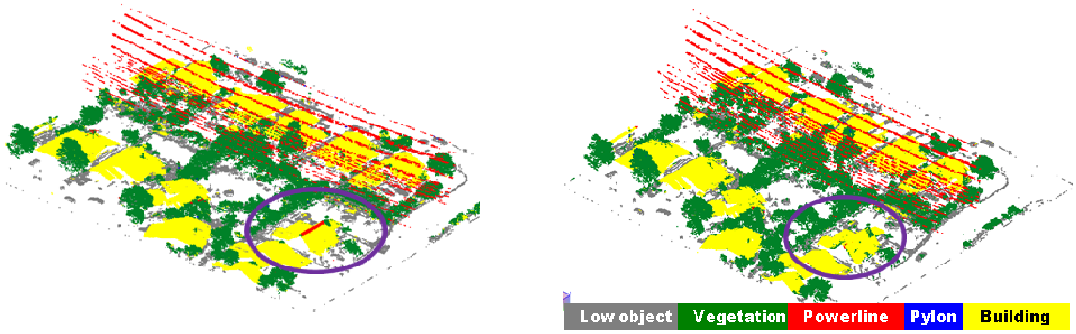
computed using neighboring points captured by either a vertical cylinder or sphere (one of the two is used depending on the feature group) generated from the targeted point. Then all the computed features are assigned to the point. This section constructed a classification model composed of a Random Forests classifier trained from a training sample (i.e., the balanced TR03) with the PBF. This system has been compared with our ensemble classification system suggested in this study by examining the classification performance of the two approaches.

Table 4-6 demonstrates the classification results produced by the two classification model. The PBF model, the column (1) in the table, presents better classification performance in an overall accuracy (i.e., the sample-weighted accuracy) compared to the OBF model, the column (2). Moreover, for the class-weighted accuracies the PBF model results in uniform accuracies with 3.18% standard deviation across all the classes, while the OBF model yields 4.09%, 3.75%, and 1.13% higher accuracies for the vegetation and wire classes than the PBF model. A 0.93% lower accuracy for the building class is shown in the OBF model. For the other classes the PBF model is considerably superior, 29.97% and 16.12% higher for the pylon and low object class respectively, to the OBF model. This comparative analysis indicates that the PBF is useful for all the classes and the OBF works for class objects representable as geometric primitives such as the wire class as line segment and the building class as plane segment. Figure 4-7 shows that the OBF model made a great improvement over a building ridge which is misclassified into the wire class by the PBF model. As the vegetation class is not likely to describe as either line segment or plane segment, it is distinguishable from the wire and building classes.

On the other hand, the OBF is not useful for a class object showing various geometric patterns such as the pylon class showing wire like pattern (e.g., pole type pylons) and vegetation like pattern (e.g., steel frame pylons), and the low object class showing building like pattern (e.g., widespread young or fresh-cut grasses, container rooftops, and car tops), wire like pattern (e.g., fences), and vegetation like pattern (e.g., grown grasses). Such that, if the two systems complement each other such as that the OBF model more focuses on identifying the wire, building, and vegetation classes, while the PBF model concentrates on distinguishing the pylon and low object classes from other three classes, this chapter would achieve an improvement in the classification results. An extended ensemble system is constructed by combining the PBF and OBF models in the same method used to fuse the outcomes of the C_L and C_P in the section 4.3.2.

Table 4-6. Classification accuracy comparison of point-based and object-based approaches

Feature type		Point-based feature (1)	Object-based feature (2)	(1) - (2)
Class-weighted accuracy (%)	Vegetation	90.20	94.29	-4.09
	Wire	93.10	95.85	-3.75
	Pylon	85.49	55.52	**+29.97**
	Building	92.92	91.99	+0.93
	Low object	88.64	72.52	**+16.12**
	Average	90.07	82.04	+8.03
Sample-weighted accuracy (%)		91.04	89.92	+1.12



(a) Point-based method (b) Object-based Method

Figure 4-7. A comparison of classification maps from the point- and object-based method

The computation cost of the object-based method is also estimated to compare it with one of the point-based method. As a result, the object-based method takes 54.18 minutes for 1 km power line corridor from the feature extraction to the classification using Random Forests as seen in Table 4-7. As aforementioned in the accuracy comparison, the point-based method shows better classification quality than the object-based method. However, the object-based method is a winner on the computation efficiency (178.75 minutes faster).

Table 4-7. Computational time of object-based method (minutes/km)

Steps	Feature extraction		Random Forests		Total
	Linear	Planar	Training	Testing	
Processing time	24.55	18.72	4.70	6.21	54.18

4.5. Summary

In this chapter, it is found that the object features including the linear and planar features are an excellent salience to discriminate the wire (linear-like), building (planar-like), and vegetation class (neither linear nor planar). A comparative analysis demonstrates that the classification quality of the object-based method is not better than one of the point-based method introduced in chapter, but the computation cost is much cheaper. On the accuracy per class, the classification accuracy for linear-like structure is 95.85% and the classification accuracy for planar-like structure is 91.99%. For pylon and low object classes an identification difficulty is still present due to their intra-class variations. To overcome the limitation, an ensemble system accommodated various classifiers and is able to distinguish such intra-class variations and have improved overall classification accuracy. Therefore, it is concluded that in order to efficiently classify the

low object and pylon class, there is a need to build the classifiers from different training samples which contain such class objects in various sizes, shapes, and so on. As a prototype of such ensemble system, next chapter successfully builds a MCS where each classifier has been built from a training sample of a specific voltage corridor and has increased the diversity in such situation. The overall classification accuracy have improved from 83.29% to 85.88% due to the inclusion of different physical characteristics of the class objects present in various voltage type transmission infrastructure. Moreover, the object-based method introduced in this chapter is compared with the point-based method from chapter 3 in terms of accuracy and computation time.

5. Classification using Multiple Classifier System

Information fusion stands for combining information extracted from different sources for data processing such as classification and recognition. The sources include sensory data, patterns, features, decisions, knowledge, classifiers and so on. Multiple Classifier System (MCS) is classified as classifier fusion. In the early section, MCS is described as an ensemble system which appeared as different names in different literature, such as classifier fusion (Dybowski et al., 2010; Geurts et al., 2006), classifier ensembles (Hamby and Hirst, 2008; Kuncheva, 2004), combination of multiple classifiers (Kittler et al., 1998; Ho et al., 1994), mixture of experts (Jordan and Jacobs, 1994; Jacobs et al., 1991), composite classifier system (Dasarathy and Sheela, 1979; Skalak, 1997), consensus aggregation (Benediktsson and Swain, 1992), committee of neural networks (Drucker et al., 1994). The above studies have reported that the MCS performs better than single classifier-based methods. This chapter employs the MCS constructed by aggregating the classifiers that have trained earlier. This chapter is organized as follows: section 5.1 describes the fundamentals of MCS and following section introduces our customized MCS. Section 5.3 demonstrates and analyzes the classification results produced by the MCS. Finally, conclusion and summary remarks are addressed in section 5.4. A part of the research work of this chapter (Section 5.2.1) has been addressed in Kim and Sohn (2011).

5.1. Introduction

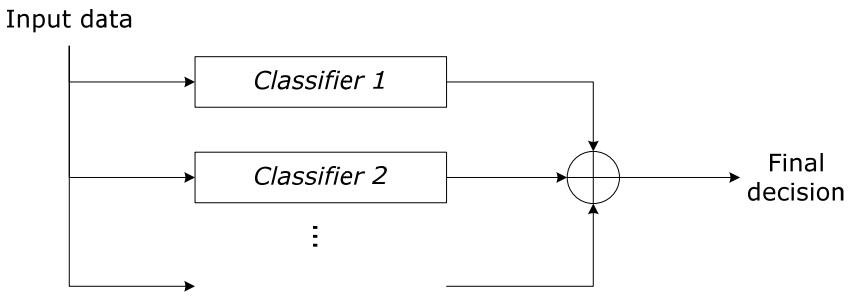
MCS is a classification system composed of a group of classifiers, where each classifier makes an independent decision, and the final decision is combined by the MCS Compared to the single classifier-based method, the MCS is more accurate and reliable with the contributions of the base classifiers making different decisions. The advantage of MCS is to reduce the risk of choosing a poor decision classifier and could be a common mistake with single classifier-based method (Dara, 2007). The fact that the classification accuracy could be increased from multiple base classifiers that makes different decisions is called diversity and diversity between classifiers is one of the critical goal for a successful MCS. Any MCS without any diversity indicate all the classifiers are identical and hence have no advantage in combining these classifiers and will be further discussed in section 5.1.2. The MCS in general follows two basic structures to construct itself: parallel and cascade (section 5.1.1 for more details) where the parallel MCS is a more common architecture. In parallel MSC, all classifiers comprising of the MCS operate in parallel and their predictions are combined to reach a final decision. On the contrary, classifiers of the sequential MCS are applied in sequence, that is, an output of a classifier is used as an input of next classifier. This decreases a problem complexity, but the performance of each classifier extremely depends on that of its previous classifier.

Another subject to be discussed in the MCS is how to efficiently combine diverse information extracted from different sources. For example, a majority vote selects most frequent class label as a prediction over all classifiers is a common combination way in

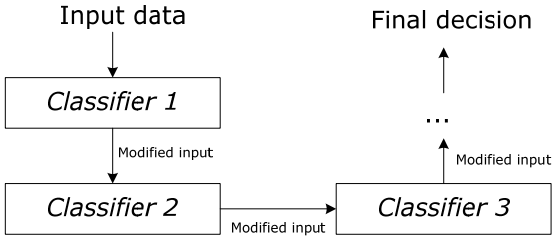
basic ensemble methods such as bagging and Random Forests. Various combining methods are introduced depending on different circumstances of the MCS.

5.1.1. **General MCS scheme**

Two fundamental architectures have been introduced to build a MCS. Most studies using MCS employed a parallel structure in the literature addressed early as illustrated in Figure 5-1(a). Each classifier in the parallel MCS is trained with a subset of the same input data (i.e., the same original sample). Each subset is independent of each other, which means its components are different from those of other subsets, these subsets can be produced by random selection from the original sample. Hence, all classifiers make different predictions for an instance of test data and they can be mutually complementary through combining the predictions. The parallel MCS can employ different combination strategies depending on the type of information produced by the classifiers as addressed in the section 5.1.3. Unlike the parallel MCS, the sequential MCS applied individual classifiers in sequence and each classifier is making its decision from the output of the previous classifier (Figure 5-1(b)). The classifier output could be a modified data set with some kind of ranking over possible classes or pre-predicted class labels (Wozniak et al., 2014; Dara, 2007), finally, the last classifier makes a final decision. This mechanism decreases the overall problem complexity by introducing a classifier at each step and each classifier refines the information in the data set. For an instance, a current classifier reduces candidate class label sets necessary to be treated by a next classifier (Xu et al., 2009). Stacked generalization (Wolpert, 1992) and cascade generalization (Gama and Brazdil, 2000) are representative MCSs following the sequential structure.



(a) Parallel MCS



(b) Sequential MCS

Figure 5-1. General framework of Multiple Classifier System

5.1.2. **Diversity**

The diversity is one of the most important characteristic of a successful MCS, it is one of the reasons why MCS demonstrates better classification compared to the single classifier-based method. MCS with high diversity allows the synthesis of different decisions from the membership classifiers whereas low diversity MCS makes similar decisions from the membership classifiers and therefore final decisions cannot be benefited from the aggregation. In this situation, the MCS works like a single classifier-based method. A key idea to maintain the diversity in the MCS is to increase the diversity between available data sources such as 1) training samples, 2) features, and 3) classifiers. To infuse such important diversity into the MCS, a traditional approach is to have diverse training subsets taken from an original data (Breiman, 1996), where diverse classifiers are trained with the respective subsets. Secondly, diverse feature sets which are extracted

under different circumstances from the original training sample produce diverse classifiers and also increase diversity. Thirdly, different types of supervised classification models such as SVM, decision tree, and others can be applied to the original data. A detailed discussion on diversity generation will be discussed in the next section. Unfortunately, there is no formal proof of a dependency between diversity metric (computed from diversity measuring methods) and accuracy improvement (Wozniak et al., 2014; Ko and Sabourin, 2013) but the diversity could be measured in a few ways.

Diversity in data subsets

Diversity can be increased in data subsets which are obtained through random sampling with a given data set. From the data set including n instances, $\mathbf{D}=\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, ns instances (ns < n) are randomly drawn to create a subset. Repeating this procedure *T* times yields *T* subsets. To have diversity, each subset needs to be different from any of other *T*-1 subsets. In other words, for two arbitrary subsets, $\mathbf{D}_i$ and $\mathbf{D}_j$, which are the i^{th} and j^{th} sampled subsets of **D** respectively, diversity between the subsets can be achieved on following occasion as seen in Eq. 5-1 (Ko and Sabourin, 2013):

$$\begin{aligned} &\forall_{i,j}, 1 \le i, j \le T, \mathbf{D}_i, \mathbf{D}_j \subset \mathbf{D} \\ &\forall_{i,j}, i \ne j, \mathbf{D}_i, \not\subseteq \mathbf{D}_j, \mathbf{D}_j, \not\subseteq \mathbf{D}_i \end{aligned} \tag{5-1}$$

Any supervised classification model can be applied to a data subset, $\mathbf{D}_i$ to train a classifier, C_i. However, typically the same kind of model is used over subsets which are taken samples from a data set. Using different types of model leads to another type of

diversity which will be discussed in the section "Diversity in classification models"; an abused mixture of diversity would cause unexpected distraction from decision making. Thus, T diverse classifiers are finally generated using the T subsets taken from the original data set, **D**.

Bagging (Breiman, 1996) is one of the most conventional ensemble learning method, it maintains the diversity by randomly subsampling a given data. K-fold cross validation can also increase diversity by splitting an original data set into K equal size subsets, then generates K models through the K times cross-validation. In each validation, a subset is used for testing a model trained with the other K-1 subsets. In K-fold cross validation a certain instance belonging to a subset cannot exist in other subsets, which is different from the Bagging. For classification problem, Bagging makes a final prediction by using the majority vote over the trained models, while K-fold cross-validation chooses the best model to classify given test data. Such combining strategies are addressed in more detail in the section 5.1.3.

