IMAGE PROCESSING AND FEATURE EXTRACTION FOR BUILDING INFORMATION MODELLING

Caitlin McHugh[1], Samsung Lim[2]

School of Civil and Environmental Engineering, University of New South Wales, Sydney, NSW 2052, Australia

[1]caitlin.mchugh@unswalumni.com, [2]s.lim@unsw.edu.au

Abstract

As-built Building Information Models (BIMs) have the potential to improve construction performance by replacing conventional documentation, facilitating greater access to site information and providing more accurate representations than models based on Computer-Aided Design (CAD) drawings. Applications using as-built BIMs to improve construction processes rely on efficient and accurate collection of data to emulate the dynamic nature of a construction site. Current methods used to collect and process data for building information models are time intensive or require specialist equipment. In contrast, applications based on computer vision only require a digital camera and can be run on a personal computer. The main aim of this study is to investigate the use of image processing to extract information about building geometries. In this paper, popular feature extraction algorithms in obtaining information about façade geometries such as corners and edges were assessed. The feasibility of identifying areas of windows from extracted geometries was also investigated, as locations and areas of windows are important in the energy analysis of existing buildings. A number of promising results were produced, however, further work is required before feature extraction can be considered as a viable alternative for collecting information for as-built BIMs.

Keywords: BIM, As-built BIM, façade geometry, feature extraction, image processing

1. INTRODUCTION

The Associated General Contractors of America define Building Information Modelling (BIM) as “the development and use of a computer to simulate the construction and operation of a facility” (Ernstrom, 2006, p.3). In contrast to BIM based on Computer-Aided Design (CAD) plans, as-built building information models provide an accurate representation of site conditions at the time of data collection. Recently a focus has been placed on applications of as-built BIM which is being used to improve construction processes and site safety. Replacing conventional paper documentation, as-built BIM can be used for the design of extensions on existing structures while recently they have been used in automated systems that check compliance with construction specifications and existing conditions (Boukamp and Akinci, 2007). Automated rule checking systems have been used to check compliance with disability and fire standards (Delis and Delis, 1995) as well as applied to geometries and attribute characteristics stored in building information models to identify and display site safety hazards (Melzner et al., 2013).

Applications based on as-built BIM also have the potential to assist with the Facilities Management (FM) of buildings. Becerik-Gerber et al. (2011) investigated applications of as-built BIM in the space management, visualisation and marketing (such as for animated walk-throughs) of existing buildings. They highlighted the importance of as-built BIM in the emergency management in buildings which can be used to locate dangers as well as identify evacuation routes and risks to emergency personnel. Key factors limiting the application of BIM in these areas were identified as large amounts of work needed to create the models and the lack of interoperability between BIM and FM systems.

Traditional methods of collecting site information include the use of tape measures and total stations which can be time consuming or require specialist equipment. More recently 3D aerial or land-based laser scanners have been used to collect information for as-built BIM. Although these technologies enable the automated collection of many point measurements from a single site, there is currently no commercial software available which is able to perform all the processes that are required to construct a building information model from a point cloud, resulting in high time and labour costs especially for large projects. Tang et al. (2010) presented a review of the technologies attempting to automate the process of constructing as-built building information models from data collected from laser scans. Recently there has been a focus on the use of computer vision feature extraction to aid with the pre-processing of data, for example, by filling in occlusions caused by foreground obstructions (Frueh et al., 2005).

The advantage of programs that rely on photogrammetry and computer vision is that they only require the use of a digital camera and personal computer, and they even have the potential to be developed to run on mobile phone applications. The purpose of this paper is to investigate the feasibility of using image processing algorithms to extract information about edges, corners and areas of windows from digital pictures for use in as-built BIM. The authors investigated the application of the Canny Edge Detector (Canny, 1986) and the Harris Corner Detector (Harris and Stephens, 1988) in obtaining information about edges and corners of buildings from digital pictures. The feasibility of identifying windows from façade edge geometries is also investigated, as these parameters are essential for the energy analysis of existing buildings.

