

# 3D City Model Generation Using Aerial Ortho-Rectified Imagery and LIDAR Data Fusion in Semi-Automatic Way

Loutfia ElSonbaty<sup>a\*</sup>, Mahmoud Hamed<sup>b</sup>, Amr Ali<sup>c</sup>, Mahmoud Salah<sup>d</sup>

<sup>a,b,c,d</sup>Faculty of Engineering Shoubra, Benha University, Cairo, Egypt

<sup>a</sup>Email: [loutfia\\_karam2001@hotmail.com](mailto:loutfia_karam2001@hotmail.com)

<sup>b</sup>Email: [prof.mahmoudhamed@yahoo.com](mailto:prof.mahmoudhamed@yahoo.com)

<sup>c</sup>Email: [amrhali@feng.bu.edu.eg](mailto:amrhali@feng.bu.edu.eg)

<sup>d</sup>Email: [engmod2000@yahoo.com](mailto:engmod2000@yahoo.com)

## Abstract

One of the most interesting research topics in the last decade is generating 3 Dimensions (3D) city model, nevertheless representing a suitable method to achieve the required rapid, automatic, accurate extraction of realistic model for large urban area used for GIS applications and photogrammetry is still a challenging issue. Consequently, a new technique and strategy that increase the efficiency for the 3D city modeling is required. The aim of this research is to develop a simple and efficient semi-automatic approach to generate a 3D city model for urban area using the fusion of LiDAR data and Ortho-rectified imagery. This integration of these data sources provides its efficiency for 3D building extraction that represents the main item in the 3D city model. This approach use both LiDAR data and imagery as the primary cue to delineate building outlines, based on pixel based classification. The third dimension is obtained automatically from normalized digital surface model nDSM, and then the 3D model is generated using multi-Faceted patch. The accuracy assessment for both height and building outlines is conducted using the referring to the ground truth. The results of the accuracy assessment stage illustrated by means of the well-known statistical methods. It is experimentally validated that the proposed approach can successfully detect different types of buildings from simple rectangle to circular –shape, when assessed in terms of different quantitative statistics criteria and visual inspection.

**Keywords:** 3D City model; LIDAR; Ortho-rectified Imagery; Data Fusion; Maximum likelihood classification ; Multifaceted patches.

---

\* Corresponding author.

## **1. Introduction**

Generating an updated and accurate 3D city model are very important topic for a lot of application, as urban planning, transportation, tourism, communication and many other application. Although aerial photogrammetry is still one of the preferred ways to obtain three-dimensional information of the earth's surface, it still has a major drawback in the degree of automation where it still low. Airborne LiDAR is considered one of the important methods used for 3D building model derivation. It able to reconstruct 3D model directly from measured 3D points, so it is improve the degree of automation but the quality of the resulted model is restricted by the ground resolution of the LiDAR data. Therefore generating 3D model with fine details and high precise geometric position from only LiDAR data is difficult [1].

It is clear that the integration between photogrammetry and LiDAR can lead to more complete and accurate product with sufficient level of automation. Different trials are carried out in order to generate 3D city model. Langue [2] developed a practical object- oriented model based method for 3d building extraction by integrating ALS points and aerial imagery.

The object-oriented building model for 3D building extraction is developed using UML by integrating data collection and construction methods, geometry and topology, semantics and properties as well as data storage and management of 3D buildings into one model. Kwak [3] developed a framework for fully-automated building model generation by integrating data-driven and model-driven methods as well as exploiting the advantages of images and LiDAR datasets, but its biggest limitation is that it can model only the types of buildings which decompose into rectangles.

Jarvis [4] outlined the integration of photogrammetric and LiDAR data, within a GIS, for the accurate reconstruction of a 3D realistic urban model in a semi-automated procedure. This paper is organized as follow. The first section presents the importance of generating 3D city model and the previous related works. Second section outlines the data used for the investigation and the methodology which gives the overall approach to extract 3D objects. Third section discusses the evaluation of the method and results, the final section draws conclusion and future works.