Diversity in feature sets

Another way to incorporate and increase diversity in a MCS is to use diverse feature sets that are extracted from different data sources or being produced by applying different feature extraction methods to one data source. For example, multiple feature sets investigated from a hyper spectral image, each of which comes from a specific band of the image (Benediktsson et al., 2005; Jimenez et al., 2005; Foody and Mathur, 2004). Typically there are two ways to handle T multiple feature sets ($f_1, f_2, \ldots, f_T$) in supervised

classification approaches. One is to combine the multiple feature sets into a feature group (*f*), then, a classification model (C_0) is built using the new feature group as shown in the Eq. 5-2. This method is traditionally used to produce land-cover (LC) or land-use (LU) classification maps from optical images (Na et al, 2010; Waske and Braun, 2009). Considering all the features as a group makes a feature treatment straightforward. On the other hand, a generated model would be complex due to overtraining too many undeserved features, e.g., a decision tree with high depth. Consequently, a strict model tends to be generated. Therefore, a feature refining step to reduce down the feature dimension would be essential to prevent the model from overtraining.

$$\begin{aligned} f &= f_1 \cup f_2 \cup \cdots \cup f_T \\ \mathrm{C}_0 &= \mathrm{train}(f) \end{aligned} \tag{5-2}$$

The second way is the training of multiple classification models (C_1, C_2, …, C_T) from corresponding feature sets, and the outcome of the models (C) are combined for a final classification as done by the MCS (Eq. 5-3). Unlike the previously described method produces a strict model, this one leads to weak models (or weak learners), each of which yields different and incomplete classification results. According to many studies (Kittler et al., 1998; Dietterich, 2000; Polikar, 2006; Dara, 2007), fusing such weak models produce better classification results compared to a single strict model.

$$\begin{aligned} \mathrm{C}_i &= \mathrm{train}(f_i), 1 \le i \le T \\ \mathrm{C} &= \mathrm{C}_1 \cup \mathrm{C}_2 \cup \cdots \cup \mathrm{C}_T \end{aligned} \tag{5-3}$$

As aforementioned in chapter 3 (3.3), feature refining is required to handle large numbers of features because having more features than needed leads to model over-fitting. Feature selection, which selects an optimal feature subset amongst a given features, and feature de-correlation, which remove correlations between the features, are representative ways to reduce the number of features. For examples, Random Forests is able to compute feature importance over all the features that can be further analyzed to serve this purpose. Based on the importance, irrelevant features are eliminated and the remaining features are used to build a classification model. PCA can be applied to calculate a metric that describe the relationships between feature pairs. A given feature set is then converted into a new feature set through a linear projection using the metric calculated and this feature refining can be employed for the MCS depending on its necessity.

Diversity in classification models

Applying various types of classification models to a given data set also encourages diversity. This is because different classification models potentially exhibits different biases, this principle produces different classification hypothesis (Wolpert, 2001). Any classification algorithms or models can be employed such as SVM, MLC (Maximum Likelihood Classifier), Random Forests, or others. For example, SVM and MLC were applied to high resolution images to detect buildings (Erener, 2013). Moreover, classifiers produced by applying the same classification model also promote diversity if the base classifiers are built using different model parameters. According to Samadzadegan et al.

(2010), combining one-against-one SVM and one-against-all SVM yielded better classification accuracies on ALS data than using a single model.

5.1.3. **Combining methods in MCS**

Chapter 4 has already addressed on combining methods at three levels: data-level, feature-level, and classifier-level. Apart from some of the common ways for information fusion, this section describes additional combining strategies that are commonly used in the MCS. Denote that this chapter introduces only basic fusion methods which are practically used in this research. For an instance in a sample to be classified, **x** is a feature vector of the instance. Let a committee of classifiers in the MCS C={C_1, C_2, …, C_i, …, C_T}. A prediction confidence value on a class (y_j) for the instance, $c_{ij}(\mathbf{x})$ can be estimated by a classifier C_i:

$$c_{ij} = P(y_j|\mathbf{x}, C_i) \tag{5-4}$$

If Random Forests is taken as a classification model, c_{ij} is calculated from classification results provided by the classifier C_i using the Eq. 5-4.

Majority voting

Majority voting is the most common method to combine classifiers in the MCS. The prediction confidence values are not used, instead, the most frequently appearing class label is assigned over all the classifiers. If each classifier has a different reliability on its performance, *weighted majority voting* can be employed by applying different weights to votes made by the individual classifiers. The final decision is made by selecting a class

label corresponding to the highest vote, for example, Random Forests classifies with majority voting scheme.

Product rule

In *product rule*, a prediction confidence (c_j) on a class y_j is represented as a product of the confidence values (c_{ij}) of the classifiers. This method would be useful when membership classifiers of the MCS are independently built (Kittler et al., 1998). If the confidence value of a certain classifier equals 0, the overall prediction confidence becomes 0. To prevent such case, all the confidence values should be larger than 0 by adding a small decimal, Δ. The final decision is made by selecting a class label with the highest prediction confidence.

$$c_j = \prod_i c_{ij} \tag{5-5}$$

Sum rule

Sum rule (Eq. 5-6) adds up the confidence values over all the classifiers for a class y_j. This rule could be used for improving classification using similar classifiers that are mutually independent (Kittler et al., 1998). *Sum rule* is the same as *average rule* when the prediction confidence c_j is divided by the number of classifiers. *Weighted sum rule* (Eq. 5-7), where different weights (w_i) are assigned to the confidence values of the individual classifiers (C_i), is a more general form. Assigning an identical weight leads for *sum rule*. Similarly, the final decision is made by selecting a class label with the highest prediction confidence.

$$c_j = \sum_i c_{ij} \tag{5-6}$$

$$c_j = \sum_i w_i \times c_{ij} \tag{5-7}$$

Maximum and minimum rule

Unlike all the confidence values across classifiers that affects the final decision in *produce rule* and *sum rule*, *maximum rule* (Eq. 5-8) and *minimum rule* (Eq. 5-9) select the best and the worst classifier respectively for each class. Then the confidence value of the selected classifier is regarded as the prediction confidence per class. The final decision is made the same way as the early introduced rules.

$$c_j = \max_i \{ c_{ij} \} \tag{5-8}$$

$$c_j = \min_i \{ c_{ij} \} \tag{5-9}$$

5.2. MCS development

Several prototypes of the MCS have been investigated to achieve a better classification quality against single classifier based methods. These prototypes includes Feature based MCS, Scene based MCS, and Extended MCS which is a combination of Feature based MSC and Scene based MSC. The following sections will discuss these MCSs in more details.

5.2.1. **Feature based MCS**

Feature based MCS (FMCS) is associated with the "diversity in feature sets" described in the section 5.1.2. It accommodates classifiers built from two different feature sets, each of which has been extracted from a single training sample but under different circumstances: point-based (PBF) and object-based feature set (OBF) introduced in the chapter 3 and 4 respectively. As discussed before (Chapter 4), classification accuracy improvement by fusing linear and planar feature set belonging to OBF, combining PBF and OBF can further enhance performance. Figure 5-2 illustrates a structure of the FMCS with the given training and testing data. Point-based, linear-based, and planar-based features are extracted on the training data. The feature extraction method was previously described, for point-based feature in chapter 3 and for linear-based and planar-based feature in chapter 4. Then, classifiers are built from the respective features using any classification model (Random Forests in this study). For an instance in the testing data each classifier estimates a confidence value per class, based on the confidence values, a final decision of the instance is made by one of the combining methods addressed in the previous section (section 5.1.3). This study employs *sum rule*, which anticipates better performance by using similar classification model, as Random Forests has been used over the feature sets.

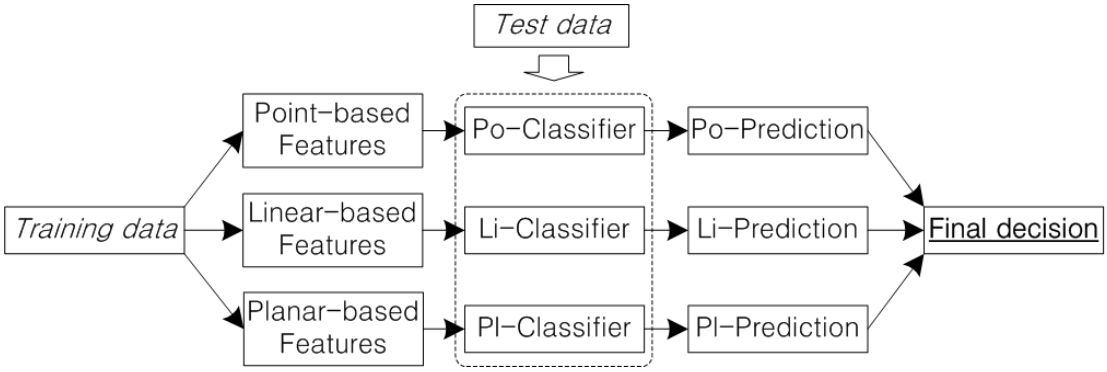


Figure 5-2. Feature-based MCS fusing classifiers built with point, linear, and planar features.

As discussed in chapter 3 and 4, the classifier (Po-Classifier) trained with point-based features is successful over all the classes, while the classifiers (Li-Classifier and Pl-Classifier) for linear-based and planar-based are successful for highlighting linear objects (i.e., the wire class) and planar objects (i.e., the building class) respectively. Thus, it is expected that the classification quality can be improved with the contributions to better identifying the wire class using the Li-Classifier, the building class using the Pl-Classifier, and the remaining classes using the Po-Classifier.

5.2.2. **Scene based MCS**

Scene based MCS (SMCS) increases the "diversity in data sets" by building diverse classifiers from different training samples with different voltage types. A basic concept of the SMCS is invented by an idea that electric structures (i.e., mostly transmission line and electric pylon) in a power line corridor have different physical characteristics, such as shape, size, material, and so on depending on carrying voltage type of the corridor. Practically, bundled wires and huge pylons are typically constructed for over 500kV corridors, while single wires and small pylons for less 69kV corridors according to the

design and construction standards of electric utilities. Such that, assuming similar scene characteristics live in similar voltage type of corridors is reasonable, this research categorize power line corridors into *Low voltage* (<69kV), *Medium voltage* (>69kV and <350kV), and *High voltage* (>350kV). A training data is taken from each type of corridors and same feature extraction method is applied to the individual training data. Then, classifiers are built using the extracted features (L-Classifier working better on low voltage, M-Classifier on medium voltage, and H-Classifier on high voltage) as seen in Figure 5-3. Again, a final classification on a given testing data is made by combining confidence values from each classifier. To combine prediction, this research employ *weighted sum rule* (Eq. 5-7) depending on the voltage type of the testing data. A higher weight is given for a classifier with similar voltage type to the testing data, lower weights for other classifiers. These weights can be estimated through learning validation sets partially taken from corresponding training data and can be empirically calibrated.

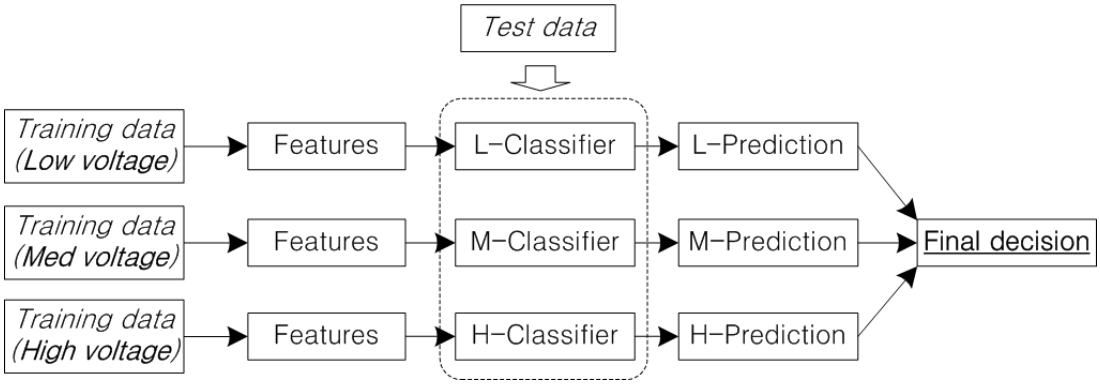


Figure 5-3. Scene-based MCS fusing classifiers trained from low, medium, and high voltage type sample.

From the SMCS, it is expected to improve classification accuracy for the wire and pylon class having intra class variations according to the voltage type. However, it would

not work for the other classes (the building, vegetation and low object classes) which are present regardless of the voltage type.

5.2.3. **Extended MCS**

Extended MCS (EMCS) is designed by fusing the FMCS and SMCS addressed in the previous sections (Section 5.2.1 and 5.2.2) for simultaneously improving the classification accuracies and solving the intra-class variations. As seen in Figure 5-4, the FMCS is applied to three training data with different voltage types (low, medium, and high voltage type). L-FMCS (low), M-FMCS (medium), and H-FMCS (high voltage FMCS) are generated by training the respective low, medium and high voltage training data as described in the section 5.2.2. As intermediate results, each FMCS makes a local decision on a given testing data. Then, a final decision is made by combining the *weighted sum rule* from the local decisions.