2. DETECTION OF BUILDING AND FAÇADE FEATURES FOR BUILDING INFORMATION MODELLING

The light detection and ranging known as lidar is one of the most common technologies being used to collect measurements of building geometries. The processes used to construct BIM models using lidar include data collection or scanning, pre-processing and modelling. A lidar scanner collects point measurements by measuring the distance from a sensor to the target as it rotates about its axis. Although lidar scanners enable the automated collection of large amounts of data from a single site, the point clouds must undergo pre-processing before meaningful geometric models can be extracted.

It is common for scans to be taken from multiple observation points due to large project size or obstructions in the line of site of the lidar scanning device. Therefore pre-processing involves transforming data collected from different axis onto a common co-ordinate system (registration) and the removal of unwanted points for example due to reflections. Some of the methods used to manually extract geometric models from the lidar point clouds include fitting geometric primitives such as surfaces or volumes by selecting data points or by using knowledge of the plan view (Haala et al., 1998).

Another method uses extrusion cross sectional surfaces. For example, plan views are constructed from horizontal and vertical cross sections before a vertical profile is constructed by extruding the horizontal cross sections within constraints of the vertical cross sections. Other methods include using triangular meshes to model more complex shapes or using objects from known databases (Campbell and Flynn, 2001). In practice the processes of extracting BIM from dense point clouds is usually manual or only semi-automated, resulting in high time and labour costs especially for large projects. Most BIM packages are unable to create a building information model directly from geometric primitives. The need to shuffle between software packages can lead to interoperability issues.

3. COMPUTER VISION FOR BUILDING INFORMATION MODELLING

Unlike laser scanning, stereo and computer vision techniques do not require specialist equipment or large amounts of manual pre-processing and are increasingly being used in the detection of building and façade features. The following sections present image-processing algorithms which can be used to extract information about geometric parameters of common architectural features.

A program was developed to extract key geometries such as edges and corners of buildings as well as identify areas of windows given an input image of a building façade. Buildings are selected from the University of New South Wales (UNSW) campus in Sydney, Australia. Run on the Python 2.7 interface, it utilised computer vision functions available in the open source computer vision library OpenCV. The processes used to extract key building geometries include image processing, corner detection, foreground extraction and edge detection. It should be noted that the proposed method is only semi automated. User interaction is required in foreground extraction, to specify parameters for corner detection and the hysteresis thresholds for edge detection.

3.1. Feature Detection

The limitations of obtaining structural information from images of building façades are well documented and have been reflected in the results of this research. These include image artefacts introduced due to camera resolution details (e.g. moiré), the effects of variations in illumination over an image, perspective distortions as well as obstructions in the foreground of images including trees and statues. These factors are unavoidable in images taken of real world scenes however their effect on results can be minimised through parameter selection methods and through image pre-processing.

Manual threshold selection for edge detection is time intensive, complicated and due to its subjective nature introduces an aspect of human error. Future works may benefit from investigating the use of automated threshold selection where thresholds are determined based on characteristics of the individual images. This is to take into account differences in architectural features and lighting conditions (Medina-Carnicer et al., 2009; Fang et al., 2009; Hancock and Kittler, 1991).

Variations in illumination over individual images also affected the results of edge detection. Specifically edges in area of shading were left undetected due to weaker gradient responses to edges. This is a significant limitation as shading is a prevailing characteristic of façades due to varying surfaces and depths. Results could be improved by calculating thresholds based on local image statistics to take into account changes in illumination due to shading and lighting. Future works may benefit from investigating the application of adaptive threshold methods such as those proposed by Khallil and Aggoun (2006) which attempts to identify thresholds based on statistical analysis of local gradient magnitude histograms.

As well as edges being undetected a number of false responses were produced due to textures on building façades. Results may be improved through the use of alternative edge detection algorithms which exploit the nature of physical edges in building façades. For example, an algorithm proposed by Gregson attempts to identify significant edges based on the assumption that they are more likely to be locally straight than noisy edges (Gregson, 1993).