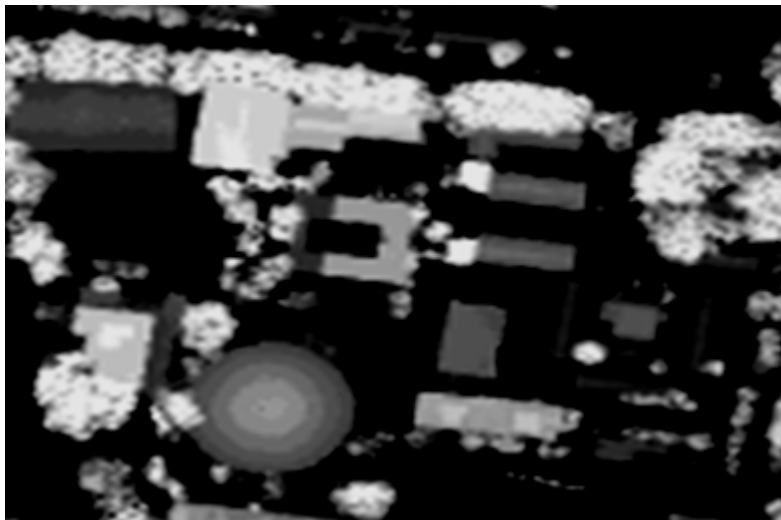
## **2. Materials**

The area is a part of university of New South Wales Campus; Sydney Australia. It is largely urban area containing residential buildings, large campus buildings, a network of main roads as well as minor road, trees, and green areas as shown in Figure 1.

LiDAR data were acquired over study area in April 2005, using an Optech ALTM1225 with a pulse repetition frequency (PRF) of 25 KHz at wavelength of 1.04  $\mu\text{m}$ . the multispectral imagery was captured by film camera by AAMHatch on June 2005 at 1:6000 scales. The film was scanned in three color bands (red, green, and blue) in tiff format, with 15  $\mu\text{m}$  pixel size (GSD of 0.096m) and radiometric resolution of 16-bit as shown in Figure 2 in form of nDSM



**Figure 1:** ortho-rectified image of the test area.



**Figure 2:** nDSM of the test area.

### 3. Methodology

#### 3.1 Data Preparation

##### 3.1.1 Image and LiDAR data co-registration

Image registration is the technique of bringing different datasets into a single coordinate system. However, multiple datasets acquired by different sensors but having the same coordinate system may still require some kind of additional pixel-to-pixel or point-to-pixel matching in order to ensure higher reliability in data fusion techniques. This kind of matching is known as image co-registration [5]. In our test the ortho rectified photograph is registered to the LiDAR intensity image using a projective transformation. Following the

transformation, the image was resampled to 1m x 1m cell size by bilinear interpolation to compensate for the difference in resolution between image and LiDAR data. The employment of the co-registration technique improves the overall alignment of both the datasets.

### 3.1.2 Reference Data

In order to evaluate the accuracy of the classifications, reference data were captured by digitizing buildings, trees, roads and ground in the orthophoto as shown in Figure 3. Class “ground” mainly corresponds to grass, parking lots and bare fields. During this process all recognized features are digitized. Adjacent buildings that were joined but obviously separated were digitized as individual buildings; otherwise, they were merged into one polygon. In order to overcome the horizontal layover problem of tall objects such as buildings, roofs were first digitized and then each roof polygon was shifted if possible so that at least one point of the polygon coincided with the corresponding point on the ground [6].



Figure 3: Reference data

### 3.1.3 Generation of Attributes

Attributes are necessary to compensate for some common problems associated with high resolution image data such as: shadows caused by tall buildings or trees; and the spectral variability within the same land-cover. Features or attributes commonly used for feature detection from aerial images and LiDAR data include height texture or surface roughness of the LiDAR data, reflectance information from aerial images or LiDAR data, the difference between first and last pulses of the LiDAR data, and shape and size of objects. The attributes calculated for predefined segments or single pixels are presented as input data for a classification method [6]. These attributes are derived from what is called Grey-Level Co-occurrence Matrix (GLCM). Grey-Level Co-occurrence Matrix GLCM stores the number of pixel neighborhoods in an image that have a particular grayscale combination. Let  $I$  be an image and let  $p$  and  $N_p$  respectively denote any arbitrary pixel and its neighbor in a given direction. If  $GL$  denotes the total number of quantized gray levels and  $gl$  denotes the individual gray

