Deriving Data Inputs for the Wildland Fire Dynamics Simulator

Report Prepared For:
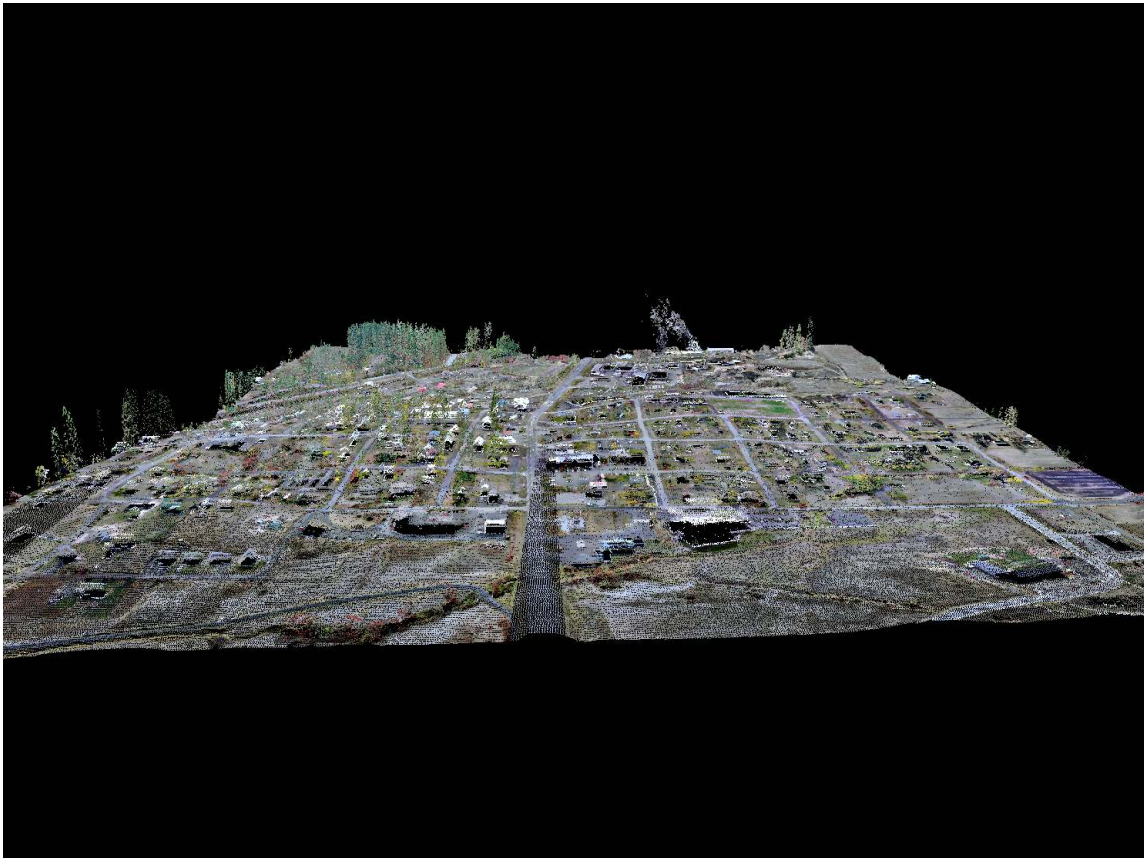
Report Prepared By:

Jason Trook, Joe Hu, Frank Roberts   **Derek McNamara**
Coeur d'Alene Tribe   **McNamara Consulting, Inc.**
PO Box 408, 850 A St.   **1112 N. 7th St.,**
Plummer, ID 83851   **Coeur d'Alene ID, 83814**

12/28/2009

Executive Summary

This document details the results of a three year study to assess the feasibility of using geospatial technologies to derive inputs for the Wildland Fire Dynamics Simulator (WFDS). This study was conducted as part of a grant (award #60NANB5D1204) from the National Institute of Standards and Technology (NIST) to the Coeur d'Alene Tribe. This grant was part of the NIST Fire Grants program. Specifically, this project focused on extracting the three-dimensional structure of man made and natural features from Light Detection and Ranging (LiDAR) data and multi-spectral imagery. A GIS based tool used to automate these feature extractions was also developed as part of this grant. The extraction of fire barrier features such as roads and parking lots from LiDAR and multi-spectral imagery using proprietary software were also assessed as part of the work conducting for this grant. Finally, this project resulted in the development of a GIS based tool to facilitate the transfer of WFDS inputs stored in GIS data sets to formats suitable to be run by WFDS.

Acknowledgements

Table of Contents

List of Tables

List of Figures

Appendix A: List of Presentations and Outreach Meetings

Appendix B: GIS LiDAR Tools User Documentation

Appendix C: List of Recorded Structural Attributes

Section 1.0 Introduction

This paper details work conducted, products produced and research results of a three year study conducted by the Coeur d'Alene Tribe and McNamara Consulting, Inc. This study was conducted through a three year grant (award #60NANB5D1204**)** to the Coeur d'Alene Tribe from the National Institute of Standards and Technology (NIST). The purpose of this study was to assess the feasibility of using geospatial technologies for deriving inputs into the NIST Building and Fire Research Laboratory's (BFRL) Wildland Fire Dynamics Simulator (WFDS). This assessment had five basic elements:

1. Provision of basic WFDS data inputs from already existing Geographic Information System (GIS) data for initial WFDS testing purposes.

2. Development of more extensive data sets across the Coeur d'Alene Tribe Reservation for more detailed testing of WFDS.

3. Research into various automated and semi-automated methods of deriving more complicated WFDS inputs from remotely sensed data and development of tools and automations to facilitate the implementation of these methods.

4. Development of prototype applications to transfer GIS data to WFDS input files.

5. Outreach to the geospatial and fire communities portraying results of this study.

The Coeur d'Alene Tribe GIS Program has a rich collection of geospatial data consisting of high resolution imagery, extensive GIS data sets portraying features across the Reservation and a state of the art GIS used to maintain, process, analyze and disseminate these data. Initial assessments began with providing basic topography and roads data to the BFRL and determination of methods to transfer this GIS data to WFDS input files. Through an initial analysis of WFDS requirements, WFDS inputs with potential for automated or semi-automated extraction from remotely sensed data were identified. It was determined that three features required some form of automation to aid in delineation and or extraction from remotely sensed data:

1. Building footprints with associated height data.

2. Tree stem locations with associated attributes of tree height, crown radius, crown base height and crown bulk density.

3. Fire barrier locations such as roads, parking lots and dirt patches.

Detailed below are the results of these assessments focusing on the examination and development of techniques to extract building, tree and fire barrier information from remotely sensed data. Early work on the project focused on the formulation of research directions appropriate for these types of extractions in the Wildland-Urban Interface (WUI). Feature extraction methodologies were also determined based on available data and tools as described below. It was determined that readily available tools could be used to extract item 3 above, but that adequate tools for the extractions of items 1 and 2 did not exist and custom development of tools was required. The specific algorithms implemented are described in sections 2 and 3 below with the functionality of the developed tools described in Appendix A.

There were a number of required WFDS inputs that were derived through previous work conducted by the Coeur d'Alene Tribe and these are briefly described in this section. In addition, a prototype application to transfer GIS data to a format suitable for WFDS is briefly described in section 5.0 with recommendations for the future use of geospatial technologies in regards to WFDS contained in section 6.0. Finally, Appendix A details a list of presentations and outreach meetings where work related to this project was described to interested parties.

1.1 Motivation

Destructive WUI fires have resulted in policy shifts in the United States, which put more emphasis on the WUI (Stewart et al., 2007[1]), with both the National Fire Plan (NFP) and the Healthy Forest Restoration Act (HFRA) providing focus on WUI mitigation (Stewart et. al., 2007[2]). This emphasis on the WUI has resulted in the need to map the WUI using GIS with a proliferation of these maps being observed (Stewart, et. al., 2007[2]). Despite the extensive use of GIS in WUI maps (e.g. Radeloff et al., 2005; Kamp and Sampson, 2005) these efforts have received criticism due to lack of precision and poor quality of input data (Stewart et al., 2007[2]). It has also been noted that disparity between WUI definitions has resulted in map discrepancies between and within political boundaries (Wilmer and Aplet, 2005). The problem is compounded in that many WUI treatments and regulations are based on limited scientific study (Mell, 2007) with, until recently, there being no physics-based wildland or WUI fire behavior models (Mell, 2008[1]). There are a number of semi-empirical wildland fire behavior models, but these do not model fires in the WUI (Evans et al., 2004[2]). Fire behavior models being developed at the BFRL are attempting to provide a scientific basis for WUI risk assessment and mitigation (Mell, 2008[1]). This could result in a more robust definition of the WUI leading to more consistent WUI mapping efforts, which could help to better protect lives and properties in these areas while focusing mitigation efforts in appropriate locations.

The WFDS is used to model WUI fires and is a modification of the BFRL Fire Dynamics Simulator (FDS) developed at NIST (Mell, 2008[2]). As described in McGrattan et al. (2009), FDS has the following characteristics:

- Computational Fluid Dynamics (CFD) model.

- The Navier-Stokes equations for low-speed, thermally-driven flow are solved numerically.

- There is an emphasis on smoke and heat transport from fires.

- A three-dimensional, rectilinear grid is used to update the approximate solution of the conservation equations of mass, momentum and energy.

- Thermal radiation is computed on the same grid as above.

- Smoke, sprinkler and fuel movement are simulated using Lagrangian particles.

BFRL has been developing FDS to predict fire spread in structures for more than 25 years, resulting in a freely distributed, well regarded and widely used model in the urban fire protection industry (Rehm et al., 2003). Modifications to FDS to include structure to structure fire spread as well as fire spread in continuous and discrete natural fuels (Rehm et al., 2003) have resulted in WFDS, the first of its kind physics-based wildland and WUI fire behavior model (Mell, 2008[1]). The prediction of fire spread in the WUI using WFDS is, however, complicated by the need to quantify the three-dimensional distribution of structures and vegetation (Evans et al., 2004[1]). Other detailed information required for WFDS includes local meteorology and the material properties of structures and vegetation (Rehm et al., 2003).   Geospatial technologies such as GIS and remote sensing allow for the creation of spatially explicit data sets of forest fuels and can enhance fire modeling (Perry, 1998). In addition, the processes of fire ignition, spread and effects are spatial in nature (He, 2007). Spatially explicit inventories of vegetation and structures across any extensive landscape, however, are labor intensive and costly (Hall et al., 2005). Remotely sensed data provides the ability to obtain spatially explicit data sets over large areas with the potential to more efficiently map ground fuels (Kean et al., 2001) and create inputs to WFDS.

While the basic requirements for WFDS inputs are understood, the determination of certain requirements for remote sensor characteristics such as optimal spatial and spectral resolution,

accuracy, and optimal sensor type(s) is in its infancy. This study represents an initial step to assess the feasibility of using remotely sensed data in conjunction with GIS for deriving inputs to WFDS. Methodologies for the extraction and derivation of WFDS inputs from remotely sensed data are examined focusing on the extraction of building, vegetative and fire barrier information as distinct entities but in an environment where the elements overlap (i.e., the WUI). In addition, the problem of transferring this GIS data to WFDS input files is examined.

1.2 Data Sources

The data sources used for this project come from three basic sources:

1. Existing GIS data.

2. Existing remotely sensed data.

3. Derivatives of the remotely sensed or GIS data.

Existing GIS data consists of ground surveys to determine the material properties and other emergency response information about structures as described in section 1.2.1. In addition, the Coeur d'Alene Tribe's GIS roads data set is used as an initial identification of fire barriers as described in section 1.2.2. Remotely sensed data comes from a combination of high-resolution active and passive sensors representing state of the art technology in sensors as well as common sensors used in many locales across the United States. The sensors used also provide imagery portraying man-made and natural features in vertical, horizontal and oblique directions, representing a unique combination of views for deriving spatially explicit WFDS inputs.

Remote sensors for extraction of spatially explicit data sets for the WFDS can fall into two general categories: passive and active sensors (Campbell, 1996). Passive remote sensors record the reflection of solar radiation or emitted energy from the earth's surface or objects on that surface (Campbell, 1996). Active remote sensors record energy emitted from the sensor and reflected back from the earth's surface or objects on that surface (Campbell, 1996). Different sensors provide variations in spatial, radiometric, spectral and temporal resolutions; data processing procedures and costs; and accuracy (Baltsavias and Gruen, 2003). Sensor data requirements are application dependent and can vary greatly (Baltsavias and Gruen, 2003). Many studies, however, have shown potential for automating the extraction of building footprint and tree stem locations from remotely sensed data (e.g. Weidener and Forstner, 1995; Brunn and Weidner, 1997; Mass, 1999; Popescu and Kini 2004). Most studies, however, tend to focus on structures and trees individually with few examining the extraction of these features concurrently in a WUI environment.

1.2.1 Structure Materials (NFPA 1144)

The data collection procedures for structures is based on the National Fire Protection Association 1144 Standard for the Protection of Life and Property from Wildfire (NFPA, 2008). This standard provides a methodology for assessing wildland fire ignition hazards around existing structures, residential developments, subdivisions and improved property or planned property improvement that will be located in a WUI environment. It also provides minimum requirements for new construction to reduce the potential of structure ignition from wildland fires. The Coeur d'Alene Tribe made modifications to this database to allow additional data to be collected specific for this project.
Data collection for structures began in October of 2001 and lasted until November of 2003. Structure locations were recorded by field technicians using ArcPad software loaded on a global positioning system (GPS) unit. ArcPad is a mobile mapping unit ideally suited for integrating GPS and GIS operations and a custom form was created to record attribute information associated

with each structure. Initial data collection focused on structures located within the combined extents of the Coeur d'Alene Tribe and Benewah County.

Twenty-eight attributes relating to structural characteristics were collected in the field. The key fields collected that related to the WFDS are: building roof type, building roofing material, and building siding material. The structures layer also contains 13 of the NFPA attributes: ROOFING, BLDGMAT, WATERSRC, ONSITEPRO, INGEGR, RDWIDTH, RDSURF, FIREACCESS, STREETSIGN, VEG, DEFSPACE, SLPSETBACK, and GASELEC. For more details of the attribute information for the structure layer, see Appendix C. Photos of structures were also collected, associating each structure by its structure ID.

The structure data was collected as part of a pre-fire survey. In order to run the Wildland-Urban Interface Fire Dynamics Simulator (WFDS), building siding material, building roof type, and building roof material are required as input data. The building information defines the surface fuel level, which combined with building feature class polygon and building digital surface model, will be inputted into the WFDS fire model simulator.

Another use of structure data is validation of the WFDS fire model simulator results. Structures data was collected after the fire and comparing the WFDS fire simulator results with the post-fire structure data will verify the correctness and accurateness of the fire model simulator. Furthermore, it can improve the algorithm of fire model. Data from structures might help to answer some question about the fire behavior and it can be used to further improve the WFDS fire model. For further examples of how the Coeur d'Alene Tribe has incorporated structures into its applications please review an article published on the Tribe in ESRI's Measuring Up press book (ESRI 2004).

1.2.2 Roads GIS

The road GIS layer is a compilation of data from various sources including, but not limited to, the United States Forest Service (USFS), Bureau of Land Management (BLM), University of Idaho, Idaho Department of Lands, and the Coeur d'Alene Tribe. All data was gathered at a scale of 1:24,000 and accuracy of the road layer is limited to the collective accuracy of the original data source. The Kootenai County roads data set was used for additional updates. Data was also improved by updating road names and by assigning address ranges to roads located in Benewah County.

Despite updates to the roads layer occurring on a regular basis recent road changes may not be reflected in the dataset. Also, there are some roads that do not have attribute information (e.g., a street name) and some roads are not spatially accurate (e.g., they do not line up with roads present on aerial imagery).

The road layer was used mainly for two purposes. First it was pre-loaded into ArcPad to help ground crews easily locate themselves when they were collecting the structure field data. The road layer typically has a street name, which will facilitate the field data collection by allowing people to identify or verify the home address. Another use of roads layer is to validate the road fire barrier generated by feature extraction tool. For instance, some streets were blocked by trees canopy coverage. The feature extraction tool might miss those roads because they are obscured by tree coverage and the road layer is a good reference data source to correct these kinds of problems.

1.2.3 NAIP

The 2004 National Agriculture Imagery Program (NAIP) imagery for the state of Idaho was used in object oriented classifications to derive a vegetation mask used in the building extraction process as well as the derivation of fire barrier features as described in section 4.0. The 2004 NAIP imagery was acquired with the Leica ADS30 digital sensor, a passive sensor; characteristics for this imagery are described in Table 1. This type of imagery is pertinent to the examination of feature extraction methodologies for the derivation of WFDS inputs due to its availability in many locations throughout the United States. The NAIP Program goals consist of providing new imagery on an annual basis with ortho-imagery acquired on a 1 to 5 year cycle (NAIP, 2002). In addition, the NAIP Program is beginning to offer a near-infrared band as an extra option, which could further aid in feature extractions to produce WFDS inputs (NAIP, 2008)[1].

Table 1 NAIP 2004 specifications.

NAIP Specifications	
Flight Height	23,000 feet
Capture Season	Full Agricultural Growth
Cloud Cover	10%
Raw Data Spatial Resolution	0.9 meters
Final Product Spatial Resolution	1 meter
Spectral Resolution	3 color bands: Optional infrared band
Radiometric Resolution	8 bit
Horizontal Accuracy	+/- 3 meters of reference digital ortho quarter quads (DOQQs)
Orthorectified	Yes (10m or 30m USGS DEM)

1.2.4 LiDAR

Light Detection and Ranging (LiDAR) data flown in February 2005 by Horizons, Inc. was used in both the building and tree extraction processes. This LiDAR data, obtained across the Coeur d'Alene Tribe Reservation, was acquired with a LH Systems ALS50 LiDAR System, an active sensor, with characteristics shown in Table 2. LiDAR data has shown an increased use over the past 15 years due to improvements in sensors, global positioning systems (GPS) and inertial mapping systems (IMS). In addition, LiDAR has become the standard for high resolution topographic mapping over large extents. There are currently no nationwide programs to acquire LiDAR data, however, a National LiDAR Initiative (Stroker et al., 2007) is being discussed with several statewide efforts being completed or in the planning stages (e.g. Pennsylvania, Iowa and Oregon).

LiDAR data provides a point cloud of data where each point represents a return from a surface feature with horizontal and vertical coordinates. In addition, LiDAR provides the signal strength of each return pulse as an 8 bit value. Finally, most LiDAR data provides information about the return number and number of returns in each laser pulse, the scan angle, edge of flight line information and GPS time. A binary public file format for the storage and exchange of this LiDAR point data, termed the LAS file format, has been created. This LAS Format Standard is maintained by the American Society for Photogrammetry and Remote Sensing (ASPRS) Standards Committee. The LAS format used for raw data processed and analyzed for this project

---

[1] A near-infrared band was available for use with this project but due to the infrared lens being pointed off nadir, taller features such as trees and buildings were shifted and this imagery was, consequently, not used in this project.

was 1.0. New LAS formats have been developed since the acquisition of LiDAR data used in this project.

1.2.5 Pictometry

The final source of remotely sensed data comes from both nadir and oblique multispectral imagery. Pictometry, Inc. acquired this imagery in the fall of 2008 across the Coeur d'Alene Tribe Reservation with characteristics shown in Table 3. Many locales across the country have Pictometry imagery available. This imagery, however, is proprietary but is usually available for purchase. Both nadir and oblique feature extractions are possible with this imagery, but for this project it was used to aid in the accuracy assessments for the building and tree extractions. This imagery was acquired after much of the processing procedures had been developed and analysis conducted for this project so WFDS input data sets were not produced from this imagery.

The oblique characteristics of this imagery in conjunction with proprietary tools developed by Pictometry, Inc., however, could allow for the manual measurements of vertical characteristics of building and vegetation. It is, consequently, believed this imagery has large potential for deriving WFDS inputs in absence of other vertical information such as LiDAR but would require more manual intervention. When vertical information is available, the Pictometry, Inc. imagery can be used to assess the accuracy of automated feature extractions and other exploratory data analysis.

Table 2 LiDAR data specifications.

LiDAR Specifications	
Flight Height (Above Mean Terrain)	6,000 feet
Field of View	25 degrees
Collection Mode	3+3
Number of Flight Lines	82
Number of Flight Line Miles	1,712
Swath Width	2,660 feet
Line Spacing	1,862 feet
Maximum Along Track Spacing	1.8 meters
Maximum Cross Track Spacing	2.6 meters
Nominal Post Spacing	2.0 meters
Number of Basestation Locations	1
Estimated Horizontal Error (1 sigma)	0.21 meters Nadir; 0.22 meters FOV
Estimated Vertical Error (1 sigma)	0.15 meters Nadir & FOV

Table 3 Pictometry imagery specifications.

Pictometry Specifications	
Flight Height	Varies
Capture Season	Fall
Cloud Cover	Fly Below Clouds
Nadir Product Spatial Resolution	0.5 to 1 foot
Oblique Spatial Resolution	0.5 to 1 foot
Spectral Resolution	3 color bands
Radiometric Resolution	8 bit
Horizontal Accuracy	Unknown
Nadir Imagery Ortho-rectified	Yes (LiDAR 1m DEM)

1.3 Processing Tools and Initial Products

Any remote sensing analysis requires certain manipulations and combinations of various image sources to make them suitable for further analysis. This section briefly describes the tools used in the building, tree and fire barrier feature extractions and the derivation of certain data inputs required for these extractions. Other derived products specific to the particular feature extraction being discussed are detailed in the respective section. There are many software packages available for this type of work where algorithms and processes employed in most commercial software packages are proprietary with specific implementations not generally being known by users.

This project utilized ERDAS Imagine 9.0 for the geo-rectification and mosaicing of images and combinations of different bands of data into multi-band images. ENVI 6.3 was employed to derive textural measures of occurrence. LiDAR digital surface and elevation models (DSM and DEM) were created using ArcGIS software and custom scripts and tools. LP360 was used for editing LiDAR point clouds, analysis, visualization, and display. Finally, Feature Analyst was used to perform various object oriented image classifications.

1.3.1 LiDAR Bare Earth DEM Generation

The first step in most LiDAR analysis projects is the segmentation of ground points from the LiDAR point cloud. This study used an open source implementation of the Multiscale Curvature Classification (Evans and Hudak, 2007) developed in the Arc Macro Language (AML) for ESRI's workstation Arc/Info. LiDAR point clouds in LAS format were converted to Arc/Info coverages and the above AML was run. This AML leaves artifacts from buildings and other man-made objects that were manually removed. The resulting ground points where then used to create a triangulated irregular network (TIN), which was converted to a 1 meter DEM raster data set using ArcGIS software. In addition, an initial DSM was created by interpolating the first return LiDAR points in the same manner as the ground points described above.

1.3.2 NAIP Geo-rectification

The NAIP imagery was geo-rectified to the LiDAR DSM. A comparison of Table 1 and Table 2 shows the difference in horizontal accuracy between the NAIP and LiDAR data. The raw LiDAR point data is purported to have a better horizontal accuracy compared to the NAIP. The created digital surface products have, however, not been assessed for horizontal or vertical accuracy and are likely less accurate than the raw point data but it is believed the horizontal accuracy of the interpolated LiDAR points would have a horizontal accuracy greater than the NAIP imagery. There are, consequently, obvious shifts in features between the NAIP and LiDAR imagery. Any fusion technique combining the NAIP and LiDAR requires proper alignment between the two image sources[2]. This was achieved by using the ERDAS Imagine Raster/Geometric Correction tools. Building corners were identified and used as the ground control points between the two sources. Building corners were identified in both images by performing a variance texture analysis of the LiDAR first return DSM and the NAIP in ERDAS Imagine. The rectification was completed with a Root Mean Square Error (RMSE) of 0.5 meters. It should also be noted that this geo-rectification does not improve errors common in digital imagery, which might distort taller objects such as buildings or trees.

---

[2] Having the imagery of interest ortho-rectified to the LiDAR data will alleviate misalignments between the two image sources and is the preferred method.

14

1.3.3 LiDAR Processing Core Library

The raw point cloud of LiDAR data was converted to grid or raster products as described above. The algorithms to extract building and tree information described in subsequent sections use these LiDAR derivatives as ancillary data in the feature extraction processes but the main algorithms operate directly on the LiDAR point cloud of data. A single LiDAR tile can contain several million points or more with processing algorithms requiring many iterative procedures. It is, consequently, a requirement to have an efficient means of point cloud access. Ordered binary trees are a classic data structure used to store and access data efficiently. Binary trees are characterized by each node in the tree having at most two branches, the left and the right branch, where the node can have only one branch or no branches, in which case it is called a leaf. The algorithms developed for this project utilize binary tree data structures implemented in a C#.NET library developed by McNamara Consulting, Inc. with custom search geometries to very quickly iterate over the point cloud of data. These libraries also contain methods to read and write to LiDAR data stored in LAS format 1.0. The LiDAR processing tools developed as part of this project, using the above library, are written in VB.NET and run in a windows environment. The feature extraction algorithms make use of certain GIS operations contained in the ArcGIS software and utilize the COM-based library known as ArcObjects. There is a user manual describing the functionality of these tools contained in Appendix B.

Section 2.0 Building Feature Extraction in the WUI

Abstract

Building information is used in many fields including planning, telecommunications and environmental sciences. There is also an emerging use of building information in Wildland-Urban Interface (WUI) assessments with a need for efficient and accurate tools and algorithms to extract building information from remotely sensed data. This section presents an automatic approach for the segmentation of above ground LiDAR points to buildings and the subsequent extraction of building footprints from the segmented point cloud. This work is conducted in the context of deriving inputs for the Wildland Fire Dynamics Simulator (WFDS). The point segmentation algorithm incorporates a landcover classification from a fusion of LiDAR and multi-spectral imagery as a mask in a unique manner with a custom laser pulse algorithm employed to remove vegetation points. A plane fitting approach is then used to segment the LiDAR point cloud to building and non-building points. The accuracy of the segmentation algorithm is assessed in a study area encompassing 449 structures in Worley, Idaho. Two methods are then examined for building boundary determination (i.e., building footprint extraction) from the segmented point cloud: a raster based method and a point based method. In addition, the accuracy of LiDAR measured building corner heights and roof apexes are compared against field measurements for 24 buildings.

The segmentation algorithm achieves an overall accuracy of 97% for distinguishing between building and non-building points given an accurate bare-earth digital elevation model. The point based method achieved producer's and consumer's accuracies of 76% and 85%, respectively, for building boundary determination. The raster based method achieved producer's and consumer's accuracies of 81% and 72%, respectively, for building boundary determination. The root mean square error (RMSE) between the LiDAR measured building corners and field measured building corners was 0.25 meters with the means showing a tendency for the LiDAR to overestimate building corner height. The RMSE between the LiDAR measured building maximum heights and field measured building maximum heights was 0.89 meters with the means showing a tendency for the LiDAR to overestimate building maximum height. The poorer results for maximum building height comparisons might be attributed to manual measurement error. Overall the methods show excellent potential for deriving the three dimension distribution of buildings in context of producing WFDS inputs. Furthermore, results can easily be improved using manual interaction and the developed tools to increase the accuracy of the building footprints. The use of LiDAR and multispectral imagery for deriving the horizontal and vertical extent of buildings for inputs to WFDS is likely the only practical method over any large area.

2.1 Previous Work

This study stems from the need for the efficient and accurate quantification of the three dimensional distribution of structures for inputs into WFDS, with the mapping of WUI structures having the additional benefit of aiding in the creation of policy maps and visual images (Stewart et al., 2007[2]). Housing density has also been shown to be one of the most sensitive parameters in evaluation of the national WUI (Stewart et al., 2007[1]). A key component of the correct delineation of the WUI is, consequently, the accurate delineation of building information. WFDS takes as inputs the horizontal and vertical extent of buildings, building siding and roofing material, and building vents such as chimneys, doors and windows. Remote sensing provides an efficient means of deriving some of these types of building information (Haithcoat et al., 2001) with active research occurring in this area for decades (Brenner, 2005). This section focuses on the extraction of the horizontal and vertical extent of buildings from remotely sensed data.

The extraction of building information from passive remote sensors has historically focused on the use of high-resolution aerial imagery (Huertas and Nevatia, 1986; Irvin and McKeown 1989) and more recently satellite imagery (Sohn et al., 2005; Shan and Lee 2005) with less attention being paid to hyperspectral data (e.g. Huertas et al., 1999) and thermal remote sensing data (e.g. Trujullo et al., 2005). Early efforts from these passive remote sensors occurred using manual methods from aerial image sources (Mayunga et al., 2005). These manual methods are labor and cost intensive with large area extraction not being feasible (Ruther et al., 2002) and have led to many attempts to automate building extraction from remotely sensed imagery (e.g. Woo et. al., 2008; Mayunga et. al., 2005; and Gulch, 2000). Due to difficulties with extraction of 3D building structures from a single 2D image (Pu, 2007) later work focused on a multiple-view approach (Fradkin et al., 1999; Noronha and Nevatia, 2001) where the use of image produced digital terrain models (DTM) was incorporated. Recently, high-resolution satellite imagery such as quickbird (Mayunga et. al., 2005) and IKONOS (Sohn and Dowman, 2001) has received attention. Regardless of the sensor used, approaches can involve image segmentation (Zimmermann, 2000), image transformations such as the 2D Hough Transform (Wei et al., 2004), other morphological operators (Baltsavias et al., 1995), the fitting of object models (Yi-Hsing and Wang, 2001) and other creative approaches or combination of approaches (Muller and Zaum, 2005). Despite the extensive research in this area, automated processing of building information from aerial imagery is still at a very early stage (Brenner, 2005). There are many reasons for this including variations in image type, scale and resolution (Wang and Tseng, 2003); variation in environments where no single approach is best (Gruen, 2000); and issues with shadows, occlusions, building lean and between scene variations.

The use of active sensors has also been examined for the extraction of building information. Interferometric Synthetic Aperture Radar (IFSAR) has been used in conjunction with other imagery to aid in building detection (Huertas et al., 1998). LiDAR data, however, has been shown to provide better shape delineation of buildings over IFSAR data due to higher resolution and lack of shadowing/layover effects (Gamba and Hoshmand, 2000). During the 1990's LiDAR data became available with improvements in the sensors and geo-referencing capabilities resulting in sufficient accuracy for various topographic mapping purposes (Brenner, 2005). LiDAR technology has been a mature technology for a decade with many companies offering various services (Baltsavias, 1999) and the technology is well established for the generation of highly accurate digital terrain models (Hollaus and Wagner, 2006). In addition, the ability to extract building information has been demonstrated by a number of studies listed below. LiDAR building extraction approaches fall into two general categories: extraction from the interpolated point cloud of data or DSM (e.g. Rottensteiner and Briese, 2002) or use of the raw point cloud of LiDAR data (Sampath and Shan, 2007).

Raster based methods turn irregularly distributed laser scanning data into grid form to reduce data requirements and enhance speed of processing (Cho et al., 2004). Raster methods for building extraction typically begin with segmentation of ground and non-ground points where the ground and non-ground points are interpolated to grids (i.e., DEM and DSM, respectively) that are differenced with building height thresholds employed (e.g. Al-Harthy and Bethel, 2002). This initial segmentation can be further expanded to remove vegetation pixels in a variety of approaches such as the use of surface roughness (Haithcoat et al., 2001), edge detectors such as Laplacian edge detectors combined with shape information (Wang, 1998), curvature based methods (Rottensteiner and Briese, 2002), the use of shape cues in a Bayesian network (Brunn and Weidner, 1997) or the identification of planar patches (Gamba et al., 2005). Traditional pixel based classifiers have also been used such as Mass (1998), who incorporated several texture measures of a LiDAR derived DSM into a maximum likelihood classification. Additionally, object oriented image classification and segmentation techniques have been used, which account for the contextual information surrounding each pixel, in the extraction of building information

17

(Ibrahim, 2005). The use of a TIN model has also been used for building boundary determination where a connected component analysis of the TIN model was used to segment individual buildings (Morgan and Habib, 2002).

Raster based methods suffer from the introduction of errors during the interpolation process (Cho et al., 2004) and have difficulty in distinguishing vegetation from buildings (McNamara, 2006). In order to overcome this introduction of error and better able segmentation of vegetation from buildings, the extraction of building information directly from the LiDAR point cloud has been attempted (McNamara, 2008; Wang and Tseng, 2004). As with raster based methods, an initial segmentation to ground/non-ground points is often performed. Additional segmentations to differentiate buildings versus other features such as vegetation have been conducted by extending the well known Hough Transform to 3D (Vosselman and Dijkman, 2001). Filin (2002) used a surface clustering technique to identify homogenous areas in the LIDAR point cloud. In addition, the high point densities achievable with LiDAR allow for the detection of planar roof faces in LiDAR point clouds (Vosselman and Dijkman, 2001). Least squares regression methods are common for fitting planar surfaces to the 3D point cloud (e.g. Zeng, 2008) with Engel et al., (2008) using the well known RANSAC technique for fitting surface planes and Wang and Tseng (2004) using an octree data structure for efficient data processing. Vosselman and Dijkman (2001) used building ground plans to aid in the building boundary determination from the segmented points. Sampath and Shan (2007) used a modified convex hull approach to trace the boundary of a grouped set of segmented LiDAR building points. A side ratio constraint based boundary tracing algorithm that utilizes a TIN model was used by Huang et al. (2008).

Integration of LiDAR and passive optical sensors has also been utilized to extract building information due to the complimentary nature of the two technologies (Brenner, 2005). Aerial images are used to add additional planar patches to LiDAR planar patch segmentation and improve the geometric quality of extracted buildings by matching LiDAR and image edges (Rottensteiner and Briese, 2003). Novacheva (2008) used planar patches found from LiDAR data in conjunction with color edge detection of aerial images to reconstruct building roofs. The use of LiDAR and aerial imagery edge detections is combined in Yong and Huayi (2008) where the missed building edges in LiDAR data are corrected with aerial imagery edges and the occluded edges in aerial imagery are corrected with LiDAR data.