Distortions in the edge image were created by obstructions concealing parts of the building façades. Future studies may attempt to address this issue through the application of algorithms (Toyama et al., 1999) and commercial software such as those implemented by Böhm (2004) which have been developed to detect and remove moving objects such as pedestrians and cars by using multiple images of an object taken from a single station. Other works attempt to remove stationary obstructions such as trees and statues. This is done by taking multiple images of the building façade from different frames and using the effect of parallax to model, then remove occlusions in the same manner as for moving obstructions.

3.2. Corner Detection

Unlike edges, corners are discrete and thus their identification is not only important for providing connectivity between edges but for providing reference points which can be used to locate and orientate objects within a real world coordinate system. Corners in a digital image are created by discontinuities in depth, surface orientation as well as changes in illumination and are characterised by high intensity changes in more than one direction. Algorithms that attempt to identify corners in images generally fall into one of two categories: template- or geometry-based methods. Zheng et al. (1999) presented a review of different corner detection algorithms applied on grey level images evaluating them in terms of detection, localisation, stability (where corners are detected at the same position in different images taken from the same object) as well as the complexity of the algorithm and speed of implementation. Template-based corner detection algorithms involve measuring the correlation between an $n \times n$ square and sub-windows of the image using templates of possible corner configurations (Davies, 2012). Alternatively, geometry-based corner detectors attempt to identify corners based on properties of differential geometry, for example, by identifying corners as points along edges where the change in intensity gradient is a maximum (Kitchen and Rosenfeld, 1980) or as topological points on the image surface (Deriche and Giraudon, 1993).

Corner points were identified in digital images of building façades using the Harris Corner detector. The Harris corner detector is based on an algorithm which examines the change in gradient as a window (w) is shifted in different directions (Moravec, 1980). A mathematical formulation for the change in intensity (I) produced by shifting a window (u, v) at a point (x, y) is given in Equation (1).

$$E(u, v) = \sum_{x,y} w(x, y)\ [I(x + u, y + v) - I(x, y)]^2 \qquad (1)$$

If there is a large intensity change in more than one direction, then the window claimed to contain a corner. If the shift in the window only causes an intensity change in one direction, then it contains an edge. Lastly, if there is no change of intensity, then the window does not contain any key features. The Harris corner detector typically uses a circular- or circularly-weighted window such as a Gaussian window where the intensity of a pixel is compared to that of pixels in the surrounding region weighted according to the Gaussian function. This reduces the influence of background noise by applying a greater weight to the intensity differences between pixels in the centre of the window.

The Harris corner detector uses a scoring system where each pixel is assigned a score R which is calculated using the eigenvalues of the Taylor expansion of Equation (1). In this scoring system, a large R corresponds to a high intensity gradient in more than one direction. Results are refined using non-maximal suppression and hysteresis thresholding. Non-maximal suppression is used to mark pixels which are a maximum compared to its eight adjacent pixels which reduces the influence of natural oscillations which occur around corners in digital images. Hysteresis threshholding is applied to mark edges with an R value above a value k which is determined empirically.

To evaluate the performance of the Harris corner detector, the authors adopted the definition of a corner presented by Mokhtarian and Mohanna where a corner is "an image point where a two dimensional change can be observed in the image" (Mokhtarian and Mohanna, 2006, p. 81). It is the authors' aim to investigate the performance of the Harris corner detector in identifying corners in building façades which can be used to locate extracted building geometries in real world coordinate systems.

Corner detection was implemented prior to foreground extraction. It was found that applying the Harris corner detector on the foreground image lead to high responses on the boundary between foreground and background. Similarly a number of false responses were also detected in façade features such as railings and louvers, which reflects the Harris Corner detector's high response to edges due to noise, pixilation and quantisation.