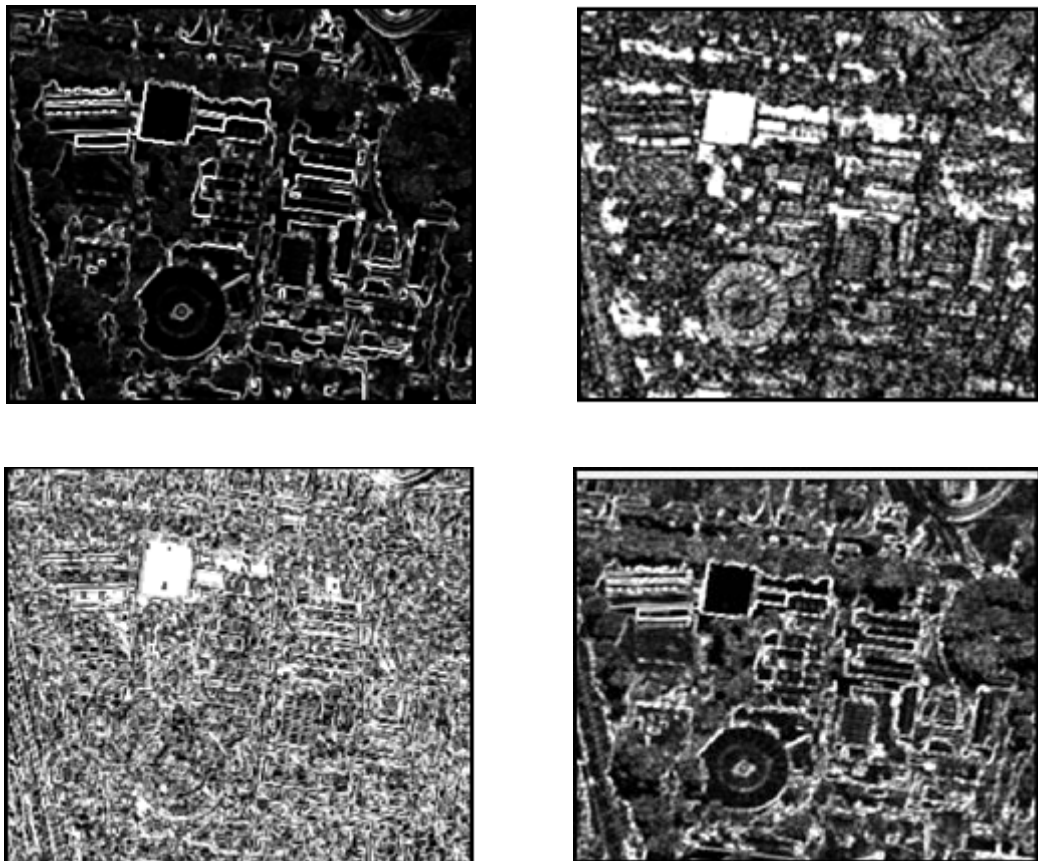
levels, where,  $gl \in \{0, \dots, GL - 1\}$ , then each component of GCM can be written as follows:

$$gcm(i, j) = \Pr(gl_p, gl_{N_p}) = (i, j)$$

$gcm(i, j)$  is the number of times the gray level of a pixel  $p$  denoted by  $gl_p$  equals  $i$ , and the gray level of its neighbor  $N_p$  denoted by  $gl_{N_p}$  equals  $j$ , as a fraction of the total number of pixels in the image. Thus, it estimates the probability that the gray level of an arbitrary pixel in an image is  $i$ , and that of its neighbor is  $j$ . One GCM matrix is generated for each possible neighborhood direction, namely,  $0_$ ,  $45_$ ,  $90_$  and  $135_$ .

Average and range of 14 features like Angular Second Moment, Contrast, Correlation, etc., are generated by combining all the four matrices to get a total of 28 features. In the GCM approach for texture extraction, color information is completely lost since only pixel gray levels are considered [7].

A set of 48 possible attributes describing the terrain form and cover were selected and shown in Table 1 and samples of possible attributes are shown in Figure 4.



**Figure 4:** sample of possible attributes

**Table 1:** The full set of the possible attributes from aerial images and LiDAR data

attributes	Red band	Blue band	Green band	Intensity	DSM	nDSM
contrast	√	√	√	√	√	√
correlation	√	√	√	√	√	√
entropy	√	√	√	√	√	√
homogeneity	√	√	√	√	√	√
mean	√	√	√	√	√	√
Second moment	√	√	√	√	√	√
similarity	√	√	√	√	√	√
variance	√	√	√	√	√	√

**Table 2:** The full set of the possible attributes, 24 attributes, from aerial images and LiDAR data

attributes	Red band	Blue band	Green band	Intensity	DSM	nDSM
correlation	√	√	√	√	√	√
entropy	√	√	√	√	√	√
mean	√	√	√	√	√	√
variance	√	√	√	√	√	√

Because of the way the texture equations derived from the GLCM are constructed, many of them are strongly correlated with one another only 24 of the 48 possible attributes were uncorrelated and hence available for the classification process as shown in Table 2 .

### 3.2 Classification

Image classification refers to the labeling of images into one of a number of predefined categories [8]. It can be said that classification divides the feature space into several classes based on decision rule. Classification is an important task within the field of computer vision; it is a challenging task in various application domains, including biomedical imaging, biometry, video-surveillance, vehicle navigation, industrial visual inspection, robot navigation, and remote sensing [9].

Remote-sensing classification is a complex process and requires consideration of many factors. The major steps of image classification may include determination of a suitable classification system, selection of training samples, image preprocessing, feature extraction, selection of suitable classification approaches, post-classification processing, and accuracy assessment.