2.2 Point Cloud Segmentation

This study used the approach of segmenting LiDAR building points where the procedures and algorithms utilized are displayed diagrammatically in Figure 1. Initial steps segment points unlikely to be buildings based on characteristics of the point under consideration and spatial relationships with ancillary data. A plane fitting approach is then employed. A bare earth DEM, which for this study was produced as described in section 1.3.1, and an input binary grid vegetation mask are input to the segmentation procedure.

As described above, general methods for segmentation of LiDAR point clouds to buildings are typically either grid or point based. Comparisons between grid and point based methods are not made for the point cloud segmentation procedures. This is because most, if not all, grid based methods result in some data loss due to aggregation, which can be particularly pronounced in buildings surrounded by vegetation. Point based methods have a greater probability of classifying building points surrounded by vegetation, a key component in WUI assessments as well as potential to identify other sources of building segmentation error described in Cheuk and Yuan (2009). Furthermore, initial research for this project as described in McNamara (2006) determined that raster based methods suffer from the problem of not identifying buildings underneath or surrounded by vegetation. This situation essentially defines the WUI and it was,

therefore, concluded that point based segmentation methods have better potential for identifying buildings surrounded by vegetation compared to raster based segmentation methods.

2.2.1 Height and Last Return Filter

The LiDAR building filter begins with a segmentation of the input LiDAR point cloud to ground and non-ground points. This is accomplished by converting the point cloud to an ArcInfo coverage data model and running the ground segmentation AML. A bare earth DEM is then created from the segmented ground points by first creating a TIN and converting this TIN to a raster data set. All subsequent steps are run on an in-memory LiDAR point cloud. The first pre-processing step prepares the input point cloud of data to only examine the last returns as possible building points. It has been shown that the use of the last return in building extraction can help to improve results (Matlkalnen, 2009). It is, however, the case that the use of the last return might in some cases miss building edges and decrease building segmentation accuracy (Matlkalnen, 2009). The first return of each two return pulse is, consequently, included in the initial filter to include possible returns from building edges. The next step is to remove LiDAR points with above ground elevation values above and below user input thresholds, 25 meters for maximum height and 2 meters for minimum height, respectively. This filtering step is conducted by determining the bare earth DEM pixel value on which each point in the input LiDAR point cloud occurs. The height value of the DEM pixel value is then subtracted from the height value of the particular LiDAR point under consideration. The point is then removed from consideration if it is above or below the input thresholds.
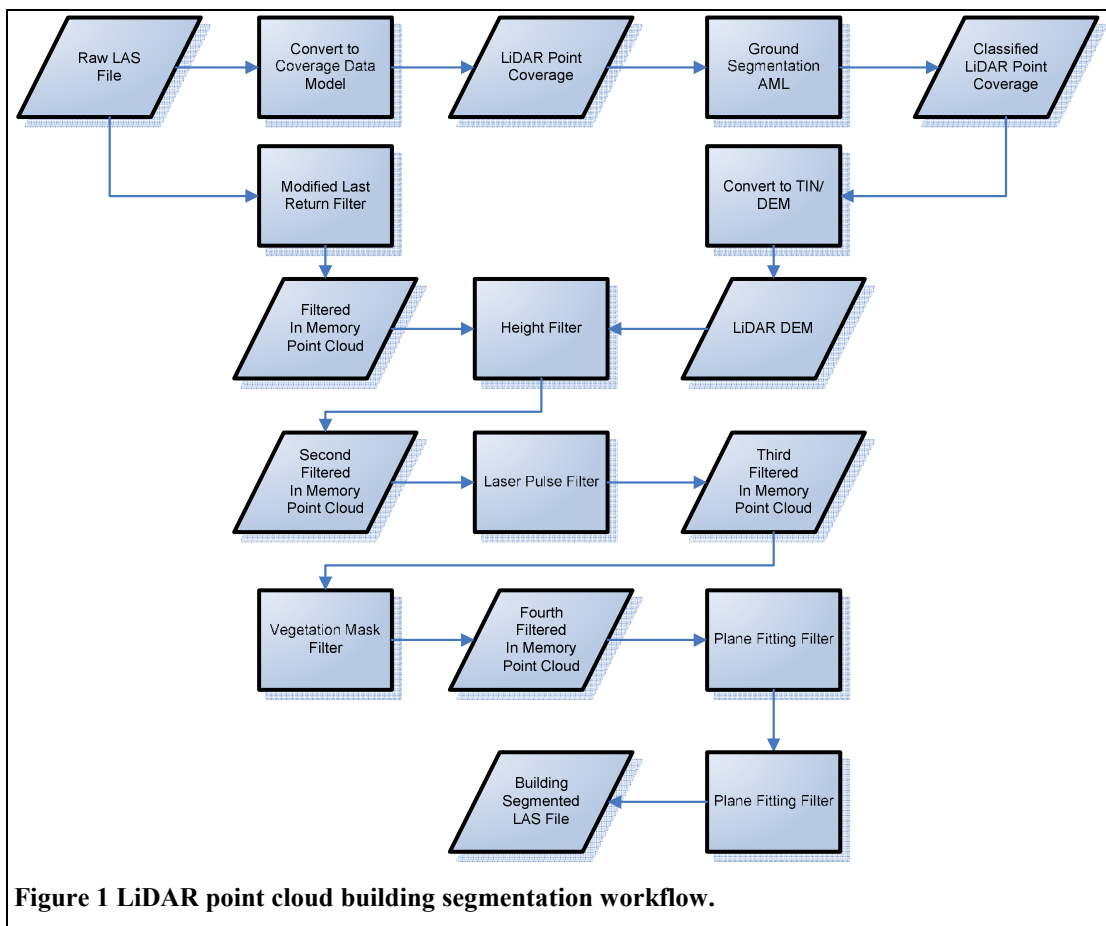
Figure 1 LiDAR point cloud building segmentation workflow.

2.2.2 Laser Pulse Filter

The next filter is used to filter first return of two return pulses that are likely not to be from a building edge. This filter considers the relationship between the first and second return in a particular laser pulse and the same first return and ground elevation as determined by an input bare earth DEM. The distance between the first and last return in a particular laser pulse is determined as well as the distance between the same first return and the ground elevation, as portrayed in the input bare earth DEM, for which this first return falls. If the difference between these two distances is greater than a threshold, representing the minimum building height as input by the user, all returns in the particular laser pulse are classified as not being a building with the exception of the last return, which could be a building covered by vegetation. This initial segmentation takes advantage of the fact that a LiDAR first return pulse when hitting dense vegetation, will not always have a last return which hits the ground. This filter does, however, in some cases filter out building points that represent edges of multi-storied buildings. This should not adversely affect the boundary determination as these points are typically interior building points. The above described method does require that the LiDAR points be organized in the order in which they were obtained from the sensor.

Figure 2 Above ground vegetation extracted from NAIP imagery and LiDAR texture measure.

2.2.3 Vegetation Mask Filter

The next filtering step involves the use of a binary vegetation grid where pixel values of 1 represent pixels classified as vegetation and pixel values of 0 represents pixels classified as non-vegetation. This filtering step removes those first return points that occur on pixels classified as vegetation. Only the first return points are removed to allow for other returns that might represent buildings underneath vegetation. This vegetation grid can be created using many approaches. For example, if color-infrared imagery is available, the use of the Normalized Difference Vegetation Index (NDVI) can be used to show patterns of vegetation (Burgan, 1996). In addition, color imagery such as the NAIP imagery can be used to distinguish vegetation from man-made objects (Walker and Briggs, 2005). Coupling this vegetation mask filter with the plane fitting described below might help to overcome spectral similarities between rooftops and roads, a common issue with extracting buildings from aerial imagery (Xie et. al., 2006), while maintaining the ability to identify buildings underneath vegetation.

For this project, an object oriented image classification approach using the Feature Analyst extension for ArcGIS was employed. Object oriented image classifications take into account

individual characteristics of each pixel as well as neighboring pixels (Walker and Briggs, 2005). The inputs to this classification were the NAIP color imagery with a variance texture measure of the LiDAR first return DSM. The three color bands of the NAIP imagery were input as spectral bands and the texture measure of the LiDAR first return DSM was input as a texture band. The "learning approach 1" method was used with the feature to extract being chosen as "tree or shrub". Several training sites were "heads-up" digitized off the NAIP imagery and the feature extraction process was run. After the first run the machine learning capabilities of Feature Analyst were utilized to remove incorrectly classified features and include missed features. Several cleanup passes were performed with training examples for incorrectly classified features and missed features "heads-up" digitized. The final result is shown in **Error! Reference source not found.** and was converted to a binary grid to be used as the input vegetation mask filter. A similar approach was taken for extracting fire barrier feature types (see Section 4).

2.2.4 Plane Fitting

After the above algorithms are run, the segmentation of the LiDAR point cloud to building and non-building points occurs by iterating through each point in the point cloud, not removed by the above procedures. A window is fit around each point with dimensions for the window initially input as the average point spacing found in the point cloud and adjusted interactively to determine optimal parameter values. This study used a moving window with side lengths of 3 meters. If there are enough points in the window (i.e., more than 3) the least-squares solution to the following equation of a plane is solved:

$$Ax + By + C = z \qquad\qquad \textbf{(1)}$$

where

       x = the X coordinate of the point,

       y = the Y coordinate of the point, and

       z = the Z coordinate of the point.

The above equation is solved by minimizing the sum of the residuals where residuals are calculated as the vertical distance of each data point to the determined plane. The slope of the determined plane is crudely examined where if either the *A* or *B* coefficients in equation 1 are greater than an input threshold the, points are not classified as buildings. A slope threshold of 0.6 (i.e., 60%) was used in this study. The equation is then evaluated for goodness of fit based on the coefficient of determination (R^2). This is calculated by first determining the total sum of squares (SS_{tot}) as follows:

$$\sum_{i=1}^{n}(z_i - z_{hat})^2 \qquad\qquad \textbf{(2)}$$

where

       z_i = the actual height value of a particular point based on equation 1and

       Z_{hat} = the mean height value of all the points.

The residual sum of squares (SS_{reg}) is then calculated as follows:

$$\sum_{i=1}^{n}(z_{pred} - z_{hat})^2 \qquad\qquad \textbf{(3)}$$

where

       Z_{pred} = the predicted height value of a particular point based on equation 1and

       Z_{hat} = the mean height value of all the points.

The R^2 is finally calculated as shown in equation 4:

$$R^2 = \frac{SS_{\text{reg}}}{SS_{\text{tot}}}.$$

(4)

If the R^2 is above an input threshold, a region growing process occurs. This study used a threshold of 0.94. The R^2, however, is not an appropriate indicator of goodness of fit for an equation of the plane when the slope is 0 (i.e. flat). For flat planes the R^2 does not predict any of the variability in the vertical direction. In order to account for this, when the values of *A* and *B* in equation 1 are less than a threshold (i.e., roofs with flat or relatively flat slopes) the R^2 is not used as the indicator of goodness of fit. Instead the standard deviation of the residuals in the Z direction is used to evaluate goodness of fit where, if this value is less than a threshold, the region growing process occurs. A threshold of 0.2 meters was used in this study.

The region growing process begins by expanding the moving window, initially in all directions, with the expansion value being equal to half the length of a side of the initial moving window. The Z distance of each point in the window to the fitted plane is calculated as follows:

|[(Point Height Value) – (Predicted Height Value from Equation 1)]| **(5)**

Outliers are removed if the value calculated in equation 5 is greater than some threshold. This study used a threshold of 0.2 meters. If no new points are added, the points in the original window are classified and the next point in the iteration is considered. If new points are added, the direction(s) in which these points occur is recorded. A new plane is fit to the points which are not removed above. The window is expanded again, only in the direction for which new points are added as shown in Figure 3 and the process continues until no new points are added. The next point in the iteration is considered and the process continues until all points, not removed in the initial filters, have been evaluated. The algorithm concludes with a re-examination of the LiDAR point cloud. If points within the input

Figure 3 Directional window example portraying expansion in only two directions.

above ground height threshold (i.e., 2 and 25 meters) are not classified as building, the other above ground points within a threshold horizontal and vertical distance, in this study 1.5m horizontal and 0.2m vertical, are examined. If the percentage of these points classified as buildings is greater than the input threshold, 50% in this study, the point is classified as a building. This last step ensures that returns from off-roof features, building sides and building edges below the roof plane are classified as buildings.

2.3 Building Boundary Determination

Two methods to delineate the building boundaries are examined. Both methods utilize the segmented LiDAR building points created using the methods described in section 2.2. One

method utilizes the LiDAR segmented building points directly and the other method works off of an interpolation of the segmented building points and the segmented ground points. Accuracy assessments are run on both of these methods using an approach similar to that described by Song and Haithcoat (2005) where ten indexes are described. Three of these indexes: shape similarity, corner difference and root mean square error, are not used in this study. These indexes are not examined because these indexes examine relationships between reference building corners and extracted building corners. Since no orthogonalization of the extracted building footprints was conducted, these measures are not appropriate. The seven indexes used in the building footprint accuracy assessments are presented in Table 4.

Table 4 Building footprint extraction accuracy assessment indexes.

Index	Calculation	Description
Detection Rate	$(6)\ \dfrac{\sum \text{Reference FP Interesected By Extracted FP}}{\sum \text{Digitized Reference FP}}$ [3]	This measure is the producer's accuracy and shows the percentage of extracted footprints coinciding with reference footprints.
Correctness	$(7)\ \dfrac{\sum \text{Extracted FP Intersected By Reference FP}}{\sum \text{Extracted FP}}$	This measure is the consumer's accuracy and can be used as an estimate of the reliability of the extracted footprints.
Averaged Matched Overlay	$(8)\ \dfrac{\sum\limits_{i}^{\text{Re }ferenceFP}\left(\dfrac{\text{Overlapping FP Area}}{\text{Reference FP Area}}\right)}{\sum \text{Correctly Extracted FP}}$	This measure represents the area in common between the traced and reference footprint, averaged across all correctly extracted footprints.
Average Area Omission Error	$(9)\ \dfrac{\sum\limits_{i}^{\text{Re }ferenceFP}\left(\dfrac{\text{Excluded Reference FP Area}}{\text{Reference FP Area}}\right)}{\sum \text{Correctly Extracted FP}}$	This measure represents the area of the reference footprint not delineated in the extracted footprint, averaged across all correctly extracted footprints.
Average Area Commission Error	$(10)\ \dfrac{\sum\limits_{i}^{\text{Re }ferenceFP}\left(\dfrac{\text{Falsely Extracted FP Area}}{\text{Reference FP Area}}\right)}{\sum \text{Correctly Extracted FP}}$	This measure represents the area in the extracted footprint that is outside the reference footprint, averaged across all correctly extracted footprints.
Average Area Difference	(11) $\dfrac{\sum\limits_{i}^{\text{Re }ferenceFP}\left(\left|\dfrac{\text{Extracted FP Area - Reference FP Area}}{\text{Reference FP Area}}\right|\right)}{\sum \text{Correctly Extracted FP}}$	This measure represents the absolute value of the difference in area between the extracted and reference footprint, averaged across all correctly extracted footprints.
Average Perimeter Difference	(12) $\dfrac{\sum\limits_{i}^{\text{Re }ferenceFP}\left(\left|\dfrac{\text{Extracted FP Length - Reference FP Length}}{\text{Reference FP Length}}\right|\right)}{\sum \text{Correctly Traced FP}}$	This measure represents the absolute value of the difference in perimeters between the extracted and reference footprint, averaged across all correctly extracted footprints.

---

[3] FP signifies building footprint.

2.3.1 Point Grouping and Boundary Tracing

The point grouping and boundary tracing approach is from Sampath and Shan (2007). The first step is to group segmented building points representing a single building into a distinct set. This is accomplished by using another region growing algorithm that begins by fitting a window around each segmented building point. Segmented building points in each window have an additional window fit around them and the process continues until no new segmented building points are found. The window dimensions are again dependent on the point spacing of the input LiDAR data where the window should have a length and width equal to about twice the point spacing. This study used a square window with side lengths of 4 meters. If the grouped set of points contains 5 or more points the boundary tracing begins.
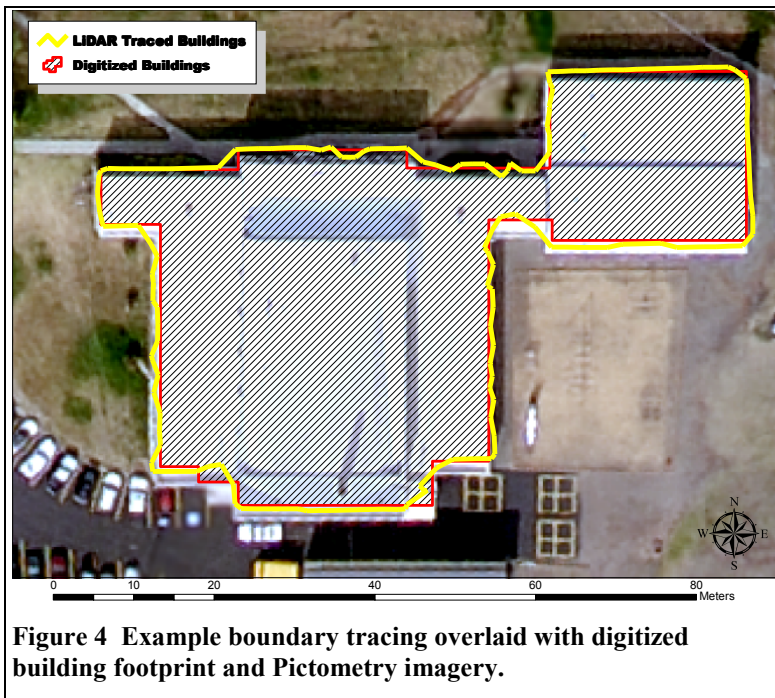
Figure 4 Example boundary tracing overlaid with digitized building footprint and Pictometry imagery.

The boundary tracing algorithm proceeds by starting with the left most point in the group of points. A window is fit around this point where the window size should be slightly larger than the point spacing of the input LAS file. This study used a square window with side lengths of 5 meters. The building segmented points in the first window are determined and line segments are formed between the initial point and every other point in the moving window. For the initial window, the angle between the vertical axis and each line segment formed above is calculated. These angles are sorted in increasing order where the point with the lowest angle is determined as the next boundary point. The same sized window is then fit around the new boundary point and segmented building points are identified. Line segments are again formed between the center point and every other point in the moving window. The line segment formed from the first boundary point to the second boundary point is also determined. The angle between this line segment and every other line segment is determined and these angles are sorted in increasing order where the point with the lowest angle is determined as the next boundary point. Each line segment determined as a boundary segment must also not intersect an already formed boundary line segment. The process continues until the original point is found as the next boundary point, thereby resulting in a traced outline of the building boundary as shown in Figure 4.

The traced footprints produce a line GIS data set that is converted to a polygon. These polygons are further filtered using an area threshold. In this study building polygons with an area less than 10 meters square (m^2) were removed from the data set. A value of 10 m^2 was chosen due to the resolution of the LiDAR data set used for this study. While the nominal post spacing of the LiDAR data is 2.0 meters, due to large overlap the post spacing is actually higher than this in

most locations and closer to 1.5 meters. Buildings, less than 9 to 10 square meters (i.e., 3 meters by 3 meters) might, therefore, often not have enough points to describe the equation of the roof plane. These small buildings, consequently, can be considered below the resolution of the LiDAR data.

2.3.2 Raster Based Method

The raster based method builds off of the common approach of using DSMs to extract building footprints. This approach is portrayed in Figure 5. In this method, the height above the ground of LiDAR points is determined by subtracting from the Z value of the respective LiDAR point, the pixel value of the DEM for which it occurs. Points with a value less than a threshold, 2 meters in this study, are set equal to 0 with segmented building points being set to the calculated above ground height.   The resulting points are converted to a TIN using the above ground height, which is then exported to a raster data set. This produces what can be termed a normalized building surface model (nBSM). This nBSM is converted to an integer grid and another height threshold is employed to set cells lower than the input height threshold (i.e., 2 meters) equal to zero. This second height threshold is required due to interpolation between building and ground points creating grid cells with values between 0 and the input height threshold. This nBSM is reclassified such that grid cells with a value of 0 are set to No Data. This raster data set is
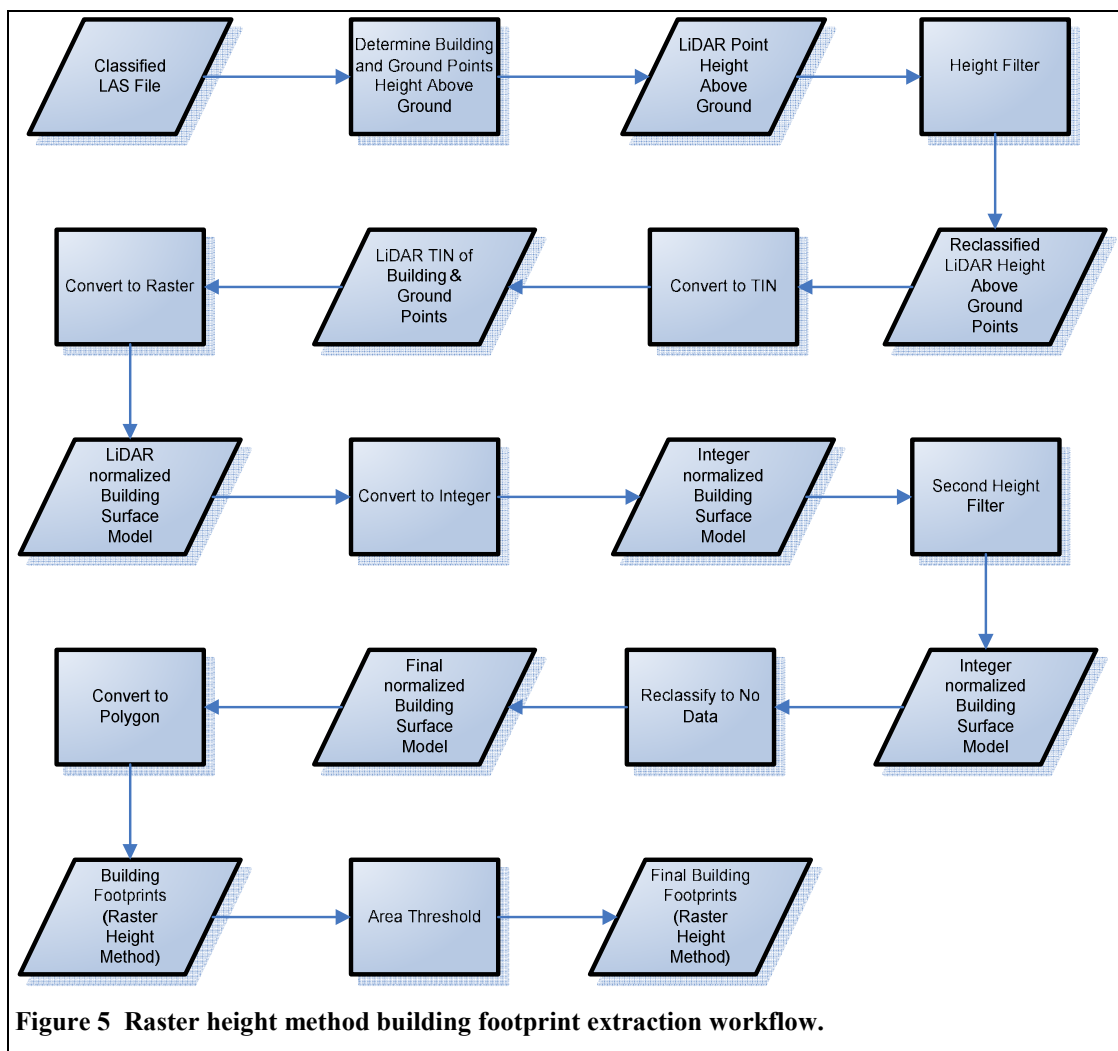
Figure 5 Raster height method building footprint extraction workflow.

converted to a polygon feature class using the ArcGIS software and an area threshold of 10 m^2 was used to produce building footprints with an example shown in Figure 6.

2.4 Building Heights

The accuracy of building heights as recorded by LiDAR was also examined in this study. Twenty-four buildings, as shown in Figure 7, were surveyed. The height above ground of each building corner and roof apex was determined. Building heights were surveyed using a Trimble GeoXH sub-foot accuracy Global Positioning System (GPS) receiver along with a Laser Technology Inc. ForestPro Laser Range Finder and Electronic Compass. The laser range finder was mounted on a tripod and the height of each building corner and apex was measured using standard procedures for the instrument. The distance from the location of the tripod to the respective building corner or roof apex was also determined using the laser range finder with the bearing to the building corner or roof apex measured using the electronic compass. The location of the tripod was measured using the GeoXH GPS where the tripod was setup in a location free of obstructions that might result in multi-path errors. Using this information the precise location of the respective building corner was determined as well as the recorded height associated with this location.

The LiDAR point cloud was classified using the ground and building segmentation algorithms described above. A DEM of the study area was derived by creating a TIN from the ground classified building points and exporting this TIN to a raster data set. The above ground height of each building classified LiDAR point was determined by subtracting from the height value of the respective point, the pixel value of the DEM for which the point occurs. The surveyed building corners or roof apexes where manually aligned with the closest LiDAR classified building point. As described below, some surveyed building corners had no LiDAR return due to occlusion from vegetation.

Figure 6 Example raster height method overlaid with digitized building footprint and Pictometry imagery.
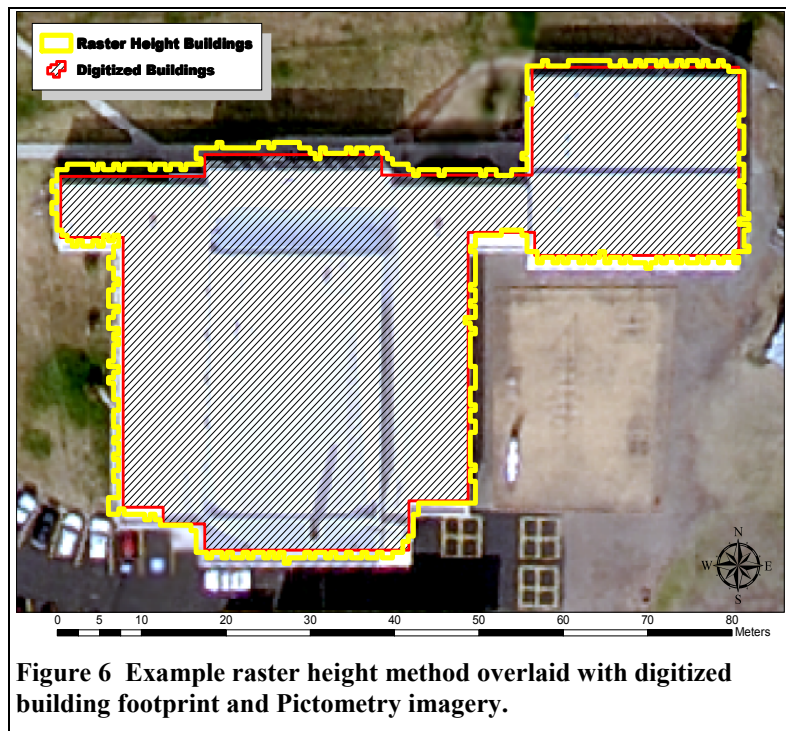
2.5 Results

This section details results for the accuracy assessments for the two methods of building boundary determination. The accuracy of the automatically extracted building boundaries were compared against building boundaries manually digitized from a combination of the NAIP imagery and LiDAR DSM. All buildings in the study area were assessed using ground surveys to ensure the digitized boundary represented an actual building. The study area, containing 449 buildings, is shown in **Error! Reference source not found.**.

26

Additionally, the accuracy assessments of LiDAR recorded building heights are presented in this section. The study area for these assessments is shown in Figure 7. This study area was chosen because it represented buildings on a sloping terrain and buildings owned by the Coeur d'Alene Tribe for which easy access was available. Measured building corners were manually paired with the closest LiDAR classified building corner, if any were recorded by the LiDAR sensor. Measured maximum building heights were paired with the maximum above ground height of the respective building to produce the assessment results presented below.

2.5.1 Building Segmentation Accuracy Assessment

Quantitative assessment of the accuracy of the building segmentation algorithm is a difficult process. This type of assessment might require exact determination of every LiDAR point returned from a building and the differentiation between returns from building sides, building roofs and off-roof features such as chimneys and events. This would be an onerous exercise where the assessments of the accuracy of the boundary tracing and raster methods presented below might provide sufficient information for the determination of the appropriateness of using the discussed building segmentation procedure. Nonetheless, a simple

Table 5 LiDAR building segmentation error matrix.

		SEGMENTED DATA				
REFERENCE DATA	CLASS	*Building (Points)*	*Non-Building (Points)*	*Row Total*	Producer's Accuracy (%)	Errors of Omission (%)
	Building (Points)	75,219	11,107	86,326	87.13	12.87
	Non-Building (Points)	12,955	936,148	949,103	98.64	1.36
	Column Total	88,174	947,255	1,035,429		
	User's Accuracy (%)	85.31	98.83		Overall Accuracy (%)	97.68
	Errors of Commission (%)	14.69	1.17		K^	0.85

error matrix was developed for the building segmentation algorithm. This error matrix was developed by first combining all segmented building points with other points within the height threshold used for this study (i.e., points greater than or equal to 2 meters above ground and points less than or equal to 25 meters above ground). Points within this set not segmented as buildings were considered segmented non-building points. The points from the entire point set, within the height threshold, which intersected the digitized building boundaries where selected and marked as reference building points. All other points, within the height threshold, were considered to be reference non-building points.

From the above data, an error matrix shown in Table 5 was created and results were derived as described in Campbell (1996). The producer's accuracy shows that about 87% of the reference building points were correctly segmented by the segmentation algorithm with about 99% of the reference non-building points correctly segmented. The user's accuracy shows that for this particular LiDAR data set about 85% of the points segmented as buildings correspond to reference building points with about 99% of the non-building points corresponding to reference non-building points. The error of omission for the segmented building points shows that about 13% of the examined LiDAR points, which were building returns in the reference data, were incorrectly excluded from the building segmentation. The error of omission for the segmented non-building points shows that about 1% of the examined LiDAR points, which were non-building returns in the reference data, were incorrectly segmented as buildings. The error of

commission for the segmented building points shows that about 15% of the examined LiDAR points were incorrectly segmented as building points with about 1% of the examined LiDAR points incorrectly segmented as non-building points, according to the reference data. The overall accuracy of the building segmentation algorithm for segmentation of building and non-building points is about 98%. Finally, the value $K^\wedge$ (kappa coefficient) can be interpreted to mean that the segmentation achieved an accuracy that is 85% better than would be expected from random assignment of points to categories of building and non-building.

2.5.2 Boundary Tracing and Raster Height Method Accuracy Assessments

The accuracy assessment results for the boundary tracing and raster height method are shown in Table 6. The detection rate shows that about 76% of the reference building boundaries contained traced polygons. The correctness rate shows that about 85% of the traced polygons represent actual buildings on the ground. The average matched overlay shows that for the traced polygons coinciding with reference buildings there is about 77% area in common, on average. The traced polygons representing reference building boundaries exclude about 23% of the reference building footprint area with about 7% of these traced polygons area not coinciding with reference footprint area, on average. On average, there is about a 19% difference in area and a 7% difference in perimeter between the traced and reference boundaries. An examination of the means of the traced footprint area (153 m^2) and the reference footprint area (175 m^2) indicates the corresponding reference footprints have a higher area compared to the traced footprints.

Table 6 Boundary tracing and raster method accuracy assessment results.

Index	Boundary Tracing (%)	Raster Height (%)
Detection Rate	76	81
Correctness	85	72
Average Matched Overlay	77	82
Average Area Omission Error	23	18
Average Area Commission Error	7	9
Average Area Difference	19	14
Average Perimeter Difference	7	29

The detection rate for the raster height method shows that about 81% of the reference building boundaries contained raster height polygons. The correctness rate shows that about 72% of the raster height polygons represent actual buildings on the ground. The average matched overlay shows that for the raster height polygons coinciding with reference buildings there is about 82% area in common, on average. The raster height polygons representing reference building boundaries exclude about 18 % of the reference building boundary area with about 9% of these raster height polygon areas not coinciding with reference footprint area, on average. On average, there is about a 14% difference in area and a 29% difference in perimeter between the raster height and reference footprints. An examination of the means of the raster height footprints area (160 m^2) and the corresponding reference footprint area (169 m^2) indicates the corresponding reference footprints have a higher area compared to the raster height footprints.

2.5.3 Building Height Accuracy Assessment

Each of the 24 buildings measured for height had the highest point on the building measured. In addition, each building corner was measured representing 129 original measurements. There was one gross outlier in the building height measurements; one building that did not have the maximum height recorded; and 8 spatial outliers in the building corners where the precise location of the measured building corner could not be aligned with the respective LiDAR building corner. These outliers were likely due to measurement error. In addition, there were 8 measured

building corners that were not observed in the LiDAR data due to being surrounded by dense vegetation.

Figure 7 Building height accuracy assessment study area.

Figure 8 shows a plot of field measured maximum building height versus LiDAR measured maximum building height. Table 7 shows the means of the field measured and LiDAR measured maximum building heights as well as the mean square error (MSE) and root mean square error (RMSE) of the field measured maximum building height versus the LiDAR measured maximum building height. Figure 9 shows a plot of field measured building corner heights versus LiDAR measured building corner heights. Table 8 shows the means of the field measured and LiDAR measured building corner heights as well as the mean square error (MSE) and root mean square error (RMSE) of the field measured building corner height versus the LiDAR measured building corner height.