The Harris corner detector requires three input parameters, the size of the input window, the value for parameter k and the kernel size of the Sobel operator which is used to perform image smoothing and to estimate the intensity gradient in each window. Manual parameter selection was a difficult and time-consuming process which involved a trade off between good corner detection and minimising the response to noise. Finding parameters which produced good results across all images was further complicated by varying façade textures of building façades.

Highly textured façades such as that shown in Figure 1 lead to the identification of a number of points of limited physical significance such as corners of bricks. A lower error rate would have been produced if parameters were selected based on the characteristics of individual images. For the purpose of this research, however, only the successful identification of a small number of key points to geo-reference the digital image of building façades is required, and all erroneous responses can be ignored. Thus the adoption of a single set of parameters for all images is considered acceptable.

Figure 1: Highly textured façades lead to high error rate as points of low physical significance such as corners of bricks were marked as corners.

The Harris corner detector was also found to produce poor localisation at certain corner junctions. Figure 2 demonstrates how the Harris Corner detector performed poorly at identifying corners which were the junction of two curved edges. Future areas of research could focus on evaluating the performance feature detection algorithms in detecting the types of corners present in building façades as well as an investigation on the affect of scaling on the accuracy and error rate of corner responses. Accuracy could be assessed by comparing the location of the detected corner to their precise two dimensional position using knowledge of 3D coordinates or to ground truths established by human judgement (Mokhtarian and Mohanna, 2006).

Figure 2: High error rate along roof of the building demonstrates how the Harris corner has a poor response to corners which are the junction of two curved edges.

3.3. Foreground Extraction

Foreground extraction was used to segment areas of interest from background noise in images. This increased the accuracy of feature extraction algorithms such as edge detection by reducing the effect of background noise. It also reduced the computational time associated with further feature detection algorithms.

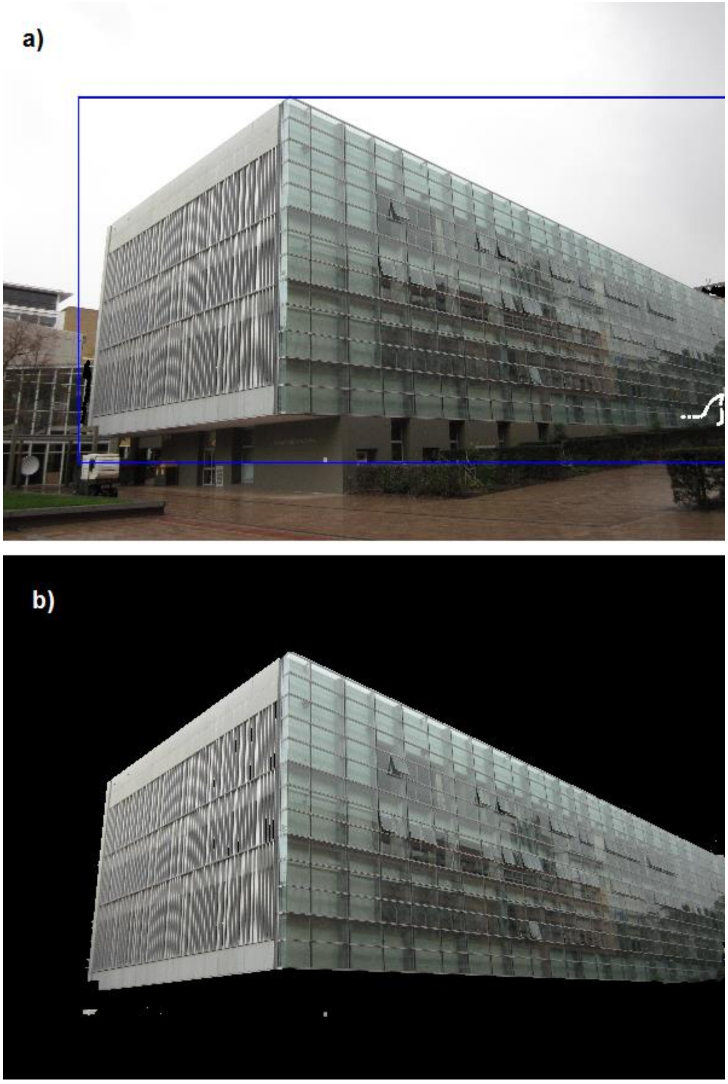
Manual foreground extraction involves tracing around the border of areas of interest. This approach is time consuming and inaccurate, however, accurate fully-automated foreground segmentation remains an unsolved problem with most algorithms in use being semi automated. Foreground extraction algorithms generally fall under two categories: contrast/edge-based segmentation or those based on colour/texture. Some popular edge-based foreground extraction methods include intelligent scissors (Mortensen and Barrett, 1995) and Magnetic Lasso. In these methods, segmentation follows definition of the boundary of the object of interest. For example, intelligent scissors require the user to define 'seed' points along the boundary of the object of interest. Dynamic programming and graph search are then used to wrap a 'live wire' around its boundary.