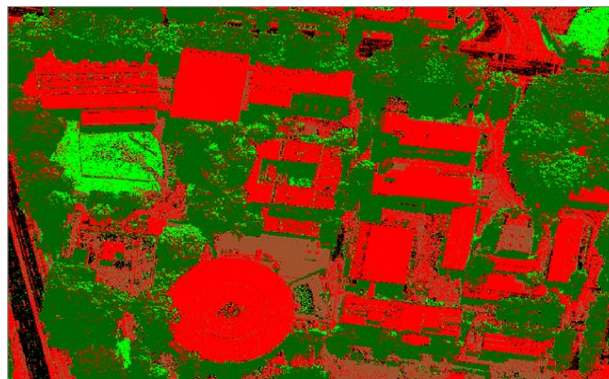
Maximum likelihood classification is used in this research. Maximum likelihood classification assumes that the statistics for each class in each band are normally distributed and calculates the probability that a given pixel belongs to a specific class. Unless you select a probability threshold, all pixels are classified.

Each pixel is assigned to the class that has the highest probability. If the highest probability is smaller than a threshold you specify, the pixel remains unclassified. The basic equation assumes that these probabilities are equal for all classes and that the input bands have normal distributions as shown in Eq. (1).

$$g_i(x) = \ln(p(w_i)) - 0.5 \ln |\Sigma_i| - 0.5(x - m_i)^t \{ \Sigma_i^{-1} (x - m_i) \} \quad (1)$$

where  $x$  is the measurement vector of the candidate pixel,  $w_i$  is a class (e.g. a building, a green space, a road, etc.),  $p(w_i)$  is the probability that the class  $w_i$  belongs to the image,  $m_i$  is the mean vector of the data in class  $w_i$ ,  $\Sigma_i$  is the covariance matrix of the data in class  $w_i$ ,  $|\Sigma_i|$  is the determinant of the covariance matrix, and  $t$  is a transposition function.

The pixel is assigned to the class  $w_i$ , for which  $g_i(x)$  is the largest [10]. The result of Maximum likelihood classification is shown in Figure 5.



**Figure 5:** Classified image.

### 3.2.1 Assessment of classification accuracy



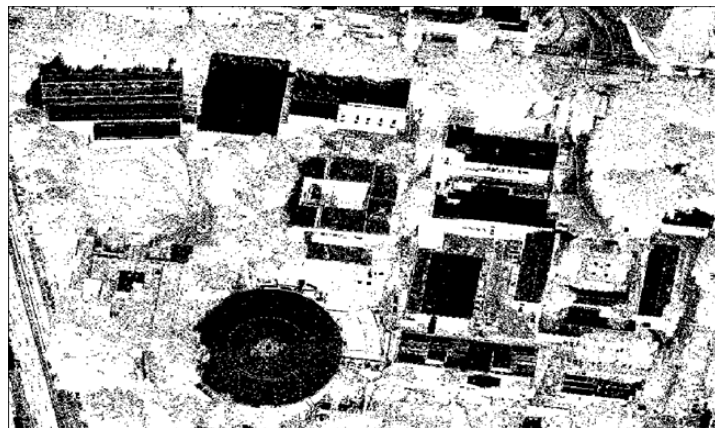
The obtained overall classification accuracy was 82.49%, while the per-class accuracies were 90.96%, 99.84%, 58.00% and 64.41% for buildings, trees, roads and grass respectively.

### 3.3 Post-processing

#### 3.3.1 Binary Image and morphologic operation

Binary images are images whose pixels have only two possible intensity values. They are normally displayed as black and white. Numerically, the two values are often 0 for black, and either 1 or 255 for white. Binary images are often produced by thresholding a grayscale or color image, in order to separate an object in the image from the background. The color of the object (usually white) is referred to as the foreground color. The rest (usually black) is referred to as the background color [11].

In order to extract the buildings from the classified image, it is converted to binary image with two indexes 1 and 0. Digital values of Buildings are often converted to 1, while digital numbers of other classes are converted to 0. The resulted binary image still has noisy feature as shown in Figure 6, so morphologic operation is needed.



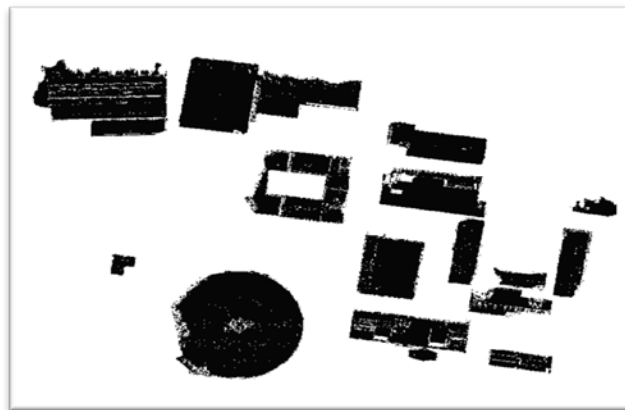
**Figure 6:** Binary image with noise.