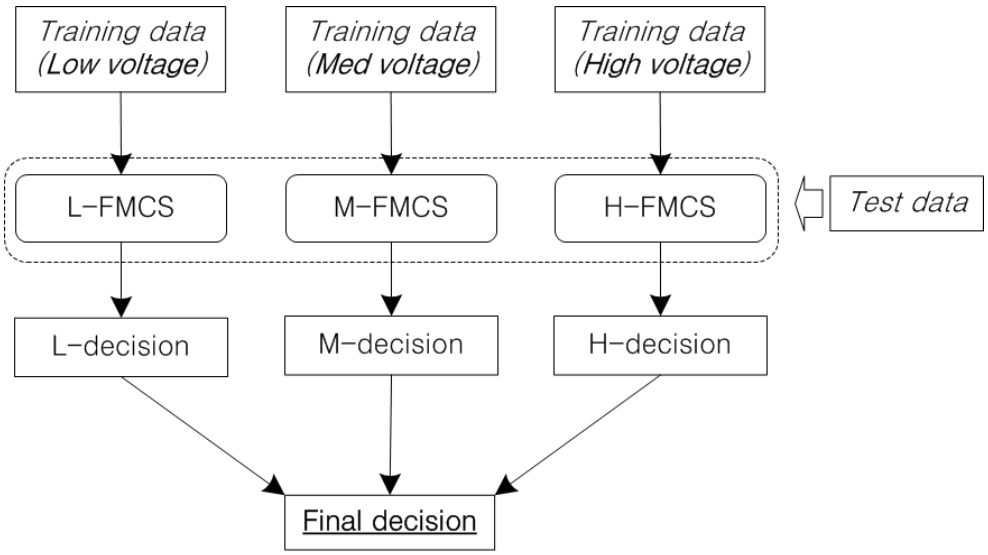


Figure 5-4. Extended MCS fusing FMCS and SMCS

5.3. MCS based Classification

5.3.1. **Experiment Setup**

This experiment uses multiple training samples taken from different voltage types of power line corridors: 69kV, 115kV, 230kV, and 500kV subset as shown in Figure 5-5. They are also independent from the regions (TL08 to TR07 located in Sacramento, California) from the test site and termed TR69, TR115, TR230, and TR500 respectively. The scene characteristics are described as follow:

69kV subset

The TR69 is a residential area where distribution lines and small pole-type pylons are constructed beside in-line houses. Buildings are detached from each other but some of them are contacted by trees at their roof edges. Trees are relatively older and larger in size compared to other samples and most of them are deciduous. Low vegetation, fence and vehicle are partially seen over the site, this site carries a different site characteristic from other samples where power line cables are in contact with trees.

115kV subset

Deciduous and coniferous trees are mixed in the TR115, but deciduous trees are predominant. It is observed that some trees touch the houses and grasses are present over the entire site, while fences and vehicles are rarely visible. Steel frame pylons support single (non-bundled) transmission lines and transmission lines and pylons are well separated from trees.

230kV subset

In this site, two pylon lines (one pylon line indicate a set of transmission lines and a pylon which are physically attached) pass in parallel. The individual transmission lines are a single (non-bundled) cable and pylons are pole type (or tubal type). Not many trees are present in this site and most of them are smaller deciduous trees compared to ones in the TR69 and TR115. Building wall is visible as well as fence and vehicle, while grass is not visible. There is no any touch of trees with the transmission lines.

500kV subset

This site contains only transmission lines, a pylon, grasses, and trees excluding buildings, fences, and vehicles. A big steel frame pylon stands with bundled transmission lines (a bundle of two cables). Trees in the TR500 among all the training samples are the smallest and the transmission lines are clearly separated with trees.

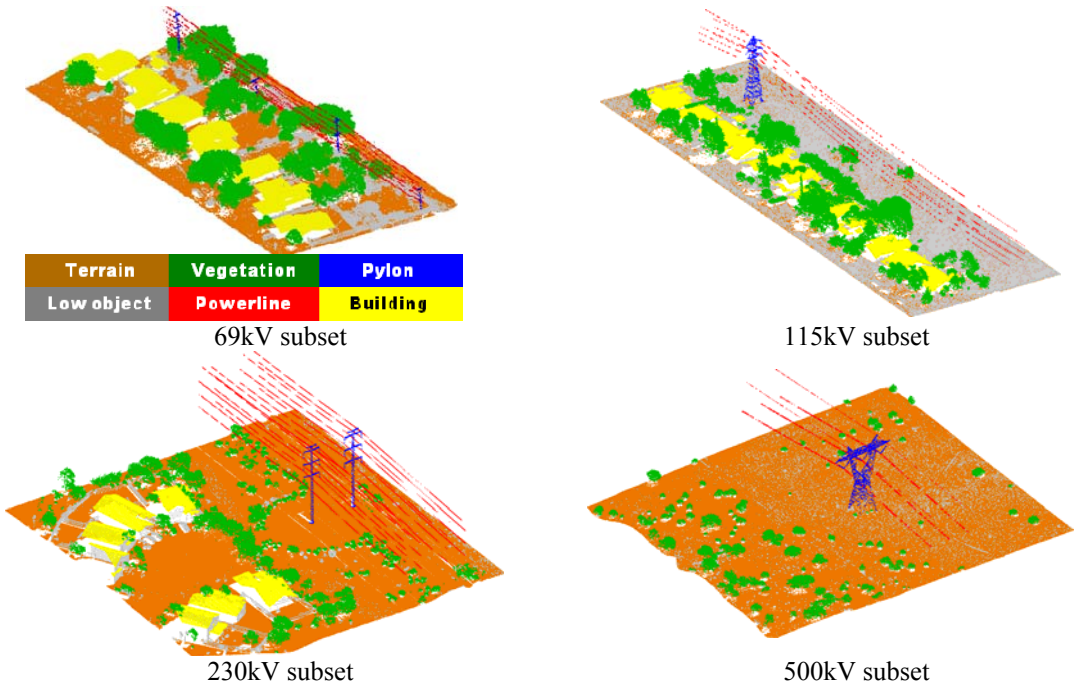


Figure 5-5. Training samples taken from 69kV, 115kV, 230kV, and 500kV corridor.

Table 5-1 shows a comparison of class-dependent characteristics over all the training sites and the test site. The bold letters in the cells represents the characteristic similarity and the symbol indicates a frequency of the corresponding class object, i.e., X (not present), Δ (<25%), Δ Δ (<50%), and O (≥50%).

Table 5-1. Class-relevant characteristics for each training site

Class		69kV site	115kV site	230kV site	500kV site	Test site
vegetation	coniferous	X	**Δ**	X	X	**Δ**
	deciduous	**O**	**O**	**O**	**O**	**O**
	height (ft)	**3 to 62**	10 to 60	3 to 25	3 to 15	**3 to 50**
	crown (ft)	**3 to 30**	**3 to 30**	1 to 12	3 to 15	**3 to 65**
wire	single	**O**	**O**	**O**	X	**O**
	multiple	X	X	X	O	O
	height (ft)	15 to 56	**30 to 100**	**35 to 100**	50 to 115	**20 to 130**
pylon	tubal	**O**	X	**O**	X	**O**
	lattice	X	**O**	X	**O**	**O**
	touch tree	**Δ**	X	X	X	**Δ**
	height (ft)	35 to 56	**100**	**100**	**115**	**85 to 130**
building	residential	**O**	**O**	**O**	X	**O**
	commercial	X	X	X	X	Δ
	sunshield	X	X	X	X	Δ
	chimney	**Δ**	**Δ**	X	X	**Δ**
	TV antenna	**Δ**	**Δ**	X	X	**Δ**
	touch tree	**Δ**	**Δ**	X	X	**Δ**
	height (ft)	10	**15**	15 to 25	-	**15**
low object	grass	X	**Δ**	**Δ**	**Δ**	**Δ**
	fence	**O**	**O**	Δ	X	**O**
	vehicle	**Δ**	**Δ**	**Δ**	X	**Δ**
	container box	X	X	X	X	Δ

The experiments were completed in the previous chapters (Chapter 3 and 4), where Random Forests is employed as a classification model to produce a classifier with given feature set. The parameters on Random Forests are set to be the same as before (*T* is 60

and F is dependent on given feature number). This chapter also perform the feature optimization and balanced learning for each training sample.

5.3.2. **Classification Results of FMCS**

In this study, two kinds of feature sets are extracted from the point cloud under different circumstances that have introduced to distinguish the targeted class objects: point-based and object-based feature. The object-based features are divided into the linear and planar. The section 4.4.3 has addressed the classifier-level fusion is more efficient in terms of the accuracy and the computational complexity compared to the feature-level one. This experiment shows an accuracy improvement by fusing the classification outputs from the classifiers, each of which is built using each feature set. For this experiment, TR03 was used as a training sample and TL08 to TR07 were used as test samples.

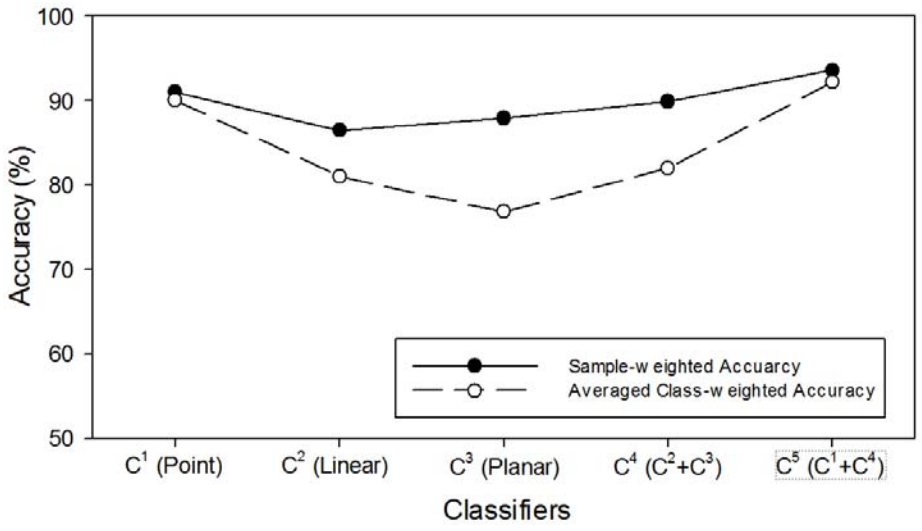


Figure 5-6. Classification accuracy comparison of each classifier

In Figure 5-6, the C^1 is a classifier trained using the PBF. The C^2 and C^3 are built from the linear and the planar features respectively, termed the OBF. The C^4 stands for a fusion of the C^2 and C^3. Finally, the C^5 is combining C^1 and C^4. The C^5 has the highest accuracy in both sample-weighted and averaged class-weighted accuracy, 2.58% and 2.17% higher in the sample-weighted and the averaged class-weighted accuracy. It is believed that a classification uncertainty has been decreased thanks to cooperation between each classifier.

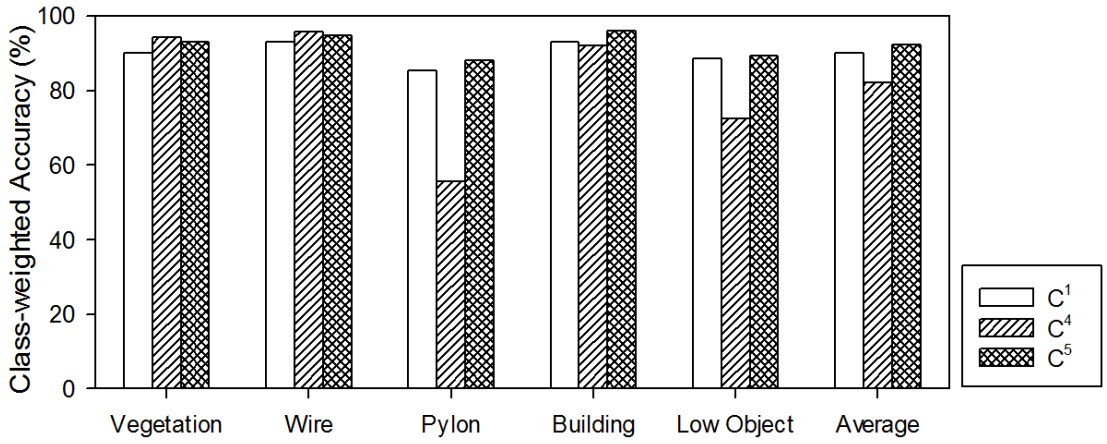


Figure 5-7. Accuracy per class of three best classifiers.

Figure 5-7 shows class-weighted accuracies resulted in by each classifier, C^1, C^4, and C^5. Compared to the C^1 and C^4, the C^5 produced the highest accuracies over all the classes except the vegetation and wire classes which had the highest accuracies in the C^4. C^1 and C^4 cooperated to reach to an agreement in terms of assigning the same class label to each instance, but for some instances confusions occurred by disagreements between the C^1 and C^4 led to the misclassification. As a result, the C^5 yielded 1.15% and 1.12%

lower accuracy for the vegetation and wire class than the C^4. Nevertheless, the C^5 recorded the highest accuracies for the other classes, 2.53% for the pylon, 3.11% for the building, and 0.63% higher for the low object class than the C^1 which showed the second highest accuracies for the three classes after the C^5.

This experiment explored the confusion matrices over the C^1, C^4, and C^5 to study a contribution of the MCS to the classification accuracy improvement. Given an element m_{ij}^k in a confusion matrix M^k, k={C^1, C^4, C^5} and *i*,*j*={vegetation, wire, pylon, building, low object}, an element Δm_{ij} of a subtracting matrix ΔM is computed as follow:

$$\begin{aligned} \Delta m_{ij} &= m_{ij}^{C5} - \min\left(m_{ij}^{C1}, m_{ij}^{C4}\right) \text{ for i=j} \\ \Delta m_{ij} &= m_{ij}^{C5} - \max\left(m_{ij}^{C1}, m_{ij}^{C4}\right) \text{ for i}\neq\text{j} \end{aligned} \tag{5-10}$$

By subtracting the matrix it shows the number of instances that are misclassified by either the C^1 or C^4 and the number of correctly classified by the C^5 in Table 5-2. As discussed for confusion matrix comparison in section 3.5.5 the positive value of diagonal elements indicates that the C^5 reassigned correct class labels to more instances and the negative value of off-diagonal elements indicates corresponding number of misclassified instances in the previous step that were corrected by the C^5. For examples, 7,103 out of 9,204 vegetation points which were misclassified into the wire class by the C^1 have been correctly classified as the vegetation class due to the contribution of C^4 (see the cell with *i*=vegetation and *j*=wire in Table 5-2). As a contribution of the C^1, for 73,926 low object points mislabelled into the vegetation class by the C^4, the C^1 helped to reassign the low

object class label to 44,188 points (see the cell with *i*=low object and *j*=vegetation in Table 5-2). Thus, combining the C^1 and C^4 (i.e., the C^5) led to a classification improvement over all the classes.