Image matting techniques attempt to segment images from the background based on the property that differences in surfaces textures in an image are related to differences in material properties in the physical world. Wang and Cohen (2008) presented a review of the different image matting approaches, evaluating them in terms of robustness, efficiency and accuracy. In most cases texture-based foreground segmentation or image matting requires the user to first define a trimap: identifying areas of sure foreground, sure background and unknown regions. Unknown pixels are then classified using properties of image smoothness and image statistics (Chuang et al., 2001; Sun et al., 2004; Grady et al., 2005).

Developed by Rother et al. (Rother et al., 2004), the authors used the 'Grab-Cut' algorithm to segment images of building façades from their backgrounds. Grab-Cut is a multistep algorithm that utilises characteristics of edges and image mattes to separate foreground objects from background images. The user is first required to identify areas of known background by placing a rectangle around the object of interest. Everything outside the rectangle is labelled as background and the remaining pixels are labelled as unknown. An initial segmentation is carried out based on Gaussian Mixture Model (Stauffer and Grimson, 1999). This is where background pixels are assumed to be represented by a mixture of Gaussian distributions. In each iteration, the image is then segmented based on the probability that the pixel belongs to the Gaussian distribution of the foreground/background. The user then has the option of touching up the segmentation by using brush strokes to constrain incorrectly labelled pixels to the foreground or background.

Images were resized before foreground extraction was performed by reducing the size of the image. This also reduced the computational time. For images of greater resolution or where large sets of images are needed to be processed this high computational time would be of concern. Figure 3 presents an example where Grab-Cut produced good results with minimal user interaction. After initial segmentation, Grab-Cut converged to produce accurate foreground segmentation. Additional user interaction was only required to constrain pixels to background near the bottom left corner of the building where a structure in the background overlapped the building of interest. Also, white brushstrokes were used to constrain foreground pixels on the bottom right hand corner of the building in an area of shading.

Figure 3: Foreground extraction. a) Original image with user input including initial segmentation and brush strokes near the bottom left and right hand corner of the building which were used to constrain pixels to background and foreground respectively, b) final segmentation.



In contrast to textured objects such as smoke and hair, due to the nature of structures and construction materials, the edges of buildings are well defined against the background (i.e. sky in Figure 3). In cases where the colour distribution of the background and foreground were similar (such as a grey or white building against an overcast sky) two situations arose. Either more iterations were required before the initial segmentation converged or the user was required to manually correct incorrectly labelled pixels. Less accurate results could be expected from pictures that are taken in highly developed areas where background and foreground pixels are of similar colour or in bushy areas where parts of the foreground are camouflaged by foliage.