Morphological operations are affecting the form, structure or shape of an object in binary image. They are used in pre or post processing (filtering, thinning, and pruning) or for getting a representation or description of the shape of objects/regions [12].

The basic operations of binary morphology are dilation, erosion, closing, and opening where the results of these operations are shown in Figure 7 and 8. As the names indicate, a dilation operation enlarges a region, while erosion makes it smaller. A closing operation can close up internal holes in a region and eliminate "bays" along the boundary. An opening operation can get rid of small portions of the region that jut out from the boundary into the background region [13]. The foundation of morphological processing is in the mathematically rigorous field of set theory; however, this level of sophistication is seldom needed. Most morphological algorithms are simple logic operations. In other words, each application requires a custom solution developed by trial - and-error. This is usually more of an art than a science [14].As mention before the set of black and white pixels



constitute a description of a binary image. Assume that only black pixels are considered, and the others are treated as a background. The primary morphological operations are dilation and erosion, and from these two, more complex morphological operations such as opening, closing, and shape decomposition can be constituted. The morphological transformation dilation (+) combines two sets using vector addition. The dilation operation can be done by performing vector addition of the pair of elements for both the sets X and B [15]. We can't get back the original image by performing erosion followed by dilation and vice versa. Instead, the result is a simplified and less detailed version of the original image. Erosion followed by dilation creates an important morphological transformation called opening. The opening of an image X by the structuring element B is denoted by  $X \circ B$  and is defined as  $X \circ B = (X (-) B) (+) B$ . Dilation followed by erosion is called closing. The closing of an image X by the structuring element B is denoted by  $X \bullet B$  and is defined as  $X \bullet B = (X (+) B) (-) B$ .



**Figure 7:** Binary image before closing operation.

Once the classified image is converted to binary image with noisy features, the fundamental operations of morphologic operation are required to merge smaller homogeneous region into larger one or deleted according to specific threshold. Therefore the buildings with area larger than the determined threshold are kept and adjacent to a larger homogenous region by a distance larger than the specific threshold so the result is a binary image without any noise



**Figure 8:** Binary image after closing operation.

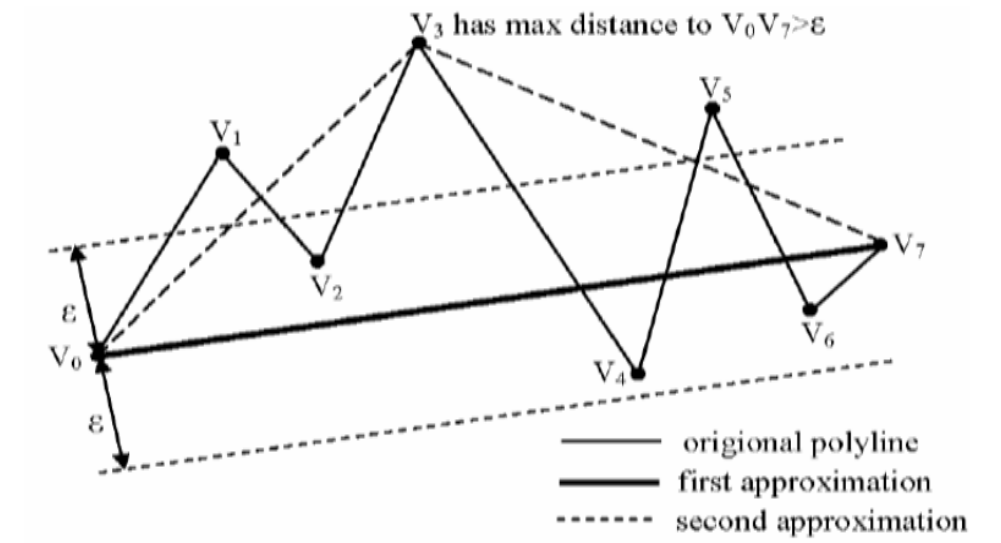
### 3.3.2 Vectorization and Generalization

In this step the smoothed binary image is converted from raster to vector format to extract building boundaries. But due to the nature of the raster process, the resulted boundaries need more processing to overcome the problems of irregularities as shown in Figure 10 and adjust the rectangularity of the polygons as shown in Figure 11.