2.6 Discussion

Overall the building segmentation algorithm showed promise for the proposed methods ability to segment LiDAR building points. Both the boundary tracing method and raster height method also showed promise for automatic extraction of building boundaries from a segmented point cloud. LiDAR derived building heights compared well to field measured building heights with manual measurement error possibly accounting for some of the discrepancies. The LiDAR data did not, however, typically penetrate through dense vegetation surrounding buildings but only a small population was examined. The developed tools used for these procedures, however, allow for quick clean-up of the segmented LiDAR point clouds, thereby allowing for even higher accuracies with some manual intervention.

Figure 8 Field measured versus LiDAR measured maximum building heights.

2.6.1 Building Segmentation

The building segmentation algorithm showed good results compared to reference data. The high producer and consumer accuracies for the segmentation of non-building points are partially a result of the study area containing significant forested areas compared to residential areas. The large number of reference non-building points compared to building points also contributed to the high overall accuracy for the building segmentation procedure. If the study area was limited to the extent of area covered
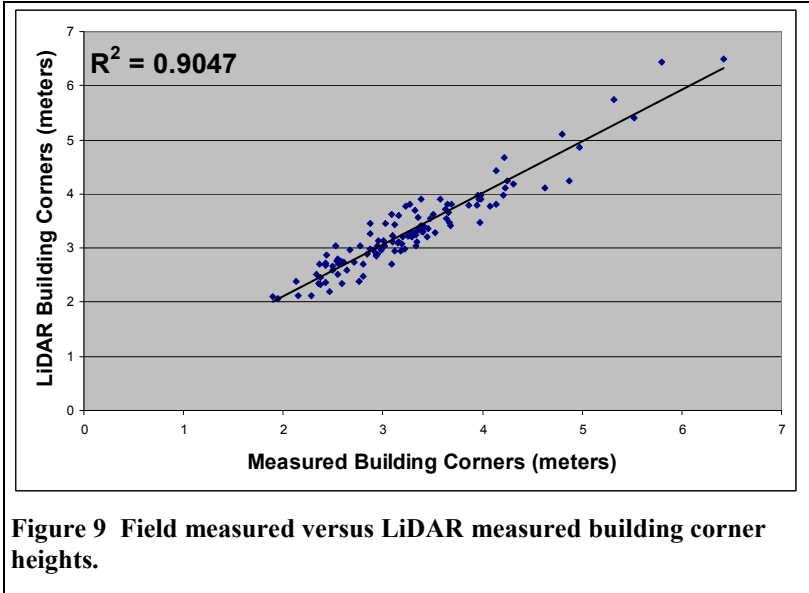
Figure 9 Field measured versus LiDAR measured building corner heights.

by buildings, the accuracies for the detection of non-building points would be lower, resulting in a lower overall accuracy. Nonetheless, the results indicate the usefulness of the approach when segmenting LiDAR point clouds in an environment containing large vegetation only areas intermixed with residential areas (i.e. the WUI). In addition, the producer and consumer accuracies for the segmentation of building points are likely lower than the actual accuracies. These lower accuracies are due to the method of classifying the reference data points. Reference building points were determined by classifying all LiDAR points within the reference building boundaries as building points. This method, consequently, included vegetation returns covering buildings and likely decreased the accuracies associated with the segmentation of building points. A number of buildings in the study area were surrounded by vegetation. Qualitative visual

Table 7 Comparisons between LiDAR and field measured maximum building height.

Measurement Technique	Mean (meters)	Mean Square Error (m²)	Root Mean Square Error (meters)
LiDAR	5.75	0.79	0.89
Manual	5.11		

Table 8 Comparisons between LiDAR and field measured building corner height.

Measurement Technique	Mean (meters)	Mean Square Error (m²)	Root Mean Square Error (meters)
LiDAR	3.33	0.06	0.25
Manual	3.28		

examination of these buildings as shown in Figure 10 indicates the segmentation procedure performed well in differentiating vegetation returns from building returns for these situations.

Additionally, the use of the input vegetation grid played a significant role in not erroneously segmenting returns as buildings in vegetative areas where the number of segmented building points increased by 30% when the vegetative binary grid was not input to the segmentation procedure. The use of the vegetation mask with the specified parameter values represents an optimized configuration for the study area. If no vegetation mask is available, adjustment of certain parameters can reduce the number of erroneously segmented building points in

Figure 10 Segmented building points covered by vegetation.

vegetative areas with minimal reduction in segmentation of actual building returns. The use of the vegetative mask also resulted in the masking of certain returns from buildings when the vegetative pixels represented deciduous trees covering buildings. This is because the LiDAR data was collected in leaf-off conditions where the first return, in some cases, was able to penetrate through the deciduous leaf-off branches. The proposed use of the vegetation mask, therefore, might be aided by the differentiation between deciduous and conifer trees when LiDAR data is collected in leaf-off conditions.
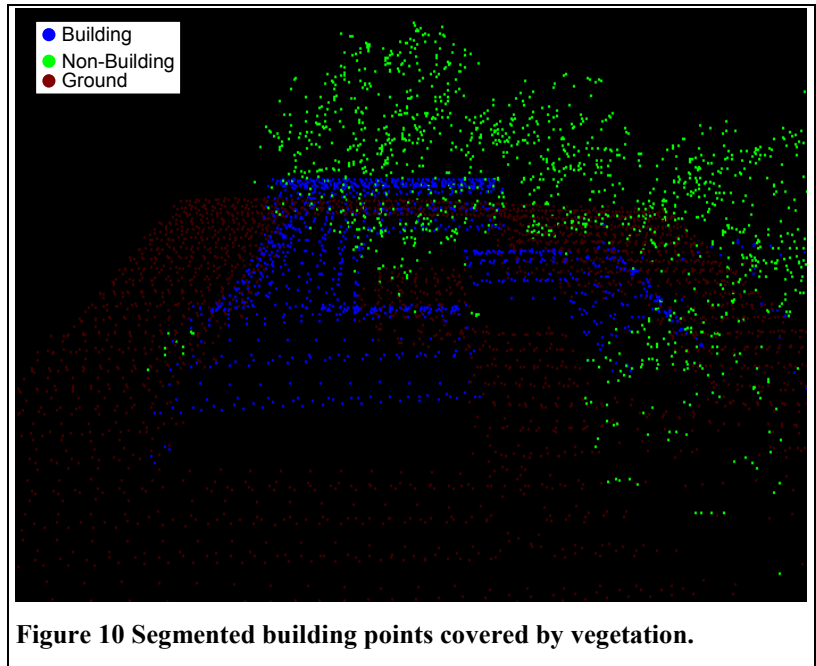
Finally, the proposed laser pulse filter reduced the vegetative points incorrectly segmented as building points by about 20% when considering only multiple return pulses. This indicates this filter is appropriate for use in environments containing significant amounts of vegetation. This use of this filter in a more urban environment containing tiered buildings, however, might not be appropriate due to this filter removing building edge points from tiered buildings. The removal of these building points is unlikely to have a detrimental effect on the extraction of the building boundaries as these points typically represent the interior of buildings. The removal of these building points, however, could affect height values associated with the building points for tiered buildings.

2.6.2 Boundary Tracing and Raster Height Method

Both the boundary tracing and raster height methods showed promising results in extracting building boundaries from a segmented LiDAR point cloud. The lower detection rate (i.e. producer's accuracy) associated with these methods compared to the producer's accuracy associated with the segmentation algorithm is largely due to the area threshold employed. Some extracted building footprints were removed from the accuracy assessment because the extracted area represented an area below the threshold employed (i.e., 10 m^2). These removed footprints, in some cases, actually represented portions of larger buildings where only part of the building returns were classified. The classified building points were included as being correct in the

31

segmentation accuracy assessment, but these points did not produce an adequate footprint to be considered correct for the extraction accuracy assessment. This indicates that smaller extracted footprints should be examined as areas for possible manual cleanup.
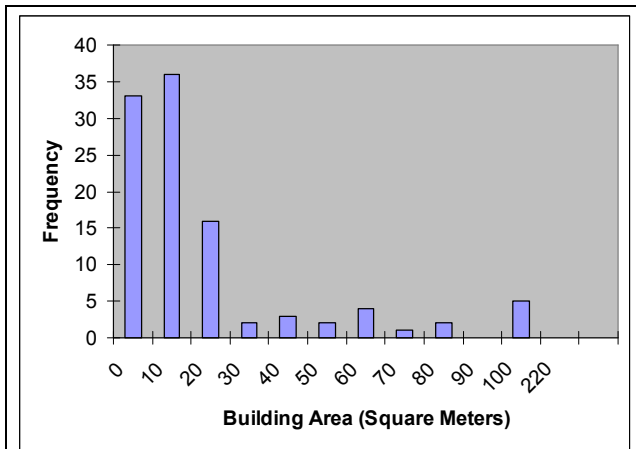
Figure 11 Distribution of missed building areas for the boundary tracing method.

Also, many of the missed building footprints had an area less than the input area threshold. The distribution of the areas of missed building footprints for the boundary tracing method is shown in Figure 11 with the distribution of the areas of missed building footprints for the raster height method shown in Figure 12. As indicated by Figure 11 and 12, the majority of missed building footprints have an area less than 30 m^2 for the boundary tracing method and 20 m^2 for the raster height method. The boundary tracing method had five missed footprints with areas greater than 100 m^2. These buildings contained segmented building points, but were only partially traced due to irregularities in the grouped building points. In addition to these missed buildings, the higher detection rate associated with the raster method can be accounted for by the raster method having an increased extracted footprint area. This is shown by the higher average matched overlay, the lower average area omission, the higher average area commission and the lower average area difference in the raster height method. This increased area for extracted footprints from the raster height method is accounted for by the differences in methods. Any interpolation of points will provide a larger area than an exact boundary tracing of these points. The magnitude of difference in area related measurements between the boundary tracing and raster height method will likely increase the larger the pixel size of the created nBSM.

Both the boundary tracing and raster height methods showed relatively large missed portions of reference buildings as demonstrated by the area of omission values of 23% and 18% for the boundary tracing and raster height methods, respectively. These large error of omission values also resulted in large values for average area difference (i.e., 19% and 14 % for the boundary tracing and raster height methods, respectively). While the effect these differences might have on WFDS runs is unknown it might
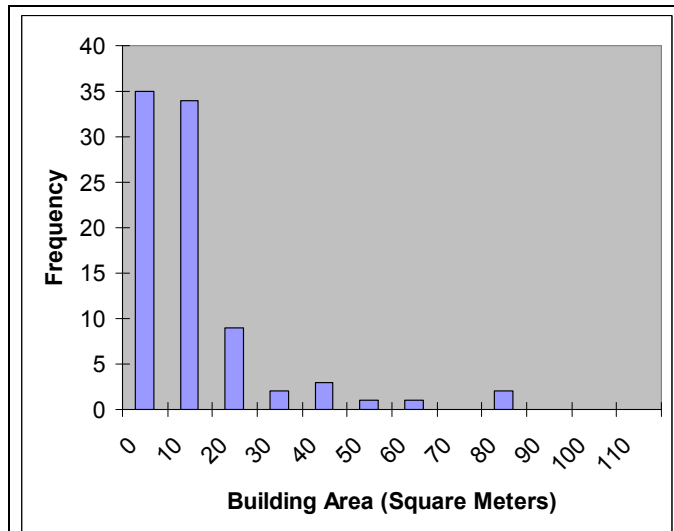
Figure 12 Distribution of missed building areas for the raster height method.

be significant. Qualitative evaluation, however, of the reference building footprints against the Pictometry imagery shows that in some cases there was a tendency for the digitizer to

overestimate building area as shown in Figure 13. This tendency to overestimate the building area might be due to the fact that reference footprints were digitized from the NAIP imagery before the Pictometry imagery was available. The NAIP imagery was of a coarser resolution and might have caused the digitized building area to be larger due to mixed pixels and resulted in the larger numbers for area of omission and area difference. While the small area of commission errors indicates that the extraction methods are not delineating large areas outside the reference footprint, these values might, however, increase if the area exaggeration in the digitized footprint was reduced. It is also likely the case that the traced boundaries will generally produce building footprints that are smaller than reference footprints. This tendency is probably a function of point density with lower point density datasets having less of a probability of exactly identifying the building edge. The tendency for the raster height method to extract a larger area footprint compared to the boundary tracing method could result in this technique being more useful in accounting for the missed building edges in lower point density datasets.

Figure 13 (A) Digitized footprint over-laid on Pictometry imagery, and (B) Digitized footprint over-laid on NAIP imagery showing mixed pixels and possible cause of area exaggeration.

The large difference between correctness rates (i.e. consumer's accuracy) for the boundary tracing and raster height methods is again likely caused by the difference in techniques between the two extraction methods where the raster based interpolation technique will produce polygons with a larger area. The tendency for the raster height method to produce larger polygons, on average, resulted in more polygons that exceeded the input area threshold compared to the boundary tracing method. This resulted in a lower correctness rate for the raster height method compared to the boundary tracing method because there were more erroneous polygons extracted.

Additionally, there was a large difference between the average perimeter differences of the boundary tracing and raster height methods. The large value for the average perimeter difference of the raster method is due to the footprints being derived from raster data. Raster data represents square pixels where the conversion of building pixels to a vector dataset produces jagged edges as shown in Figure 14. The jagged nature of the raster height method extracted footprints might be irrelevant to inputs for WFDS as all input building footprints are converted to grids for WFDS.

33

The small value for the average perimeter difference of the boundary tracing method, however, might indicate the methods ability to correctly extract the shape of the buildings despite the extracted area being underestimated.

2.6.3 Building Heights

Overall the LiDAR measured building heights showed good correspondence to the field measured building heights. The LiDAR measured building corners showed a better correspondence with the field measured data compared to the LiDAR measured maximum building heights. In both cases the RMSE were less than 1 meter where most WFDS experiments over larger scales would have resolutions of 1 meter or greater. This indicates the LiDAR performs as well as the field measurements for determining

Figure 14 Jagged nature of raster height extracted footprints.

building height, with regard to inputs for WFDS. In addition, field measurements of building heights are labor and time intensive to obtain and essentially impractical over any large area.

The poorer results for the LiDAR measured maximum building height is thought to be due to human measurement error. This is not shown conclusively but evidence is provided in the fact that the means of the LiDAR measured maximum building height is greater than the field measured maximum building height. It would be expected that the LiDAR measured maximum building height would typically be lower than the actual maximum building height due to the unlikely probability of the laser pulse hitting the exact roof apex. A comparison of the means shows there is a tendency for the LiDAR to over estimate building height. It is possible that this overestimation using LiDAR data is caused by improper segmentation of ground points or errors introduced by interpolation of these ground points. If errors with ground classified points or interpolation errors caused errors in building heights, a larger magnitude of difference between the mean measured building corner heights would be expected than what is shown in Table 8. This is not the case and while the mean of the LiDAR measured building corners is greater than the mean of the field measured building corners, the magnitude of difference is smaller. In addition, it is expected that the LiDAR would tend to overestimate building corner height, for sloped roofs[4], due to the unlikely probability of the LiDAR pulse hitting the exact building corner.

Finally, every building corner that was covered by conifer vegetation did not have any LiDAR building returns below the vegetation. LiDAR returns underneath deciduous vegetation did, however, have LiDAR building returns. There were only 8 building corners surrounded by conifer vegetation with 6 building corners surrounded by deciduous vegetation. This small sample size does not lead to quantitative conclusions about the ability of LiDAR to penetrate through dense vegetation, but qualitative evaluation of other areas shows similar results. It is possible, however, through manual intervention to add points representing these missed building corners based on assumptions derived from other building returns.

---

[4] All of the roofs in the study area were sloping.

2.7 Conclusion

Section 2 of this report presents an algorithm for automatic extraction of the three dimensional distribution of buildings from LiDAR and multispectral data. The LiDAR segmentation algorithm first employs thinning algorithms that utilize vegetation grids derived from multispectral data as well as incorporating a custom laser pulse filter. Both of these thinning algorithms improve results of the plane fitting segmentation procedure, which ultimately segments the LiDAR above ground points to building and non-building categories. The results of this segmentation algorithm compared against reference data indicate the usefulness of this algorithm in segmenting building points in a WUI environment. Additionally, the unique capabilities of the tools developed to perform the building segmentation allow for rapid cleanup of the data and improved results.

From the segmented point cloud, two building boundary determination methods (i.e. boundary tracing and raster height) are evaluated against reference data. Similar results are obtained for each method where both methods underestimated building area. This underestimation might, in part, be due to an overestimation of building area during the manual digitization process. The raster method better represented the area of the reference footprint compared to the boundary tracing method. The raster method, however, had a lower correctness rate due in part to the larger area of extracted footprints and subsequent inclusion of more polygons above the input area threshold.

The use of one method over another for deriving inputs to WFDS likely depends on the environment and the characteristics of the input LiDAR data. When point densities are low the raster method might account for some of the under estimation of building area. With high point density datasets the boundary tracing method might not have as significant of an underestimation of building area and might produce better correctness rates. The raster method is also much simpler to implement and requires less processing time than the boundary tracing method. The shape similarities between the extracted and reference footprints were, however, not examined in this study, except rudimentarily in the average perimeter difference measure. This measure showed better results for the boundary tracing method. The effect changes in building footprint shape would have on WFDS outputs is unknown but suspected to be negligible if appropriate three dimensional areas are described. If shape proved to be important, the boundary tracing method could prove superior and adjustments made to account for underestimation of building area. Both techniques showed excellent promise for deriving building information inputs to WFDS. The techniques might also prove complimentary where different methods can be used for different environments and data characteristics.

Comparisons between field measured and LiDAR measured building heights were also promising and likely within the resolution of most WFDS landscape scale model runs. Additionally, it might be the case that the LiDAR measured building heights are more consistent and less prone to human error but further study is required to confirm this. The comparisons made in this report do not account for all measurement errors inherent in each system but the techniques used here for field measurements are felt to be more prone to human error than LiDAR measurements derived from quality equipment and trained operators. This study also confirmed recent findings by Cheuk and Yuan (2009) where lack of laser penetration through vegetation is a potential error in the determination of building height.

Overall the use of LiDAR and multispectral data for automatic extraction of building information in a WUI environment appears to be the only practical method over any large area. The effect the reduced area of the automatically extracted footprints might have on WFDS model runs needs to be examined, however, to determine if the accuracies presented in this section are sufficient for producing reliable WFDS outputs. If the reduction of area in the LiDAR extracted footprints

could be quantified, adjustments could be made for WFDS inputs. Additionally, other sensitivity analyses of WFDS would aid in the interpretation of these results in the context of deriving building information inputs for WFDS. Finally, the algorithms presented here have applications beyond WFDS and could be used for hazard and property assessments, urban growth analysis and other areas. Future efforts beyond WFDS sensitivity analyses should focus on examination of differences in building segmentation and footprint extraction from datasets with varying point densities; improvements to the plane fitting technique to measure the orthogonal distance to the identified plane instead of the vertical distance; distinct segmentation of off-roof features, building sides and building edges; and orthogonalization of building footprints to produce spatial representations that better match the shape of the actual building.

Section 3.0 Tree Extraction in the WUI

Abstract

The automatic extraction of tree stem locations from remotely sensed data can be of great benefit to various land management disciplines. Many organizations and programs such as the Continuous Forest Inventory (CFI) program survey tree stem locations throughout the country. Accurate and automatic extraction of tree stem locations with associated attributes can also be used to characterize the WUI and as inputs for fire behavior fuel models. This section presents an automatic approach for the extraction of tree stem locations and associated attributes of height above ground, crown radius and height to live crown (HLC). This work is conducted in the context of deriving inputs for the Wildland Fire Dynamics Simulator (WFDS). The tree extraction algorithm operates directly on the raw LiDAR point cloud classified for building returns. The algorithm derives the horizontal and vertical coordinates of the potential tree stem location with attributes for crown width and HLC. The accuracy of the tree extraction algorithm is assessed against surveyed trees in three urban plots and five wildland plots in Worley and Plummer, Idaho.

The tree extraction algorithm correctly extracted 27% to 97% of the trees found in the various study areas. Generally, the algorithm did very well at extracting dominant trees but in most cases missed sapling trees or those trees with a diameter at breast height less than 4 inches. Additionally, the algorithm did not extract trees whose tree tops where within the canopy of a taller tree. LiDAR derived tree heights compared well to surveyed tree heights with root mean square errors (RMSE) varying between 0.9 to 2.10 meters for the different study sites with the means of the LiDAR derived heights tending to be less than the surveyed tree heights. This discrepancy is due in part to the time difference between the ground and LiDAR surveys (i.e., two years) and the well documented trend of LiDAR to underestimate tree height. Comparisons between LiDAR derived crown radius and surveyed crown radius showed RMSEs between 0.62 and 1.20 meters with the means for the LiDAR derived crown radii tending to be less than the surveyed crown radii. This is likely due to a tendency for the crown radii algorithm to overestimate crown radii. RMSEs for comparisons of HLC varied between 1.09 and 6.92 for the various study sites. The algorithm did not extract HLC from the LiDAR data very well possibly due to the inability of the algorithm to distinguish between dead and live branches as well as the inability of the algorithm to differentiate returns from different trees. Results from the tree extraction algorithm can be enhanced through manual editing and the potential for using LiDAR for deriving tree inputs to WFDS is demonstrated in some cases. Nonetheless, examinations of other methods for deriving the 3D structure of vegetation for inputs to WFDS are recommended.

3.1 Previous Work

This study stems from the need for the efficient and accurate quantification of the three dimensional distribution of vegetation for inputs into WFDS. WFDS takes as inputs the horizontal and vertical extent of vegetation as well as material properties for the vegetation of interest. Vegetation can be input into WFDS as points with values specified for the horizontal and vertical coordinates, crown width, crown base height, diameter at breast height, crown bulk density and other physical properties of the vegetation under consideration. Alternatively, vegetation can be input into WFDS as rectilinear elements where each 3D grid cell contains the same attributes as described above. As with building extraction, remote sensing provides an efficient means of deriving some of these types of vegetation information. This section focuses on the extraction of tree stem locations as points with associated attributes of crown radius, HLC and tree height.

Techniques to estimate vegetation characteristics from remotely sensed data have been used for many years with the estimation of forest biophysical parameters historically occurring from passive optical sensors (Hall et al, 2005). A combination of field data and aerial photography has traditionally been used to map crown bulk density, crown closure, and canopy height (Riano et al., 2003). The use of empirical methods with passive sensors has been used to map crown closure, above ground biomass, crown bulk density, and structural stage classes (e.g., Cohen and Spies, 1992). These empirical methods have been shown to be non-linear where above ground biomass saturates at approximately 100 Mg/ha (Cohen and Spies, 1992). Additionally, passive optical sensors cannot discriminate the vertical structure of biomass (Hall et al, 2005). Finally, the detection of individual tree crowns (ITC) has been achieved from aerial imagery (e.g. Gougeon, 1997) and satellite imagery.

Recent technological developments have resulted in the increased availability of active sensor data such as LiDAR, which provides 3D data. In addition, LiDAR does not saturate at high biomass (Riano et al., 2003). LiDAR data could become the main technology for the mapping of forest biophysical variables (Thomas et al., 2006) due to the high degree of accuracy demonstrated in mapping these variables (e.g. Lim and Treitz 2004; Hopkins et al. 2005; Anderson et al., 2005). Forest biophysical variables such as tree height and crown width have been measured at the stand level (Hall et al., 2005), the plot level (Holmgren et al., 2003; Hyyppa et al., 2001; Lim and Treitz, 2004), and the individual tree level (Chen et al., 2006; Coops et al., 2004; Holmgren and Persson, 2004; Persson et al., 2002; Roberts et al., 2005). Studies focusing on deriving crown base height and crown bulk density are fewer and typically use allometric relationships or statistical inference to estimate these variables (Popescu and Zhao, 2008). Many of these methods (e.g., Anderson et al., 2005) derive forest biophysical variables using a raster data model where each cell in the raster represents a 3D area of the land surface (Perry, 1998). It is less common to characterize the vertical space of forest canopies and attempts focus on voxel[5] based approaches (Chasmer et al., 2004; Parker, 1995; Weishampel et al., 1997) and height bins (Naesset 2004; Popescu and Zhao, 2008). Finally, many studies have examined the synergistic use of active and passive sensors for vegetative mapping (Hill and Thomson 2005; Packalen and Maltamo 2007) where the combination or fusion of passive and active sensor data can be demonstrated to improve classification accuracy (Walter, 2005).

LiDAR has also been used for the extraction of tree stem locations (e.g. Rowell et al., 2006; Popescu and Kini, 2004; and Solberg, 2006) where the LiDAR surface points are typically interpolated to a raster data set termed a canopy height model (CHM). Both Rowell et al. (2006) and Popescu and Kini (2004) use a local maximum algorithm and user input relationships between tree height and crown width to extract tree stem locations. Solberg (2006) also uses a local maximum to identify potential tree stem locations where a local maximum is defined as a grid cell having higher Z values than its eight neighboring grid cells. Falkowski et al. (2006) use spatial wavelet analysis (SWA) to extract tree stem locations independent of any allometric relationship between tree height and crown width. Results reported by Falkowski et al. (2006) compared favorably to those described in Popescu and Kini (2004). The use of the raw point cloud of LiDAR data has also been used for the extraction of tree stem locations. Barilotti et al. (2009) used a morphologic analysis of the laser point distribution to identify single tree stem locations. The use of cluster analysis was used by Morsdorf et al. (2003) to identify individual tree stem locations.

Popsecu and Kini (2004) used a median filter on the CHM along with fitting a fifth degree polynomial to crown profiles corresponding to local maximum to calculate crown width. The local minimum of the fitted polynomial were determined and the distance between these

---

[5] In WFDS terms a voxel can be thought of as a three-dimensional grid cell.

minimum calculated to determine crown width. Crown base height was not calculated by Popescu and Kini (2004) but in Popescu and Zhao (2008) a voxel based method combined with Fourier and wavelet filtering and a similar polynomial fitting technique was used to determine crown base height. Solberg (2006) used a region growing algorithm beginning with the identified local maximum to calculate crown width. Crown base height was then calculated by Solberg (2006) using the deciles of the height distributions with crown base height being determined from the decile with the largest distance to the neighboring decile. In Falkowski et al. (2006) the size of the wavelet allows for direct calculation of the crown width.

3.2 Methods

The inputs and workflows used for the identification of tree stem locations and associated attributes in this study are shown diagrammatically in Figure 15. The tree stem extraction algorithm takes as inputs a bare earth DEM, canopy height model (CHM) and LAS file with building points classified. User inputs include an allometric relationship between tree height and crown width, minimum tree height, minimum number of points required for a tree stem location, minimum and maximum crown widths, a canopy search radius length, and the minimum number of correct profiles. All these inputs will be discussed in detail in the sections below. The tree stem extraction beginnings with an initial thinning algorithm to remove points clustered together at the end of the scan line. A simple minimum tree height filter is then employed with the identification of local maximum occurring. The crown width algorithm proceeds with the tree stem extraction concluding with the height to live crown algorithm. Tree stem locations with associated attributes are output to a user designated text file.

3.2.1 Local Maximum Filter

LiDAR returns at the end of a scan line are often clustered where this clustering will sometimes result in many points returned from pole features such as telephone poles. In order to remove these points from consideration and alleviate the potential of extracting pole features as trees an initial thinning algorithm is employed. This thinning algorithm searches for points clustered in a small radius with a large range in height values. This study used a search radius of 0.5 meters, a minimum number of points of 5 and a minimum height range of 5 meters. With these pole points removed an initial minimum tree height filter is employed to exclude potential tree stem locations lower than a user input height threshold. The height threshold used for each study site was the minimum tree height surveyed for that location. The smallest tree surveyed in the study was 1.14 meters tall.

With the LAS file thinned as described above, the identification of potential tree stem locations proceeds. A local maximum filter is employed that takes as an input an allometric relationship between tree height and crown width. This study used an allometric relationship from Tiede et al. (2005) that was not calibrated for specific species or tree structures. This relationship is shown in the equation below:

$$CW = 1.54 + (0.123*TH) \qquad \textbf{(13)}$$

where

$\qquad$ CW = crown width, and

$\qquad$ TH = tree height.

The developed tree extraction tool is capable of taking up to a third order polynomial representing the allometric relationship between tree height and crown width.

The motivation for the use of a local maximum filter comes from Popescu and Kini (2004). This algorithm, however, operates directly on the point cloud of data instead of the interpolated surface

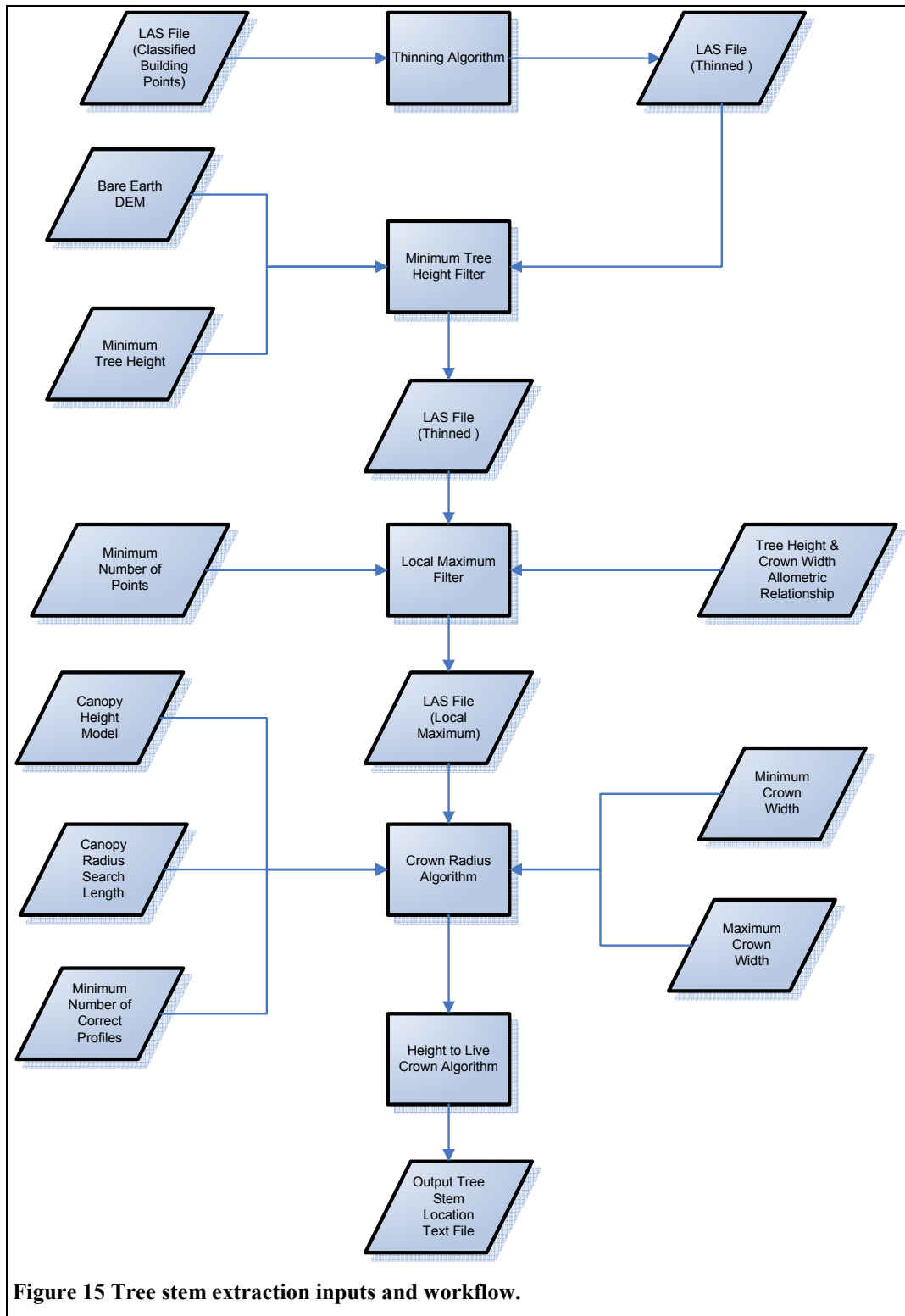points or CHM as in Popescu and Kini (2004). The identification of local maximum proceeds by

Figure 15 Tree stem extraction inputs and workflow.

iterating through each point in the LiDAR point cloud and if the point in the iteration is the highest point in the variable search window it is flagged as a tree top. In order to filter out local maximum, which represent pole features, the number of points below the identified local

maximum point is recorded and if this number of points is below the user input threshold, three points in this study, the search window is expanded by 33% (Personal Communication, Eric Rowell). If new points are added such that the total number is greater than the threshold the point is flagged as a tree top. This is an advantage of using the point data directly compared to the canopy height model (CHM), which cannot determine points beneath the identified local maximum. This is the main difference between the algorithm used by Popsecu and Kini (2004) and the algorithm presented in this paper for identifying potential tree stem locations. Once the potential tree stem locations have been identified the derivation of crown radius begins.