Another interesting observation was that the algorithm did not perform well when different areas of the foreground were of high contrast in colour such as the area of shadow adjacent to a light area illuminated by direct sunlight. This is because when the intensity varies abruptly it does not conform to the Gaussian distribution (Kim et al., 2007). One drawback of Grab-Cut is that user error can be easily introduced to the algorithm through the initial segmentation. The rectangle which is used to define the area of known background should be as close a fit to the building edges as possible. If the area inside the rectangle contains too much background area it may take many iterations or further user interaction before correct segmentation is reached. Future works would benefit from changing the tool available to the user to define the initial segmentation. For example, area of known background could be defined using straight-line segments rather than a rectangle.

3.4. Edge Detection

Edge detection algorithms are commonly used to extract details of building boundaries as they less likely to be completely obscured by obstructions then features such as corners. Edge detection algorithms use changes in intensity to identify physical changes in depth, illumination and reflection. A review of different edge detection algorithms can be found in (Ziou and Tabbone, 1998) and (Davis, 1975). Most edge detection algorithms include the following processes: application of a smoothing filter to reduce the effects of intensity changes due to image noise, application of a gradient operator and a process that attempts to label the edge by suppressing false edges. There are studies that attempt to rank the relative performance of different edge detection algorithms (Heath et al., 1996) however their performance is related to image properties including edge types and image noise.

Edges of building geometries were extracted using the Canny edge detector (Canny, 1986), an algorithm which involves the following processes.

a. Noise reduction
b. Calculation of gradient magnitudes and directions
c. Non-maximum suppression
d. Hysteresis thresholding

Noise reduction is performed by convolving a discrete approximation of the Gaussian function known as the Gaussian filter with the original image. The intensity gradients are then calculated across the image using the Sobel operator (Sobel and Feldman, 1968). Non-maximum suppression is used to mark edges that are to be considered to be part of an edge. This is where the gradient of each pixel is compared with that of the two adjacent pixels. If the gradient of the pixel is greater than the gradient of the two pixels in the gradient direction, then it is marked as an edge.

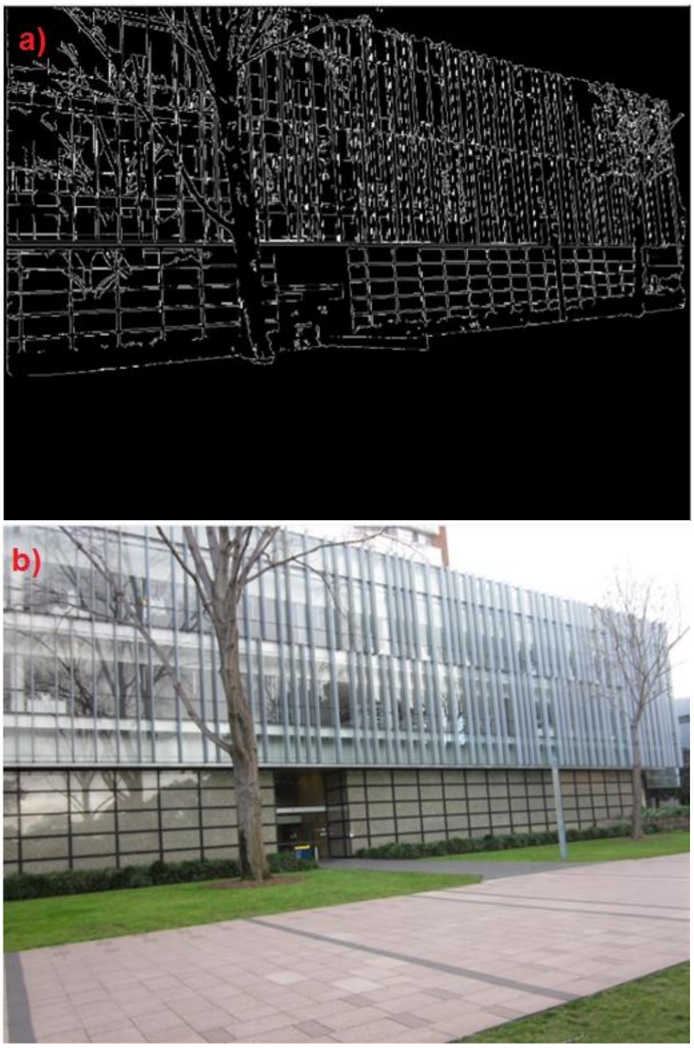
Edges are determined from the resulting binary image by hysteresis thresholding which is used to mark true edges while removing false edges which may be due to noise, changes in colour or changes in lighting. The intensity gradient of each pixel is compared to an upper and lower threshold. If the gradient is above the upper threshold it is marked as a definite edge and if it is below the minimum threshold it is marked as a non-edge. Any pixels with gradients between the upper and lower thresholds are only marked as an edge if they are connected to a pixel with a gradient above the upper threshold. The thresholds are most commonly decided on an ad hoc basis and are dependent on image characteristics such as lighting and level of noise.