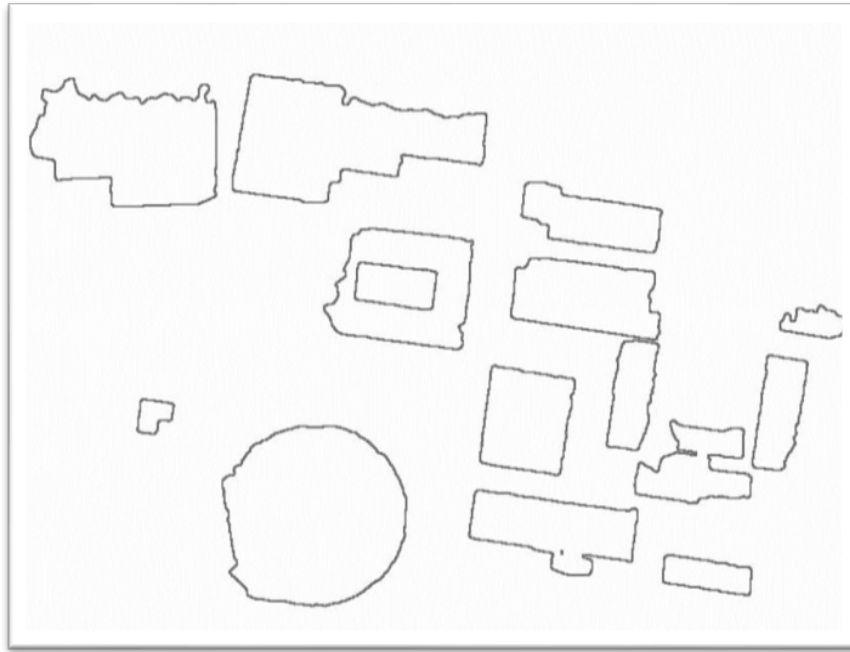
This is done through generalization process. One of the most common generalization algorithms is The Ramer–Douglas–Peucker algorithm (RDP) is an algorithm for reducing the number of points in a curve that is approximated by a series of points. The initial form of the algorithm was independently suggested in 1972 by Urs Ramer and 1973 by David Douglas and Thomas Peucker and several others in the following decade [6].

This algorithm is also known under the names Douglas–Peucker algorithm, iterative end-point fit algorithm and split-and-merge algorithm. The algorithm recursively divides the line. Initially it is given all the points between the first and last point. It automatically marks the first and last point to be kept. It then finds the point that is furthest from the line segment with the first and last points as end points (this point is obviously furthest on the curve from the approximating line segment between the end points). If the point is closer than user defined value (tolerance=  $\epsilon$ ) to the line segment then any points not currently marked to be kept can be discarded without the simplified curve being worse than  $\epsilon$  [16].

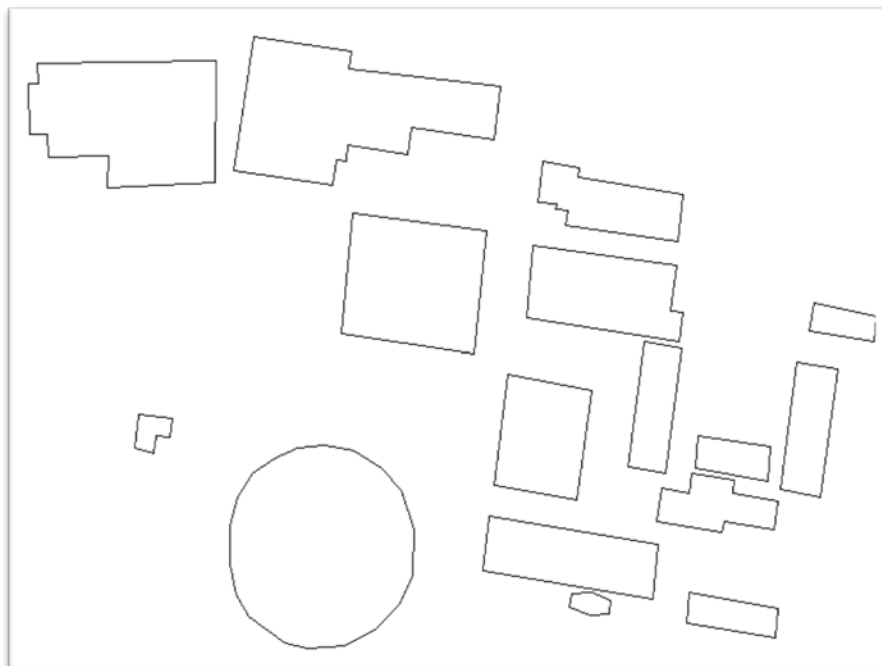
If the point furthest from the line segment is greater than  $\epsilon$  from the approximation then that point must be kept. The algorithm recursively calls itself with the first point and the worst point and then with the worst point and the last point (which includes marking the worst point being marked as kept). When the recursion is completed a new output curve can be generated consisting of all (and only) those points that have been marked as kept. This concept is shown in Figure 9[16].