Table 5-2. Subtracted confusion matrix of C^5 to (C^1 and C^4)

Class		Predicted				
		Vegetation	Wire	Pylon	Building	Low object
Actual	Vegetation	**38,399**	-7,103	-296	-8,869	-22,131
	Wire	-1,655	**2,307**	-811	170	-739
	Pylon	-5,156	-791	**6,229**	-69	-213
	Building	-28,290	-2,531	-13	**43,675**	-21,700
	Low object	-44,188	-370	-4	-33,745	**78,304**

Table 5-3 shows a confusion matrix of the classification result performed by the C_5 over all the test sites (TL08 to TR07). The C^5 resulted in 93.62% sample-weighted and 92.24% class-weighted accuracy, which are higher than those of the C^1 (91.04%, 90.07%) and C^4 (89.92%, 82.04%). However, it is observed that some classification errors caused by confusion between class objects. As addressed in the early sections (section 3.5.4 and 4.4.4) and shown in Table 5-3, most confusion occurred when two different class objects locate adjacent to each together such as shrubs living under trees, power line cables intersecting with pylons, buildings surrounded by trees, and so on. In addition, other confusion happened from a class object showing complex spatial features, i.e., the low object class containing grass (vegetation-like), container box (building-like), fence (partially wire-like) and vehicle (partially building-like). Finally, insufficient points to extract salient features confused the classifier such as regions of tree tops and power lines with a low point density.

Table 5-3. Confusion matrix of C^5

Class		Predicted					Omission error (%)
		Vegetation	Wire	Pylon	Building	Low object	
Actual	Vegetation	**1,213,848**	2,101	1,875	8,454	76,911	6.86
	Wire	5,499	**133,989**	601	995	361	5.27
	Pylon	1,046	1,148	**16,872**	1	101	11.98
	Building	27,379	63	58	**1,039,016**	15,454	3.97
	Low object	29,738	141	46	20,253	**417,396**	10.73
Commission error (%)		4.98	2.51	13.26	2.78	18.19	

5.3.3. **Classification Results of SMCS**

The previous experiment verified that combining multiple classifiers built using various sets of features extracted from same training sample contribute to improving classification quality. This experiment suggests a classifier fusion to overcome the intra-class variations by fusing classifiers trained with training samples with different scene characteristics: TR69, TR115, TR230, and TR500 sample. As seen in the metrics from Table 5-1, each training sample presents disparate characteristics on a shape and size for a class object and is taken in a different corridor independently from the test sites (TL08 to TR07). To explore an effectiveness of the scene based combination, this experiment excluded the feature based combination, hence it is only using the point feature set, that is, all the classifiers were made from TR69, TR115, TR230 and TR500 with point features: C^1_{TR69}, C^1_{TR115}, C^1_{TR230} and C^1_{TR500}. As notified in the early section, a C^1 is a classifier built with the point feature set. Then, it is applied to all the test sites (TL08 to TR07).

Table 5-4 summarizes the class-weighted and sample-weighted accuracies for all the single classifiers and the combined classifiers. “All” in the table presents a combination across all the classifiers, i.e., $w_{TR69} = w_{TR115} = w_{TR230} = w_{TR500} = 1$ in the Eq. 5-7,

while "$C^1_{TR115+TR230}$" indicates combining the C^1_{TR115} and C^1_{TR230}, i.e., $w_{TR69} = w_{TR500} = 0$ and $w_{TR115} = w_{TR230} = 1$. For the $C^1_{TR115+TR230}$, it is combined based on prior information on the test sites where 115kV and 230kV transmission lines run together. The spatial characteristics of an object in a corridor scene are associated with its voltage type are regarded. For an instance, it is expected that the transmission lines from two power line corridors have similar geometric features if the voltage type of the corridors is same. As seen in Table 5-4, the best classifier is the C^1_{TR69} for the vegetation and wire class, the C^1_{TR500} for the pylon class, and the C^1_{TR115} for the building and low object class. The combined classifiers (i.e., "All" and "$C^1_{TR115+TR230}$") do not seem to be the best for any specific class, but they show somewhat good accuracies over all the classes. As it is expected, the $C^1_{TR115+TR230}$ demonstrates the best performance in both a sample-weighted and an average of class-weighted accuracies which is a requirement for a good classifier. For the vegetation class, the C^1_{TR69} and C^1_{TR115}, whose training samples (TR69 and TR115) mainly contain tall and widely branched deciduous trees similarly to the test sites, therefore, resulted a higher classification accuracies compared to the other classifiers whose training samples have mostly small trees as seen in Figure 5-5. On the other hand, the wire class showed mostly equable accuracies over the classifiers. The C^1_{TR69} failed to identify the pylon class because it can detect only small pylons which is not similar to the ones found in the test sites, while the others work well. Especially, the C^1_{TR500} is the best for the pylon class because it is corresponding to the training sample (TR500) includes a pylon showing similar metric in the size to ones in the test sites. The building class was better classified by the C^1_{TR69} and C^1_{TR115} compared to the C^1_{TR230}.

This is because some of buildings in the TR69 and TR115 touched the surrounding trees like ones found in the test sites, unlike the test sites, buildings in the TR230 are well separated from tree. Additionally, the C^1_{TR500} is not able to identify the building class due to no building point in the TR500. For the low object class, the C^1_{TR69} and C^1_{TR115} demonstrated an impressive performance compared to the others. As addressed in the section 3.5.3, like the test sites, the TR69 and TR115 have more various objects (i.e., grass, fence, and car) than the other two samples having only grass.

Table 5-4. Classification accuracies of single classifiers and feature fused classifiers

Classifier		C^1_{TR69}	C^1_{TR115}	C^1_{TR230}	C^1_{TR500}	All	$C^1_{TR115+TR230}$
Class-weighted accuracy (%)	Vegetation	**88.23**	84.59	58.35	37.27	87.10	86.58
	Wire	**71.63**	69.21	65.08	51.41	73.44	68.14
	Pylon	0.00	79.34	88.02	**94.27**	81.89	84.19
	Building	86.17	**92.34**	80.54	0.00	88.09	91.71
	Low object	88.11	**88.50**	51.63	50.68	85.94	87.42
	Average	86.13	87.22	65.78	27.00	86.60	**87.67**
Sample-weighted accuracy (%)		66.83	82.80	68.72	46.73	83.29	**83.61**

Table 5-5 gives a confusion matrix on the classification result produced by the $C^1_{TR115+TR230}$. It yielded 87.67% class-weighted and 83.61% sample-weighed accuracy which is relatively lower compared to classifiers trained with a training subset taken from the test sites. As aforementioned, it is observed that confusion mostly happened between two class objects touching each other. Especially, this classifier made considerably many commission errors on the pylon class. This is because of overtraining the pylon class which caused misclassifying trees and transmission lines when they are vertically overlapped.

Table 5-5. Confusion matrix of $C^1_{TR115+TR230}$

Class		Predicted					Omission error (%)
		Vegetation	Wire	Pylon	Building	Low object	
Actual	Vegetation	**1,128,249**	4,987	55,542	20,180	94,231	13.42
	Wire	13,602	**96,380**	28,170	3,120	173	31.86
	Pylon	886	2,026	**16,137**	53	66	15.81
	Building	61,843	2	445	**992,228**	27,452	8.29
	Low object	42,495	80	1,135	15,121	**408,743**	12.58
Commission error (%)		9.53	6.86	84.09	3.73	22.98	

5.3.4. **Classification Results of EMCS**

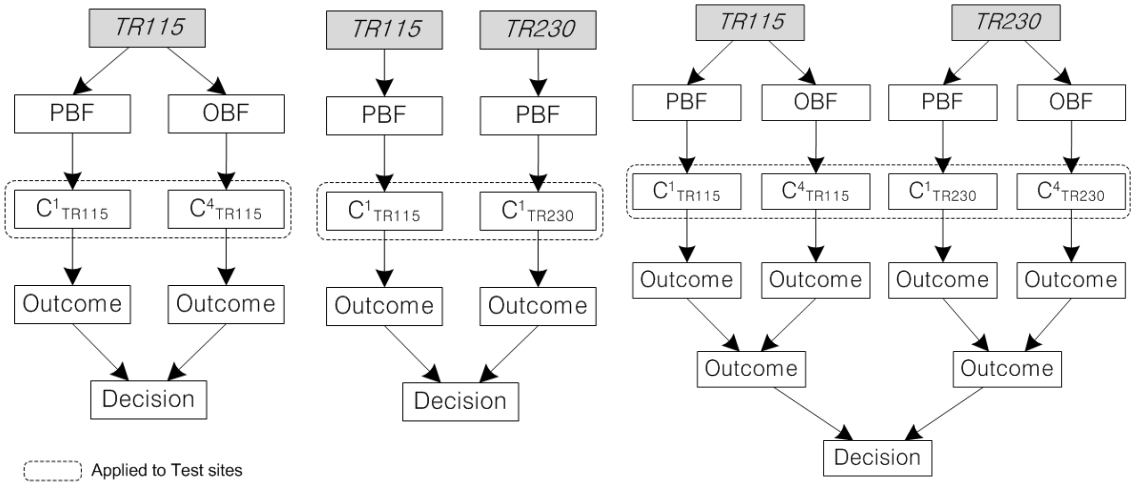
This section calculated the accuracy assessment over all the FMCSs produced from the training sample (the TR69, TR115, TR230, and TR500), which are the C^5_{TR69}, C^5_{TR115}, C^5_{TR230}, and C^5_{TR500}. This experiment is done in order to explore which classifier or which classifier combination is the best in using the PBF and OBF. As the same results showed in the section 5.3.3, a classifier fusion ($C^5_{TR115+TR230}$) is the highest in both sample-weighted and averaged class-weighted accuracy (see table 5-6). Hence, this experiment has selected TR115 and TR230 as training samples to classify the test sites.

Table 5-6. Classification accuracies of single classifiers and scene fused classifiers

Classifier		C^5_{TR69}	C^5_{TR115}	C^5_{TR230}	C^5_{TR500}	$C^5_{TR115+TR230}$
Class-weighted accuracy (%)	Vegetation	89.74	85.43	75.84	72.88	**90.42**
	Wire	**79.88**	71.13	70.65	59.63	73.32
	Pylon	0.00	77.78	89.07	**94.98**	84.80
	Building	88.14	**92.59**	84.05	0.00	91.40
	Low object	89.22	**90.13**	52.96	55.51	89.46
	Average	69.40	83.29	74.51	50.60	**85.88**
Sample-weighted accuracy (%)		88.05	88.01	75.08	30.56	**89.79**

Then, a prototype of an EMCS by fusing the FMCS and SMCS for a classification performance improvement based on the selected training samples is built. The performance of each MCS is evaluated. For the FMCS, this experiment fused two

classifiers trained with the PBF and OBF from the TR115 which shows the highest classification accuracy (Figure 5-8(a)). For the SMCS, two classifiers trained with the PBFs from the respective TR115 and TR230 (Figure 5-8(b)) are combined. The EMCS combined the FMCSs made from the respective TR115 and TR230 (Figure 5-8(c)).



(a) FMCS (b) SMCS (c) EMCS

Figure 5-8. Classification work flow of each MCS (FMCS, SMCS, and EMCS)

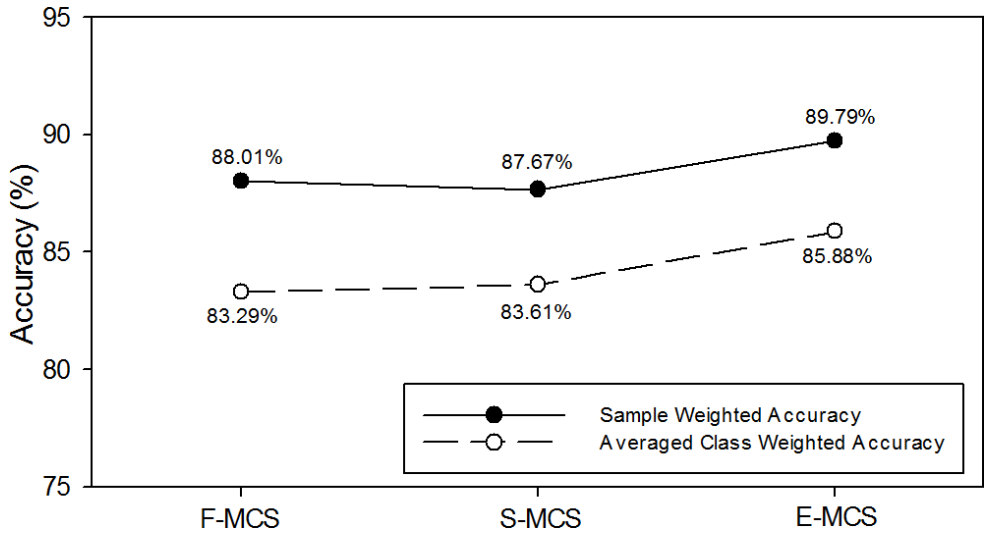


Figure 5-9. Accuracy comparison of each MCS

Figure 5-9 depicts the sampled-weighted and averaged class-weighted accuracy produced from each MCS by applying it to the test sites. As expected, the EMCS maintained the highest values in two accuracies, 89.79% sample-weighted and 85.88% class-weighted accuracy. The classification performance is somewhat satisfactory even if the test set is independent of the training set, which means they are taken from different power line corridors. This typically happens in a practical classification requiring no training on a new data set to be classified. However, as shown in the confusion matrix of the classification on Table 5-7 and the classification map on Figure 5-10 done by the EMCS, the EMCS ($C^5_{TR115+TR230}$) caused considerable commission errors of the pylon class as the $C^1_{TR115+TR230}$ committed in the section 5.3.2. Nevertheless, the EMCS decreased 15% commission error rate compared to the $C^1_{TR115+TR230}$ (compare the commission error rates for the pylon class in Table 5-3 and Table 5-7). Most of the errors occurred over transmission line and tree which are overlapped each other in vertical as seen in Figure 5-10. This is because the training samples selected in the early step, the TR115 and TR230, do not treat such overlaps in height, which means transmission line and tree in the samples maintain adequate clearance each other. Such overlaps occasionally happen in the test sites. Unlike the TR115 and TR230, the TR69 includes trees closely living under transmission lines. As seen in Table 5-7, this led to relatively higher classification accuracies for the vegetation and wire class than classifiers trained from other training samples. For other classes, similar main confusion observed from the results in the chapter 3 and 4 has occurred. Some of tree trunks close to the ground surface were misclassified into the low object class. On the contrary, the EMCS

incorrectly assigned the vegetation class to some fences surrounding with trees. Most omission errors of the building class were observed over building walls, while most commission errors happened over container box roofs. Minor errors have been found over distribution lines and small pylons holding them.