In this research the high and low hysteresis thresholds were selected manually for each image. Deciding the threshold values was a trade off between a good localisation and high error rate. Too high a threshold missed some of key edges, and too low a threshold resulted in noise e.g. building textures being marked as edges. Optimal threshold will depend on characteristics of the input image. Sample images were taken from around a university campus where buildings were constructed during different time periods, possessing different architectural features and reflecting a need for different hysteresis thresholds.

Rather than manually select hysteresis threshold for each building, which is time consuming and introduces user error, future works aimed at the automated extraction of building geometries from images would benefit from the use of unsupervised threshold selecting algorithms. Unsupervised methods of threshold estimation include parametric methods, where selection of parameters is based on image statistics such as the standard deviation of background noise (Hancock and Kittler, 1991). Alternatively, the effect of building textures could be reduced by increasing the size of the Gaussian kernel, however, this may have the effect of reducing the accuracy of edge localization.

The most significant factor affecting the results of edge detection was the presence of trees and other obstructions obscuring parts of the building. On highly reflective surfaces the reflections of objects in the foreground were also detected as edges. This is seen in Figure 4 where the reflection of the tree in the foreground was detected by the canny edge detector.

Figure 4: Edge detection. a) Edges detected with Canny edge detection illustrating how the reflection of a tree is detected as edges by the Canny edge detector, b) original image.

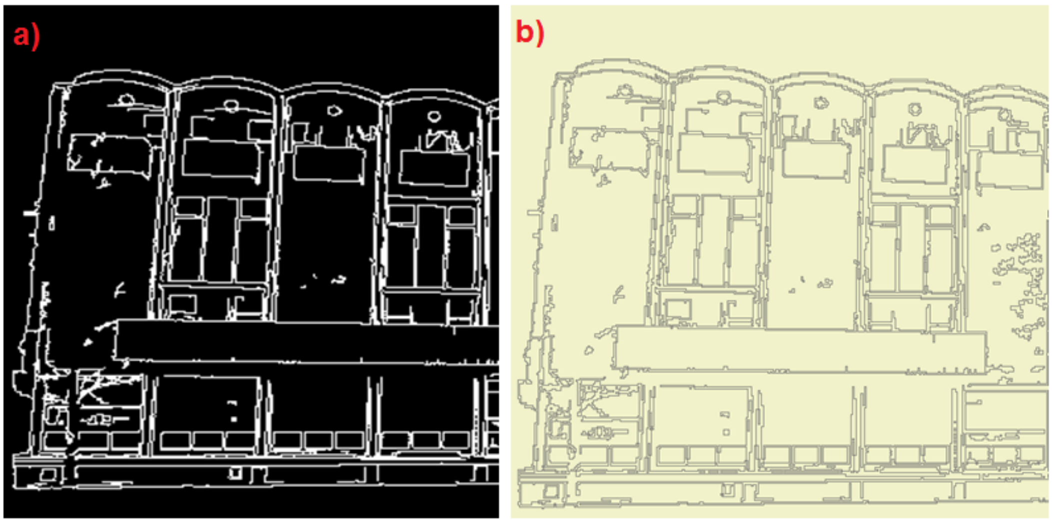


4. WINDOW DETECTION

The location and areas of windows relative to features of the building envelope has a significant impact on the energy use in the operation of a building through influence on insulation, lighting as well as passive heating and cooling (Persson et al., 2006; Ghisi and Tinker, 2005). With operational energy responsible for up to 90% of a buildings energy demand over its life (Ramesh et al., 2010) the ability to extract information about areas of windows is of high importance in the energy analysis of individual buildings as well as the buildings in a geographic area.