3.2.2 Crown Radius Algorithm

The approach for determining crown radius uses an input CHM. As described in Popescu and Kini (2004) performing a median filter on this CHM might help improve results. This is because
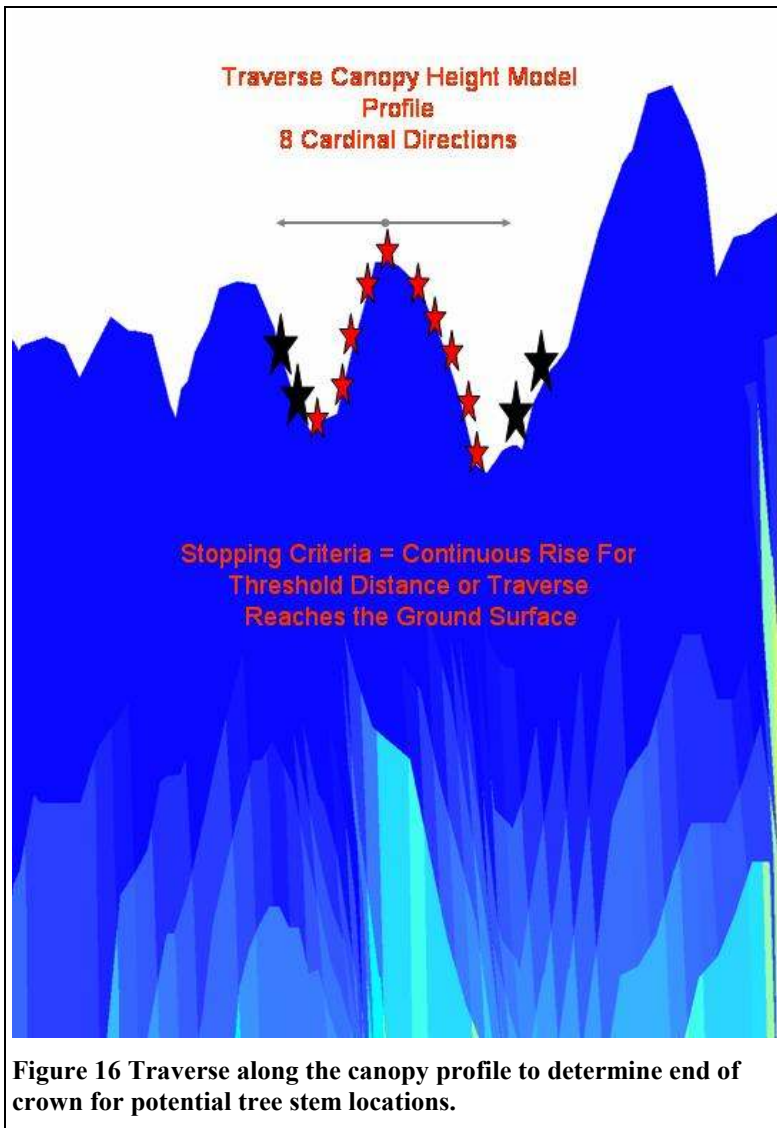
Figure 16 Traverse along the canopy profile to determine end of crown for potential tree stem locations.

the median filter will remove anomalies present in most canopies while preserving the edge of the canopies under consideration. This study used a 3X3 median filter on the input CHM implemented in ENVI 6.3. While the use of a smoothed CHM was derived from work conducted by Popescu and Kini (2004) the approach for determining crown radii used in this study is different and is independent of the input allometric relationship between tree height and crown width. For each identified potential tree top point a traverse along the crown profile in all 8 cardinal directions is conducted as shown in Figure 16. Each value in the CHM is recorded along the traverse and if the next value of the CHM is greater than the previous value or if the value of the CHM is less than the user input minimum tree height the location is marked. If the above situation occurs consecutively over an input

distance specified in the input canopy search radius length, the initially flagged location is determined to be the end of canopy point for the particular direction of the traverse. This study used a canopy search radius length of 2.5 meters. The difference between the flagged values in

the north and south direction, or the east and west direction are calculated. In addition, the CHM profiles are also traversed in the Southeast/Northwest and Northeast/Southwest directions and the crown radii in these directions are calculated as above. These 8 crown radii values are averaged to determine the crown radii for the potential tree stem location under consideration. This algorithm will sometimes result in a large overestimate of crown radius. This occurs when the traverse of the canopy profile extends into another trees canopy while continuing to decrease in height. Each tree is checked to see if its crown encompasses other trees to adjust for this error. If 4 or more trees are encompassed within the crown of a particular tree, the crown radius in the particular direction of the encompassed tree is adjusted to be the midpoint between the original tree and the encompassed tree. In order to prevent the CHM traverse from exceeding the potential maximum crown width found in the area of interest, an input maximum crown width is entered where if exceeded the CHM traverse will be terminated. This study used a maximum crown width of 24 meters. Additionally, a minimum crown width is also input to again alleviate misidentification of pole features not filter out by the procedures described above. This study used a minimum crown width of 1 meter.

In addition to the above, a method to remove the edge of canopy or mid canopy points incorrectly extracted as trees has been developed. Due to the complex nature of tree canopies a branch or portion of the canopy, which spikes above the rest of the canopy for the particular tree under consideration, is often identified as a local maximum and incorrectly output as a tree stem location. These incorrectly identified tree stem locations often occur at the edge of a stand of trees, a situation that essentially defines the WUI. An attempt has, therefore, been made to identify these points concurrent to measuring the crown radii of the tree. This is accomplished by recording the value of the CHM for the particular point identified as a potential tree top location during the traverse of the CHM profile. For the next four values of the CHM in the traverse it is determined if the majority of these values are greater than the initial CHM value. If this condition is met the particular direction of the traverse is flagged. This value is input to the algorithm as the minimum number of correct profiles. This study used a minimum number of correct profiles of 7 indicating that the above described scenario can only occur once in the eight canopy profile traverses for the originally identified tree stem location to be output as a final tree stem location.

3.2.3 Height to Live Crown Algorithm

The algorithm used to determine HLC is similar to that described by Holmgren and Persson (2004) where all laser points below a particular flagged tree top point and within the identified crown width as described above are grouped into 0.5m height bins. Each bin is categorized as 0 or 1 where bins with a 0 value have less than 1% of the total number of non-ground laser points within all the bins and all others have a 1 value (Holmgren and Persson, 2004). The HLC was then determined as the distance from ground of the lowest laser data point above the highest 0 classified bin found. Some modifications are made to the above described algorithm. The first modification is a result of tree crown radius not always being consistent in all directions. Grouping all points below the flagged tree top point and within the crown width could, consequently, result in incorrect estimates of crown base height. To account for this the location below flagged tree top points is divided into four quadrants and the average crown radii for each particular quadrant is used to determine the laser points within that quadrant. In addition, instead of using the lowest laser data point above the highest 0 classified bin found, the bins are searched for a continuous occurrence, for a threshold number of times, of 0 value bins. When this occurs the lowest laser point in the next 1 value bin, after the occurrence of the 0 value bins, is marked as the crown base height. This step is conducted in an attempt to alleviate the influence of under-story vegetation.

3.3 Results

This section details results for the accuracy assessments of the tree stem extraction algorithm with associated attributes. The accuracy of the automatically extracted tree stem locations and associated attributes were compared against tree stem locations and associated attributes surveyed on the ground during the summer of 2007. A total of eight plots were surveyed representing urban and wildland forest stands. Three study sites were located in an urban setting and the other sites were located in a wildland setting. The Worley Park study site shown in Figure 18 represents a typical urban scene found in many WUI environments. This study site contained three deciduous trees, 57 conifer trees, four shrubs, six telephone poles, four basketball hoops, four light poles, one totem pole, and four pieces of playground equipment. The Plummer High School study site is shown in Figure 17 and contains 41 conifer trees with a building and several telephone poles on the edge of the study site. The Plummer Housing study site shown in Figure 19 contained a mix of deciduous and conifer trees. The five wildland tree extraction study sites are shown in Figure 20 and Figure 21. These plots were 11.3 meter diameter circular plots coinciding with continuous forest inventory (CFI) plots. In all, the above study sites contained Ponderosa Pine, Douglas Fir, Western Red Cedar, Aspen, Grand Fir, Western Red Hemlock and Black Locust tree species. These trees ranged from fully mature trees to saplings categorized by a diameter at breast height (DBH) less than 4 inches.

Figure 18 Worley Park WUI tree extraction study site.

Figure 17 Plummer High School tree extraction study site.

The location and height above ground of each tree stem location was surveyed using a Trimble GeoXH sub-foot accuracy GPS receiver along with a Laser Technology Inc. ForestPro Laser Range Finder and Electronic Compass. The laser range finder was mounted on a tripod and the height of each tree was measured using standard procedures for the instrument. The distance from the location of the tripod to the respective tree stem location was also determined using the laser range finder with the bearing to the tree stem location measured using the electronic compass. The location of the tripod

Figure 19 Plummer Housing tree extraction study site.

was recorded using the GeoXH GPS where the tripod was setup in a location free of obstructions, as practical, which might result in multi-path errors. All GPS recorded points were differentially

Figure 20 Wildland tree extraction study sites (plots 38 and 39_40).

corrected to an accuracy of within 1 meter or less using the Trimble GPS Analyst Extension. In many cases this required repeated measurements of the tripod locations, which were flagged in the field. Using this information the location of the tree stem was determined as well as the recorded height associated with this location. Additionally, for each tree surveyed the HLC, DBH and crown radius were measured. Crown radius was determined by measuring the major crown width and minor crown width of each tree, averaging the two

measurements and dividing the result by two. HLC was measured using the ForestPro Laser Range Finder to measure the height above crown of the first live vegetation. DBH was measured at 4.5 feet above ground using a diameter tape or d-tape.

Despite the high accuracy of the equipment used for surveying the tree stem locations it was sometimes difficult to precisely register the surveyed trees with the LiDAR extracted tree stem locations. This is due to many factors including the combined inherent error of the raw LiDAR data with the GPS and laser range finder, the fact that the LiDAR observed tree top does not always coincide with the measured tree stem location at the base of the tree due to tree tilt and human error. Consequently, a large amount of manual effort was required to re-

Figure 21 Wildland tree extraction study sites (plots 33, 36 and 37).

adjust the surveyed trees to the LiDAR extracted tree stem locations. For some plots, due to the density of the trees, precise tree by tree comparison of the surveyed trees to the LiDAR extracted trees was not feasible and distributions were compared.

3.3.1 Tree Stem Location Accuracy Assessment

The accuracy assessment results for individual tree stem locations are shown in Table 9. For each study site the percentage of correctly extracted trees is reported along with the number of false positives. The percentage of correctly extracted trees is determined by dividing the number of LiDAR extracted trees that coincide with a surveyed tree by the total number of surveyed trees in each plot. False positives are tree stem locations extracted from the LiDAR data that do not correspond to an actual tree stem location on the ground. The accuracy of the trees

Table 9 Accuracy assessments for individual tree stem locations.

Plot	Correctly Extracted Trees	False Positives
Worley Park	97% (59 of 61)	5
Plummer Housing	~85% (~172 of 201)	~16
Plummer High School	85% (35 of 41)	1
Plot 33	27% (9 of 34)	0
Plot 36	37% (10 of 27)	1
Plot 37	84% (16 of 19)	3
Plot 38	83% (5 of 6)	7
Plot 39_40	30% (14 of 46)	1

extracted from the Plummer Housing site listed in Table 9 is an approximation. This is because the density of trees for the Plummer Housing site was too dense to allow for exact alignment of surveyed trees to LiDAR extracted trees on a tree by tree basis and the numbers were, consequently, estimated.

3.3.2 Tree Height Accuracy Assessment

The accuracy assessment results for tree height are shown in Table 10. Only those trees that were extracted correctly from the LiDAR data were considered in this comparison. For each plot the average tree heights as recorded by the LiDAR tree extraction algorithm and the average surveyed tree heights are reported. Additionally, the MSE and RMSE are reported for each site. The Worley Park site had three gross outliers removed likely due to measurement error where the three trees were in the interior of the plot and caused difficulty in the measurement of height values. The Worley Park site also had two trees that were removed between the time of the LiDAR survey and the ground survey. The Plummer High School site had two gross errors removed likely due to measurement error where the recorded surveyed height value was erroneously high. For all plots the average tree height for the surveyed trees showed a tendency to be higher than the average tree height for the LiDAR derived trees.

Table 10 Tree height accuracy assessment results.

Plot	LiDAR Mean Height	Surveyed Mean Height	Mean Square Error (m^2)	Root Mean Square Error (m)
Worley Park[6]	32.76	32.95	4.41	2.10
Plummer High School[7]	28.46	29.59	3.20	1.79
Plot 33	11.17	11.85	2.01	1.44
Plot 36	3.62	4.44	0.83	0.91
Plot 37	22.74	23.50	1.15	1.07
Plot 38	19.38	21.38	4.63	2.15
Plot 39_40	21.78	22.30	1.65	1.28

A precise tree by tree comparison for the assessment of accuracy of the LiDAR derived tree heights in the Plummer Housing site was not possible due to the difficulty of aligning the LiDAR derived trees with the surveyed trees. The distributions of the LiDAR derived tree heights and the surveyed tree heights were, consequently, compared as shown in Figure 22. For the Plummer Housing site the LiDAR extracted trees that were determined to be false positives through visual analysis were removed from the comparison. The mean height of the surveyed trees was 20.40 m and the mean height of the LiDAR extracted trees was 19.35. The Plummer Housing plot also showed a tendency for the surveyed tree heights to be higher than the LiDAR extracted tree heights.

---

[6] Three gross outliers removed due to likely measurement error. Additionally, two trees where removed between the time of the LiDAR survey and the ground survey.
[7] Two gross outliers removed due to likely measurement error.
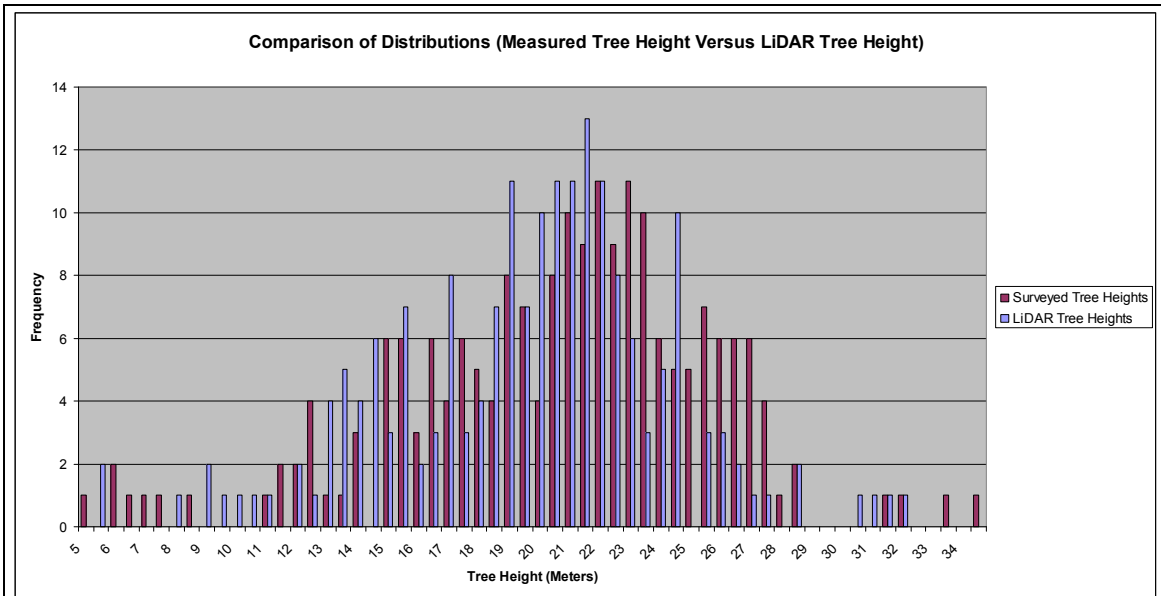
Figure 22 Distributions of LiDAR extracted tree heights and surveyed tree heights.

3.3.3 Crown Radius Accuracy Assessment

The accuracy assessment results for crown radius are shown in Table 11. Only those trees that were extracted correctly from the LiDAR data were considered in this comparison. For each plot the average crown radius as recorded by the LiDAR tree extraction algorithm and the average surveyed crown radius are reported. Additionally, the MSE and RMSE are reported for each site. For most plots the mean crown radii showed a tendency to be higher for the LiDAR extracted crown radii compared to the surveyed crown radii. The Worley Park site did not show this tendency.

Table 11 Crown Radius accuracy assessment results.

Plot	LiDAR Mean Crown Radius (m)	Surveyed Mean Crown Radius (m)	Mean Square Error (m^2)	Root Mean Square Error (m)
Worley Park	4.35	4.56	1.21	1.1
Plummer High School	4.07	4.02	0.78	0.88
Plot 33	2.42	2.17	0.39	0.62
Plot 36	1.91	0.93	1.36	1.16
Plot 37	2.96	2.77	0.85	0.92
Plot 38	4.19	3.73	0.76	0.87
Plot 39_40	3.26	2.55	1.43	1.20

A precise tree by tree comparison for the assessment of accuracy of the LiDAR derived crown radii in the Plummer Housing site was not possible due to the difficulty of aligning the LiDAR derived trees with the surveyed trees. The distributions of the LiDAR derived crown radii and the surveyed crown radii were, consequently, compared as shown in Figure 23. For the Plummer Housing site the LiDAR extracted trees that were determined to be false positives through visual analysis were removed from the comparison. The mean crown radius of the surveyed trees was 2.58 meters and the mean crown radius of the LiDAR extracted trees was 2.66 meters. The Plummer Housing site showed a similar tendency as the other plots for the LiDAR crown radii to be greater than the surveyed crown radii.
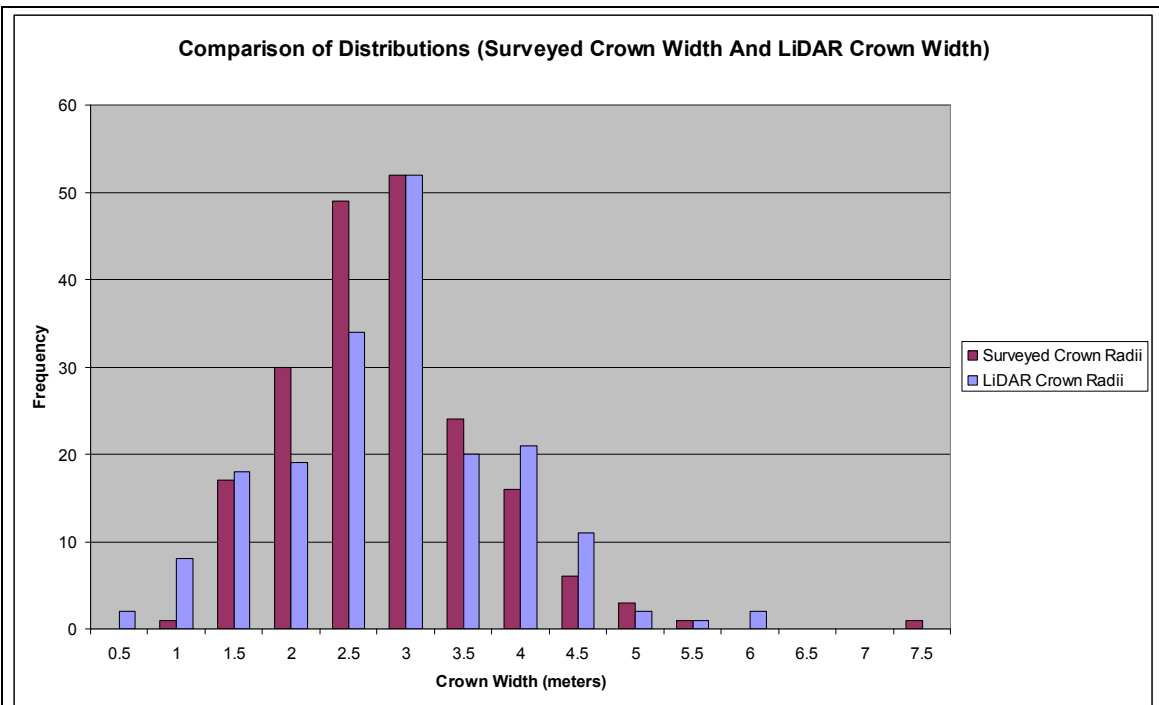
Figure 23 Distributions of LiDAR extracted crown radii and surveyed crown radii.

3.3.4 Height to Live Crown

The accuracy assessment results for HLC are shown in Table 12. Only those trees that were extracted correctly from the LiDAR data were considered in this comparison. For each plot the average HLC as recorded by the LiDAR tree extraction algorithm and the average surveyed HLC are reported. Additionally, the MSE and RMSE are reported for each site. In contrast to the height and crown radii comparisons, there was no apparent tendency for the mean HLC of the LiDAR extracted trees to be systematically different compared to the surveyed HLC.

Table 12 Height to live crown accuracy assessment results.

Plot	LiDAR Mean Height to Live Crown (m)	Surveyed Mean Height to Live Crown (m)	Mean Square Error (m²)	Root Mean Square Error (m)
Worley Park	8.82	6.54	47.88	6.92
Plummer High School	6.10	5.91	29.39	5.42
Plot 33	2.94	4.07	19.44	4.41
Plot 36	2.09	1.65	1.19	1.09
Plot 37	8.22	11.70	43.10	6.57
Plot 38	2.42	4.21	7.11	2.67
Plot 39_40	11.55	12.91	7.93	2.82

A precise tree by tree comparison for the assessment of accuracy of the LiDAR derived HLC in the Plummer Housing site was not possible due to the difficulty of aligning the LiDAR derived trees with the surveyed trees. The distributions of the LiDAR derived HLC and the surveyed HLC were, consequently, compared as shown in Figure 24. For the Plummer Housing site the LiDAR extracted trees that were determined to be false positives through visual analysis were removed from the comparison. The mean HLC of the surveyed trees was 7.82 meters and the mean HLC of the LiDAR extracted trees was 5.95 meters.
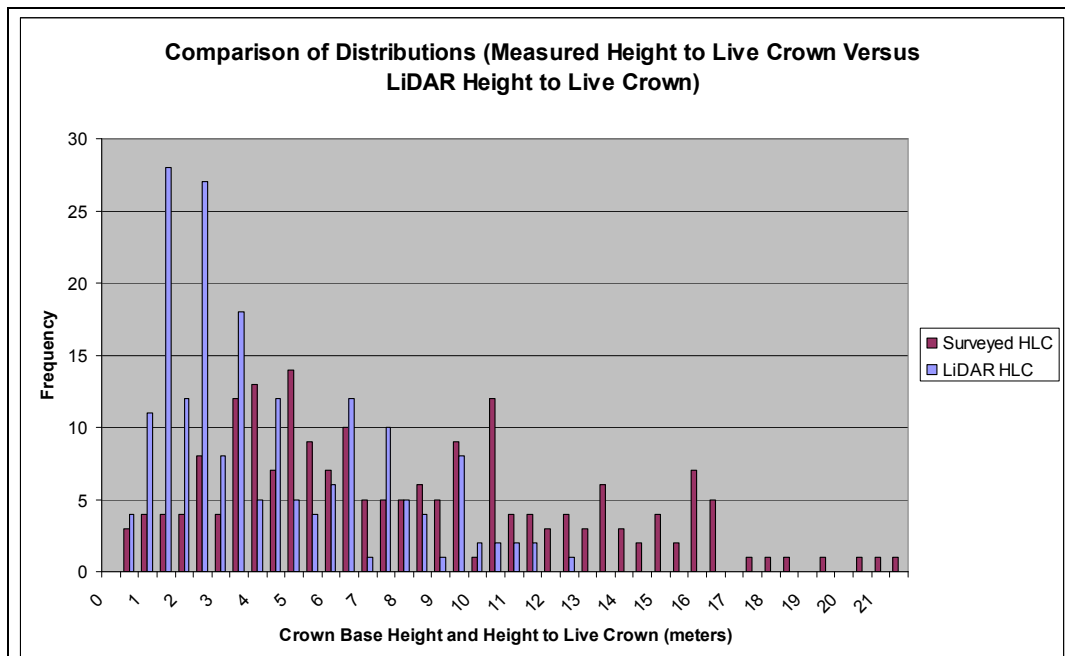

Figure 24 Distributions of LiDAR extracted HLC and surveyed HLC.

3.4 Discussion

Results for the extraction of tree stem locations with associated attributes varied between sites. Many of the results can be cleaned up through manual intervention, thereby improving results. Manual cleanup for certain missed trees, however, is likely only possible when coupled with field data, particularly sub-dominant trees. While the results varied from site to site the use of the algorithms in certain conditions certainly shows promise and can be used for various applications. The effect of point density was also not examined in this study and higher density LiDAR data sets will also likely increase the accuracy of results.

3.4.1 Tree Stem Location

Figure 25 Saplings in plot 36 that were not extracted well.

The accuracy assessment results for the tree stem locations varied widely from site to site ranging from a low of 27% to a high of 97%. This discrepancy in results is largely accounted for by the age class of species present in each site. Those sites with low percentages for correctly extracted trees had a large number of samplings. If saplings are removed from the analysis the plot 33 will improve from 27% (9 of 34 trees correctly extracted) to 66% (4 of 6 trees correctly extracted). Plot 39_40 will improve from 30% (14 of 46 trees correctly extracted) to 82% (14 of 17 trees correctly extracted). Plot 36 was dominated by sapling trees as shown in Figure 25. These saplings could serve as ladder fuels for wildland fire and the correct extraction of these types of trees could be important for fire modeling. It is, however, possible higher density LiDAR data could improve these results for saplings but this was not assessed in this study.

There are two other situations where the algorithm failed to correctly extract trees. As shown in Figure 26,
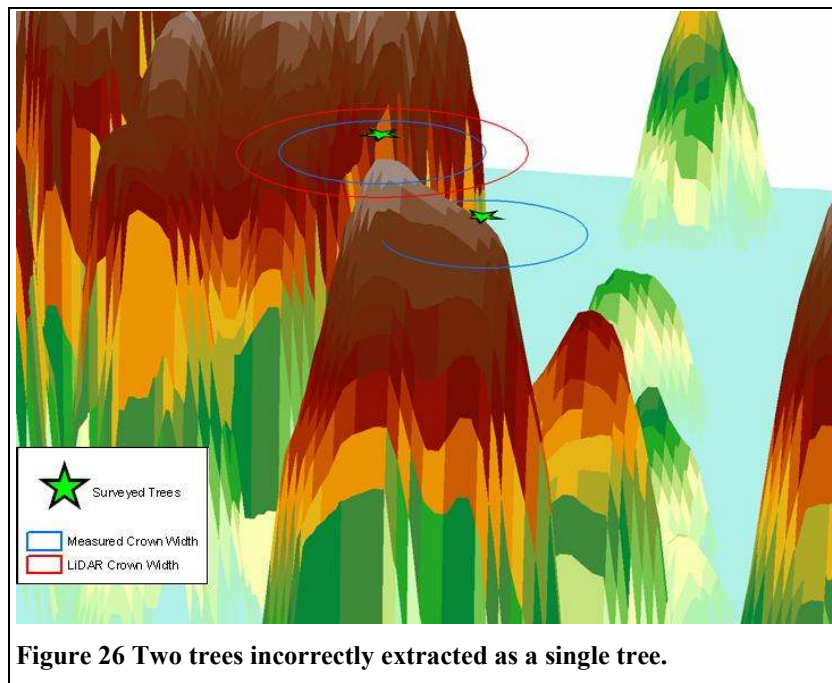

Figure 26 Two trees incorrectly extracted as a single tree.

when two trees are very close together the algorithm will often extract one tree instead of two. The resultant crown width usually encompasses both trees so from a fire modeling perspective it is unknown the result of this type of incorrect feature extraction. It is likely this situation would be alleviated to some extent with higher point density LiDAR data. The algorithm also fails to extract trees whose tree tops are covered by the canopy of a taller tree. It does not appear that higher point density LiDAR data would correct this situation.

3.4.2 Tree Height

The accuracy assessments for tree heights showed reasonable correspondence between the LiDAR derived tree height and the surveyed tree height. The discrepancies where in part due to the time difference between the LiDAR acquisition (i.e. February 2005) and the ground surveys (i.e. summer of 2007). Additionally, it is well known that LiDAR has a tendency to underestimate tree height. This is due to the unlikely probability of the laser pulse hitting the exact apex of the tree crown. Typically, the laser pulse will be returned from the shoulder portions of the tree crown, resulting in some underestimation of tree height. Additionally, slope steepness can affect the resulting tree height. It is also known that the accuracy of LiDAR data decreases with steep slopes and it is possible this or some other anomaly in the creation of the LiDAR derived ground surface could have resulted in a decrease in LiDAR derived tree height.

3.4.3 Crown Radius

The accuracy assessments for crown radius showed a tendency for the LiDAR algorithm to over estimate crown radius compared to ground measurements. The cause of this over estimation is understood and portrayed in Figure 27. In some circumstances, the traverse of the canopy profile will continue into the canopy of another tree while continuing to decrease in height. While the algorithm attempts to account for this by considering the interaction of adjacent canopies this situation will still occur and result in a tendency to over estimate crown radius in some circumstances. This tendency was likely

Figure 27 The cause of over estimation of crown radius.

offset somewhat by the inability of the LiDAR data to measure overlapping canopies between two tree crowns. Ground surveys, however, are able to measure crown radii that are overlapping between different tree canopies. Additionally, examination of the LiDAR derived and ground surveyed mean crown radii at the Worley Park site does not show the same tendency as other plots. This could possibly be due to the fact that the situation described above and portrayed in Figure 27 did not occur as frequently in Worley Park due to the openness of the tree stand. The large discrepancies between LiDAR mean crown radii and ground surveyed crown radii in some plots such as plot 36 is likely due to the large number of saplings and the LiDAR algorithm

grouping many saplings into one tree, thereby resulting in a large over estimation of crown radii. Finally, for stand alone trees or trees on the edge of a stand it is likely that laser returns from the canopy might not portray the exact extent of the canopy due to the unlikely probability of the laser pulse being returned from the exact canopy edge. Increased LiDAR point densities could alleviate this problem.

3.4.3 Height to Live Crown

The accuracy assessment results comparing LiDAR derived HLC to ground surveyed HLC showed large discrepancies between the two measures. There did not appear to be any systematic tendency between the two measures among the different plots. There are, however, some observations about the field data and LiDAR data that can be made. First, LiDAR data cannot distinguish between live and dead vegetation. While the intensity values of LiDAR data when averaged over some set area have shown an ability to distinguish between conifer and deciduous vegetation (e.g. Anderson et al. 2005), this ability at a single branch level would be difficult even at very high point densities. Second, LiDAR data might have a tendency to over estimate HLC. This is due to the unlikely probability of the laser pulse hitting the exact lowest live branch at the lowest point from the ground. This over estimation would be compounded in dense canopies where the laser pulse is intercepted by upper level branches before reaching the lowest vegetation. Next, the field measured HLC did not measure live crown beneath a respective tree that was from an adjacent tree. LiDAR data cannot distinguish from which tree branch vegetation comes and would only consider the lowest returns as described in section 3.2.3. The above factors could result in a combination of over estimation or under estimation depending on the circumstances. The combination of these factors could result in the observance of no systematic tendency between the LiDAR derived HLC and the ground surveyed HLC where certain plots had one factor occurring and other plots a different factor.

3.5 Conclusions

Section 3 of this report presents an algorithm for automatic extraction of tree stem locations with associated attributes of tree height, crown radius and HLC. The tree extraction algorithm uses the common method of identifying local maximum based on allometric relationships between tree height and crown width. This algorithm differs from previous algorithms in that it operates directly on the raw point cloud. This has benefits, particularly in regards to tree extraction in a WUI environment that contains other features that could be confused with trees such as telephone poles. The crown width algorithm is unique in the literature and has the advantage of being independent of the relationship between tree height and height to live crown. The HLC algorithm is similar to previous algorithms presented in the literature.

The accuracy of tree extractions from LiDAR data using the above described algorithm varied between plots. Generally speaking the presented algorithm worked well in extracting dominant tree species in a well spaced stand of trees. The algorithm will likely never identify sub-dominant trees that have tree tops completely encompassed by a dominant tree. It is probably the case that no algorithm could identify these types of trees from LiDAR data. Additionally, this algorithm will not identify sub-dominant trees in a distinct under-story from the dominant trees when the canopy of the dominant trees obscures the downward view of the sub-dominant trees. In theory, the algorithm could be modified to account for this situation. LiDAR derived tree heights and measured tree heights compared well given the time difference between measurement dates. The crown radius algorithm also had mixed results and the tendency to over estimate crown width was identified. The algorithm could be modified to alleviate this situation, thereby improving results. The worst results were seen with the HLC algorithm. There are a number of factors that could

account for the poor results and likely further field data collection and analysis would be required to improve the algorithm.

An important consideration when determining the effectiveness of LiDAR in deriving tree inputs to WFDS is the time and money it takes to survey this information on the ground. Surveying trees for inputs to WFDS is likely impractical over any large area. Additionally, vegetation could be input to WFDS as rectilinear elements and extraction of vegetation inputs to WFDS in this manner from LiDAR data could be simpler than precise tree by tree comparisons. It is also important to understand what the reported discrepancies between LiDAR derived vegetation information and ground surveys means from a WFDS model output standpoint. Various WFDS simulations might help to better understand this. Nonetheless, in certain circumstances such as open, even-aged stands of trees the use of LiDAR and the above described algorithm shows promise for extracting tree inputs to WFDS.

Section 4.0 Fire Barrier Extraction in the WUI

Abstract

Fire barriers are an importance component of Wildland Urban Interface (WUI) assessments. They ultimately distinguish between combustible features and non-combustible features and can be used to analyze and quantify risks to individual structures as well as entire communities based on the presence/absence of fire barriers and their spatial relationship to each other. This work will help in deriving inputs for the Wildland Fire Dynamics Simulator (WFDS) and can help to facilitate or refine fire modeling exercises.

Utilization of an automated method for extracting fire barrier feature types was done using Feature Analyst software. Several experiments were conducted to optimize isolating fire barrier feature types: feature extraction using different imagery resolutions and feature extraction using different band combinations. Overall accuracy is comparable between NAIP imagery (1 meter resolution) and Pictometry imagery (1 foot resolution) and 3 bands vs. the addition of a texture band. The feature extraction software was successful at separating fire barrier feature types from combustible features showing automated methods for feature extraction has the potential for deriving WFDS inputs in a timely and efficient manner. However, both users accuracy (errors of commission) and producers accuracy (errors of omission) was greater than desired when distinguishing between different fire barrier feature types, pavement features and gravel features. In particular, gravel features had high errors of commission (87-90%) and high errors of omission (48-79%) compared to pavement features (34-53% commission; 59-63% omission). A reduction in error rates can be achieved with improvements to the software and changes to the methodology.