Table 5-7. Confusion matrix of EMCS

Class		Predicted					Omission error (%)
		Vegetation	Wire	Pylon	Building	Low object	
Actual	Vegetation	**1,178,400**	687	14,311	10,359	99,432	9.58
	Wire	11,452	**103,713**	22,827	3,119	334	26.68
	Pylon	827	1,814	**16,254**	36	237	15.20
	Building	42,967	2	168	**988,881**	49,952	8.60
	Low object	35,449	68	370	13,381	**418,306**	10.54
Commission error (%)		7.15	2.42	69.86	2.65	26.39	

On the computational cost, the EMCS takes 258.93 minutes to classify 1 km of the power line corridor as seen in Table 5-8. For each test site, the feature extraction is performed once and the computational cost linearly increases depending on the number of selected classification models to classify the test site. Apart from the classification, it takes 246.67 minutes to build a single classification model from the corresponding training sample, that is, 986.68 minutes (16.44 hours) are estimated for producing a classification model pool accommodating four classifiers. However, the suggested MCS system supposes that the model pool is already built, so the processing time for building the model pool is excluded from the computational cost estimation.

Table 5-8. Computational cost of EMCS (minutes/km)

Steps	Feature extraction		Classification for each classifier		MCS	Total (Two classifiers)
	Point	Object	Point	Object		
Processing time	194.65	43.27	4.02	6.21	0.55	258.93

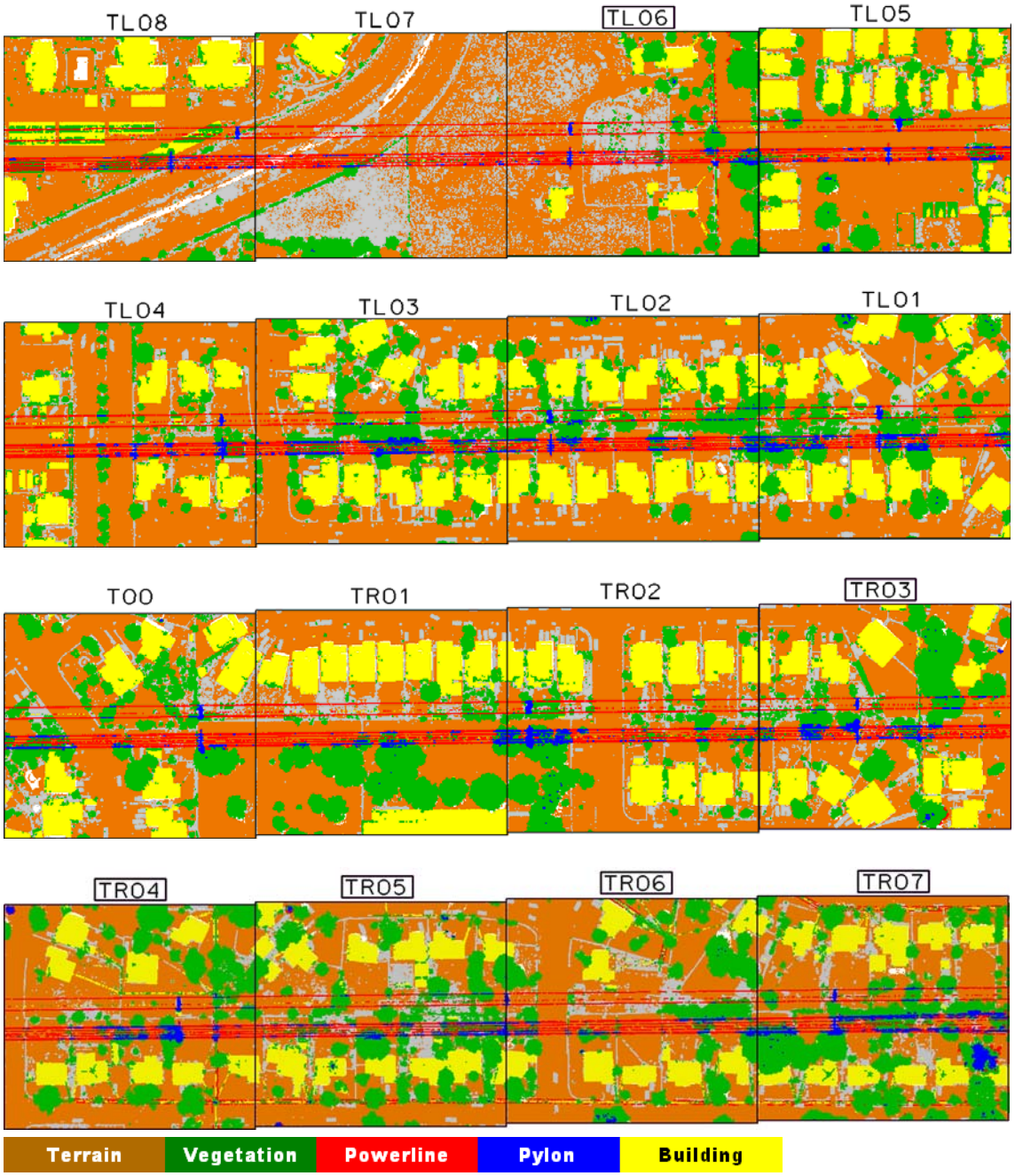


Figure 5-10. Final classification map of EMCS

5.4. Summary

This chapter employed MCS to achieve an improvement in classification by cooperating the classifiers built from features described in chapter 3 and 4, called FMCS. According to the experimental results, FMCS outperformed any single classifiers amongst the committee of classifiers and brought 2.58% and 2.17% increase in the respective sample-weighted and class-weighted accuracy. In spite of such successful classification performance, FMCS has a limitation in treating intra-class variations which are typically observed over power line corridors, e.g., variance in size, shape, material, and so on for an object class. To solve the problem, this chapter therefore constructed SMCS where each of membership classifiers is trained with samples taken from different power line environments, i.e., corridors with different carrying voltages. An optimal combination of the voltage-dependent classifiers according to the voltage type of an incoming corridor led to better classification accuracy than any other single classifiers. In addition, a hierarchical parallel combination of FMCS and SMCS as seen in Figure 5-8, termed EMCS, which is the suggested classification system in this dissertation, outperformed all other classification models, single classifiers, FMCS, and SMCS. Moreover, the EMCS resulted in 89.79% sample-weighted and 85.88% class-weighted accuracy without additional training samples. A benefit of the EMCS is to select optimal pre-trained classifiers and apply them to a new power line corridor. Nevertheless, it results in a similar classification quality as using training samples taken from the new corridor to be classified. In addition to the accuracy, once a classification model pool is already constructed, the processing time takes 4.3 hours to process 1 km power line

corridor, which is more cost-efficient compared with manual classification works. The processing time for manual classification depends on human knowledge and experience, but as aforementioned in section 3.4.1, two days were taken to manually produce the classification reference of 1.9 km corridor LiDAR, which means 8.4 hours taken per kilometer.

6. Conclusions

This study examined three classification approaches to identify key corridor objects, power line, pylon, building, vegetation, and remaining objects (named low object), from LiDAR data. The first two methods utilized two different feature sets, called point- (chapter 3) and object-based feature set (chapter 4), extracted by applying the feature extraction in different circumstances and built a classifier from each set. Ensemble technique is employed to make a more intelligent decision by fusing the classifiers already built in chapter 3 and 4 (chapter 5).

In chapter 3, diverse LiDAR features were developed and investigated and the potentials of a supervised classification method was examined for identifying power line corridor objects. Random Forests was employed to build a supervised classifier using the features. A point-based method, which extracts total 21 features for each single LiDAR point, was proposed. Some of feature groups including the vertical-related feature were newly designed, particularly for characterizing pylons and wires. The experimental results suggested that it is important to train the classifier with class-balanced training samples. Compared to unbalanced learning, training from balanced data showed 1.33% and 4.44% higher learning performance in sample-weighted and class-weighted accuracy respectively. It is realized that balanced learning resulted in almost equivalent accuracy across all the classes. A sensitivity analysis of classifiers trained with different samples has been conducted. This study found that an optimal classification model needs to be trained with a training sample containing more diverse objects representing intra-class

variations. The optimal classifier showed high classification results in both sample-weighted (91.04%) and class-weighted (90.07%) accuracy. The performance of point-based classifier was compared to a grid-based classifier. The experiments confirmed that the point-based classifier shows 4.86% and 5.74% higher in the respective sample-weighted and class-weighted accuracy than the grid-based classifier. This result suggests that the point-based classifier is more suitable for discriminating vertical overlapping of multiple objects. Even though the proposed supervised classifier has demonstrated its success in corridor scene classification, the classifier still produced misclassification errors, especially regions where sufficient neighboring points are not collected to exhibit their distinctive characteristic such as building roof ridges, hips, and eaves. Thus, future investigation is necessary to further rectify those errors by introducing more features, called object-based features described in chapter 4, extracted with different perspectives.

Unlike chapter 3 utilizing the point-based features, chapter 4 proposed to use object-based features extracted from point groups, each of which is produced by partitioning a point cloud into meaningful point groups (or point segments). This object-based method is expected to mitigate “salt and pepper” effect which is typically observed in the point-based method and improve classification accuracy for class objects exhibiting homogeneity such as building roof and transmission line. For generating meaningful point groups, this study performed point segmentation twice (in terms of linearity and planarity) and derived different features (i.e., linear and planar features) from each segmentation result. Those features are excellent salience to separate power line (linear-like), building (planar-like) and vegetation (neither linear nor planar). Practically, the

experiments demonstrated higher classification accuracies for the wire (95.85%), and building (91.99%), and vegetation class (94.29%) compared to other classes having an identification difficulty due to their intra-class variations. This chapter introduced an ensemble system where the combining method selection is critical. For the classification ensemble system built in this study, the classifier-level fusion, which combines classifiers built from multiple feature sets, is more efficient than the feature-level fusion, which groups multiple feature sets into a new feature set, considering both high accuracy and low computation cost. A comparative analysis with classification using point-based feature (chapter 3) and object-based feature (chapter 4) indicated that the point-based feature is useful for all the power line corridor objects, while the object-based feature better works for class objects showing line-likeness, planar-likeness, and randomness geometry such as power line (+3.75% better), building (-0.93% similar), and vegetation (+4.09% better). In addition to the class per accuracy, the classification method in this chapter resulted in 89.92% sample-weighted and 82.04% class-weighted accuracy overall. Even though the classification method in chapter 3 showed higher accuracy than one in chapter 4, the major goal of this study is to achieve more classification improvement by combining their results based on ensemble framework, referred as MCS (Multiple Classifier System) which is introduced in chapter 5.

Chapter 5 introduces MCS, also referred as classifier fusion, to improve classification quality by cooperating pre-built classifiers each other, one classifier trained from point-based features (PBF) in chapter 3 and another classifier from object-based features (OBF) in chapter 4. Consequently, MCS, termed as FMCS, yielded 2.58% and

2.17% higher in respective sample-weighted and class-weighted accuracy compared to the best one amongst the single classifiers. Another issue in this chapter is unavailability of pre-classified training samples to build new classifiers for every incoming datasets. Therefore, this study applies an ensemble of pre-built classifiers (SMCS), each of which is trained with samples from different power line environments, i.e., corridors with different carrying voltages, to unlabeled corridors. An experiment demonstrates that the SMCS, where voltage-dependent classifiers are optimally combined according to the voltage type of the incoming corridor, produced the best classification results in both sample- (0.81% higher) and class-weighted accuracy (0.45% higher) compared to any other single classifiers. In addition, a hierarchical parallel combination of FMCS and SMCS as seen in Figure 5-8, termed EMCS, which is the suggested classification system in this dissertation, outperformed all other classification models, single classifiers, FMCS, and SMCS. Moreover, the EMCS resulted in 89.79% sample-weighted and 85.88% class-weighted accuracy without additional training samples. A benefit of the EMCS is to select optimal pre-trained classifiers and apply them to a new power line corridor. Nevertheless, it results in a similar classification quality as using training samples taken from the new corridor to be classified.

This study provides a pipeline for classification of power line corridor LiDAR: segmentation, feature extraction, feature de-correlation, feature selection, balanced learning, outcome combining, and decision. These procedures have been over 90% automated for efficient data processing against massive point cloud. According to a throughput test in chapter 3, approximately 4 hours are taken to classify 1km of corridor.

However, the computation cost can be dramatically decreased by General-Purpose Graphic Process Unit (GPGPU) integrated computing system. In addition to the automation in classification, the suggested ensemble system demonstrated the robustness in classification if optimal classifiers are chosen from a classifier pool depending on scene characteristics of new incoming dataset. Combining the optimal classifiers achieved similar classification performance without any additional training over the new dataset. Moreover, this dissertation showed a potential which the classifier pool based system is applicable for practical classification if a plenty of existing knowledge obtained from past classification works is available. A database accommodating more diverse classifiers can be constructed using the knowledge so that it classifies power line corridors where objects with diversity live.