**Figure 9:** Smoothing a line segment with the Douglas–Peucker algorithm



**Figure 10:** Buildings map after raster to vector conversion.



**Figure 11:** Buildings map after generalization.

### ***3.4 Horizontal Accuracy Assessment***

In order to evaluate the planimetric accuracy of the resulted vector map, three GCPs were determined by field surveys. The GCPs were selected to be evenly distributed throughout the study area as shown in Figure 12, and a comparison was carried out between GPS observations and the extracted building data coordinates as shown in

Table 3.



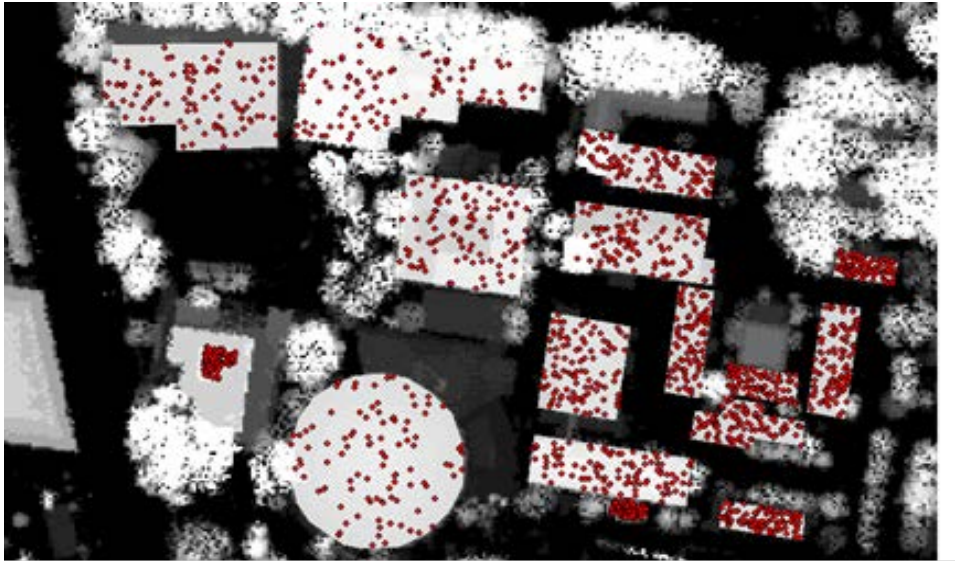
**Figure 12:** Distribution of the GPS control points.

**Table 3:** The accuracy estimate of the building vectorization process

Emap	Nmap	EGPS	NGPS	$\Delta E$	$\Delta N$	$\sqrt{\Delta E^2 + \Delta N^2}$
336011.97	6245795.00	336011.44	6245794.54	-0.53	-0.46	0.7018
336196.91	6245778.50	336196.64	6245777.47	-0.27	-1.03	1.0648
336164.00	6.245700.50	336163.01	6245701.74	-0.99	0.97	1.3860
Mean				0.59	0.82	
RMS				1.18		

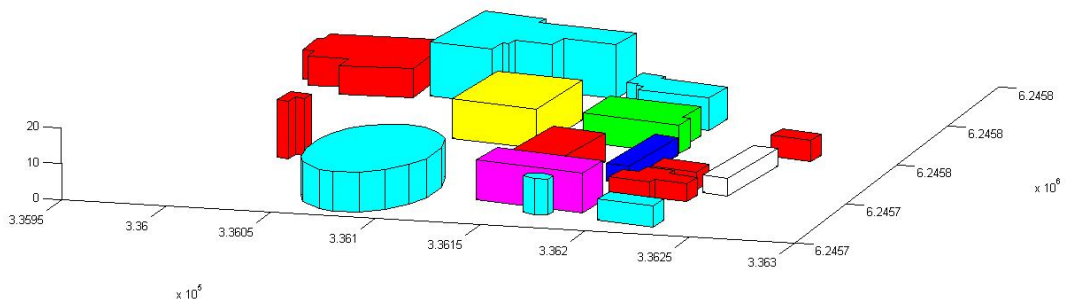
### 3.5 3D model construction

In order to construct 3D model two main items must be exist, building outlines and height information. In the proposed methodology, to determine the heights of the buildings aset of random sample points, constrained by building footprints are generated as shown in Figure 13 and the result is a feature class containing groups of points, one group for each building. Elevation information extracted from elevation surface can be added to each point as an attribute then; the elevation information is summarized to generate a single value for each building.