4.1 Previous Work

This study stems from the need for the efficient and accurate quantification of fire barriers for input into WFDS. A crucial component of the WFDS model is the distinction of fire barriers from combustible material within a study area. Depending on the region of interest different feature types may represent a potential fire barrier (e.g., roads, lawns, swimming pools, boulders). Two fire barrier feature types were identified for this project: pavement and gravel features. Feature extraction is a method where a user can 'extract' desired features from high-resolution imagery. Each cell of the raster image is analyzed and classified according to feature type. Feature Analyst is a commercial feature extraction software system that leverages multiple object recognition attributes, using semi-automated to automated workflows, to accelerate the collection of features from imagery (Blundell, 1999). Feature Analyst was used as the only feature extraction software through the whole data processing stage. The key components of the Feature Analyst system includes workflow, user interface, and modeling approach for delivering automated feature extraction (AFE) capability to users collecting geospatial intelligence from satellite and aerial imagery sources (Opitz, 1996).

Feature Analyst is an efficient tool to extract features from imagery. In this exercise two layers were extracted (pavement and gravel features). The multiple layers were dissolved into a single polygon feature layer representing a fire barrier. The GeoWFDS fire model input grid layer is a three dimensional grid layer and a DEM was used to obtain height information for the fire barrier feature layer. The fire barrier layer was used as a mask to clip the bare earth DEM resulting in raster layer that had both the fire barrier shape and height information. Next, the GeoWFDS fire model exporting tool assigned the proper surface fuel values. The raster layer will be the final input layer into the WFDS fire model.

4.2 Methods

There are two methods for extracting critical objects in remotely sensed images: manual and automated methods. In the manual method, the features are hand-digitized and manually

attributed.   Although the manual method is still the predominant method for geospatial data production, it involves high labor and costs as well a large time commitment. The Feature Analyst approach to object-recognition and feature extraction is automated and overcomes these shortcomings by using inductive learning algorithms and techniques to model the feature-recognition process, rather than explicitly writing a software program (Maloof, 1998; Burl, 1998). The user gives the system (computer program) a sample of training features from the image. The system then automatically develops a model that correlates known data (such as spectral or spatial signatures) with targeted outputs (i.e., the features or objects of interest) (Blundell, 2006). The learned model then automatically classifies and extracts the remaining targets or objects. The accompanying workflow and metadata (information on spectral bandwidth, date and time stamp, etc.) can be used to quickly compose new models for changing target conditions such as geographic locations or hours of the day (Blundell, 2006). This approach leverages the natural ability of humans to recognize objects in complex scenes (Blundell, 2006).

4.2.1 Feature Analysis Workflow

In the Feature Analyst system, the image analyst creates feature extraction models simply by classifying the objects of interest in a small subset of the image or images (Opitz and Blundell, 1999). Since the user is not required to be familiar with programming, a person with little computer knowledge can effectively create visual models for the tasks under consideration (Opitz, 1999). In addition, different users can focus on the different features of interest, with the system dynamically learning these features (Opitz, 1999). The Feature Analyst workflow includes the following steps:

> 1. Using the Feature Analyst standard (default) toolbar the GIS staff sets up a learning environment for features to be extracted. First, samples of feature types representing fire barriers (e.g., pavement features) are manually digitized. These samples serve as training examples for calibration when extracting a particular feature type from an image. The learning algorithm uses the characteristics (i.e., reflectance) of each pixel within each sample to help derive a shapefile of the feature type of interest across the entire study area. Next, using a menu driven toolbar, learning tools are used to select the type of feature to be extracted (e.g., narrow linear features) and the imagery input bands (Figure 28).

> 2. Features are extracted using Feature Analyst software. All of the learning parameters are set behind the scenes and are calibrated to increase the accuracy of the extraction results and reduce the noise. Four types of feature runs were conducted for pavement and gravel feature types: extraction using NAIP imagery (1 meter resolution), extraction using NAIP imagery with an added texture band, extraction using Pictometry (1 ft resolution), and extraction using Pictometry with an added texture band. Depending upon the input imagery the process took between 1 minute and 20 minutes to complete.

> 3. Results of the outputs are visually inspected. Results were dependent upon the quality and resolution of the input imagery and how well the training samples represented the feature types.

> 4. The GIS staff cleaned up the data. Feature Analyst has built in cleaning tools but we found manually correcting mistakes by removing wrongly classified features and digitizing any missing features was just as time efficient as having the software complete the process. This final step is essential for quality control but it also takes the most time to complete.

5. An accuracy assessment was completed to evaluate the performance of the feature extraction software. A comparison was made utilizing different configuration settings for the software.
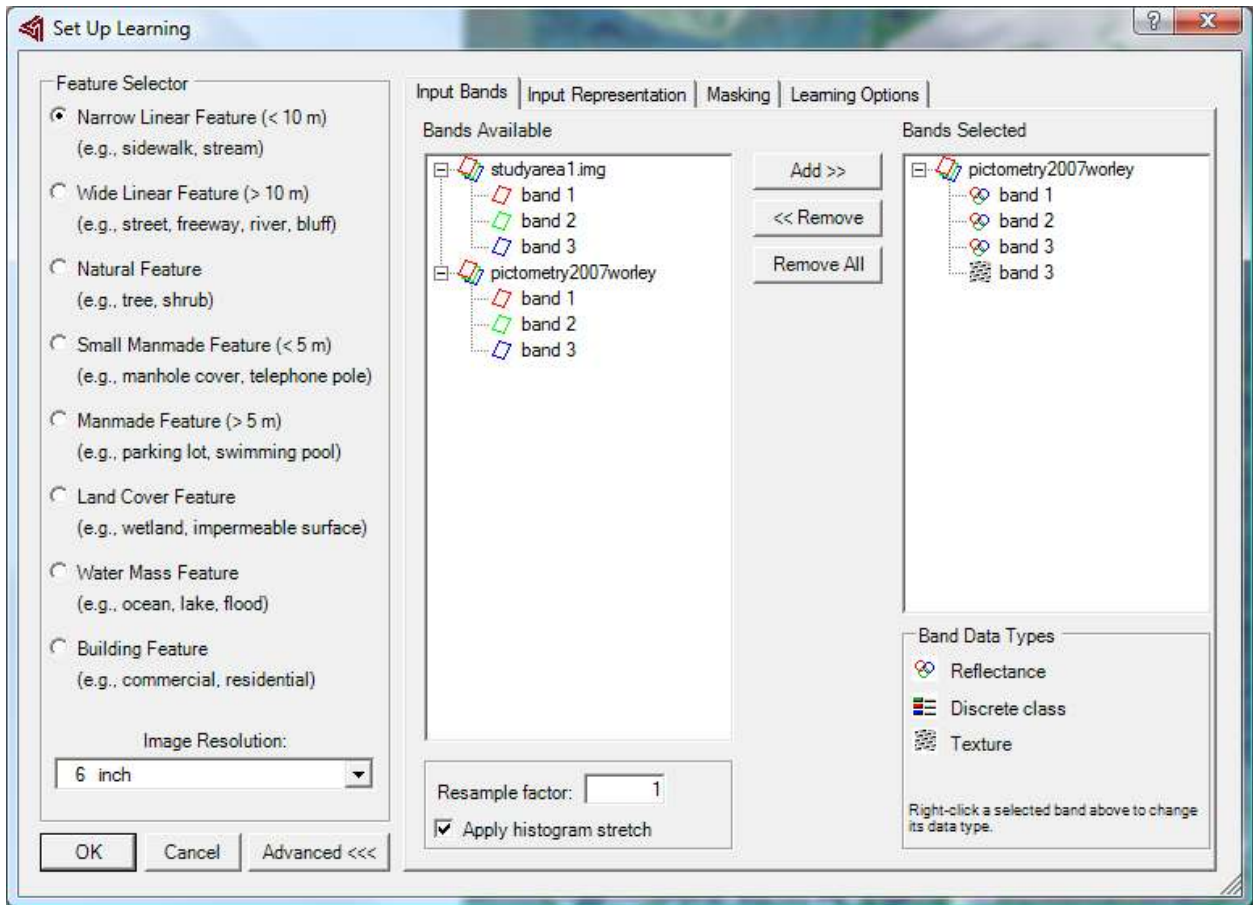
Figure 28 Example Interface of Feature Analyst Software

4.3 Results

This section details results for the two methods of extracting fire barrier feature types. The accuracy of features extracted using NAIP imagery was compared with using higher resolution Pictometry imagery and the input image. Also, the accuracy of automatically extracting pavement and gravel features using a reflectance band was compared against extracting features using both a reflectance band and a texture band.

4.3.1 Visual Inspection of Results

Inspection of pavement derived features showed a high degree of agreement between NAIP extracted features and Pictometry extracted features. The overall area for both datasets was similar and the degree of overlap was strong. Areas of disagreement mostly occurred in non-pavement areas that were misclassified by either of the two datasets. Also, Pictometry had more refined results compared to the coarser NAIP imagery (Figure 29). This came as no surprise because of the differences in resolution of the two imagery datasets. Across the entire study area

it appears that commission errors for pavement features were larger with the Pictometry derived dataset versus the NAIP dataset. There are a number of exposed bare earth patches that were incorrectly classified as pavement using Pictometry imagery.

The addition of a texture band showed noticeable results for the NAIP derived dataset because areas of commission were significantly reduced. However, addition of a texture band to the high resolution imagery showed no visual changes in the dataset nor significant changes in errors of commission or omission. Overall Feature Analyst was an effective tool for extracting pavement features.

With respect to gravel features, Feature Analyst performed poorly. The dataset for gravel features (both NAIP and Pictometry) closely mirrors the pavement dataset. In other words Feature Analyst had a difficult time distinguishing gravel from pavement features. Addition of a texture band did not improve the accuracy of detecting gravel features. Overall the NAIP derived gravel dataset overestimates the presence of gravel features while the Pictometry derived gravel dataset underestimates the amount of gravel features.

4.3.2 Pavement and Gravel Features Accuracy Assessment

An error matrix was produced to quantify the results of extracting fire barrier feature types using an automated method. It can identify overall errors rates as well as error of commission and errors of omission. For this study an error matrix was generated using

Figure 29 Feature extraction using coarser NAIP imagery misclassified a vehicle and parking lot lines as pavement; Pictometry results corrected identified non-pavement features.

methods described in Campbell (1996). A total of 4 error matrices were produced to quantify how well Feature Analyst performed with different datasets: 3 band NAIP image, 4 band NAIP image, 3 band Pictometry image and 4 band Pictometry image (Table 13).

Results show overall accuracy results are better using Pictometry (87-91%) imagery compared to the more coarse resolution of NAIP imagery (83%). However, closer inspection illustrates NAIP outperforms Pictometry for certain features. For instance, NAIP has a much better user's accuracy for pavement features (57-66%) compared to Pictometry derived features (47-48%). Also, the producer's accuracy is better for NAIP for gravel features (49-52%) compared to Pictometry (21-23%). A comparison between 3 and 4 band products (within the same image dataset) reveals minimal change in the user's and producer's accuracy. Overall accuracy rates also show little variance but the kappa analysis is greatly improved when adding a texture band on the Pictometry dataset; no change was observed for NAIP. Finally, the value $K^\wedge$ (kappa coefficient) is a measure to determine how the classification compares to a random assignment of points.
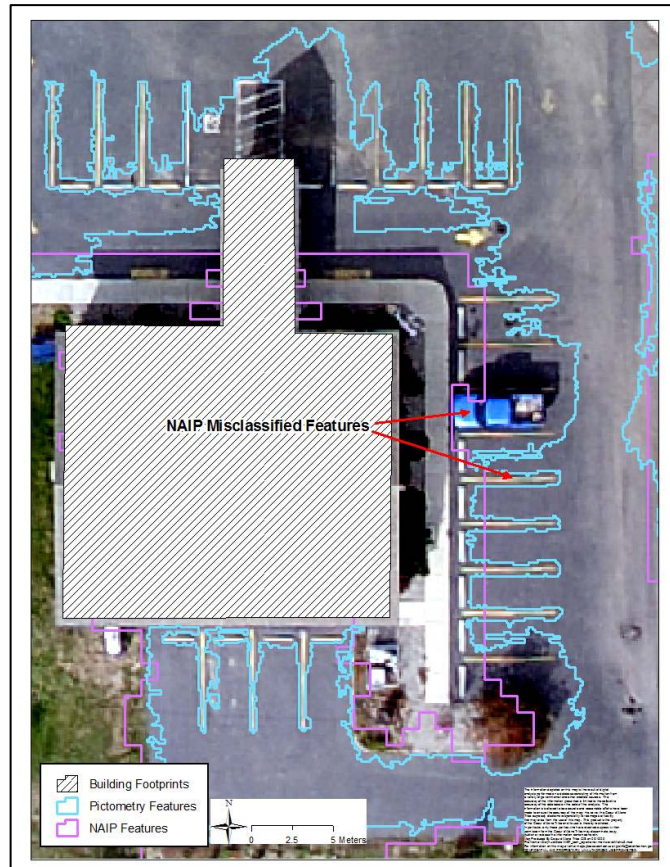
With the exception of the Pictometry 4 band results, overall kappa values are less than 0.5. In other words, the classification of fire barrier features is no better than randomly assigning a value to each point.

Table 13 Error Matrix results from Feature Analyst

a) NAIP 3 band

	SEGMENTED DATA					
CLASS	Pavement	Gravel	Other	Row Total	Producer's Accuracy (%)	Errors of Ommission (%)
Pavement	88	107	37	232	37.93%	62.07%
Gravel	16	53	32	101	52.48%	47.52%
Other	49	278	2508	2835	88.47%	11.53%
Column Total	153	438	2577	3168		
User's Accuracy (%)	57.52%	12.10%	97.32%		Overall Accuracy (%)	83.62%
Errors of Commission (%)	42.48%	87.90%	2.68%		K^	0.38

(REFERENCE DATA)

b) Pictometry 3 band

	SEGMENTED DATA					
CLASS	Pavement	Gravel	Other	Row Total	Producer's Accuracy (%)	Errors of Ommission (%)
Pavement	92	97	33	222	41.44%	58.56%
Gravel	24	23	62	109	21.10%	78.90%
Other	78	100	2672	2850	93.75%	6.25%
Column Total	194	220	2767	3181		
User's Accuracy (%)	47.42%	10.45%	96.57%		Overall Accuracy (%)	87.61%
Errors of Commission (%)	52.58%	89.55%	3.43%		K^	0.42

(REFERENCE DATA)

c) NAIP 4 band

	SEGMENTED DATA					
CLASS	Pavement	Gravel	Other	Row Total	Producer's Accuracy (%)	Errors of Ommission (%)
Pavement	87	110	38	235	37.02%	62.98%
Gravel	17	50	35	102	49.02%	50.98%
Other	27	304	2485	2816	88.25%	11.75%
Column Total	131	464	2558	3153		
User's Accuracy (%)	66.41%	10.78%	97.15%		Overall Accuracy (%)	83.16%
Errors of Commission (%)	33.59%	89.22%	2.85%		K^	0.37

(REFERENCE DATA)

d) Pictometry 4 band

	SEGMENTED DATA					
CLASS	Pavement	Gravel	Other	Row Total	Producer's Accuracy (%)	Errors of Ommission (%)
Pavement	87	93	38	218	39.91%	60.09%
Gravel	23	25	60	108	23.15%	76.85%
Other	69	97	2678	2844	94.16%	5.84%
Column Total	179	215	2776	3170		
User's Accuracy (%)	48.60%	11.63%	96.47%		Overall Accuracy (%)	91.36%
Errors of Commission (%)	51.40%	88.37%	3.53%		K^	0.58

(REFERENCE DATA)

4.4 Discussion

Our study showed mixed results using Feature Analyst to extract fire barrier feature types. Datasets derived from imagery using a coarser resolution (1 meter) produced results that matched, or even exceeded, a higher resolution dataset (<1 foot). High resolution imagery was more accurate classifying features that align correctly with matching features in the imagery but overall errors of commission and omission were higher with Pictometry datasets because of a greater number of misclassified polygons. In terms of overall performance a higher resolution image produces better accuracy results and thus is preferred but if costs are a concern a user should see comparable results using a free dataset, such as NAIP imagery, for their analysis. Feature extraction was done using the reflectance band (3 color bands) along with the addition of a 4th band, texture. Adding a texture band had a negligible effect at best when discerning pavement or gravel feature types. We expected greater improvements in accuracy but both visual inspection of the dataset and analysis of the error matrix failed to show noticeable results.

When comparing results between feature types the overall accuracy was better for pavement features versus results from gravel feature extraction. Results of gravel features often overlapped or coincided with pavement features (Figure 30) and Feature Analyst had a difficult time distinguishing actual gravel features from either pavement or other bare earth feature types. Within the study area there are many bare earth features that closely resemble pavement and

gravel feature types. For instance, plowed agricultural fields and other exposed dirt features are similar in both reflectance and texture values. Despite having a representative training sample of gravel features their spectral makeup was too generic for Feature Analyst to produce a quality dataset. This lead to both high commission and omission error rates. Meanwhile, pavement features tend to be darker in color thus making them more distinguishable from other features types.

The final product for this exercise was to produce a single fire barrier feature layer from an assortment of non-combustible feature types. Feature extraction software seemed to be capable, in addition to being efficient, as a means of classifying pavement and gravel features from features at risk from fires (structures or vegetation). The difficultly lies in distinguishing between gravel, pavement, and bare earth features because of similar spectral and texture compositions. High quality data and refinements in feature extraction software will help to reduce error rates when classifying these feature types.
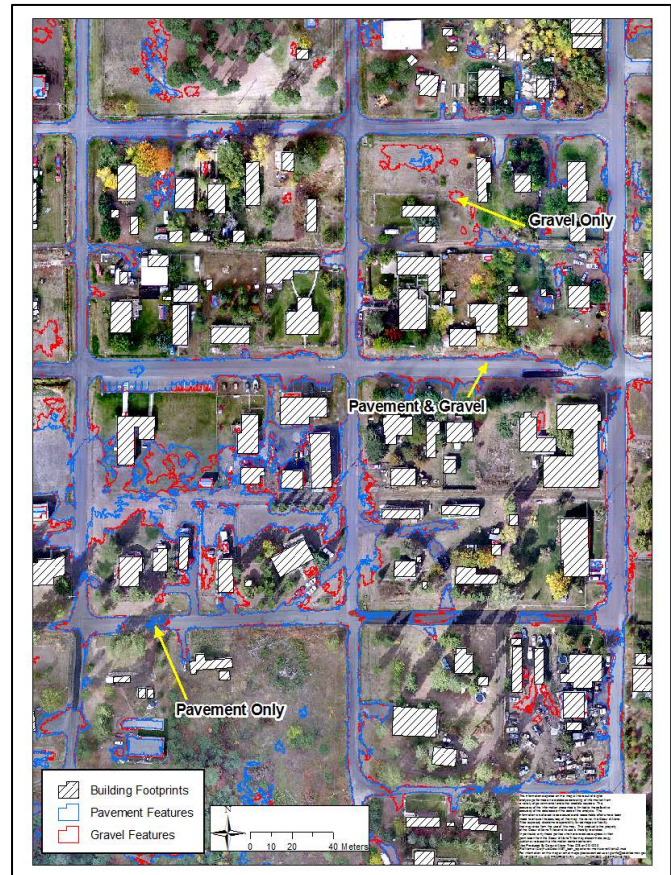

Figure 30 Results for pavement and gravel features. Note the high degree of overlapping between the two.

4.5 Conclusions

Section 4 of this report discussed extracting fire barrier feature types from imagery using an automated approach versus the traditional method of manually classifying features. Our experiments show that Feature Analyst software can be effectively used to extract certain feature types. Comparisons between different sources of imagery revealed minimal differences in a visual comparison and slightly more accurate results, in terms of accuracy rates, using NAIP imagery. In terms of extracting feature types, the addition of a texture band did not reduce the degree of misclassification errors. Overall error matrix results were exceeded what was expected but the software was effective at identifying pavement feature types and discerning combustible from non-combustible features. However, quality control is necessary to ensure the end product meets accuracy requirements.

Section 5.0 GIS to WFDS Data Transfer

The application of the above described geospatial technologies resulted in the creation of geospatial data sets describing the properties and spatial extent of man-made and natural features to be input to WFDS. Methods, however, to transfer this geospatial data to a format suitable to be used as input files for WFDS were required. As the WFDS inputs were derived using a GIS it was determined that a GIS would be a logical choice as a platform to base the development of a software application to facilitate the transfer of the WFDS inputs stored in GIS format to WFDS input files. A GIS was also a logical choice because of the potential for writing WFDS model outputs back to the GIS for subsequent spatial analysis of WFDS model runs.

Initially a free GIS viewer developed by ESRI called ArcReader was modified to create an application to allow for the transfer of GIS data to WFDS input files. This simple application shown in Figure allowed users to view the WFDS inputs in GIS format along with ancillary data such as aerial imagery. Users could navigate around the data and enter a point in the map window representing the bottom left extent of the WFDS input file to be created. The resultant form that appeared allowed users to specify the extent and other characteristics for the WFDS input file to be created. This application required the use of licensed ESRI software to create the files used for the application and had a number of limitations including a 2 gigabyte size limit for the GIS database representing the WFDS inputs. This application was used as a stop-gap measure to provide NIST personnel with a simple to use interface to create small WFDS input files from the GIS data sets created for this project.

As the project progressed a more sophisticated application was required to allow for creation of WFDS input files of any size. This application was again built off of the ESRI GIS platform and required users to have an authorized license of this software. In addition to the development of

Figure 31 Initial application to create WFDS input files from GIS inputs.

the software application, a custom spatial database schema termed a geodatabase was developed to allow for the consistent storage of WFDS inputs in GIS format. This can facilitate data transfer amongst researchers and the consistent creation of WFDS inputs. The software application was termed GEOWFDS and takes raster and vector GIS data with associated tabular data and creates WFDS inputs files over a user specified extent. In addition, GEOWFDS facilitates the linking of WFDS model run information, output to thermocouples contained in the WFDS input file, back to the GIS for further analysis. Finally, GEOWFDS allows for the inputting of distinct vertical layers of vegetation as well as vegetation as points as described in section 3.0. Deriving vegetation as vertical layers or height bins might be a more robust method of deriving the 3D structure of vegetation from LiDAR data compared to extracting tree stem locations, in some circumstances.

A user manual was developed by McNamara Consulting, Inc. detailing all of the functionality of the GEOWFDS program. In addition, tutorials were developed demonstrating the proper usage of GEOWFDS with both high and coarse resolution data sets. The user manual, tutorial and example data can be downloaded at:

http://www.mcnamara-consulting.com/Default.aspx?Page=Software/GEOWFDS.html

Section 6.0 Recommendations

The use of geospatial technologies for derivation of inputs to WFDS is likely the only practical means of obtaining these inputs over any significant area. Whether WFDS inputs are derived using manual GPS surveys, manual photogrammetric techniques or completely automatic extractions from remote sensing data, the use of WFDS in understanding fire behavior in the WUI can be enhanced through the use of geospatial technologies. The use of information systems for generating inputs to FDS has been occurring for sometime (e.g. DXF2FDS and PyroSim). The WFDS-GIS linkage created for this grant extends this use to GIS to allow for further exploitation of geospatial data sets that are becoming more prevalent in the WUI. While this project is the first known assessment of deriving inputs for WFDS, it touches upon only a few aspects of deriving these inputs and does not examine some WFDS inputs such as terrain and the material properties of vegetation. Additionally, only a few methods of deriving the WFDS inputs have been examined and other methods could hold promise for more efficient and effective creation of WFDS inputs. Finally, the actual level of accuracy and precision required for WFDS inputs remains unanswered. This section provides recommendations for future research related to the use of geospatial technologies with WFDS.

Terrain is a key input to WFDS and future studies could focus on ascertaining the importance of the following items related to deriving terrain information for WFDS:

- Assessing the resolution required to produce valid WFDS model outputs.

- Determining the importance of creating digital terrain models (DTM)[8] from DEMs.

- Interpolation and conversion methods used in creating the input DEM and converting to WFDS input data structures.

Assessments involving the resolution required to produce valid WFDS runs could aid in cost effectiveness of creating WFDS terrain inputs where coarser resolution inputs would be cheaper to create and more readily available. Creating DTMs that incorporate sharp geographic features like ridgelines and other topographic features could be important to producing valid WFDS model outputs. For instance, it is common practice when using DEMs for hydrologic modeling to ensure the resultant DTM does not contain spurious sinks or spikes that interfere with the proper modeling of water flow across the landscape. The importance of ensuring these sinks and spikes[9] are not present in DEMs used for creating WFDS inputs is not known by the authors. Finally, various interpolation routines used to create the DEMs could be superior to others. This question is being actively studied as related to creating DEMs from LiDAR data.

This study also did not examine the extraction of the material properties of vegetation including the derivation of crown bulk density from LiDAR data. The extraction of this type of information is also being studied extensively in the literature (e.g. Skowronski et al. 2007). As with many other inputs the effectiveness of the use of LiDAR data in deriving this WFDS input largely depends on the level of accuracy and precision required. Nonetheless, the potential that LiDAR has demonstrated in deriving the crown bulk density of vegetation is demonstrated by the availability of open source tools capable of deriving this information from LiDAR data (e.g. United States Forest Service Canopy Fuel Estimator).

---

[8] A DTM incorporates features that represent sharp breaks in the landscape such as ridgelines that might be smoothed or lost when creating a DEM.
[9] The DEMs created for this study where checked for any large-scale anomalies.

6.1 Building Extraction

The use of LiDAR data and multi-spectral imagery for the extraction of the 3D structure of buildings in the WUI showed great promise. The derivation of the 3D structure of buildings from high-resolution remotely sensed data can be achieved with reasonable accuracy using automated techniques. The extraction of these inputs with extremely high accuracy is also feasible when manual intervention by experienced specialists occurs. It is recommended that future studies on extracting building information inputs for WFDS focus on the following items:

- Assessing the resolution required to produce valid WFDS outputs.

- Ascertaining the effectiveness of using multi-spectral data from other portions of the electromagnetic spectrum, including infra-red data and hyper-spectral data.

- Automatic extraction of the material properties of buildings from remotely sensed data.

Coarser resolution inputs to WFDS could be cheaper and easier to obtain. Extracting the 3D structure of buildings from coarser resolution LiDAR data than what was used in this study, however, could result in the need for more manual intervention to produce accurate results. The use of other types of multi-spectral data will likely increase the accuracy of automatic building extractions due to infra-red and hyper-spectral data being better able to distinguish vegetation from man-made features compared to color data. Finally, the use of remotely sensed data for extracting the material properties of buildings could greatly reduce the costs of deriving WFDS inputs.

This study used material properties of buildings derived from ground surveys as described in section 1.2.1. These ground surveys are time and labor intensive. WFDS inputs for material properties of buildings come from the NFPA form 1144a. Technically, roof properties can only be determined by on the ground surveys because the categories depend on how the roof was constructed internally. Nonetheless, the availability of using oblique imagery for determining building siding material could be possible and alleviate ground survey costs. Additionally, the use of remotely sensed data for determining certain roof properties such as distinguishing between asphalt, tile or wood shingle roof types could be possible and also reduce ground survey costs.

6.2 Vegetation Extraction

The use of LiDAR for extracting tree stem locations showed promise for even-aged stand of trees in open environments. The LiDAR tree extraction results where not as promising for sub-dominant trees. Additionally, this study did not examine extraction of other types of vegetation such as trees and shrubs. As such it is recommended that future studies on extracting vegetation information inputs for WFDS focus on the following items:

- Assessing the resolution required to produce valid WFDS outputs.

- Using remotely sensed data for deriving the material properties of vegetation (e.g. crown bulk density).

- Extracting vegetation as 3D rectilinear elements.

As with all other WFDS inputs, obtaining these with coarser resolution remotely sensed data could help to reduce costs. The use of LiDAR with other types of remotely sensed imagery for the derivation of the material properties of vegetation such as crown bulk density is important for deriving WFDS inputs. Finally, it is recommended to examine voxel based approaches to extracting vegetation. These approaches are likely required to extract certain types of vegetation such as shrubs and could prove simpler then precise tree stem extractions.

6.3 Fire Barrier Extraction

Overall, the use of an automated method for extracting fire barrier feature types showed promise. Feature extraction software was able to successfully identify fire barriers from combustible features. However, when it came to identifying pavement features from gravel features the rates of error were larger than desired. It is recommended that future studies on an automated method of extracting fire barrier feature types focus on the following items:

- Inspect methods to smooth features resulting in less jagged edges.

- Time efficient way for editing/cleaning results (i.e., eliminating 'donut holes' appearing in the middle of a feature.

In addition, an assessment extracting features using an infrared/near infrared band may is recommended to see if error rates are reduced when distinguishing between specific feature types. 2009 NAIP imagery has just been recently released. It consists of a fourth, infrared band, that would make this analysis possible.

6.4 WFDS-GIS Linkage

The development of the linkage between WFDS and a GIS was required to facilitate the transfer of information from GIS data sets to a format capable of being used by WFDS. The presented application represents a first step in that direction. It is recommended that future work related to the development of the WFDS-GIS linkage focus on the following items:

- Incorporating the ability to create WFDS inputs of wind from GIS data sets representing wind vectors.

- Transferring data directly from the industry standard LAS LiDAR format to WFDS input files.

- Reading WFDS model outputs back to the GIS for further spatial analysis.

The current version of GEOWFDS does not facilitate the incorporation of wind vectors, which are often times an important component of understanding fire behavior. Allowing the direct transfer of information stored in the LiDAR point cloud, or LAS file, to WFDS could reduce data processing needs and better facilitate the transfer of vegetation as voxels to WFDS input files. Finally, the ability to read information contained in WFDS output files back to a GIS could allow for the exploiting of the spatial analysis capabilities inherent in GIS and might better help in understanding fire behavior.

Section 7.0 References

Al-Harthy, A., and J. Bethel. 2002. "Heuristic Filtering and 3D Feature Extraction." *Proceedings of ISPRS Commission III Symposium.* Graz, Austria. September 9-13, 2002.

Anderson H.E., R.J. McGaughey, S.E. Reutebuch. 2005. "Estimating Forest Canopy Fuel Parameters using LIDAR Data." *Remote Sensing of Environment.* 94(2005): pp. 441-449.

Baltsavias, E., S. Mason, and Dd. Stallmann. 1995. "Use of DTMs/DSMs and Orthoimages to Support Building Extraction." *Automation of Digitial Terrain Model Generation and Man-Made Object Extraction from Aerial Images.* Institute of Geodesy and Photogrammetry. Zurich, Switzerland.

Baltsavias, E.P., and A. Gruen. 2003. *Remotely Sensed Cities.* London: Taylor and Francis, London, pp. 474-482.

Barilotti, A., F. Crosilla, and F. Sepic. 2009. "Curvature Analysis of Lidar Data for Single Tree Species Classification in Alpine Latitude Forests." *ISPRS Workshop: Laserscanning 09.* Paris, France. September 1-2, 2009.

Blundell, Opitz. 2006. "Object recognition and feature extraction from imagery". 1st International Conference on Object-based Image Analysis (OBIA 2006)

Brenner C., 2005. "Building Reconstruction from Images and Laser Scanning." *International Journal of Applied Earth Observation and Geoinformation.* 6, pp. 187-198.

Brunn, A., and U. Weidner, 1997. "Extracting Buildings from Digital Surface Models." *International Archives of Photogrammetry and Remote Sensing,* 32(3-4): pp. 27-34.

Burgan, R.E., and R.A. Hartford. 1993. "Monitoring Vegetation Greenness with Satellite Data." *General Technical Report INT-297.* Ogden, UT. Department of Agriculture, Forest Service, Intermountain Research Station.

Campbell, J. B., 1996. *Introduction to Remote Sensing, 2nd ed.* New York: The Guilford Press. pp. 32-33.

Chasmer, L., C. Hopkins, P. Treitz. 2004. "Assessing the Three-Dimensional Frequency Distribution of Airborne and Ground-Based Lidar Data for Red Pine and Mixed Deciduous Forest Plots." *International Archives of Photogrammetry, Remote Sensing and Spatial Infromation Sciences.* 36(2004): pp. 66-70.

Chen, L., T. Teo, C. Hsieh, and J. Rau. 2006. "Reconstruction of Building Models with Curvilinear Boundaries from Laser Scanner and Aerial Imagery." *Advances in Image and Video Technology.* Springer Berlin/Heidelberg. December 9, 2006.

Cheuk, M.L., and M. Yuan. 2009. "Assessing Spatial Uncertainty of Lidar-Derived Building Model: A Case Study in Downtown Oklahoma City." *Photogrammetric Engineering and Remote Sensing.* 75(3): pp. 257-269.

Chen, Q., D. Baidocchi, P. Gong, and M. Kelly. 2006. "Isolating Individual Trees in a Savanna Woodland using Small Footprint Lidar Data." *Photogrammetric Engineering and Remote Sensing.* 72(8): pp. 923-932.

Cho, W., Y. Jwa, H. Chang, and S. Lee. 2004. "Pseudo-Grid Based Building Extraction Using Airborne LIDAR Data." *ISPRS Congress Istanbul 2004.* Istanbul, Turkey. July 12-23, 2004.

Cohen, W.B., and T.A. Spies. 1992. "Estimating Structural Attributes of Douglas-Fir/Western Hemlock Forest Stands from Landsat and Spot Imagery." *Remote Sensing of Environment.* 41(1992): pp. 1-17.

Coops, N.C., M.A. Wulder, D.S. Culvenor, and B. St-Onge. 2004. "Comparison of Forest Attributes Extracted from Fine Spatial Resolution Multipsectral and Lidar Data." *Canadian Journal of Remote Sensing.* 30(6): pp. 855-866.

Engels, J., H. Arefi, and M. Hahn. 2008. "Generation of Roof Topologies Using Plane Fitting with RANSAC." *ISPRS Congress Beijing 2008.* Beijing, China. July 3-11, 2008.