There are two limitations with the method suggested in this dissertation: an empirical feature categorization relevant to class and the utilization of only Random Forests as a base classifier. In chapter 3, for the relevant feature selection, the point-based features have been categorized into four groups depending on their class relevance, which means a feature presents a salient characteristic for a specific class. However, the categorization has been done by an empirical experience obtained through visual investigations and analyses for each class. As a future work, therefore, this study takes a class dependent feature selection method which measures the feature-feature and feature-class correlations based on the mutual information theory (Zhou and Wang, 2009) for the feature selection. Secondly, this dissertation takes only Random Forests as a base classifier to build a classification model without any comparative analysis with other

classification algorithms such as SVM, bagging, boosting and so on. This is because the dissertation more focuses on developing features suitable for the power line corridor objects rather than the comparison of the algorithms. This is remained as another future work. Besides the limitation, this dissertation suggested developing an ensemble system (or MCS) with numbers of diverse classifiers for power line corridor classification. However, a difficulty in obtaining sufficient pre-classified training samples brought limited number of classifiers; for now each classifier is able to treat a categorized corridor by its carrying voltage, i.e., three voltage typed classifiers. A future work is to categorize the classifiers in more detail by collecting much larger samples and introducing exemplars, an exemplar indicates a sample including a class object with a specific characteristic criterion. For example, pylon exemplars (E_p) in Figure 6-1 are categorized into pole-typed, lattice-typed, H-shaped, and other-shaped exemplars and diverse pylon classifiers are individually built from those exemplars as seen in Figure 6-2. For new corridor data best relevant classifiers to the corridor are selected and its classification is done by the same mechanism of MCS addressed through chapter 4 and 5 (Figure 6-3). This optimal combination of exemplar-based classifiers is expected to bring more accurate and precise classification results compared to that of voltage-based classifiers introduced in chapter 5.

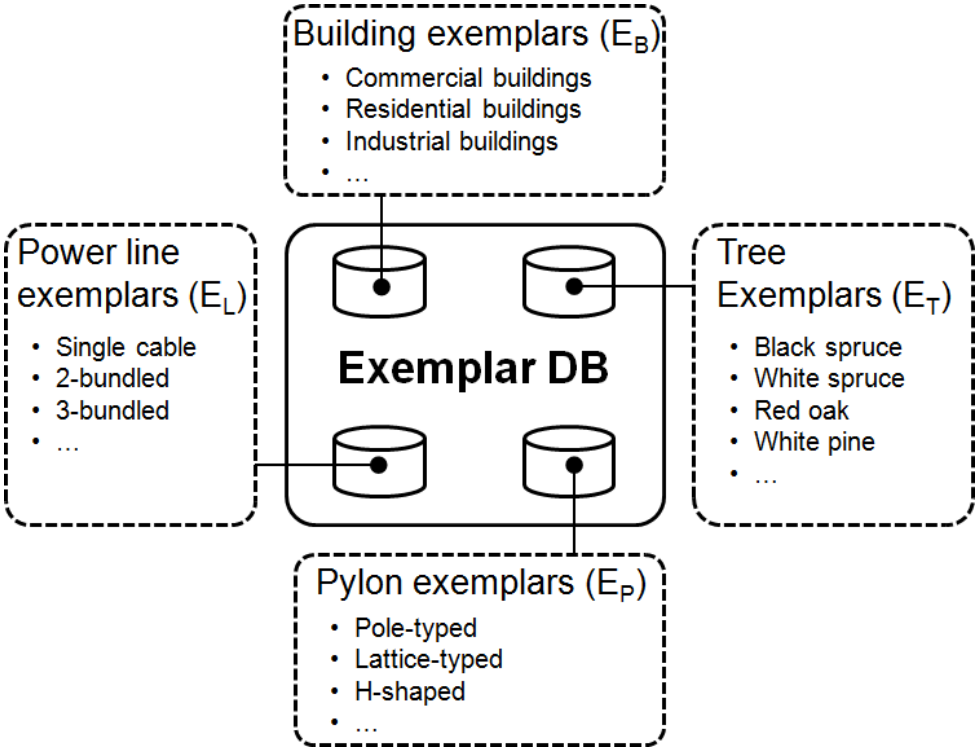


Figure 6-1. Exemplar database for building, tree, power line, and pylon

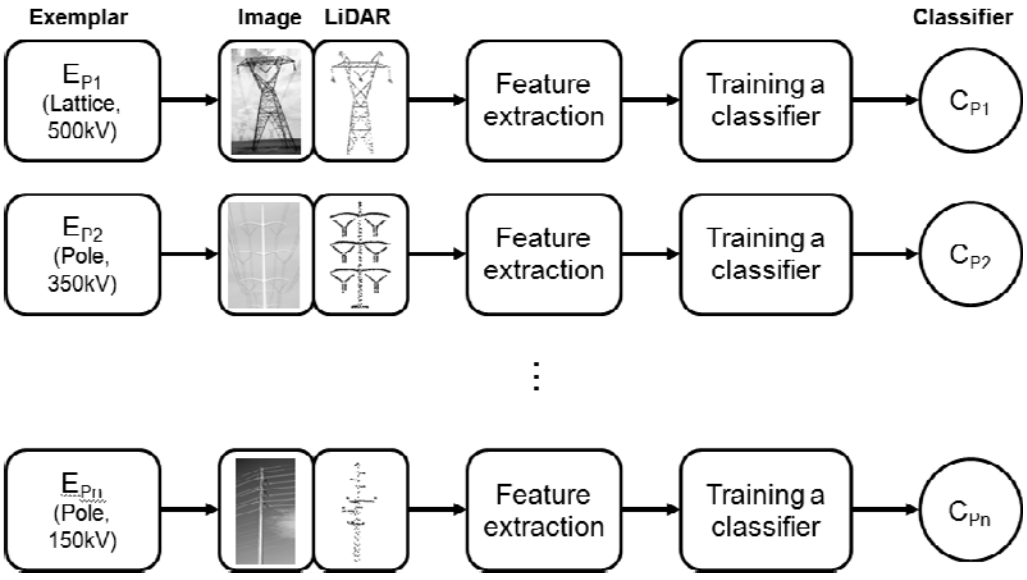


Figure 6-2. Pylon classifier building from exemplars

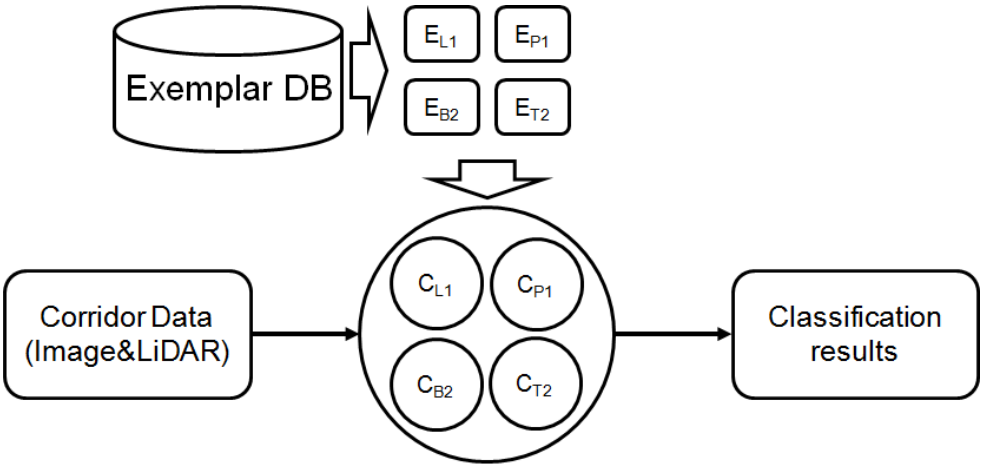


Figure 6-3. Classification of new corridor data using optimally selected classifiers

Reference

Adam P. Young and Scott A. Ashford, 2006. Application of Airborne LIDAR for Seacliff Volumetric Change and Beach-Sediment Budget Contributions. Journal of Coastal Research: 22 (2): pp. 307 – 318

Antonarakis, A.S., K.S. Richards, and J. Brasington, 2008. Object-based land cover classification using airborne LiDAR, Remote Sensing of Environment, 112(6): 2988-2998

Applanix, Applanix-A Trimble Company. www.applanix.com (accessed on June 22, 2013)

Axelsson, P. (1999). Processing of laser scanner data—algorithms and applications. ISPRS Journal of Photogrammetry and Remote Sensing, 54(2), 138-147

Bang, K., 2010. Alternative methodologies for LiDAR system calibration, PhD thesis.

Biosca, J. M. and J. L. Lerma, 2008. Unsupervised robust planar segmentation of terrestrial laser scanner point clouds based on fuzzy clustering methods, ISPRS Journal of Photogrammetry & Remote Sensing, 63, 84-98

Bishop, C.. 2006. Pattern Recognition and Machine Learning. Springer. 2006.

Benediktsson, J. A., Swain, P. H., 1992. Consensus theoretic classification methods, IEEE Trans. on Systems, Man and Cybernetics, vol. 22, no. 4, pp. 688–704

Benediktsson J.A., Palmason, J.A., Sveinsson, J.R., 2005. Classification of hyperspectral data from urban areas based on extended morphological profiles, IEEE Transactions on Geoscience and Remote Sensing, 42 (2005), pp. 480–491

Brasington J, Langham J, Rumsby B. 2003. Methodological sensitivity of morphometric estimates of coarse fluvial sediment transport. Geomorphology 53(3/4): 299–316

Breiman, L., 1996, Bagging Predictors, Machine Learning, Vol. 24, No. 2, pp. 123-140

Breiman, L., 2001. Random forests, Machine Learning, 45(1): 5-32

Bremer, M., Wichmann, V. and Rutzinger, M., 2013. Eigenvalue and graph-based object extraction from mobile laser scanning point clouds. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. II-5/W2, pp. 55–60

Brenner, C., 2009. Extraction of Features from Mobile Laser Scanning Data for Future Driver Assistance Systems. Advances in GIScience, Lecture Notes in Geoinformation and Cartography, Springer (2009) pp. 25–42

Brovelli, M., Cannata, M., and Longoni, U. 2004. Lidar data filtering and DTM interpolation within GRASS. Transactions in GIS 8(2), 155–174

Cai, H., Rasdorf, W., 2008. Modeling road centerlines and predicting lengths in 3-D using LIDAR point cloud and planimetric road centerline data, Computer-Aided Civil and Infrastructure Engineering, 23(3), 157–73

Carlberg, M., P. Gao, G. Chen, and A. Zakhor, 2009. Classifying urban landscape in aerial LiDAR using 3D shape analysis, Proceeding of Image Processing (ICIP), 2009 16th IEEE International Conference, 7-10 November 2009, pp. 1701-1704

Chan, R. H., Ho, C. W., & Nikolova, M. (2005). Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization. Image Processing, IEEE Transactions on, 14(10), 1479-1485

Chawla, N.V., K.W. Bowyer, L.O. Hall, and W.P. Kegelmeyer, 2002. Smote: Synthetic minority oversampling technique, Journal of Artificial Intelligence Research, 16, pp. 321–357

Chen, C., A., Liaw, and L. Breiman, 2004. Using Random Forests to learn imbalanced data, Technical Report, Department of Statistics, University of California

Cheng, L., J. Gong, M. Li, Y. Liu, 2011. 3d building model reconstruction from multi-view aerial imagery and LIDAR data, Photogrammetric Engineering and Remote Sensing, 77 (2) (2011), pp. 125–139

Chehata, N., L. Guo, and C. Mallet, 2009. Airborne lidar feature selection for urban classification using Random Forests, Proceeding of Laser scanning 2009, IAPRS, Vol. XXXVIII, Part 3/W8, 1-2 September 2009, Paris, France, pp. 207-212

Choi, Y. W. , Y. W. Jang , H. J. Lee and G. S. Cho, 2008. Three-dimensional LiDAR data classifying to extract road points in urban area, IEEE Geosci. Remote Sens. Lett., 5(4), pp.725 -729

Chust, G., Maitane Grande, Ibon Galparsoro, Adolfo Uriarte, Ángel Borja, 2010. Capabilities of the bathymetric Hawk Eye LiDAR for coastal habitat mapping: A case study within a Basque estuary, Estuarine, Coastal and Shelf Science, 89(3), pp. 200-213

Chust, G., I. Galparsoro, A. Borja, J. Franco, A. Uriarte, 2008.Coastal and estuarine habitat mapping, using LIDAR height and intensity and multi-spectral imagery, Estuarine, Coastal and Shelf Science, 78 (4) (2008), pp. 633–643

Collin, A., Bernard Long, Phillippe Archambault, 2012. Merging land-marine realms: Spatial patterns of seamless coastal habitats using a multispectral LiDAR, Remote Sensing of Environment, Volume 123, pp. 390-399

Coops, N.C., A. Varhola, C. W. Bater, P. Teti, S. Boon, N. Goodwin, and M. Weiler, 2009. Assessing differences in tree and stand structure following beetle infestation using lidar data, Canadian Journal of Remote Sensing, 35(6), pp. 497-508

Côté, J.F., J.L. Widlowski, R.A. Fournier, M.M. Verstraete, 2009. The structural and radiative consistency of three-dimensional tree reconstructions from terrestrial lidar, Remote Sensing of Environment, 113 (5) (2009), pp. 1067–1081

Coulston, J.W., G.G. Mosen, B.T. Wilson, M.V. Flnco, W.B. Cohen, and C.K. Brewer, 2012. Modeling percent tree canopy cover: A pilot study, Photogrammetric Engineering & Remote Sensing, 78(7):715–727

Crawford, M., Kim, W., 2009. Manifold learning for multi-classifier systems via ensembles. In: Benediktsson, J.A., Kitller, J., Roli, F. (Eds.), MCS, Lecture Notes in Computer Science, vol. 5519. Springer, Berlin, pp. 519-528

Dara, R.A., 2007. Cooperative Training in Multiple Classifier Systems; thesis :Doctor of Philosophy; Waterloo, Ontario, Canada

Dasarathy, B. V. and Sheela, B. B., 1979. Composite classifier system design: Concepts and methodology, Proceedings of the IEEE, vol. 67, no. 5, pp.708 -713 1979

De Rose, R.C., Les R. Basher, 2011. Measurement of river bank and cliff erosion from sequential LIDAR and historical aerial photography, Geomorphology, 126(1-2), pp. 132-147

Dietterich, T., 2000,. Ensemble methods in machine learning. In J. Kittler, & F. Roli (Eds.), Multiple classifier systems, Vol. 1857 of Lecture Notes in Computer Science (pp. 1–15). Cagliari, Italy, Springer

Dold, C. and Brenner, C., 2004. Automatic Matching of Terres-trial Scan Data as a Basis for the Generation of Detailed 3D City Models. In: O. Altan (ed.), Proc. of the XXth ISPRS Congress, IAPRS, Vol. XXXV-B3.