**Figure 13:** Distribution of random sample points.

MATLAB code is used to construct 3D model. Multi-Faceted Patches are used as follow; if a building will be constructed, x, y, and z coordinate arguments are specified as matrix. MATLAB draws one polygon per column, producing a single patch with multiple faces [17]. Based on the previous concept, the 3D building model is constructed as shown in Figure 14.



**Figure 14:** sixteen building extracted from the study area.

### 3.6 Vertical accuracy

The vertical accuracy of the constructed 3D building model is within 15- 20 cm which match with vertical accuracy of LiDAR.

### 4. Conclusion and Recommendations

This paper discusses the potential of LIDAR/ aerial ortho-rectified imagery data fusion in order to construct 3D city model. In the proposed methodology a data classification was carried out and the overall accuracy reached to 82.49% and the horizontal accuracy of the building outlines was 1.18mt while vertical accuracy was within

15- 20 cm as that of LiDAR. While the limitation of the proposed methodology is the degree of level of details (LOD), it cannot reach to more than LOD1 which represent a simple block model, In order to maximize the benefits of the proposed 3D model reconstruction, it is recommended to use other method of classification as rule based classification and investigate the possibility of dealing with other building shape like gable building and shed roof building and increase degree of automation and degree of details.

### **Acknowledgements**

I wish to acknowledge AAMHatch for the provision of the UNSW dataset.

### **Reference**

- [1] L. Cheng, L. Gong, M. Li, Y. Liu. "3D Building Model Reconstruction from Multi-View Aerial Imagery and LiDAR Data." *Photogrammetric Engineering and Remote Sensing*, vol. 77, PP. 125-139, 2011.
- [2] W. Langue. "Object-Oriented Model based 3D Building Extraction using Airborne Laser Scanning Points and Aerial Imagery." MSc. Dissertation, Institute of Geo-Information Science and Earth Observation, ITC, Netherlands, 2007.
- [3] E. Kwak. "Automatic 3D Building Model Generation by Integrating LiDAR and Aerial Images Using a Hybrid Approach". Ph.D. Dissertation, Department Of Geomatics Engineering, University of Calgary, Canada, 2013.
- [4] A. Jarvis. "Integration of Photogrammetric and LiDAR Data for Accurate Reconstruction and Visualization of Urban Environments." MSc. Dissertation, Department Of Geomatics Engineering, University of Calgary, Canada, 2008.
- [5] A. Fikri. "A Methodology for Processing Raw LiDAR Data to Support Urban Flood Modeling Framework." Ph.D. Dissertation, Institute for Water Education, Delft University of Technology, Netherlands, 2012.
- [6] M. Salah. Towards Automatic Feature Extraction from High Resolution Digital Imagery and Lidar Data for GIS Applications. Ph.D. Dissertation, Department Of Surveying Engineering, University of Benha, Egypt, 2010.
- [7] A. Vadivel, S. Sural, K. Majumdar." An Integrated Color and Intensity Co-Occurance Matrix". *Pattern recognition letters*. vol. 28, pp.974-983, 2007.
- [8] P. Kamavisdar, S. Saluja, S. Agrawal. "A Survey on Image Classification Approaches and Techniques." *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 2, pp. 1005-1009, Jan. 2013.
- [9] D. Lu, Q. Weng. "A Survey of Image Classification Methods and Techniques for Improving Classification Performance." *International Journal of Remote Sensing*, vol. 28, pp. 823-870, Mar. 2007.

- [10] J.A. Richards. Remote Sensing Digital Image Analysis. Berlin: Springer-Verlag, 2013, pp. 247-252.
- [11] P. Fisher, S. Perkins, A. Walker, E. Wolfart. Hypermedia Image Processing Reference .New York: J. Wiley & Sons, 1996, pp. 117-145.
- [12] The MathWorks. User's guide for Matlab. The MathWorks, Inc., 2013. <http://www.mathworks.com/help/images/morphology-fundamentals-dilation-and-erosion.html>
- [13] L. G. Shapiro, G. Stockman. Computer Vision .New Jersey: Prentice Hall, 2011, pp. 78-86.
  
- [14] S.W. Steven. The Scientist and Engineer's Guide to Digital Signal Processing. San Diego: California Technical Publishing, 1998, pp. 436-442.
- [15] G. Amalorpavam, T. Harish, J. Kumari, M. Suresha. "Analysis of Digital Images Using Morphological Operations." International Journal of Computer Science & Information Technology (IJCSIT), vol. 5, pp.145-159, Feb. 2013.
- [16] D. Douglas, T. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. The Canadian Cartographer, 1973, pp.112-122.
- [17] The MathWorks. 3-D Visualization. The MathWorks, Inc., 2015. [http://www.mathworks.com/help/pdf\\_doc/matlab/visualize.pdf](http://www.mathworks.com/help/pdf_doc/matlab/visualize.pdf)