[1]Evans, D., E. McPherson, and J. Wallace, 2004. "Physics-Based Modeling of Community Fires." *International Interflam Conference,* Edinburgh, Scotland, 12(2004): pp. 11.

[2]Evans, D.D., R.G. Rehm and E.S. Baker. 2004. "Physics-Based Modeling for WUI Fire Spread – Simplified Model Algorithm for Ignition of Structures by Burning Vegetation." Gaithersburg, MD: Fire Research Division, Building and Fire Research Laboratory, National Institute of Standards and Technology.

Evans, J. S., and A.T. Hudak. 2007. "A Multiscale Curvature Algorithm for Classifying Discreet Return LiDAR in Forested Environments." *IEEE Transactions on Geoscience and Remote Sensing.* 45(4): pp. 1029-1038.

Falkowski, M.J., A. Smith, A.T. Hudak, P.E. Gessler, L.A. Vierling, and N.L. Crookston. "Automated Estimation of Individual Conifer Tree Height and Crown Diameter Via Two-Dimensional Spatial Wavelet Analysis of LiDAR Data." *Canadian Journal of Remote Sensing.* 32(2): pp. 153-161.

Fradkin, M., M. Roux and H. Matre. 1999. "Building Detection from Multiple Views." *ISPRS Conference on Automatic Extraction of GIS Objects from Digital Imagery.* Technische Universitat Munchen, Germany. September 8-10, 1999.

Gamba, P., and B. Houshmand. 2000. "Digital Surface Models and Building Extraction: A Comparison of IFSAR and LIDAR Data." *IEEE Transactions on Geoscience and Remote Sensing.* 38(4): pp. 1959-1968.

Gamba, P., F. Dell'Acqua, and M. Cesari. 2005. "Three-Dimensional Object Recognition in LIDAR Data Using a Planar Patch Approach." *3rd International Symposium Remote Sensing and Data Fusion Over Urban Areas (URBAN 2005).* Tempe, AZ. March 14-16, 2005.

Gougeon, F. 1997. "Recognizing the Forest from the Trees: Individual Tree Crown Delineation, Classification and Regrouping for Inventory Purposes." *Proceedings of*

Airborne Remote Sensing Conference and Exhibition. Copenhagen, Denmark, 2(1997): pp. 807-815.

Gruen, A. 2000. "Potential and Limitations of High-Resolution Satellite Imagery." *Proceedings of the Asian Conference on Remote Sensing.* Taipei, Taiwan. December 4-8, 2000.

Gulch, E. 2000. "Digital Systems for Automated Cartographic Feature Extraction." *International Archives of Photogrammetry and Remote Sensing and Spatial Information Sciences.* Vol XXXIII, Part B2, pp. 241-255.

Haithcoat, T., W. Song, and J. Hipple. 2001. "Building Extraction – LIDAR." *R&D Program for NASA/ICREST Studies.* September 16, 2001.

Hall, S.A., I.C. Burke, D.O. Box, M.R. Kaufmann, and J.M. Stoker. 2005. "Estimating Stand Structure Using Discrete-Return Lidar: an Example from Low Density, Fire Prone Ponderosa Pine Forests.*" Forest Ecology and Management,* 208(1-3): pp. 189-209.

He, H.S., 2008. "Forest Landscape Models: Definitions, Characterization, and Classification." *Forest Ecology and Management,* 254(2008): pp. 484-498.

Hill, R.A., and A.G. Thomson. 2005. "Mapping Woodland Species Composition and Structure Using Airborne Spectral and LiDAR Data." *International Journal of Remote Sensing.* 26(2005): pp. 3763-3779.

Hollaus, W., and W. Wagner. 2006. "Operational Use of Airborne Laser Scanning for Forestry Applications in Complex Mountainous Terrain." *9th Internnational Symposium on High Mountain Remote Sensing Cartography.* Graz and Hohe Tauern National Park, Austria. September 14-22, 2006. pp. 19-26.

Holmgren, J., M. Nilsson, and H. Olsson. 2003. "Estimation of Tree Height and Stem Volume on Plots Using Airborne Laser Scanning." *Forest Science.* 49(2003): pp. 419-428.

Holmgren, J., and A. Persson. 2004. "Identifying Species of Individual Trees Using Airborne Laser Scanner." *Remote Sensing of Environment.* 90(4): pp. 415-423.

Huang, X., X. Cheng, F. Zhang, J. Gong. 2008. "Side Ration Contrain Based Precise Boundary Tracing Algorithm for Discrete Point Clouds." *ISPRS Congress Beijing 2008.* Beijing, China. July 3-11, 2008.

Huertas A., and R. Nevatia, 1988. "Detecting Building in Aerial Images." *Computer Vision, Graphics and Image Processing,* 41(2): pp. 131-152.

Huertas, A., Z. Kim, and R. Nevatia. 1998. "Use of IFSAR with Intensity Images for Automatic Building Modeling." *DARPA Image Understanding Workshop.* Monterey, CA. November, 1998.

Huertas, A., R. Nevatia, D. Landgrebe. 1999. "Use of Hyperspectral Data with Intensity Images for Automatic Building Modeling." *U.S. Army Research Office Grant No. DAAH04-96-1-04444.*

Hyypa, J., O. Kelle, M. Lehikoinen, and M. Inkinen. 2001. "A Segmenetation-Based Method to Retrieve Stem Volume Estimates from 3-D Tree Height Models Produced by Laser Scanners." *IEEE Transactions on Geosciences and Remote Sensing.* 39(2001): pp. 969-975.

Ibrahim, S. 2005. "Feature Extraction and 3D City Modeling Using Airborne LIDAR and High-Resolution Digital Orthophotos." The University of Texas at Dallas. March, 2009. Geographic Information Sciences. Fall, 2005. <http://epps.utdallas.edu/mgis/prj_mstrs/2005/Fall/Ibrahim/Sulafa%20Ibrahim_MastersWebsite_Fall2005/index.htm>

Irvin, R., and D. Mckeown, 1989. "Methods for Exploiting the Relationship Between Buildings and their Shadows in Aerial Imagery." *IEEE Transactions on Systems, Man and Cybernetics,* 19(6): pp. 1564-1575.

Kamp, M, and N. Sampson. "Using GIS to Identify Potential Wildland-Urban Interface Areas Based on Population Density." General Conservation Topics. March, 2009. The Sampson Group. Unknown Date.

< http://www.sampsongroup.com/Papers/wui_paper.pdf>

Kean, R.E., R. Burgan, J. Wagtendok, 2001. "Mapping Wildland Fuels for Fire Management Across Multiple Scales: Integrating Remote Sensing, GIS, and Biophysical Modeling." *International Journal of Wildland Fire,*, 10(2001): pp. 301-319.

Lim, K., and P. Treitz. 2004. "Estimation of Aboveground Forest Biomass from Airborne Discrete Return Laser Scanner Data Using Canopy-Based Quantile Estimators." *Scandinavian Journal of Forest Researc.* 19(2004): pp. 558-570.

Lunetta, R. S., J. G. Lyon. 2004. Remote sensing and GIS accuracy assessment

Mass, G-H. 1999. "The Potential of Height Texture Measures for the Segmentation of Airborne Laser Scanner Data." *4th Airborne Remote Sensing Conference and Exhibition*, Ontario, Canada.

Matlkalnen, L., J. Hyyppa, H. Kaartlnen. 2009. "Comparison Between First Pulse and Last Pulse Laser Scanner Data in the Automatic Detection of Buildings." *Photogrammetric Engineering & Remote Sensing.* 75(2): pp. 133-146.

Mayunga, S.D., Y. Zhang, and D.J. Coleman. 2005. "Semi-Automatic Building Extraction Utilizing Quickbird Imagery." *Proceedings of the ISPRS Workshop CMRT 2005 Object Extraction for 3D City Models, Road Databases and Traffic Monitoring – Concepts, Algorithms and Evaluation.* Vienna, Austria. 29-30 Aug 2005.

McGrattan, K.B., B. Klein, S. Hostikka, and J. Floyd. (2009). "Fire Dynamics Simulator (Version 5) User's Guide." *NIST Special Publication 1019-5.* National Institute of Standards and technology, Gaithersburg, MD.

McNamara, D. (2006) "Extracting Building Footprints from LiDAR and Aerial Imagery in the Wildland Urban Interface." 2006 *Northwest GIS Conference*. Spokane, WA.

McNamara, D. (2008) "Using LiDAR and Multispectral Data for Feature Extraction in the Wildland Urban Interface." *2008 Intermountain GIS Conference.* Missoula, MT.

Mell, W, 2007. "Wildland Urban Interface Fires." *National Institute of Standards and Technology Fire Research Division 2007 Annual Fire Conference.* April 4-5, 2007. Gaithersbug, MD.

[1]Mell, W. "Wildland-Urban Interface (WUI) Fires – Experiements and Modeling." Building and Fire Research Laboratory. February, 2009. NIST. June 2008. <http://www2.bfrl.nist.gov/userpages/wmell/PUBLIC/WUI/NIST_WUI_overview.pdf>.

[2]Mell. W. "Wildland-Urban Interface and Wildland Fires." Building and Fire Research Laboratory. February, 2009. NIST. July 2008.

< http://www2.bfrl.nist.gov/userpages/wmell/public.html>.

Morgan, M., and A. Habib. 2002. "Interpolation of LiDAR Data and Automatic Building Extraction." *ACSM-ASPRS Annual Conference Proceedings.* April 19-26, 2002.

Morsdorf, F., E. Meier, B. Allgower, and D. Nuesch. 2003. "Clustering in Airborne Laser Scanning Raw Data for Segmentation of Single Trees." *Proceedings of the ISPRS working group III/3 workshop 3-D reconstruction from airborne laserscanner and InSAR data.* Dresden, Germany. October 8-10, 2003.

Muller, S., and D. W. Zaum. 2005. "Robust Building Detection in Aerial Images." *Proceedings of the ISPRS Workshop CMRT 2005 Object Extraction for 3Dd City Models, Road Databases and Traffic Monitoring – Concepts, Algorithms and Evaluation.* Vienna, Austria, Aug. 29-30, 2005.

Nasset, E. 2004. "Practical Large-Scale Forest Stand Inventory Using a Small-Footprint Airborne Scanning Laser." *Scandinavian Journal of Forest Research.* 52(2004): pp. 49-56.

National Agriculture Imagery Program. 2002. "The Farm Service Agency's National Agricultural Imagery Program INAIP)." February, 2009. NAIP. November 12-14 Reston, VA. <http://www.fsa.usda.gov/Internet/FSA_File/naip_er_usgs_pp.ppt>.

National Agriculture Imagery Program. "National Agriculture Imagery Program (NAIP) Information Sheet." February, 2009. NAIP. May, 2008. < http://www.fsa.usda.gov/Internet/FSA_File/naip_2007_infosheetpdf.pdf>.

National Fire Protection Association. 2008. *NFPA 1144 Standard for Reducing Structure Ignition Hazards from Wildland Fire.* Quincy, MA: NFPA.

Nevatia, R., C. Lin, and A. Huertas. 1997. "A System for Building Detection from Aerial Images." February, 2009. Institute for Robotics and Intelligent Systems. 1997. < http://iris.usc.edu/Outlines/papers/1997/huertas-ascona97.pdf>.

Novacheva, A. 2008. "Building Roof Reconstruction from LIDAR Data and Aerial Images Through Plane Extraction and Colour Edge Detection." *ISPRS Congress Beijing 2008.* Beijing, China. July 3-11, 2008.

Opitz, D., J. Shavlik. 1996a. "Generating Accurate and Diverse Members of a Neural-Network Ensemble." Advances in Neural Information Processing Systems 8: 535-541.

Opitz, D., R. Maclin. 1999. "Popular Ensemble Methods: An Empirical Study." Journal of Artificial Intelligence Research. 11 (1): 169-198.

Opitz, D. 1999. "Feature Selection for Ensembles." Proceedings of the 16th National Conference on Artificial Intelligence. 379-384.

Opitz, D., W. Bain. 1999. "Experiments on Learning to Extract Features from Digital Images" IASTED Signal and Image Processing.

Opitz, D., S. Blundell. 1999. "An Intelligent User Interface for Feature Extraction From Remotely Sensed Images." Proc. of the American Society for Photogrammetry and Remote Sensing, 171-177.

Opitz, D., J. Shavlik. 1999. "Actively Searching for an Effective Neural-Network Ensemble." Springer-Verlag Series on Perspective in Neural Computing. 79-97.

Packalen, P., and M. Maltamo. 2007. "The k-MSM Method for the Prediction of Species-Specific Stand Attributes Using Airborne Laser Scanning and Aerial Photographs." *Remote Sensing of Environment.* 109(3): pp. 328-341.

Parker, G.G. 1995. "Structure and Microclimate of Forest Canopies." In M.D. Lowman and N.M. Nadkami (Eds.), *Forest Canopies,* (pp. 73-106). New York: Academic Press.

Perry, G.L.W., 1998. "Current Approaches to Modeling the Spread of Wildfire: a Review." *Progress in Physical Geography,* 22(1998): pp. 222-245.

Perrson, A., J. Holmgren, and U. Soeman. 2002. "Detecting and Measuring Individual Trees Using an Airborne Laser Scanner." *Photogrammetric Engineering and Remote Sensing."* 79(2002): pp. 105-115.

Popescu, P.C., and A.U. Kini, 2004. "TREEVAW: a Versatile Tool for Analyzing Forest Canopy LIDAR DATA – a Preview with an Eye Towards Future." *ASPRS Images to Decision: Remote Sensing Foundation for GIS Applications,* Kansas City, Missouri, 12-16 September 2004.

Popescu, P.C., and K. Zhao. 2008. "A Voxel-Based Lidar Method for Estimating Crown Base Height for Deciduous Pine Trees." *Remote Sensing of Environment.* 112(2008): pp. 767-781.

Pu, S. "Automatic Building Modeling from Terrestrial Laser Scanning". <u>International Institute for Geo-information Science and Earth Observation.</u> February, 2009. ITC. 2007. < http://www.itc.nl/personal/pushi/docs/automatic2007.pdf>.

Radeloff, V.C., R.B. Hammer, S.I. Stewart, J.S. Fried, S.S. Holcomb, and J.F. McKeefry. 2005. "The Wildland-Urban Interface in the United States." *Ecological Applications.* 15(3), 2005, pp. 799-805.

Rehm, R., D. Evans, W. Mell, S. Hostikka, K. McGrattan, G. Forney, C. Bouldin, E. Baker. 2003. "Neighborhood-Scale Fire Spread." *5th Symposium on Fire & Forest Meteorology.* Orlando, Florida. Nov. 16-20, 2003.

Riano, D., E. Meier, B. Allgower, E. Chuvieco, S.L. Ustin. 2003. "Modeling Airborne Laser Scanning Data for the Spatial Generation of Critical Forest Parameters in Fire Behavior Modeling." *Remote Sensing of Environment.* 86(2003): pp. 177-186.

Roberts, F. 2004. Tribe work efficiently saves time, money, and lives while preserving a culture. In C. Thomas & M. Ospina (Eds), Measuring Up: The Business Case for GIS: pp. 32-34. Redlands, CA.

Roberts, S.D., T.J. Dean, D.L. Evans, J.W. McCombs, R.L. Harrington, and P.A. Glass. 2005. "Estimating Individual Tree Leaf Area in Loblolly Pine Plantations Using Lidar-Derived Measurements of Height and Crown Dimensions." *Forest Ecology and Management.* 213(2005): pp. 54-70.

Rottensteiner, F., and C. Briese. 2002. "A New Method for Building Extraction in Urban Areas from High-Resolution LIDAR Data." *Photogrammetric Engineering & Remote Sensing.* 73(7): pp. 805-812.

Rottensteiner, F., and C. Briese. 2003. "Automatic Generation of Building Models from LIDAR Data and the Integration of Aerial Images." *Proceedings of the ISPRS Working Group III Workshop: 3D Reconstruction from Airborne Laserscanner and InSAR data."* Dresden, Germany. October 8-10, 2003.

Rowell, E., C. Seielstad, L. Vierling, L. Queen, and W. Shepperd. 2006. "Using Laser Altimetry-based Segmentation to Refine Automated Tree Identification in Managed Forests of the Black Hills, South Dakota." *Photogrammetric Engineering & Remote Sensing.* 72(12): pp. 1379-1388.

Ruther, H., M. M. Hagai, and E.G. Mtalo. 2002. "Application of Snakes and Dynamic Programming Optimization in Modeling of Buildings in Informal Settlement Areas." *ISPRS Journal of Photogrammetry & Remote Sensing.* 56(2002): pp. 269-282.

Sampath, A., and J. Shan. 2007. "Building Boundary Tracing and Regularization from Airborne Lidar Point Clouds." *Photogrammetric Engineering and Remote Sensing.* 73(7): pp. 805-812.

Shan, J., and S.D. Lee, 2005. "Quality of Building Extraction from IKONOS Imagery." *Journal of Survey Engineering,* 131(1): pp. 27-32.

Skowronskui, N.S., K.L. Clark, R. Nelson, J. Hom, and M. Patterson. 2007. "Remotely Sensed Measurements of Forest Structure and Fuel Loads in the Pinelands of New Jersey." *Remote Sensing of Environment.* 108: pp. 123-129.

Sohn, G., and I. Dowman. 2001. "Extraction of Buildings from High-Resolution Satellite Data." *Automated Extraction of Man-Made Objects from Aerial and Space Images (III).* Lisse: Balkema Publishers. pp. 345-355.

Sohn, H.G., C.H. Park, H.S. Kim, and J. Heo, 2005. "3-D Building Extraction Using IKONOS Multispectral Images." *Geoscience and Remote Sensing Symposium,* (2(25-29): pp. 1432-1434.

Solberg, S. 2006. "Single Tree Segmentation Using Airborne Laser Scanner Data in a Structurally Heterogeneous Spruce Forest." *Photogrammetric Engineering & Remote Sensing.* 72(12): pp. 1369-1378.

Song, W. and T.L. Haithcoat. 2005. "Development of Comprehensive Accuracy Assessment Indexes for Building Footprint Extraction." *IEEE Transactions on Geoscience and Remote Sensing.* 43(2).

[1]Stewart, S.I., V.C. Radeloff, R.B. Hammer, and T.J. Hawbaker. 2007. "Defining the Wildland Urbann Interface." *Journal of Forestry.* 105: pp. 201-207.

[2]Stewart, S.I., B. Wilmer, R.B. Hammer, G.H. Aplet, T.J. Hawbaker, C. Miller, and V.C. Radeloff. 2007. "Wildland-Urban Interface Maps Vary with Purpose and Context." *Rural Studies Program Working Paper Series.* Rural Studies Program, Oregon State University. December, 2007.

Stoker, J., J. Parrish, D. Gisclair, D. Harding, R. Haugerud, M. Flood, H. Andersen, K. Schuckman, D. Mauna, P. Rooney, K. Waters, A. Habib, E. Wiggins, B. Ellingson, B. Jones, S. Nechero, A. Nayegandhi, T. Saultz and G. Lee. 2007. "Report of the First National Lidar Initiative Meeting." February 14-16, 2007. Reston, VA.

Suveg, I., and G. Vosselman. 2004. "Reconstruction of 3D Building Models from Images and Maps." *ISPRS Journal of Photogrammetry and Remote Sensing.* 58(3-4): pp. 202-224.

Thomas, V., P. Treitz, J.H. McCaughey, and I. Morrison. 2006. "Mapping Stand-Level Forest Biophysical Variables for a Mixed Wood Boreal Forest Using Lidar: an Examination of Scanning Density." *Canadian Journal of Forest Resources.* 36(2006): pp. 34-47.

Tiede, D., G. Hochleitner, and T. Blaschke. 2005. "A Full GIS-Based Workflow for Tree Identification and Delineation Using Laser Scanning." *The International Archives of Photogrammetry, Remote Sensing and Spatial Information.* Vol. XXXVI, Part 3/W24, Vienna, pp. 9-14.

Trujillo, L., G. Olague, R. Hammoud, B. Hernandez. 2005. "Automatic Feature Localization in Thermal Images for Facial Expression Recognition." *IEEE Computer Society Conference on Computer Vision and Pattern Recognition.* (25)25: pp. 14.

Vosselman, G., and S. Dijkman. 2001. "3D Building Model Reconstruction from Point Clouds and Ground Plans." *International Archives of Photogrammetry and Remote Sensing.* Annapolis, MD. October 22-24, 2001.

Walker, J.S., and J.M. Briggs. 2005. "An Object-Oriented Classification of an Arid Urban Forest with True-Color Aerial Photography." *3rd International Symposium Remote Sensing and Data Fusion Over Urban Areas (URBAN 2005).* Tempe, AZ. March 14-16 2005.

Walter, V. 2005. "Object-based Classification of Integrated Multispectral and LIDAR Data for Change Detection and Quality Control in Urban Areas." *3rd International Symposium Remote Sensing and Data Fusion Over Urban Areas (URBAN 2005).* Tempe, AZ.

Wang, Z. 1998. "Extracting Building Information From LIDAR Data." *ISPRS Commission III Symposium on Object Recognition and Scene Classification from Multi-Spectral and Multi-Sensor Pixels.* Columbus, Ohio.

Wang, S., and Y.H. Tseng. 2003. "Semi-automated Building Extraction Based on CSG Model-Image Fitting." *Photogrammetry Engineering & Remote Sensing.* 69(2): pp. 171-180.

Wang, M. and Y.H. Tseng. 2004. "LIDAR Data Segmentation and Classification Based on Octree Structure." *The proceedings of the XXth Congress of ISPRS.* Vol. XXXV, Part B3, pp. 308-313.

Wei, Y., Z. Zhao, and J. Song. 2004. "Urban Building Extractioin from High-Resolution Satellite Panchromatic Image Using Clustering and Edge Detection." *Proceedings: 2004 IEEE International.* 3(20-24): pp. 2008-2010.

Weidner, U., and W. Forstner, 1995. "Towards automatic building extraction from high-resolution digital elevation models." *ISPRS Journal of Photogrammetry and Remote Sensing*, 50(4): pp. 38-49.

Weishampel, J.F., D.J. Harding, J.C. Boutet, J.B. Drake. 1997. "Analysis of Laser Altimeter Waveforms for Forested Ecosystems of Central Florida." *Proceedings of SPIE: Advances in Laser Remote Sensing for Terrestrial and Oceanographic Applications.* 3059(1997): pp. 184-189.

Wilmer, B. and G. Aplet. "Targeting the Community Fire Planning Zone: Mapping Matters." The Wilderness Society. March, 2009. TWS. May 1, 2005.

Woo, D., Q. Nguyen, Q. Nguyen Tran, D. Park, and Y. Jung. 2008. "Building Detection and Reconstruction from Aerial Images." *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences.* Vol. XXXVII. Part B3b. pp. 713 – 718.

Xie, M., K. Fu, and Y. Wu. 2006. "Building Recognition and Reconstruction from Aerial Imagery and LiDAR Data." *International Conference on Radar.* Shanghai, China. Oct. 16-19, 2006.

Yi-Hsing, T., and S. Wang. 2003. "Semi-automated Building Extraction Based on CSG Model-Image Fitting." *Photogrammetric Engineering and Remote Sensing.* 69(2): pp. 171-180.

Zehng, Q. 2008. "Data Filtering and Feature Extraction of Urban Typical Objects from Airborne LIDAR Point Cloud." *ISPRS Congress Beijing 2008.* Beijing, China. July 3-11, 2008.

Zimmermann, P. 2000. "A New Framework for Automatic Building Detection Analyzing Multiple CUE Data." *IAPRS.* Vol. XXXIII. Amsterdam.

APPENDIX A: LIST OF PRESENTATIONS AND OUTREACH MEETINGS

Hu, J. 2009. "3-D Wildland Modeling and Fire Assessment System." *NIST Annual Fire Conference.* Gaithersburg, MD. April 28 - April 30, 2009.

Hu, J. 2009. "3-D Wildland Modeling and Fire Assessment System." *Intermountain GIS Conference.* Coeur d'Alene, ID. April 6-10, 2009.

McNamara, D. 2008. "Using LiDAR and Multi-spectral Data for Feature Extraction in the Wildland Urban Interface." *Intermountain GIS Conference.* Missoula, MT. April 7-11, 2008.

McNamara, D. 2008. "Feature Extraction in the Wildland Urban Interface (WUI): Case Study for Coupling GIS with WFDS." *NIST Annual Fire Conference.* Gaithersburg, MD. March 31- April 2, 2008.

McNamara, D. 2008. "Feature Extraction in the Wildland Urban Interface (WUI)." *Inland Northwest Growth and Yield Technical Meeting.* Spokane, WA. January 15, 2008.

McNamara, D. 2007. "Enhancing the Wildland Fire Dynamics Simulator for Modeling Wildland Urban Interface (WUI) Fires. *2007 Environmental Systems Research Institute (ESRI) International User Conference.* San Diego, CA. June 18-22, 2007.

McNamara, D. 2007. "Presentation to the Spokane Tribe on LiDAR Data". *Spokane Tribe Forestry Department.* Wellpinit, WA. September 27, 2007.

McNamara, D. 2007. "Coeur d'Alene Tribe LiDAR Data." *2nd Annual Indigenous Mapping Network Gathering.* Coeur d'Alene, ID. August 20-22, 2007.

McNamara, D. 2007. "Deriving Data Inputs for the Wildland Fire Dynamics Simulator." *April 2007 Washington Geographic Information Council Meeting.* Spokane, WA. April 4, 2007.

McNamara, D. 2007. "Deriving Data Inputs for the Wildland Fire Dynamics Simulator." *NIST Annual Fire Conference.* Gaithersburg, MD. April 4-5, 2007.

McNamara, D. 2006. "Extracting Building Footprints in the Wildland Urban Interface." *Northwest ESRI GIS Users Conference.* Spokane, WA. September 13-15, 2006.

McNamara, D. 2006. "Enhancing the Fire Dynamics Simulator for Modeling WUI Fires." *NIST Annual Fire Conference.* Gaithersburg, MD. April 3-4, 2006.

APPENDIX B: GIS LIDAR TOOLS USER DOCUMENTATION

GIS LiDAR Tools Documentation
Coeur d'Alene Tribe

Tools For LiDAR Processing

Derek McNamara

9/19/2008
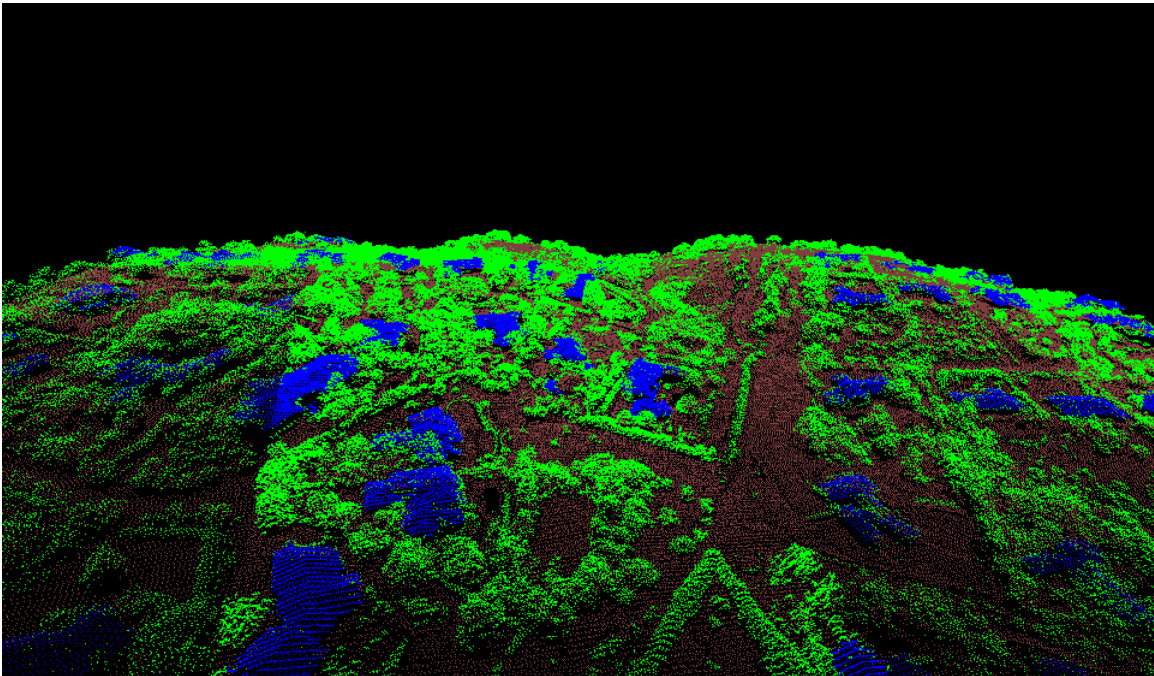
This document describes the functionality contained in the GIS LiDAR Tools application.

Disclaimer

GIS Light Intensity Detection and Ranging (LiDAR) Tools version 1.0 is a free distribution software application for processing LiDAR data. GIS LiDAR Tools was not developed as commercial-grade software. The authors, the Coeur d'Alene Tribe and McNamara Consulting, are not responsible for any misuse of the package, for the accuracy of the results, and for any consequences of running the software. In addition, although we have attempted to locate and eliminate programming errors[10] that might hamper proper use of this software package, no software can be deemed truly "bug-free." Therefore, we make no guarantees; either expressed or implied, about the software's reliability, and cannot be held responsible for any damages incurred to the user as a result of using this software.

Acknowledgements

These tools were developed as a result of a grant received by the Coeur d'Alene Tribe from the Building and Fire Research Laboratory (BFRL) of the National Institute of Standards and Technology (NIST). The purpose of this grant was to research and develop methods for extracting man-made and natural features from remotely sensed data for input into the Wildland Fire Dynamics Simulator (WFDS). In addition to NIST, these tools could not have been developed without the efforts of Douglas McNamara and Gale McNamara of McNamara Consulting Inc., who assisted in the development of the algorithms included in these tools as well as the development of a framework to efficiently process a LiDAR point cloud of data. Also, Jason Robert's ASCII to Shape utility, which utilizes the shapelib developed by Frank Warmerdam, is included in these tools. Eric Rowell of the National Center for Landscape Fire Dynamics also provided expertise and many of the ideas for the tree extraction algorithm were a result of his work as well as other cited below.   Finally, QCoherent software provided their software in exchange for algorithms utilized in these tools. This software proved invaluable in refining results produced from these tools.

---

[10] There are some known errors with workarounds listed in the appendix.

Executive Summary

These tools contain functionality to view, process, and analyze LiDAR data. The current implementation of these tools will only work with LiDAR data in LAS version 1.0. The functionality of the tools contained within this software is described below. These tools were developed as a research effort to explore methods of deriving inputs to the WFDS from remotely sensed LiDAR data in conjunction with multispectral and other ancillary GIS data. The tools described herein are not necessarily designed for processing LiDAR data over large areas but for smaller sized projects in certain locales they have proved beneficial for filtering LiDAR building points and extracting tree stem locations. They have also failed in some circumstances and have generally not been tested on LiDAR datasets with large point densities. Nonetheless, these tools help demonstrate the potential of processing LiDAR data in a GIS environment.

Requirements

The filtering tools contained within this software require an Environmental Systems Research Institute (ESRI) ArcInfo license as well as an ESRI Spatial Analyst License. The windows based LiDAR 3-D data viewer, which allows for viewing of LiDAR data in a 3-D environment as well as some export and tiling functionality, does not require any ESRI licenses. These tools are memory intensive, particularly when coupled with ESRI software so a machine with lots of memory would help in using these tools.

Contents

Figures

1.0 ArcGIS Based Tools

1.1 Getting Started

The ArcGIS based tools are activated, after installation, by adding the LiDARTools toolbar to your ArcMap document. After installation this can be accomplished by right clicking on a grey area in your ArcMap document and selecting the LiDARTools item. This will bring up a toolbar as shown below.

All of the tools within the toolbar are disabled until you load a valid LAS file. Loading the LAS file is accomplished by selecting the "Load LAS File Version 1.0" item from the "Load LAS File" menu item under the LiDAR Tools menu as shown below.
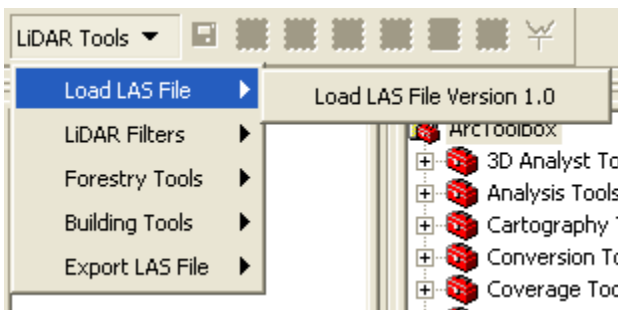
If the selected file is a valid LAS version 1.0 file the LAS file should be loaded into memory and the ArcMap map window will zoom to the extent of the LAS file. It is possible that the LAS file might need to be subset due to size and memory limitations. See the "Windows Based Tools" section of the document for instructions on sub-setting LAS files. Once the LAS file is loaded the various tools and menu items contained within the LiDAR tools toolbar should become enabled as shown below.

The various tools contained within the LiDARTools application can be run on the entire loaded LAS file or on a subset. Tools run on the entire loaded LAS file are accessed through the "LiDAR Tools" menu while tools run on subsets of the LAS file are accessed using the command buttons, with the exception of the save icon, which saves the entire loaded LAS file, and the W/T icon, which will open up the windows based tools.

2.0 LIDAR FILTERS

2.1 Building Filter

The LiDAR building filter attempts to classify LiDAR points representing laser returns from structures.   Building points are classified using the approach of fitting planes to the 3-D point cloud of LiDAR data as described in other research (Wang and Tseng, 2004; Tovari and Pfeifer. 2005; and Verma et al.,) with additional custom segmentation algorithms incorporated as described below. The LiDAR building filter changes the classification value of most LiDAR points. To account for this there is a tool in the windows based tools that allows users to select two LAS files (e.g. the original LAS file and the same LAS file classified for building points and saved to a new file) and set the points not classified as buildings back to their original classification.