Drucker, H., Cortes, C., Jackel, L. D., LeCun, Y., Vapnik, V., 1994. Boosting and other ensemble methods, Neural Computation, vol. 6, no. 6, pp. 1289–1301

Durand, M., K. M. Andreadis, D. E. Alsdorf, D. P. Lettenmaier, D. Moller, and M. Wilson, 2008. Estimation of bathymetric depth and slope from data assimilation of swath altimetry into a hydrodynamic model, Geophys. Res. Lett., 35, L20401

Dybowski, J. N., Heider, D., Homann, D., 2010. Prediction of co-receptor usage of HIV-1 from geno-type. PLoS Comput Biol 6: e1000743. doi: 10.1371/journal.pcbi.1000743

Efron B. and Tibshirani, R.J, 1993. An Introduction of the Bootstrap. Chapman and Hall/CRC: New York

Elwell, R., and Polikar, R., 2011. Incremental learning of concept drift in nonstationary environments. IEEE Transactions on Neural Networks 22 (10), pp. 1517–1531

Erener, A., 2013. Classification method, spectral diversity, band combination and accuracy assessment evaluation for urban feature detection, International Journal of Applied Earth Observation and Geoinformation, 21 (2013), pp. 397–408

Filin, S., and Pfeifer, N. 2006. Segmentation of airborne laser scanning data using a slope adaptive neighborhood. ISPRS Journal of Photogrammetry and Remote Sensing 60, 71.80

Flood, M., 2011. Workflow challenges on airborne lidar electrical transmission projects, Photogrammetric Engineering & Remote Sensing,77(5): 438-443

Foody, G.M., Mathur, A., 2004. Toward intelligent training of supervised image classifications: Directing training data acquisition for SVM classification, Remote Sensing of Environment, 93 (2004), pp. 107–117

Forlani G., C. Nardinocchi, M. Scaioni, and P. Zingaretti, 2006. Complete classification of raw LIDAR data and 3D reconstruction of building, Pattern Analysis and Applications, 8(4): 357-374

Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. Journal of computer and system sciences, 55(1), 119-139

Fugro LADS, Airborne LiDAR Bathymetry manufactured by Fugro. http://www.fugrolads.com/download/datasheets/Fugro-LADS-Mk3 (accessed on July 10, 2013)

Gama, J., Brazdil, P., 2000. Cascade generalization, Machine Learning, vol. 41, no. 3, pp. 315-343

Gesch, D., Wilson, R., 2001. Development of a seamless multisource topographic/bathymetric elevation model of Tampa Bay, Marine Technology Society Journal, 35 (4), pp. 58–64

Geurts, P., Ernst, D., Wehenkel, L., 2006. Extremely randomized trees. Machine Learning 63 (1), 3–42

Guo, L., N. Chehata, C. Mallet, and S. Boukir, 2011. Relevance of airborne lidar and multispectral image data for urban scene classification using Random Forests, ISPRS Journal of Photogrammetry and Remote Sensing, 66(1): 56-66

Guo, Q., Li W., Liu, D. and Chen, J., 2012. A framework for supervised image classification with incomplete training samples, Photogrammetric Engineering & Remote Sensing, 78(6): 595-604

Habib, A; Bang, K; Kersting, A; Lee, D-C, 2009. Error Budget of Lidar Systems and Quality Control of the Derived Data. Photogramm. Eng. Remote Sensing 2009, 75, 1093–1108

Habib, A., A.P. Kersting, A. Ruifanga, M. Al-Durgham, C. Kim, D.C. Lee, 2008. Lidar strip adjustment using conjugate linear features in overlapping strips, International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 37 (Part B1) (2008), pp. 385–390

Hamby S. E., Hirst J. D., 2008. Prediction of glycosylation sites using random forests. BMC Bioinformatics. 9(1):500

Hansen, L. K., and P. Salamon, 1990. Neural Network Ensembles. IEEE Transactions on Pattern Analysis and Machine Intelligence, 12(10), pp. 993–1001

Hay, G.J., Castilla, G., Wulder, M.A., Ruiz, J.R., 2005. An automated object-based approach for the multiscale image segmentation of forest scenes International Journal of Applied Earth Observation and Geoinformation, 7 (4), pp. 339–359

Hebel, M., Arens, M., Stilla, U., 2013. Change detection in urban areas by object-based analysis and on-the-fly comparison of multi-view ALS data, ISPRS Journal of Photogrammetry and Remote Sensing, Volume 86, pp. 52-64

Hetcht, R., Meinel, G., and Buchroithner, M. F., 2008. Estimation of urban green volume based on single-pulse LiDAR data. – IEEE Transactions on Geoscience and Remote Sensing, Vol. 46 (11): 3832-3840

Hilary F. Stockdonf, Asbury H. Sallenger Jr., Jeffrey H. List and Rob A. Holman, 2002. Estimation of Shoreline Position and Change Using Airborne Topographic Lidar Data, Journal of Coastal Research, 18(3) , pp. 502-513

Hill, R.A., S. A. Hinsley, and D. L. A. Gaveau, 2002. Mapping forest pattern and structure at a landscape scale using airborne laser scanning technology, Avian Landscape Ecology: Pure and Applied Issues in the Large-Scale Ecology of Birds, pp. 60-67

Ho, T.K., Hull, J.J., Srihari, S.N., 1994. Decision combination in multiple classifier systems, IEEE Trans. on Pattern Analy. Machine Intel., vol. 16, no. 1, pp. 66–75, 1994

Hough, P., 1962: Method and Means for Recognizing Complex Patterns. U.S. Patent 3.069.654

Hyde, P., R. Dubayah, B. Peterson, J. B. Blair, M. Hofton, C. Hunsaker, R. Knox, and W. Walker, 2005. Mapping forest structure for wildlife habitat analysis using waveform lidar: Validation of montane ecosystems, Remote Sensing of Environment, 96(3-4), pp. 427-437

Ituen, I. and Sohn, G., 2010. The Way Forward: Advances in Maintaining Right-Of-Way of Transmission Lines. GEOMATICA, 64(40):451-462

Jackowski, K., 2013, Fixed-size ensemble classifier system evolutionarily adapted to a recurring context with an unlimited pool of classifiers, Pattern Analysis and Application, 1-16

Jacobs, R. A., Jordan, M. I., Nowlan, S. J., Hinton, G. E., 1991. Adaptive mixtures of local experts, Neural Computation, vol. 3, pp. 79-87

Jimenez, L.O., Rivera-Medina, J.L., Rodriguez-Diaz, E., Arzuaga-Cruz, E., Ramirez-Velez, M., 2005. Integration of spatial and spectral information by means of unsupervised extraction and classification for homogenous objects applied to multispectral and hyperspectral data, IEEE Transactions on Geoscience and Remote Sensing, 43 (2005), pp. 844–851

Jordan M. J., Jacobs, R. A., 1994. Hierarchical mixtures of experts and the EM algorithm, Neural Computation, vol. 6, no. 2, pp. 181-214

Jwa, Y., and G. Sohn. 2012. A piecewise catenary curve model growing for 3D power line reconstruction. Photogrammetric Engineering & Remote Sensing, 78(12): 1227 – 1240

Jwa, Y., Sohn, G., Kim, H. B., 2009. Automatic detection and modeling of powerline from airborne laser scanning data. ISPRS Laserscanning 2009, September 1-4, Paris

Kim, E., and G. Medioni, 2011. Urban Scene Understanding from aerial and ground LIDAR data, Machine Vision and Applications (MVA), 22(4): 691 – 703

Kim, H. B., Sohn, G., 2013, Point-based Power Line Corridor Scene Classification using Random Forests, Photogrammetric Engineering & Remote Sensing, 79(9): 821-833

Kim, H.B. and Sohn, G., 2011. Random forest based multiple classifier system for power-line scene classification. ISPRS Workshop on Laserscanning 2011, 29-31 August 2011, the University of Calgary, Calgary, Alberta, Canada. International Archives of Photogrammetry and Remote Sensing, 38(5/W12)

Kim, H.B. and Sohn, G., 2010. 3D Classification of Power-line Scene from Airborne Laser Scanning Data. Photogrammetric Computer Vision (PCV) 2010, September 1-3, Paris, France. International Archives of Photogrammetry and Remote Sensing, 38(3A):126-132.

Kim, Y., Z. Q. Yang, W. B. Cohen, D. Pflugmacher, C. L. Lauver, and J. L. Vankat, 2009. Distinguishing between live and dead standing tree biomass on the north rim of grand canyon national park, USA using small-footprint lidar data, Remote Sensing of Environment, 113(11), pp. 2499-2510

Kittler, J., Hatef, M., Duin, R., & Matas, J. 1998. On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(3), 226–239

Ko, A. H., Sabourin, R., 2013. Single Classifier-based Multiple Classification Scheme for weak classifiers: An experimental comparison, Expert Systems with Applications, 40(9), pp. 3606-3622

Ko, C., Sohn, G. and T.K., Remmel, 2013. A spatial Analysis of Geometric Features Derived from High-Density Airborne LiDAR Data for Tree Species Classification. Canadian Journal of Remote Sensing. 39(s1): S73-S85, 10.5589/m13-024

Kulawardhana, R.W., Sorin C. Popescu, Rusty A. Feagin, 2014. Fusion of lidar and multispectral data to quantify salt marsh carbon stocks, Remote Sensing of Environment, Volume 154, pp. 345-357

Kuncheva, L. I., 2004. Classifier ensembles for changing environments, 5th Int. Workshop on Multiple Classifier Systems, Lecture Notes in Computer Science, F. Roli, J. Kittler, and T. Windeatt, Eds., vol. 3077, pp. 1–15

Lalonde, J., N. Vandapel, D. Huber and M. Hebert, 2006. Natural terrain classification using three-dimensional ladar data for ground robot mobility, Journal of Field Robotics, 23(10): 839 – 861

Lamonaca, A., Corona, P., Barbati, 2008. A. Exploring forest structural complexity by multi-scale segmentation of VHR imagery. Remote Sensing of Environment, doi:10.1016/j.rse.2008.01.017

Lee, D.H., K.M. Lee, S.U. Lee, 2008. Fusion of lidar and imagery for reliable building extraction, Photogrammetric Engineering and Remote Sensing, 74 (2) (2008), pp. 215–225

Leica AHAB, Airborne LiDAR Bathymetry manufactured by Leica. http://www.airbornehydro.com/sites/default/files/Leica%20AHAB%20HawkEye%20III.pdf (accessed on July 10, 2013)

Leica, Leica Geosystems. http://www.leica-geosystems.us (accessed on June 22, 2013)

Lehtomäki, M., A. Jaakkola, J. Hyyppä, A. Kukko, and H. Kaartinen, 2010. Detection of Vertical Pole-Like Objects in a Road Environment Using Vehicle-Based Laser Scanning Data, Remote Sensing, 2(3): 641-664

Liang, J., Zhang, J., Deng, K., Liu, Z. and Zhi, Q. 2011. A new power-line extraction method based on airborne LiDAR point cloud data, International Symposium on Image and Data Fusion, 9-11 Aug., 2011, pp. 1-4

Lim, E.H. and D., Suter, 2009. 3d terrestrial lidar classifications with super-voxels and multi-scale conditional random fields, Computer-Aided Design, 41(10): 701-710

Lin, C., G. Thomson, C.S. Lo, and M.S. Yang, 2011. A Multi-level Morphological Active Contour Algorithm for Delineating Tree Crowns in Mountainous Forest, Photogrammetric Engineering & Remote Sensing, 77(3): 241-249

Livingston, F., 2005. Implementing Breiman's Random Forest Algorithm into Weka, ECE591Q Machine Learning Conference Papers, Nov. 27

Lodha, S.K., D. Fitzpatrick and D.P. Helmbold, 2007. Aerial lidar data classification using expectation-maximization, Proceedings of SPIE Conference on Vision Geometry, XIV, volume 6499, January 2007

Lodha, S. K., Fitzpatrick, D. M. and Helmbold, D. P., 2007a. Aerial Lidar Data Classification using AdaBoost. In: International Conference on 3-D Digital Imaging and Modeling, Montreal, pp. 435–442.

Lodha, S. K., Kreps, E. J., Helmbold, D. P. and Fitzpatrick, D., 2006. Aerial LiDAR Data Classification Using Support Vector Machines (SVM). In: International Symposium on 3D Data Processing, Visualization and Transmission, IEEE, Chapel Hill, NC, pp. 567–574.

Lu, W. L., K.P. Murphy, J.J. Little, A. Sheffer, and H. Fu, 2009. A hybrid conditional random field for estimating the underlying ground surface from airborne Lidar data, IEEE Geoscience and Remote Sensing, 47(82): 2913-2922

Mason, D.C., M.S. Horritt, N.M. Hunter, P.D. Bates, 2007. Use of fused airborne scanning laser altimetry and digital map data for urban flood modelling, J Hydrol Process, 21 (2007), pp. 1436–1447

Matikainen L, Hyyppä J, Ahokas E, Markelin L, Kaartinen H, 2010. Automatic Detection of Buildings and Changes in Buildings for Updating of Maps. Remote Sensing. 2010; 2(5):1217-1248

McCormick, M.P., 2005. Airborne and spaceborne lidar. In: Weitkamp CE Lidar Range-resolved optical remote sensing of the atmosphere. Springer series in Optical sciences, vol 102, chap 13. Springer Science, Heidelberg, UK, pp 355–397.

McLaughlin, R. A., 2006. Extracting transmission lines from airborne LIDAR data, IEEE Geoscience and Remote Sensing Letters, April 2006, Vol. 3, No. 2, pp. 222-226

Melzer, T., and C. Briese, 2004. Extraction and modeling of power lines from ALS point clouds, Proceedings of 28th Workshop of the Austrian Association for Pattern Recognition, pp.47-54

MIKE2.0, Big Data Definition, http://mike2.openmethodology.org/wiki/Big_Data_Definition#Read_More (accessed on July 10, 2013)

Mills, S.J., M. P. Gerardo , Z. Li , J. Cai , R. Hayward , L. Mejias and R. Walker, 2010. Evaluation of aerial remote sensing techniques for vegetation management in power line corridors, IEEE Trans. Geosci. Remote Sens., 48(9), pp.3379 -3390

Mitche D., Spiegielhalter C.J., Taylor C., Machine Learning, Neural and Statistical Classification, Ellis Horwood 1994

Mitra, N. J., Nguyen, A., Guibas, L. (2004). Estimating surface normals in noisy point cloud data. International Journal of Computational Geometry & Applications, 14(04n05), 261-276

Moretto, J., E. Rigon, L. Mao, F. Delai, L. Picco, M.A. Lenzi, 2014. Short-term geomorphic analysis in a disturbed fluvial environment by fusion of LiDAR, colour bathymetry and dGPS surveys, CATENA, Volume 122, pp. 180-195

Muhlbaier, D., and Polikar, R., "An ensemble approach for incremental learning in nonstationary environments," in Lecture Notes in Computer Science, M. Haindl and F. Roli, Eds. Berlin: Springer-Verlag, 2007, vol. 4472, pp. 490-500

Na, X., S. Zhang, X. Li, H. Yu, and C. Liu, 2010. Improved Land Cover Mapping using Random Forests Combined with Landsat Thematic Mapper Imagery and Ancillary Geographic Data, Photogrammetric Engineering & Remote Sensing, 76(7): 833-840

Narayanan, R., Sohn, G., Kim, H.B. and Miller, J.R., 2011. A soft classification of mixed seabed objects based on fuzzy clustering analysis using airborne LiDAR bathymetry data. Journal of Applied Remote Sensing. 5, 053534 (2011); doi:10.1117/1.3595267

Nayegandhi, A., and J. C. Brock, 2009. Remote Sensing and Geospatial Technologies for Coastal Ecosystem Assessment and Management, pp.365 -389

Neuenschwander, A.L., L.A. Magruder, and M. Tyler, 2009. Landcover classification of small-footprint, full-waveform lidar data, Journal of Applied Remote Sensing, 3(1):033544-033544

Niemeyer, J., J.D. Wegner, C. Mallet, F. Rottensteiner and U. Soergel, 2011. Conditional Random Fields for Urban Scene Classification with Full Waveform LiDAR Data. In: Stilla, Rottensteiner, Mayer, Jutzi, Butenuth (eds.), Photogrammetric Image Analysis (PIA), LNCS 6952, Springer, Heidelberg, pp. 233-244

Niemeyer, J., F. Rottensteiner, U. Soergel, 2014. Contextual classification of lidar data and building object detection in urban areas, ISPRS Journal of Photogrammetry and Remote Sensing, Volume 87, pp. 152-165

Optech, Optech company. http://www.optech.com (accessed on 22 June, 2013)

Pfeifer, N., Stadler, P., and Briese, C. 2001. Derivation of digital terrain models in the SCOP++ environment. OEEPE Workshop on Airborne Laserscanning and Interferometric SAR for Detailed Digital Elevation Models, Stockholm

Polikar, R., 2006, Ensemble Based Systems in Decision Making, IEEE Circuits and Systems Magazine, 6(3), pp. 21-45

Pu, S., Martin Rutzinger, George Vosselman, Sander Oude Elberink, 2011. Recognizing basic structures from mobile laser scanning data for road inventory studies, ISPRS Journal of Photogrammetry and Remote Sensing, 66(6), pp. S28-S39

Pu, S. and Vosselman, M.G., 2007. Extracting windows from terrestrial laser scanning, ISPRS workshop : Laser scanning 2007 and SilviLaser 2007 International Society for Photogrammetry and Remote Sensing (ISPRS), Espoo, Finland, pp. 320-325

Rabbani, T., F. van den Heuvel, 2005. Efficient Hough Transform for Automatic Detection of Cylinders in Point Clouds. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences XXXVI–3/W19, pp. 60-65

Raciti, S.M., Lucy R. Hutyra, Jared D. Newell, 2014. Mapping carbon storage in urban trees with multi-source remote sensing data: Relationships between biomass, land use,

and demographics in Boston neighborhoods, Science of The Total Environment, Volumes 500–501, 1 December 2014, Pages 72-83

Riano, D., E. Meier, B. Allgower, E. Chuvieco, and S. L. Ustin, 2003. Modeling airborne laser scanning data for the spatial generation of critical forest parameters in re behavior modeling, Remote Sensing of Environment, vol. 86(2), pp. 177-186

Richter, A., D. Faust, and H. Maas, 2013. Dune cliff erosion and beach width change at the northern and southern spits of Sylt detected with multi-temporal lidar, Catena, 103, 103–111

RIEGL, RIEGL USA. http://www.rieglusa.com (accessed on 22 June, 2013)

Rodriguez, J. J., and Kuncheva, L. I., 2006. Rotation Forest: A New Classifier Ensemble Method. IEEE Transactions on Pattern Analysis and Machine Intelligence 28, pp. 1619-1630.

Rodriguez-Galiano, V.F., B. Ghimire, E. PArdo-Iguzquiza, M. Chica-Olmo, and R.G. Congalton, 2012. Incorporating the Downscaled Landsat TM Thermal Band in Land-cover Classification using Random Forest, Photogrammetric Engineering & Remote Sensing, 78(2): 129-137

Rottensteiner, F., Trinder, J., Clode, S., Kubik, K., 2005. Using the Dempster Shafer method for the fusion of LIDAR data and multi-spectral images for building detection, Information Fusion 6(4), 283-300

Rutzinger, M., B. Höfle, M. Hollaus, and N. Pfeifer, 2008. Object-Based Point Cloud Analysis of Full-Waveform Airborne Laser Scanning Data for Urban Vegetation Classification. Sensors, 8(8): 4505-4528

Rutzinger, M., Höfle, B., Pfeifer, N., Geist, T. & Stötter, J., 2006. Object-based analysis of airborne laser scanning data for natural hazard purposes using open source components. In: International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences. Salzburg, Austria Vol. 36(part 4/C42), pp. digital media.

Saeys, Y., Inza, I., & Larra˜naga, P., 2007. A review of feature selection techniques in bioinformatics. Bioinformatics, 23, pp. 2507–2517.

Sallenger, A.H, Jr., W. B. Krabill, R. N. Swift, J. Brock, J. List, Mark Hansen, R. A. Holman, S. Manizade, J. Sontag, A. Meredith, K. Morgan, J. K. Yunkel, E. B. Frederick and H. Stockdon, 2003. Evaluation of Airborne Topographic Lidar for Quantifying Beach Changes, Journal of Coastal Research, 19(1) , pp. 125-133

Samadzadegan, F., Bigdeli, B., and Ramzi, R., 2010. A Multiple Classifier System for Classification of LIDAR Remote Sensing Data Using Multi-class SVM, Multiple Classifier Systems 9th International Workshop, MCS 2010, Cairo, Egypt, April 7-9, Proceedings

Samadzadegan, F., Hasani, H. and Schenk, T., 2012. Determination of optimum classifier and feature subset in hyperspectral images based on ant colony system, Photogrammetric Engineering & Remote Sensing, 78(12): 1261-1273

Sampath, A. and Shan, J., 2010. Segmentation and reconstruction of polyhedral building roofs from aerial Lidar point clouds, IEEE Transactions on geoscience and remote sensing 48(3): 1554-1567

Schapire, R. E.. 1990. The Strength of Weak Learnability. Mach. Learn. 5, 2 (July 1990), pp. 197-227.

Schnabel, R., Wahl, R. & Klein, R., 2007, Efficient RANSAC for point-cloud shape detection, Computer Graphics Forum, 26 (2), 214–26

Sithole, G. 2005. Segmentation and classification of airborne laser scanner data, Dissertation, TU Delft, ISBN 90 6132 292 8, Publications on Geodesy of the Netherlands Commission of Geodesy, Vol. 59

Sithole, G. and Vosselman, G., 2004. Experimental comparison of filter algorithms for bare-Earth extraction from airborne laser scanning point clouds. ISPRS Journal of Photogrammetry & Remote Sensing, 59(1-2): 85-101.

Skalak, D., 1997. Prototype selection for composite nearest neighbor classifiers, Ph.D. dissertation. Dept. of Computer Science, Technical Report 96-89, University of Massachusetts, Amherst, Massachusetts

Smeeckaert, J., Clément Mallet, Nicolas David, Nesrine Chehata, Antonio Ferraz, 2013. Large-scale classification of water areas using airborne topographic lidar data, Remote Sensing of Environment, Volume 138, Pages 134-148

Sohn, G., Jwa, Y., Kim, H.B., 2012. Automatic power line scene classification and reconstruction using airborne LiDAR data. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume I-3:167-172.

Sohn, G., Dowman, I., 2007. Data fusion of high-resolution satellite imagery and LiDAR data for automatic building extraction. ISPRS J Photogramm Remote Sens 62:43–63

Sohn, G. and Dowman, I., 2008. A Model-based Approach for Reconstructing Terrain Surface from Airborne LiDAR Data. The Photogrammetric Record, 23(122):170-193

Stockdon, H.F., A.H. Sallenger, J.H. List, R.A. Holman, 2002. Estimation of shoreline position and change from airborne scanning LIDAR data, Journal of Coastal Research, 18 (2002), pp. 502–513

Sutton, C.D., 2005. Classification and regression trees, bagging, and boosting, Handbook of Statistics: Data Mining and Data Visualization (C. R. Rao et al., editors), Elsevier Publishing, Amsterdam, Netherlands, Vol. 24, pp. 303-329

Terrascan, Terrasolid Inc. http://www.terrasolid.com/home.php (accessed on June 22, 2013).
Microstation, Bentley. http://www.bentley.com/en-US/Products/MicroStation (accessed on June 22, 2013)

Ussyshkin, R. V., Smith, R. B., 2007. A new approach for assessing lidar data accuracy for corridor mapping applications. 5th International Symposium on Mobile Mapping Technology, 29-31 May, Padua, Italy, CD-ROM.

Verma, V., R. Kumar and S. Hsu, 2006. 3D building detection and modeling from aerial lidar data. IEEE Conference on Computer Vision and Pattern Recognition, October 2006, pp. 2213-2220

Vosselman, G. 2000. Slope based filtering of laser altimetry data. IAPRS XXXIII, B3/2, Amsterdam

Vosselman, G., and Dijkman, S., 2001. 3D building model reconstruction from point clouds and ground plans. International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences, 34(3/W4), 37-44

Wang, M. and Tseng, Y., 2011. Incremental segmentation of lidar point clouds with an octree-structured voxel space. Photogramm. , 26, 32–57

Waske, B., and M. Braun, 2009. Classifier ensembles for land cover mapping using multitemporal SAR imagery, ISPRS Journal of Photogrammetry and Remote Sensing, 64(5): 450–457

Wolpert, D. H., 1992. Stacked Generalization," Neural Networks, vol. 5, no. 2, pp. 241-260

Wolpert, D.H., 2001. The supervised learning no-free-lunch theorems. In: Proceedings of the 6th Online World Conference on Soft Computing in Industrial Applications, pp. 25–42

Wozniak, M., Graña, M., Corchado, E., 2014. A survey of multiple classifier system as hybrid systems, Information Fusion, 16 (2014), pp. 3–17

Xu, J.-W., Singh, V., Govindaraju, V., Neogi, D., 2009. A cascade multiple classifier system for document categorization. In J. Benediktsson, J. Kittler, and F. Roli, editors, Multiple Classifier Systems, volume 5519 of Lecture Notes in Computer Science, pages 458-467. Springer Berlin / Heidelberg

Yang, B., and Dong, Z., 2013. A shape-based segmentation method for mobile laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, *81*, pp. 19-30.

Yao, W., Y. Wei, 2013. Detection of 3-D individual trees in urban areas by combining airborne LiDAR data and imagery, IEEE Geoscience and Remote Sensing Letters, 10 (2013), pp. 1355–1359

Yao, W., Krzystek, P., Heurich, M., 2012. Tree species classification and estimation of stem volume and DBH based on single tree extraction by exploiting airborne full-waveform LiDAR data, Remote Sensing of Environment, Volume 123, pp. 368-380

Yao, W., S. Hinz, U. Stilla, 2011. Extraction and motion estimation of vehicles in single-pass airborne LiDAR data towards urban traffic analysis, ISPRS Journal of Photogrammetry and Remote Sensing, 66 (3), pp. 260–271

Yao, W., and U. Stilla, 2011a. Comparison of two methods for vehicle extraction from airborne lidar data toward motion analysis, IEEE Geosci. Remote Sens. Lett., 8(4), pp.607 -611

Yao, W., Hinz, S., Stilla, W., 2009. Object extraction based on 3d-segmentation of LiDAR data by combining mean shift with normalized cuts: two examples from urban areas. In: Proceedings of 2009 Joint Urban Remote Sensing Event (URBAN2009 - URS2009), Shanghai, China

You, R., Lin, B., 2011. A quality prediction method for building model reconstruction using lidar data and topographic maps, IEEE Trans. Geosci. Remote Sens., 49(9), pp.3471-3480

Yu, B., H. Liu, J. Wu, Y. Hu, L. Zhang, 2010. Automated derivation of urban building density information using airborne LiDAR data and object-based method, Landscape and Urban Planning, 98 (3–4) (2010), pp. 210–219

Yu, X., J. Hyyppä, M. Vastaranta, M. Holopainen, and R. Viitala, 2011. Predicting individual tree attributes from airborne laser point clouds based on the random forests technique, ISPRS Journal of Photogrammetry and Remote Sensing, 66(1):28–37

Zhang, J., Sohn, G. and Bredif, M., 2014. A hybrid framework for single tree detection from airborne laser scanning data: a case study in temperate mature coniferous forests in Ontario, Canada. ISPRS Journal of Photogrammetry and Remote Sensing, 98:44-57

Zhang, J. and Sohn, G., 2010. A Markov Random Field Model for Individual Tree Detection from Airborne Laser Scanning Data. Photogrammetric Computer Vision (PCV) 2010, September 1-3, Paris, France. International Archives of Photogrammetry and Remote Sensing, 38(3A):120-125.

Zhou N, Lipo W (2009) Class-dependant feature selection for face recognition. Adv Neuro-Inf Process Lecture Notes Comput Sci 5507, pp.551-558

Zhou, Z., 2012. Ensemble Methods: Foundations and Algorithms. Boca Raton, FL, USA: Chapman & Hall/CRC