Before the plane fitting algorithm is employed an optional initial segmentation of the LiDAR points is employed by considering the relationship between the first and last point, or return, in a particular laser pulse and the same first return and ground elevation as determined by an input bare earth digital elevation model. The distance between the first and last return in a particular laser pulse is determined as well as the distance between the same first return and the ground elevation, as portrayed in the input bare earth DEM, for which this first return falls. If the difference between these two distances is greater than a threshold, representing the minimum building height as input by the user, all returns in the particular laser pulse are classified as not being a building with the exception of the last return, which could be a building covered by vegetation. This initial segmentation takes advantage of the fact that a LiDAR first return pulse when hitting dense vegetation, will, in some cases, not have a last return which hits the ground. Before utilizing the above algorithm the input LAS file must be run through a command line executable. This executable is found in the installation folder and is called LAS.exe. Make sure that neither ArcGIS nor the "Windows Based Tools" are running when running this command line utility. The syntax for this utility is <LAS LASIN LASOUT> where LASIN is the input LAS file and LASOUT is the output LAS file.

Another optional segmentation algorithm employed uses a vegetation mask grid where 1's represent vegetated pixels. This segmentation algorithm classifies all those LiDAR first return points, falling on vegetated pixels in the input vegetation mask, as non-building points. The final segmentation algorithm employed before running the plane fitting algorithm removes points based on minimum and maximum height values input by the user.

After the above segmentations are run the plane fitting algorithm is employed, which iterates through each point in the point cloud not removed in the above algorithms. For

each iteration, an initial window is fit around the particular point and an equation of a plane is fit to the points within this window. The slope between the initial point and each other point in the window is evaluated and only those points with a slope less than a threshold, representing the maximum slope of a building roof, are included. A plane is than fit to these points and the sum of squared residuals is calculated for the fitted plane. If the fitted plane is a "good" fit based on the sum of squared residuals the window is initially expanded in all directions. Outliers are removed based on a threshold, which represents the Z distance of each point to the fitted plane. If no new points are added, no segmentation occurs and the next point in the iteration is considered. If new points are added the direction in which these points occur is recorded. A new plane is fit to the points which are not removed above. The window is expanded again, only in the direction for which new points are added as shown in figure 1 and the process continues until no new points are added. Through the implementation of this algorithm each planar surface, not representing the ground surface, in the loaded point cloud is classified as a building.
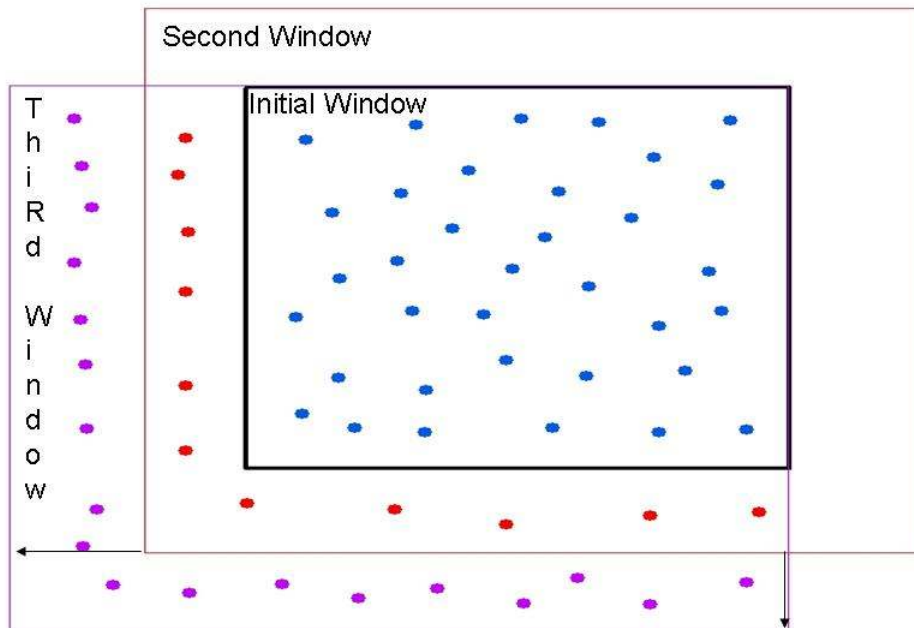
Figure 28 Directional window example portraying expansion in only two directions.

The building filter would typically be run first on the entire loaded LAS file, which can be accomplished by selecting the "LiDAR Building Filter" command under the "LiDAR Filters" menu item in the "LiDAR Tools" menu as shown below.
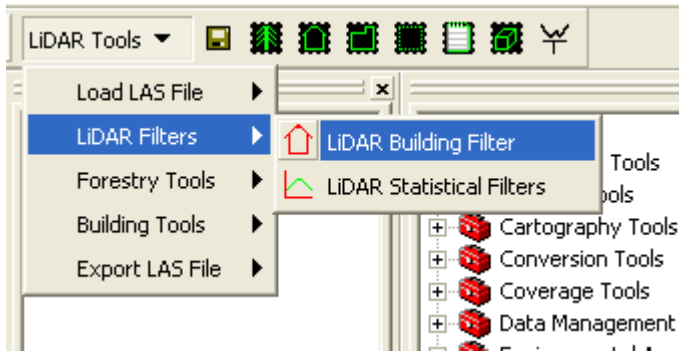
After running the filter on the entire file there is typically some cleanup required and the filter can be run on selected locations by clicking on the "Classify Building Points in Selected Extent" button as shown below and drawing a rectangle around the extent the user wishes to run the building filter. Parameters can be changed to classify missed building points or remove points erroneously classified as a building.

Either of the methods shown above will bring up the form shown below.

Each parameter in the above form is described below.

- **Select the Bare Earth Grid:** This is a bare earth digital elevation model encompassing the extent of the loaded LAS file. Generally, this input grid should have an integer cell size and unexpected results might occur if this is not the case.
- **Use Vegetation Mask:** This checkbox would be checked if there is a binary vegetation grid covering the extent of the loaded LAS file where vegetated pixels would have a value of 1. If this checkbox is checked a vegetation grid must be input in the combo box below the check box. Vegetated pixels in this grid should have a pixel value of 1. As with the bare earth grid this grid should have an integer cell size.

- **Running Thinning Algorithm (Vegetated Areas):** This check box indicates if the user would like to run the thinning algorithm described above that utilizes the LiDAR pulse grouping. This thinning algorithm would be run in areas of high vegetation. This thinning algorithm would generally not be run in an urban area with tall buildings and little vegetation. In addition, this thinning algorithm might erroneously remove building edge points of multi-storied buildings. The LAS.exe should be run on the input LAS file if using this algorithm.
- **Exclude Points with a Scan Angle Greater Than:** This check box, if checked, allows users to enter a scan angle in the associated text box. Points with an absolute value for the scan angle greater than the user input value will not be considered.
- **Enter Minimum Building Height:** This would be the minimum height expected for a building. This would be determined by subtracting the height value of a bare earth DEM pixel from the height value of a particular LiDAR point occurring on that pixel. Any LiDAR points that are below the input value would not be classified as a building point. This value would be in the units of the loaded LAS file height or Z values.
- **Enter Maximum Building Height:** This would be the maximum height expected for a building. This would be determined by subtracting the height value of a bare earth DEM pixel from the height value of a particular LiDAR point occurring on that pixel. Any LiDAR points that are above the input value would not be classified as a building point. This value would be in the units of the loaded LAS file height or Z values.
- **Enter Moving Window Size:** This is the size of the moving window fit around each LiDAR point. The value represents ½ of the length of each side of the moving window (e.g., an input value of 2 would result in a moving window of 4 X 4). Initially, users can enter a value about the size of the point spacing of the input LAS file and increase or decrease from there based on results. This value would be in the units of the loaded LAS file's X and Y values.
- **Enter Predicted Height Threshold:** This is the value used to remove outliers in the fitted plane. As described above outliers are removed based on a threshold, which represents the Z distance of each point to the fitted plane. This threshold is input in this text box. A good starting value would be slightly larger than the vertical accuracy of the input LAS file. This value would be in the units of the LiDAR height values.
- **Enter Slope Threshold:** As described above, the slope between the initial point and each other point in the window is evaluated and only those points with a slope less than this input value, representing the maximum slope of a building roof, are included.
- **Enter Number of Points Required for Plane:** This input parameter determines the number of points required for a plane. Fitting an equation of a plane requires at least three points. In a LiDAR point cloud, however, buildings would typically be represented by more than three points. This input value would, consequently never be lower than three and typically greater than 5. This parameter can be adjusted where building points might be of a lower density due to being surrounded by vegetation or other factors. Adjusting this parameter along with

the moving window size can help classify missed building points. Examine a typical building and see how many points are found in the selected moving window size. A value somewhat less than this would be a good place to start.

- **Enter % Cleanup:** This slider cleans up missed building points in a brute force manner. For each building point not filtered out by the initial thinning algorithms and not classified as a building a moving window is fit around the particular point. Where the moving window is a cube with dimensions equal to the user input moving window size. All of the points within this cube are examined. If the percentage of points classified as buildings, within this cube, is greater than the cleanup value the particular point under consideration is classified as a building.

2.2 Statistical Filters

Statistical filters employ a moving window around each LiDAR point and change either height or intensity values for each point in the LiDAR point cloud to the value of the statistical measure for the points in the window. Points can be excluded based on a distance from the ground as portrayed in the input bare earth DEM or all the points can be used. These filters can be used to create interesting measures of the LiDAR data. These filters can only be run on the entire point data set. Running a statistical filter on the entire loaded LAS file is accomplished by selecting the "LiDAR Statistical Filter" command under the "LiDAR Filters" menu item in the "LiDAR Tools" menu as shown below.
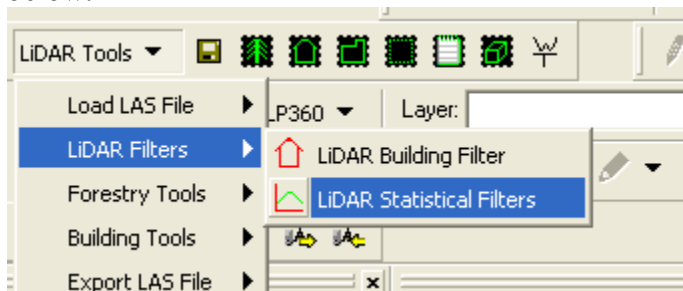
Clicking on the above command button will bring up the form shown below.

Each parameter in the above form is described below.

- **Field To Filter:** This is the field for which the statistical filter will be applied. This is either the height field or the intensity field in the loaded LAS file.
- **Statistic Type:** This is the statistical measure applied to each LiDAR point based on height or intensity values in the surrounding window. Available statistical measures are: mean, median, maximum, minimum, standard deviation and variance. The chosen statistical measure is applied to the points in the window and the particular point in the iteration is set to the value of the statistical measure.
- **Exclude Points By Height Threshold:** This check box, if checked, will require the user to enter a bare earth DEM and a minimum height to exclude. Points with a value below this minimum height, representing the height value of the particular

point minus the bare earth DEM pixel value for which the point falls, will not be included in the statistical measure calculation.

- **Select the Bare Earth Grid:** This is the input bare earth DEM encompassing the extent of the loaded LAS file.
- **Minimum Height to Exclude:** This is the minimum height to exclude where points with a value below this minimum height will not be included in the statistical measure calculation.
- **Use Rectangle Filter:** If this check box is checked the moving window will be a rectangle with the height and width equal to the user input values in the text boxes below this check box.
  - **Height:** The height, or length in the Y direction, of the moving window.
  - **Width:** The width, or length in the X direction of the moving window.
- **Use Circle Filter:** If this check box is checked the moving window will be a circle with a radius equal to the user input value in the text below this check box.
  - **Radius:** The radius of the circular moving window.

3.0 FORESTRY TOOLS

3.1 LiDAR Tree Stem Location

The LiDAR Tree Stem Location tools attempt to identify tree stem locations with associated attributes of tree height, crown width, crown base height and deciduous versus conifer trees (leaf-off data only). The Tree Stem Location tool works in association with the building filter in that points that were classified as buildings using the building filter are not included in the extraction of tree stem locations. Unlike the building filter, however, the resulting LAS file is not meant to be saved or utilized for different purposes, generally speaking. This is because many of the attributes in the resulting LAS file are altered. Certain attributes are used to store results of crown width, crown base height and other intermediate derivatives during the processing to save memory. Consequently, only the resulting text file should be used for further analysis.

Before the identification of tree stem locations begins an initial thinning algorithm is employed to help remove pole features such as those shown in Figure 2. Sometimes, particularly at the edge of scan lines pole features such as tree trunks or telephone poles will have LiDAR returns. It is beneficial to remove these points as they are sometimes incorrectly returned as trees. The initial thinning algorithm looks for points with similar horizontal coordinates and a large range in height variations and removes these points from consideration for tree stem locations.

The approach used in identifying tree stem locations uses a user input allometric relationship between crown width and tree height where tree height is the independent variable and crown width is the dependent variable. Using this input relationship local maximum are identified in the point cloud of LiDAR data by fitting a moving circular window of variable size to each point in the LiDAR point cloud. This approach was

mainly derived from work conducted by Popescu and Kini (2004). The identification of local maximum proceeds by iterating through each point in the LiDAR data, which is above a user defined threshold (minimum tree height) from the interpolated ground surface (input bare earth DEM). If the point in the iteration is the highest point in the search window it is flagged as a tree top. In order to filter out high points, which represent pole features such as power line poles, the number of points below the identified local maximum point is recorded and if this number of points is below the user input threshold the search window is expanded by 33% (Personal Communication, Eric Rowell). If new points are added such that the total number is greater than the threshold the point is flagged as a tree top. This is an advantage of using the point data directly compared to the canopy height model (CHM), which cannot determine points beneath the identified local maximum. Once the potential tree top locations have been identified the derivation of crown diameter begins.
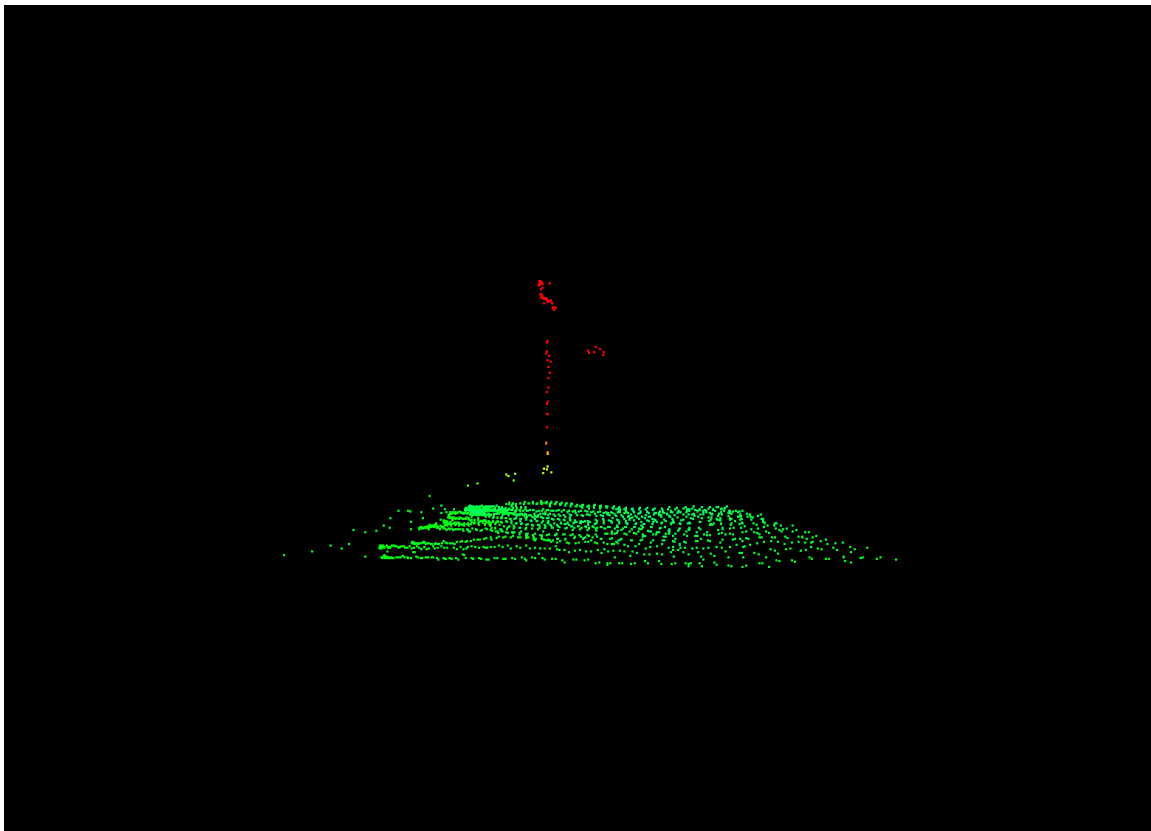
Figure 29 Laser returns from a telephone pole.

The approach for determining crown width uses an input CHM. As described in Popescu and Kini (2004) performing a median filter on this CHM might help improve results. While the use of a CHM was derived from work conducted by Popescu and Kini (2004) the approach for determining crown width in LiDAR tools is different. For each identified potential tree top point a traverse along the crown profile in all 8 cardinal directions is conducted. Each value in the CHM is recorded along the traverse and if the next value of the CHM is greater than the previous value or if the value of the CHM is

less than the user input minimum tree height the location is marked. If this occurs a consecutive threshold number of times (see description for "Canopy Radius Search Length" input parameter) the initially flagged value is determined to be the end of canopy point for the particular direction of the traverse. The difference between the flagged values in the north and south direction, or east and west direction are calculated. In addition, the CHM profiles are also traversed in the Southeast/Northwest and Northeast/Southwest directions and the crown widths in these directions are calculated as above. These 4 crown width values are averaged to determine the crown width for the tree. This value is divided by 2 to create the crown radius for the tree. This algorithm will sometimes result in a large overestimate of crown width. This occurs when the traverse of the canopy profile extends into another trees canopy while continuing to decrease in height. Each tree is checked to see if its crown encompasses other trees to adjust for this error. If 4 or more trees are encompassed within the crown of a particular tree, the crown radius in the particular direction of the encompassed tree is adjusted to be the midpoint between the original tree and the encompassed tree.

In addition to the above, a method to remove the edge of canopy or mid canopy points incorrectly extracted as trees has been developed. Due to the complex nature of tree canopies a branch or portion of the canopy, which spikes above the rest of the canopy for the particular tree under consideration, is often identified as a local maximum and incorrectly identified as a tree. An attempt has been made to identify these points concurrent to measuring the crown width of the tree. This is accomplished by recording the value of the CHM for the particular point identified as a potential tree top location during the traverse of the CHM profile. For the next four values of the CHM in the traverse it is determined if the majority of these values are greater than the initial CHM value. If this condition is met the particular direction of the traverse is flagged. The user can enter the number of times this condition is not met for the potential point to be recorded as a tree stem location as described in the "Minimum Number of Correct Canopy Profiles" parameter below.

The algorithm used to determine crown base height is similar to that described by Holmgren and Persson (2004) where all laser points below a particular flagged tree top point and within the identified crown width are grouped into 0.5m height bins. Each bin is categorized as 0 or 1 where bins with a 0 value have less than 1% of the total number of non-ground laser points within all the bins and all others have a 1 value (Holmgren and Persson, 2004). The crown base height was then determined as the distance from ground of the lowest laser data point above the highest 0 classified bin found. Some modifications are made to the above described algorithm. The first modification is a result of tree crown width not always being consistent in all directions. Grouping all points below the flagged tree top point and within the crown width could, consequently, result in incorrect estimates of crown base height. To account for this the location below flagged tree top points is divided into four quadrants and the average crown radii for each particular quadrant is used to determine the laser points within that quadrant. In addition, instead of using the lowest laser data point above the highest 0 classified bin found, the bins are searched for a continuous occurrence, for a threshold number of times, of 0 value bins. When this occurs the lowest laser point in the next 1 value bin, after the occurrence

of the 0 value bins, is marked as the crown base height. This step is conducted to attempt to alleviate the influence of understory vegetation.

Finally, the discussed tree extraction algorithm utilizes the LiDAR intensity data to differentiate between conifer trees and deciduous trees. For each point in the LiDAR data, a measure of the signal strength of the returning laser pulse is recorded as the intensity value, ranging between 0 and 255. The surface from which the laser pulse is reflecting off of will cause variations of the intensity values, which have been shown to be useful in differentiating between conifer and deciduous species (McGaughey and Andersen, 2005). This distinction in intensity values between conifer and deciduous trees would likely only occur for LiDAR data collected during leaf-off conditions. It is unlikely that the distinction would be as apparent with data collected during leaf-on conditions. Also, different environments might require different calibrations to distinguish between conifer and deciduous trees and this can be accomplished as described below.

Running the tree filter is accomplished after loading the LAS file. When trying to extract trees in an urban or residential setting containing structures it is recommended to run the building filter before running the tree filter. The input CHM should than be created from the LiDAR returns not classified as buildings. The tree stem extraction tool can be run on the entire loaded LAS file by selecting the "LiDAR Tree Stem Location" tool in the "Forestry Tools" menu item under the "LiDAR Tools" menu as shown below.
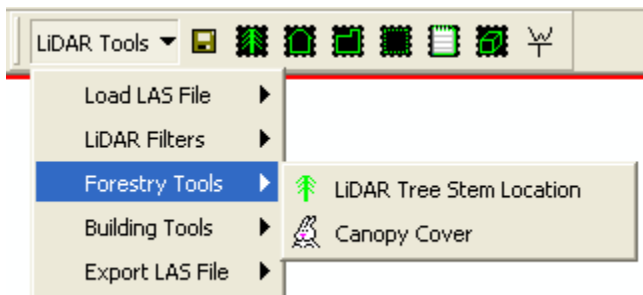
Alternatively, the tree stem extraction tool can be run on a user defined extent by selecting the "Extract Tree Stem Locations in Selected Extent" button as shown below and drawing a rectangle to define the extent.

Either method will bring up the form shown below.

Each parameter in the above form is described below.

- **Select Bare Earth Grid:** This is a bare earth digital elevation model encompassing the extent of the loaded LAS file. Generally, this input grid should have an integer cell size and unexpected results might occur if this is not the case.
- **Select Canopy Height Grid:** This is a canopy height model encompassing the extent of the loaded LAS file. If the area of interest contains buildings better

results might occur if the canopy height grid is derived using returns with building points filtered out. In addition, as mentioned above, running the canopy height grid through a median filter might help remove noise from the resulting tree stem locations. It is also recommended to use the maximum height value for the point in each pixel instead of interpolating all of the first return points. Finally, it is recommended to derive the canopy height model at a lower resolution than would typically be used for deriving other raster products from the same data such as a bare earth DEM. This helps identify subtleties in the surface model to distinguish trees that are close together. This could be at least half the typical resolution and probably as much as a quarter. The smaller the cell size of the input CHM the smaller the size of the loaded LAS file that can be processed. Generally, an input CHM with a cell size of 0.5m can be run on about 1 square kilometer of data at about a 1 meter post-spacing.

- **Shrink Search Radius if No Tree Found:** Checking this search check box will result in a shrinking of the search radius by 33% if no tree is found for the particular point under consideration in the first search. This parameter would generally be checked for areas that contain trees that are very close together. This is included to make the search more dynamic in an attempt to account for trees with different relationships between tree height and crown width.

- **Crown Width (CW) Allometric Relationship:** Three values can be entered here representing the coefficients of a third degree polynomial equation to predict tree crown width from tree height. The use of a third degree polynomial is recommended in Popescu and Kini (2004). The default equation used comes from Tiede et al. (2005). Users can use equations from the literature or derive their own from field data.

- **Enter Minimum Tree Height:** The minimum height required for a point to be a tree. This would be in the same units as the height values for the loaded LAS file.

- **Enter Minimum Crown Width:** LiDAR points with a derived crown width lower than this number will not be returned as tree top points. This would be in the same units as the horizontal coordinates of the loaded LAS file and is meant as a mechanism to filter out smaller trees and potentially pole features.

- **Enter Maximum Crown Width:** This parameter is used as stopping criteria in the crown width derivation. This would be in the same units as the horizontal coordinates of the load LAS file. If the traverse along the canopy height model exceeds this value the crown radius, in the respective direction, would be set to this value. A known limitation of this algorithm is that it sometimes over estimates crown width. This overestimation occurs when the traverse of the canopy height profile goes into another tree or vegetative canopy while elevations continue to decrease. This variable ensures the traverse of the canopy height profile does not continue beyond what should be expected.

- **Enter Median Intensity Value for Deciduous Trees:** LiDAR intensity data collected in leaf-off has shown potential for distinguishing between conifer and deciduous trees. The median value entered here is used to distinguish conifer versus deciduous trees. A group of points, representing a particular tree, whose median value is below this threshold will be classified as a deciduous tree, otherwise it will be classified as a conifer tree. This tool will output the median

values for trees in the resulting text file. This can be used to help calibrate which value to use for the median.

- **Enter Canopy Radius Search Length:** This value is used as a threshold to determine the distance used for a continuous rise in the CHM to determine the end of canopy point for an individual tree. The user enters this as a length in LAS horizontal units and this is converted to pixels of the input canopy height model. A good starting value would be slightly larger than the post spacing of the input LAS file.
- **Enter Minimum Number of Correct Canopy Profiles (1-8):** This parameter will help control the number of stem locations returned where the lower the number the more tree points returned. As described above, for each traverse in the canopy height profile it is recorded if the canopy rises directly from the flagged tree top point. An ideal tree would always have the canopy height decrease in all eight directions. Under certain circumstances, such as a tall tree next to a smaller tree, the canopy height model might decrease in one or two of the directions. Consequently, a good starting number might be 6 or 7. If too many points are being returned this number can be raised.
- **Enter Minimum Number of Points Required For A Tree:** This parameter represents the number of points below a particular point flagged as a tree top plus the tree top point. At least one point is required but a higher number can be used to filter out certain features such as telephone poles.
- **Output Text File:** The output text file where the tree stem location attributes will be stored. This is a comma delimited text file that contains the following attributes.
  - **X:** the x coordinate of the flagged tree top point in horizontal units of the LAS file from which the output text file was produced..
  - **Y:** the y coordinate of the flagged tree top point in horizontal units of the LAS file from which the output text file was produced..
  - **Z:** the original height value in the LAS file from which the output text file was produced.
  - **Hgt:** the height above ground of the tree top point.
  - **CR:** the crown radius of the tree in horizontal units of the LAS file from which the output text file was produced.
  - **CBH:** The crown base height of the tree top point in vertical units of the LAS file from which the output text file was produced.
  - **Genus:** The genus of the tree: deciduous, conifer or unclassified.
  - **IntMin:** The minimum intensity value of the points found within the derived crown base height and crown width of the respective tree top point.
  - **IntMax:** The maximum intensity value of the points found within the derived crown base height and crown width of the respective tree top point.
  - **IntAvg:** The average intensity value of the points found within the derived crown base height and crown width of the respective tree top point.
  - **IntMed:** The median intensity value of the points found within the derived crown base height and crown width of the respective tree top point.
  - **NorthCR:** The crown radius of the respective tree top point in the north direction in horizontal units of the LAS file from which the output text file

was produced. A value of -99 indicates no crown radius was determined due to an initial rise in the canopy height model.

- **SouthCR:** The crown radius of the respective tree top point in the south direction in horizontal units of the LAS file from which the output text file was produced. A value of -99 indicates no crown radius was determined due to an initial rise in the canopy height model.
- **EastCR:** The crown radius of the respective tree top point in the east direction in horizontal units of the LAS file from which the output text file was produced. A value of -99 indicates no crown radius was determined due to an initial rise in the canopy height model.
- **WestCR:** The crown radius of the respective tree top point in the east direction in horizontal units of the LAS file from which the output text file was produced. A value of -99 indicates no crown radius was determined due to an initial rise in the canopy height model.
- **NECR:** The crown radius of the respective tree top point in the northeast direction in horizontal units of the LAS file from which the output text file was produced. A value of -99 indicates no crown radius was determined due to an initial rise in the canopy height model.
- **SECR:** The crown radius of the respective tree top point in the southeast direction in horizontal units of the LAS file from which the output text file was produced. A value of -99 indicates no crown radius was determined due to an initial rise in the canopy height model.
- **SWCR:** The crown radius of the respective tree top point in the southwest direction in horizontal units of the LAS file from which the output text file was produced. A value of -99 indicates no crown radius was determined due to an initial rise in the canopy height model.
- **NWCR:** The crown radius of the respective tree top point in the northwest direction in horizontal units of the LAS file from which the output text file was produced. A value of -99 indicates no crown radius was determined due to an initial rise in the canopy height model.
- **HitRatio:** This measure represents the percentage of points within a tree crown that do not hit the ground. This measure could have potential for vegetation identification.
- **EndPnt:** Due to the fact that LiDAR tools only allows for loading of one LAS tile at a time there are sometimes trees along the edge of the tile that are returned where the traverse along the canopy profile did not extend beyond the tile. This variable marks those trees with a value of "Yes" to indicate they are along the edge of the tile.

4.0 BUILDING TOOLS

4.1 Extract Building Footprints

The "Extract Building Footprints" tool attempts to group LiDAR classified building points, representing a single building, and trace the boundary of these points. The methodology for creating this boundary tracing functionality is described in Sampath and Shan (2007) with some modifications to the boundary tracing algorithm.

The first step in this algorithm is to iterate through all of the LiDAR points classified as a building using the "Building Filter" and for each classified building point a moving window, whose size is input by the user, is fit around the point and all points classified as a building are grouped together. For each of these points moving windows are fit of the same size as the original and all points are grouped into a single building. This process continues until no new points are added.

The next step uses a modified convex hull approach to trace the boundaries of these points (Sampath and Shan, 2007). This algorithm starts with the left most point in a group of classified building points determined as described above. From this point all the points within a separate moving window are determined. Vectors are created between the initial point and all other points in the window. These points are then sorted in increasing order of the clockwise angle from the vertical axis. The point that corresponds to the least angle is chosen as the next point in the iteration. This process continues until the boundary is determined with the exception that for each other point, except the first, the points are sorted in increasing order of the clockwise angle from the angle of the vector perpendicular to the first line drawn. This helps to alleviate the non-uniformity of LiDAR spacing that might occur between scan lines.

Running the "Extract Building Footprints" tools should be done after the building points have been classified. As with most of the LiDAR Tools the footprint extraction can be run on the entire loaded LAS file or on a subset of the file. Running the "Extract Building Footprints" tool on the entire file is accomplished by clicking the "Extract Building Footprints" command under the "Building Tools" menu item under the "LiDAR Tools" menu as shown below.
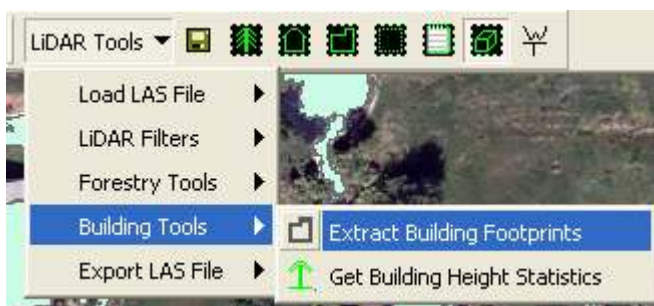
Alternatively, the tool can be run on a subset of the loaded LAS file by clicking on the "Extract Building Footprints" command on the LiDAR Tools toolbar as shown below.

Either of the procedures above will bring up the form shown below.

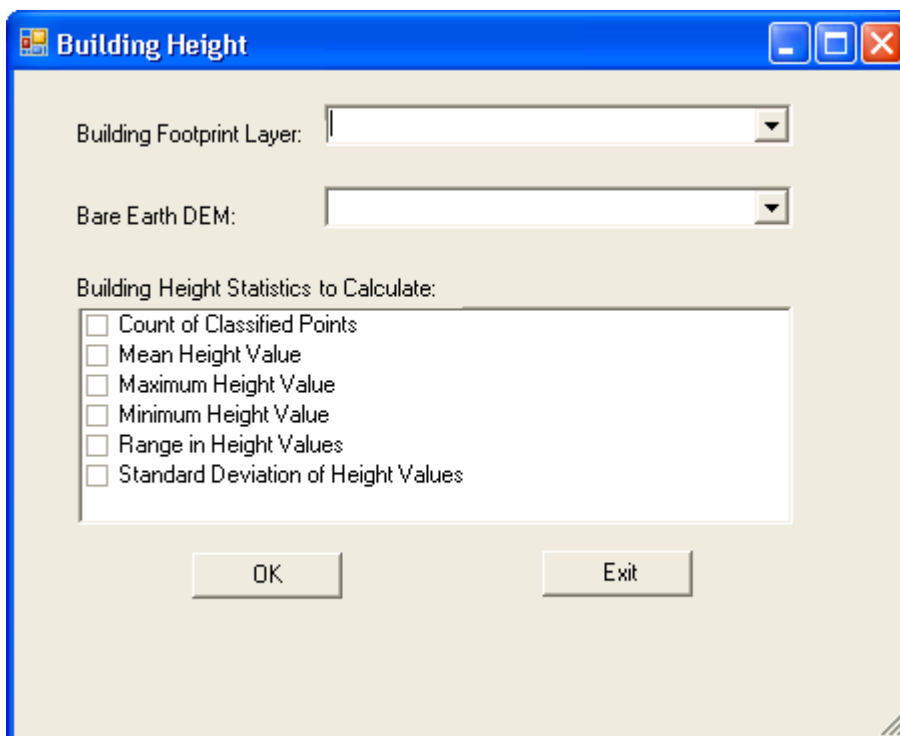Each of the parameters in the above form are described below.

- **Select Output Shapefile:** This is the path to the output line shapefile representing the traced building points. This line shapefile can be converted to a polygon shapefile.
- **Enter Window Size for Grouping Adjacent Building Points:** This parameter represents the moving window size used to group adjacent classified building points. A good starting value for this parameter would be about the post spacing of the input LAS file. A moving window size that is too large will tend to group adjacent buildings into one and a moving window size that is too small will not properly trace the boundaries of the grouped points.
- **Enter the Window Size For Building Boundary Determinization:** This parameter represents the moving window size used in the boundary tracing. This parameter should be set slightly larger than the point spacing of the input LAS file.
- **Enter Minimum Points Required for a Building:** This parameter determines the number of grouped points required for a building. This parameter helps remove erroneously classified points along the edge of the scan line and other locations.

4.2 Get Building Height Statistics

The "Get Building Height Statistics" tool returns height statistics, for the user selected polygon, of the users choosing from the classified building points in the loaded LAS file. This tool will obtain statistics for any polygons containing some LiDAR classified building points (i.e., classification = 6). This tool is run by selecting the "Extract Building Height Statistics" command under the "Building Tools" menu item in the "LiDAR Tools" menu as shown below.
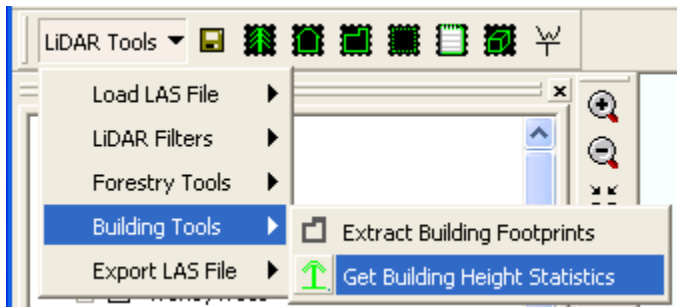
This will bring up the form shown below.

Each of the parameters in the above form are described below.

- **Building Footprint Layer:** A polygon feature class for which statistical attributes will be added and/or calculated.
- **Bare Earth DEM:** An input bare earth DEM encompassing the loaded LAS file.
- **Building Height Statistics to Calculate:** This group of check boxes lists the statistical measures to calculate for each building polygon containing building points.
  - **Count of Classified Points:** Checking this check box will add a field to the input feature class called "BldCount" and return the number of LiDAR points classified as a building for each polygon.
  - **Mean Height Value:** Checking this check box will add a field to the input feature class called "BldMean" and return the average height of all points classified as a building for each polygon..
  - **Maximum Height Value:** Checking this check box will add a field to the input feature class called "BldMax" and return the maximum height value of all points classified as a building for each polygon.
  - **Minimum Height Value:** Checking this check box will add a field to the input feature class called "BldMin" and return the minimum height value of all points classified as a building for each polygon.
  - **Range in Height Values:** Checking this check box will add a field to the input feature class called "BldRng" and return the range of height values of all points classified as a building for each polygon.
  - **Standard Deviation of Height Values:** Checking this check box will add a field to the input feature class called "BldStDev" and return the standard deviation of height values of all points classified as a building for each polygon.

If the attributes already exist in the input polygon feature class the values in the attribute will be overwritten.

# 5.0 EXPORT and SAVE TOOLS

5.1 Export to Text

There are tools in the ArcMap toolbar to allow users to export certain spatial and attribute subsets of the LAS file to text files as well as sub-setting the loaded LAS file to a smaller extent. Users can export the contents of the loaded LAS file to text by selecting the "Export Loaded LAS File to Text" command under the "Export LAS File" menu item in the "LiDAR Tools" menu as shown below.
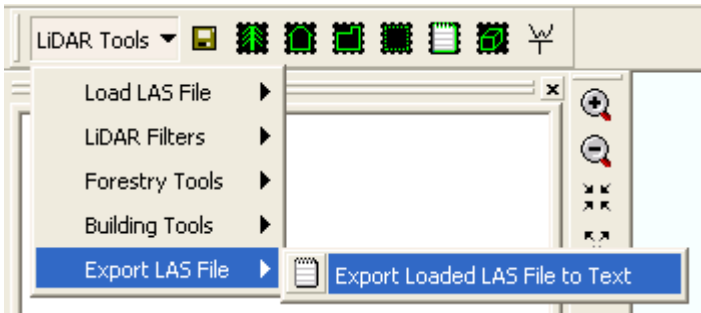
Optionally, a smaller subset of the loaded LAS file can be exported to a text file using the "Export to Text File in Select Extent" command as shown below.

Performing either of the above operations will bring up the form shown below.
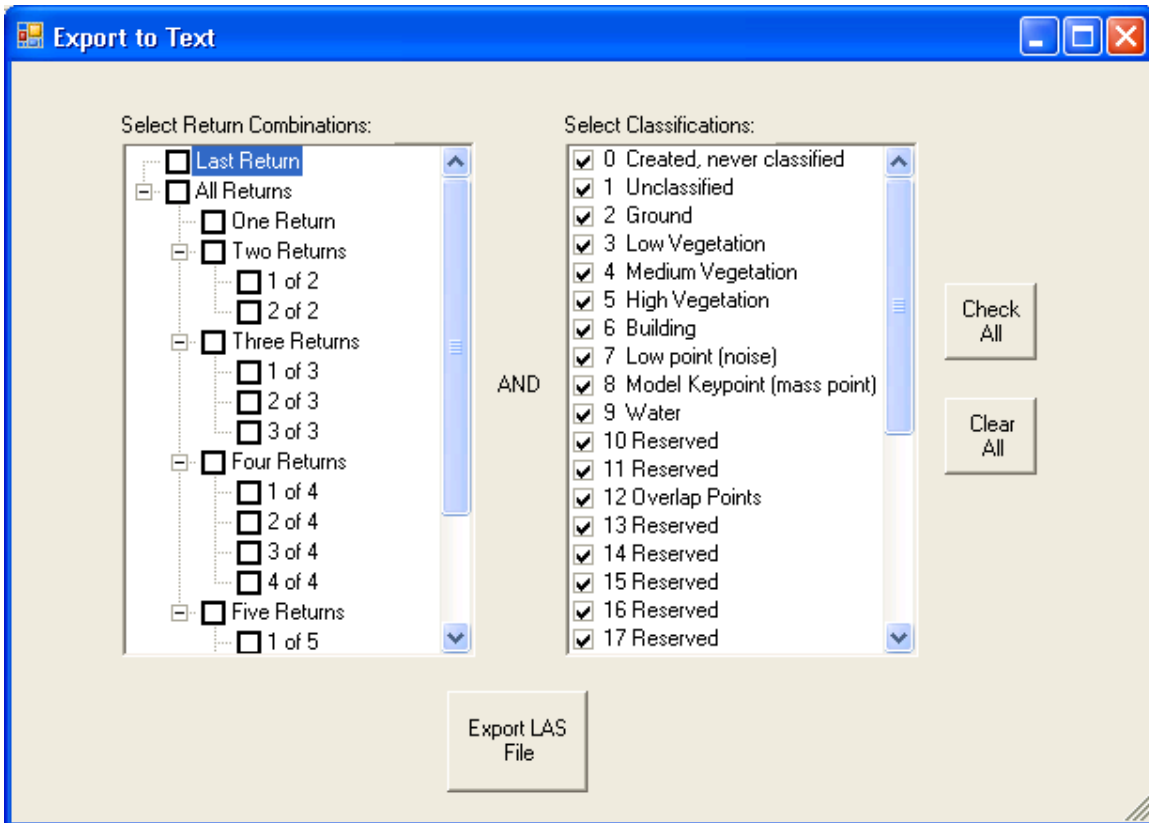
In the form above users can select various return combinations and classification attributes to choose which points will be output to the entered text file.

5.2 Subset LAS File

The subset LAS file can be used to create a new LAS file comprising of LiDAR points within the user defined extent. This functionality is accomplished by selecting the "Subset LAS File" command shown below.

A rectangle can then be drawn to subset the LAS file in this user drawn extent.

5.3 Save LAS File

LAS files are loaded into memory so changes made are not automatically saved. The "Save Loaded LAS File" command shown below will prompt the user for a location to save the LAS file. This can be the same as the originally load LAS file.

6.0 3-D VIEWER

6.1 Activating the Viewer

The 3-D viewer is mainly included to provide a quick means to view the results of the building classifications and cleanup the results. The LiDAR Tools GIS toolbar contains functionality to view the point cloud of LiDAR data in a 3-D environment. This is accomplished by clicking on the "LAS Viewer" command as shown below and drawing a rectangle around the extent of the portion of the LAS file to be viewed.

This will bring up a form as shown below, portraying the point cloud of data.
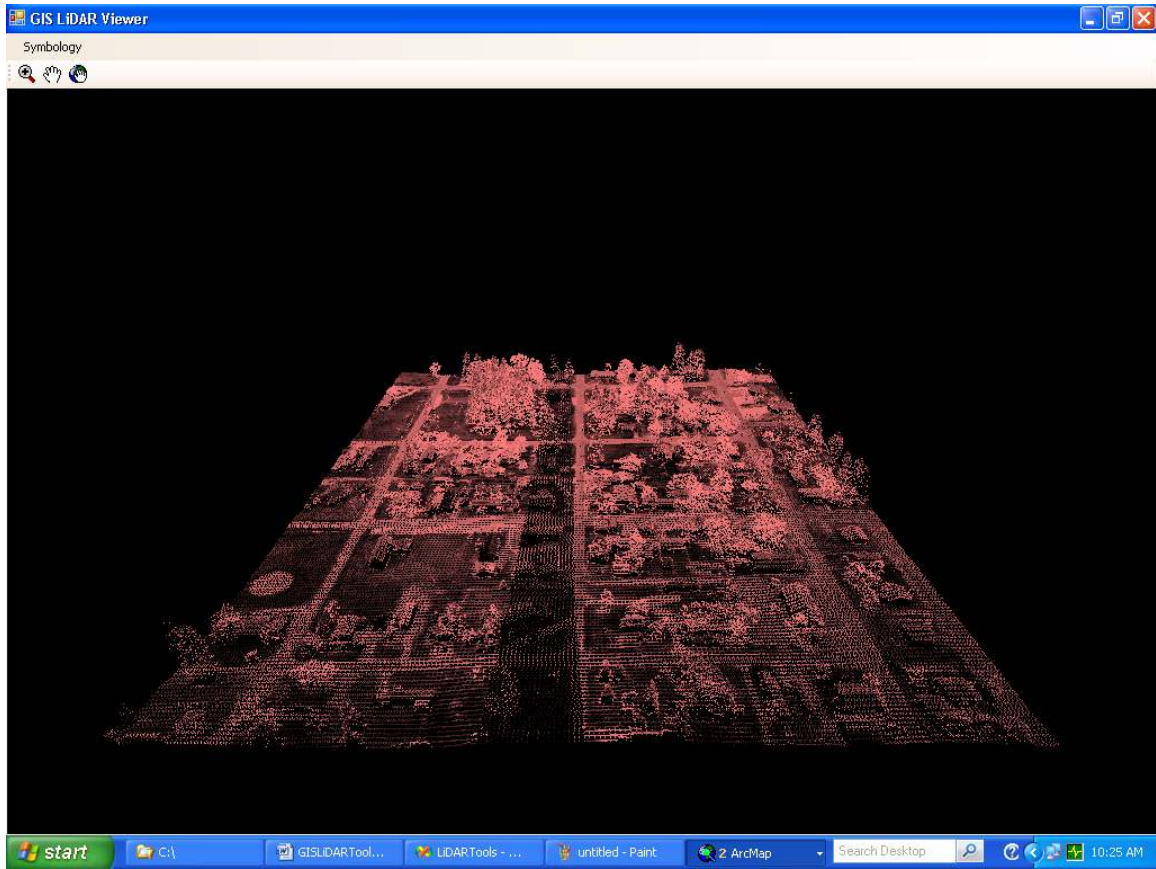
6.2 Viewer Navigation Controls

Users can zoom, pan or orbit around this viewer using three icons contained on the form above and shown below.
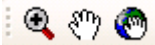

The buttons above, from left to right, are the zoom, pan and orbit buttons. These navigation tools are activated by clicking on the tool of interest. Within the viewer users can right click and drag the mouse to zoom, pan or navigate depending on the desired functionality. These controls are very sensitive and can use some improvement in this regard.

6.1 Viewer Symbology

The symbology menu allows for alteration of point feature symbology by classification value or intensity. The symbolization of classified building points is accomplished by selecting the "Classification Symbology" menu item under the "Symbology" menu as shown below.
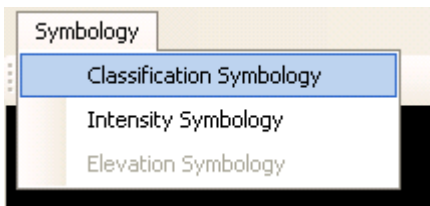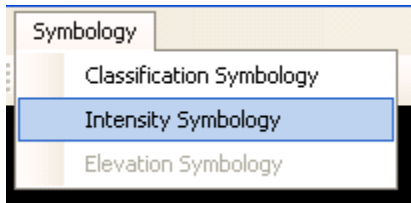
This will bring up the form shown below.
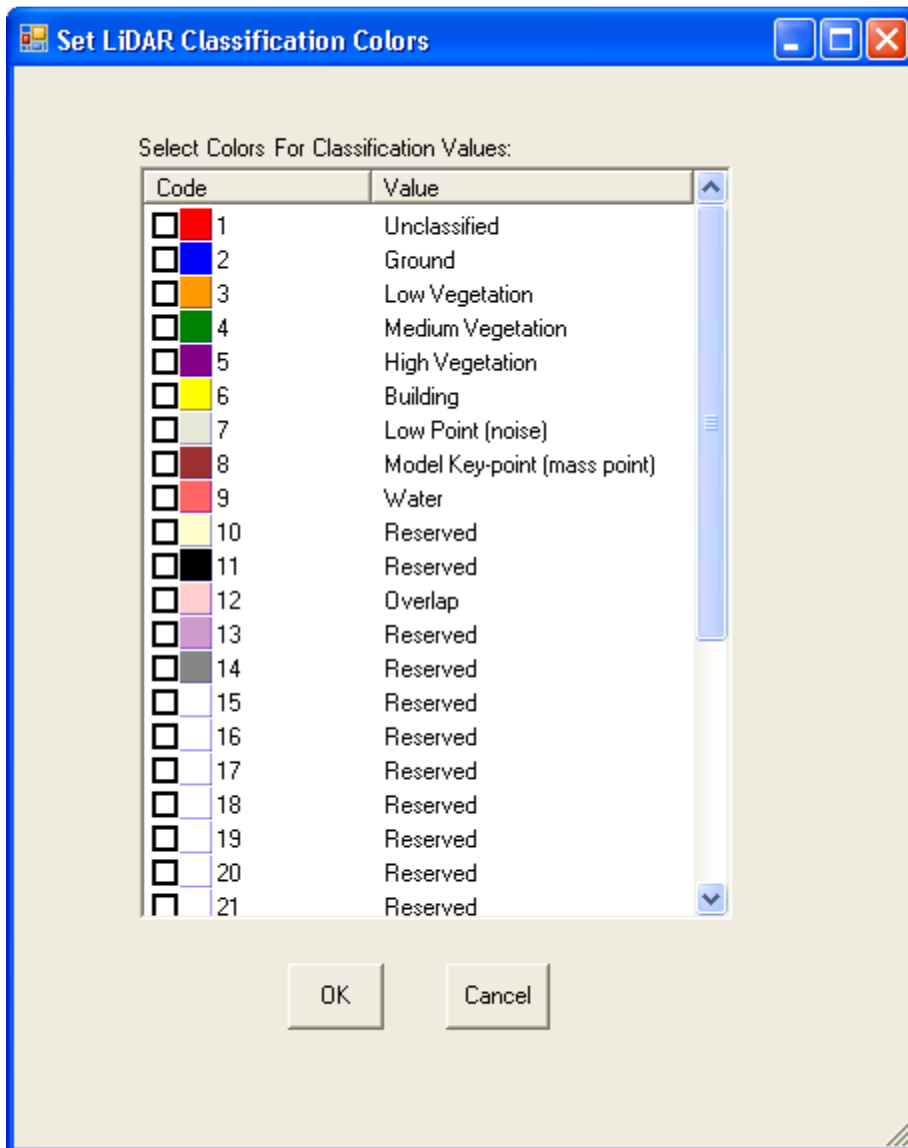
Check boxes can be checked to determine the color for which to display points with the respective classification. Any points with classification values not checked will be displayed in grey. Colors can be changed by clicking on the color box next to the particular point classification.

The symbolization of building points by intensity values is accomplished by selecting the "Intensity Symbology" menu item under the "Symbology" menu as shown below.

This will bring up the form shown below.
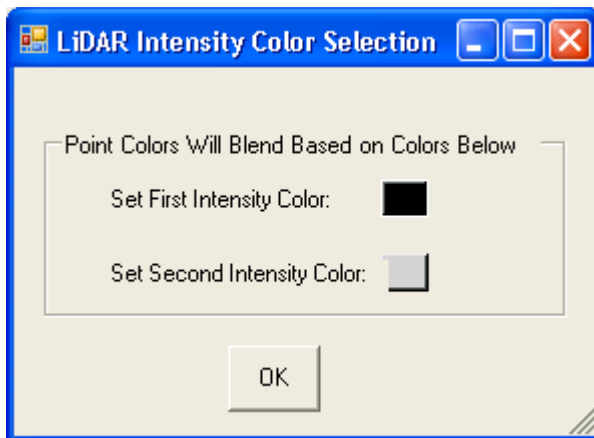
Two intensity colors are selected here and points will be symbolized with a color ramp, representing colors between the two selected colors, based on intensity value of the point.

7.0 WINDOWS BASED TOOLS

7.1 Activating the Windows Viewer

There is a separate windows based utility include in the installation. This utility is activated by going to the installation folder and clicking the "LIDARViewer.exe" to bring up the viewer shown below.
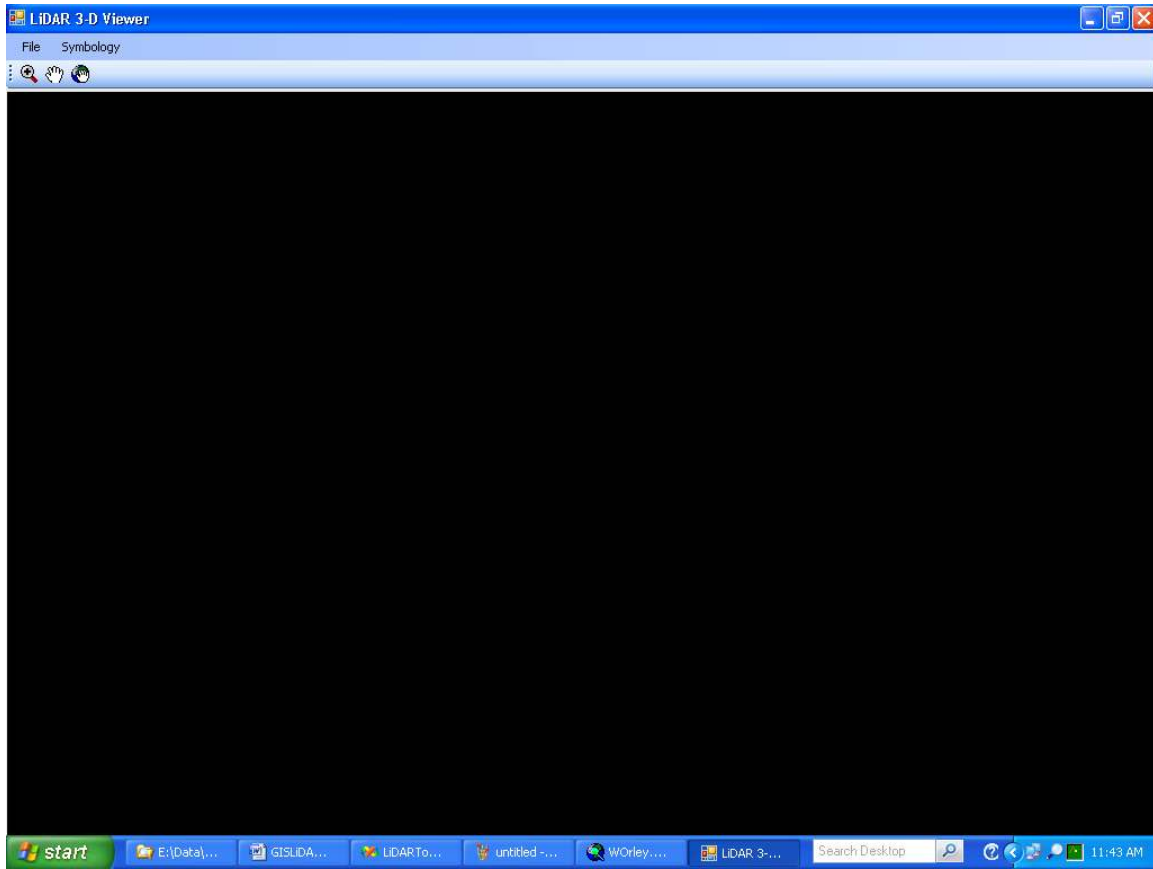
As with the GIS tools the first task is typically to load an LAS file[11]. This is accomplished by clicking the "Load LAS" command in the "Load LAS File" menu item under the "File" menu as shown below.
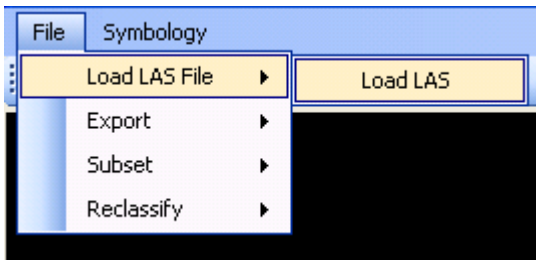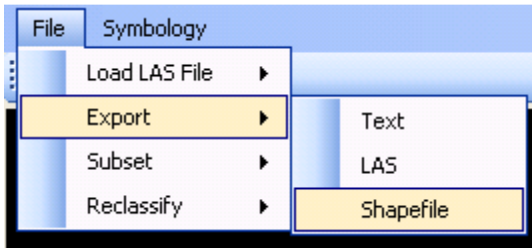

7.2 Export Functionality

After loading the LAS file the points are displayed in a 3-D environment. The symbology can be changed in the same manner as with the GIS viewer. As with the GIS Tools users can export to an LAS file or text file. Additionally, this tool allows for

---

[11] This viewer can load a substantially bigger LAS file than the GIS tools.

export to shapefile[12] by clicking the "Export to Shapefile" command on the "Export"
menu item in the "File" menu as shown below.

7.3 Subset Functionality

The load LAS file can be subset into quarters or a user defined extent using the "1/4s" or
"User Defined Extent" commands in the "Subset" menu item found in the "File" menu as
shown below.
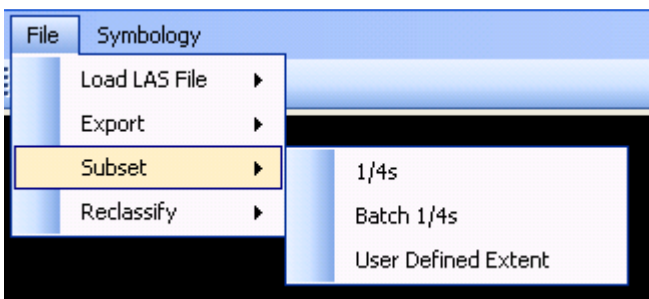
Additionally, a set of LAS files located in a single directory can be subset into quarters
using the "Batch 1/4s" command shown above. This tool, shown below, does not require
any LAS file to be loaded.

---
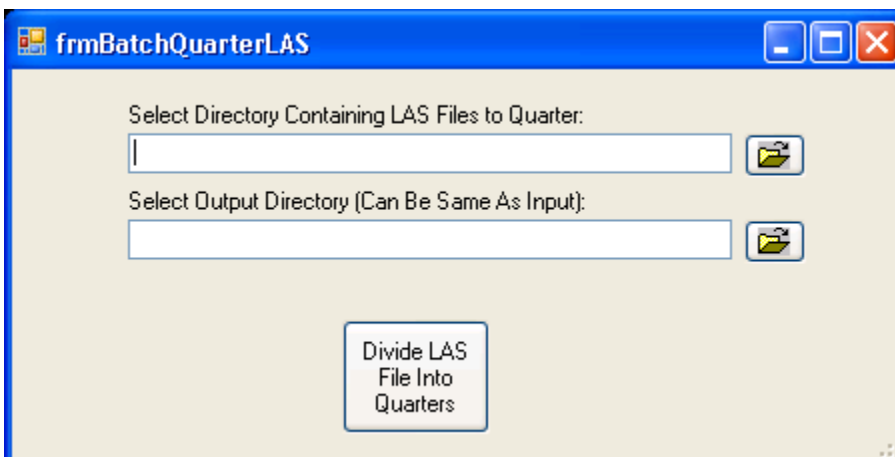
[12] This functionality uses the ascii2shp utility provided by Jason Robert2.

Two directories are selected where the first contains LAS files to subset and the second represents where the results of the subset will be stored. Output files will be named the same as the input files with a "_#" appended to the end of the name where # represents integer values from 1 to 4. Tile 1 will be the northwest tile, tile 2 will be the northeast tile, tile 3 will be the southwest tile and tile 4 will be the southeast tile.

7.4 Reclassification Functionality

Changing classification values for points altered by the building filter can be accomplished by clicking on the "By Files" command in the "Reclassify" menu item in the "File" menu as shown below.
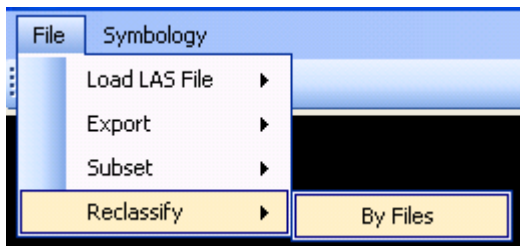
This will bring up the form shown below.
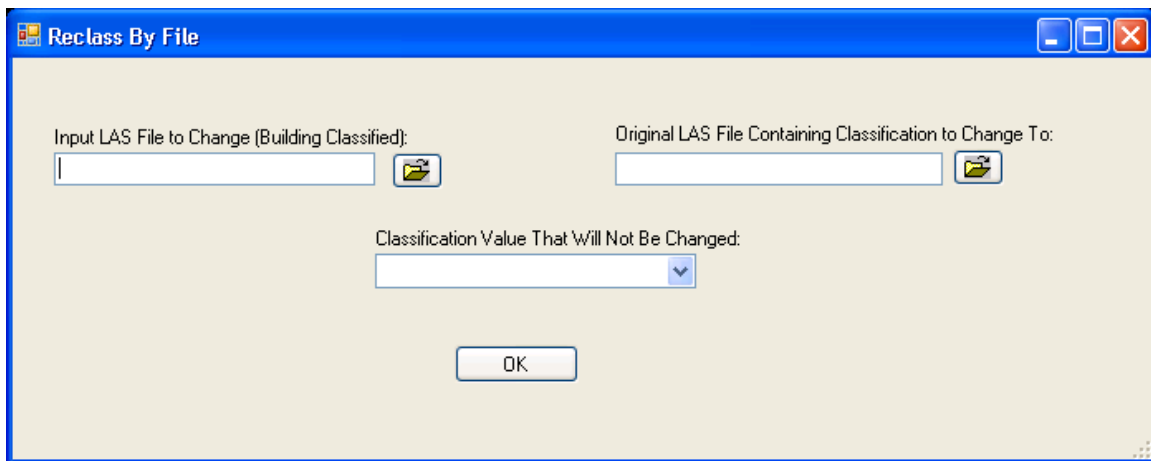
The various text and list boxes are described below.

- **Input LAS File to Change:** This would be the LAS file altered by the building filter.
- **Original LAS File Containing Classification to Change To:** This would be the original LAS file from which the "Input LAS File to Change" was derived.
- **Classification Value That Will Not Be Changed:** This is the classification value for which points in the "Input LAS File to Change" will not be changed to the corresponding point in the "Original LAS File".

Literature Cited

Holmgren, J., and A. Persson, 2004. Identifying species of individual trees using airborne laser scanner, *Remote Sensing of Environment,* 90(4): 415-423.

McGaughey, R.J., H.,E. Andersen, and S.E. Reutebuch. Comparing Estimates of Canopy Structure Derived from Leaf-On and Leaf-Off Airborne Laser Scanner Data. Silviscan: LIDAR Applications in Forest Assessment and Inventory, September 29 - October 1, 2005, Blacksburg, VA.

Popescu, P.C., and A.U. Kini. TREEVAW: A Versatile Tool for Analyzing Forest Canopy LIDAR Data - A Preview with an Eye Towards the Future. Presented at the SPRS Images to Decision: Remote Sensing Foundation for GIS Applications, Kansas City, Missouru, 12-16 September 2004.

Sampath, A., and J. Shan. Building Boundary Tracing and Regularization from Airborne Lidar Point Clouds. Photogrammetric Engineering and Remote Sensing. July, 2007.

Tiede, D., G. Hochleitner, and T. Blaschke. A Full GIS-Based Workflow For Tree Identification and Tree Crown Delineation Using Laser Scanning. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences. Vol. XXXVI, Part 3/W24, Vienna, pp. 9 - 14.

Tovari, D. and N. Pfeifer, 2005. Segmentation Based Robust Interpolation – A New Approach to Laser Data Filtering, IAPRS Vol XXXVI, 3/W3, Proceedings of Laserscanning 2005, Enschede, The Netherlands, pp. 79-84, ISSN 1682-1750

Verma, V., R. Kumar, and S. Hsu, 2006. 3D Building Detection and Modeling from Aerial LIDAR Data, Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol 2, pp. 2213-2220

Wang, M. and Y.H. Tseng, 2004. LIDAR data Segmentation and Classification based on Octree Structure, The proceedings of the XXth Congress of ISPRS, Vol. XXXV, Part B3, pp.308-313.

Appendix A

Trouble Shooting & Known Bugs

Table 13 Known bugs associated with LiDAR Tools.

Known Bug	Error	Work Around
Grids named in the table of contents to something other than the dataset name (i.e., alias) will not work with LiDAR Tools.	RasterDataset:get_Count	Do not use aliases or rename grids in the table of contents.
Loading a large LAS file could cause the application to hang.	Application hangs.	Subset the LAS file to smaller size using the Windows Tools or some other utility.
During the "Partitioning Data" process in the building or tree filters you run out of memory	System.OutOfMemoryException	Close all other applications, remove some of the not need datasets from ArcMap, or subset the LAS file.
When trying to use the 3-D viewer in ArcMap you get an "Error Opening Form".	Attempted to Read or Write Protected Memory	Close the application and all other applications and try again; subset the LAS file.

APPENDIX C: LIST OF RECORDED STRUCTURAL ATTRIBUTES

Table 1: Structure Database Field Collection Attributes

Field	Definition	Source
UNIV_CODE	Universal code. Universal numbering system describing building type for Cartographic Feature Files.	United States Forest Service
HOUSE_NUM	House Number. This is the number assigned by the respective county or city addressing systems	Respective County
ADDRESS	Address for the structure.	Respective County
CROSSRD	Cross road. Last intersection between a county road and the road leading to the structure.	Coeur d'Alene Tribe GIS
BLDGTYPE	Building Type. Describes the type of building	Coeur d'Alene Tribe GIS
OCCUPANCY	Describes the occupancy status of the building.	Coeur d'Alene Tribe GIS
COMMSTAT	Commercial status. Describes the commercial status of the building.	Coeur d'Alene Tribe GIS
BLDGMAT	Building material. Describes what the building material is like, particularly the siding, eaves, and deck.	NFPA 1144
ROOFING	Describes the roofing type.	NFPA 1144
WATERSRC	Describes what type of water source is available to firefighters.	NFPA 1144
ONSITEPRO	On site provisions. Indicates whether or not there are sprinklers inside the buildings. Most likely only to be found in public buildings.	NFPA 1144
INGEGR	Ingress/Egress. Describes how many roads can be used to get to the building.	NFPA 1144
RDWIDTH	Road width. Describes how wide the access road is to the structure.	NFPA 1144 (Modified by Coeur d'Alene Tribe GIS)
RDSURF	Road surface. Describes how wide the access road is to the structure.	NFPA 1144
FIREACCESS	Describes how far from the access road the building is as well as the maneuverability of an emergency vehicle on the property.	NFPA 1144 (Modified by Coeur d'Alene Tribe GIS)
STREETSIGN	Indicates if there is a reflectorized street sign	NFPA 1144

	present.	
VEG	Vegetation. Describes both the vegetation type and density.	NFPA 1144
DEFSPACE	Defensible space. Describes how much available space there is for emergency crews to operate.	NFPA 1144
SLPSETBACK	Slope setback. Indicates how far away the 30% slope is.	NFPA 1144
GASELEC	Gas and electric. Describes where the gas and electricity are coming into the building.	NFPA 1144
BURNBAR	Burn barrel. Indicates whether a burn barrel exists on the property or not.	Coeur d'Alene Tribe GIS
FUELTANK	Indicates whether or not fuel tanks exist on the property and if so, what kind.	Coeur d'Alene Tribe GIS
PHOTOID	This field is used for internal management and is the number of the original image taken of the structure.	Coeur d'Alene Tribe GIS
ADDPOST	This attribute describes if the address is posted on the structure.	Coeur d'Alene Tribe GIS
LOCKGATE	This attribute describes if there is a locked gate present.	Coeur d'Alene Tribe GIS
COLLECTOR	Name or initials of person collecting the data.	Coeur d'Alene Tribe GIS
DATE	The date the structure was inventoried.	Coeur d'Alene Tribe GIS
COMMENT	Any relevant information that would help distinguish the building or other pertinent information about the structure.	Coeur d'Alene Tribe GIS