Laser scans often perform poorly in the detection of fine architectural features. This is due to limited data points or non-reflective surfaces such as windows and balconies. Works that attempt to identify building features such as windows generally focus on identifying a single feature type by exploiting properties of their regular repeated shape (Shaw and Barnes, 2006; Sirmacek et al., 2011) or elements that satisfy certain symmetry rules (Wenzel et al., 2008). A limitation of these methods is that further input is required in the presence of irregular shapes. Another approach involves segmenting the façade based on horizontal and vertical projections of façade edges occurring above a specified thresholds (Lee and Nevatia, 2004). These projections segment the block into a series of rectangles which are then classified as window or façade based on the size, colour and gradient content of each block. This approach uses the assumption that the strongest edges of a façade occur at windows.

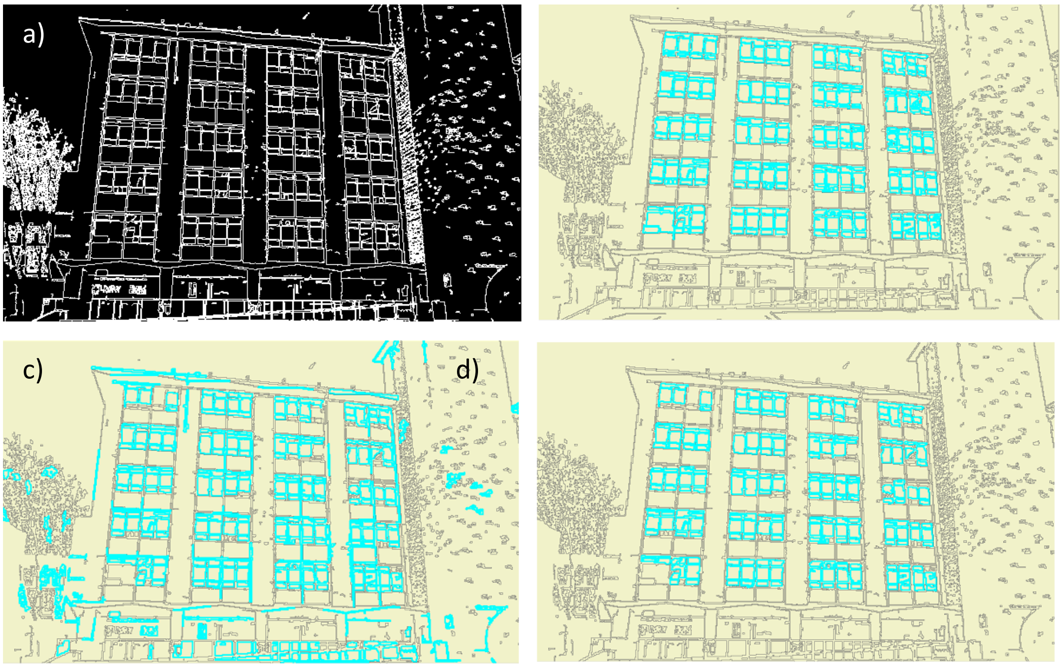
Figure 5: Vectorisation. a) Original window detection image, b) vectorised image demonstrating how polygons were created around pixels marked as edges.



Following vectorisation of edge images of building façades, the authors attempted to identify areas of windows by selected polygons based on properties of area and perimeter. Vectorisation was performed by creating polygons out of contiguous cells with the same value on the binary edge image. This method performed well in maintaining information at edge junctions, however, a concerning feature was that polygons were created around pixels marked as edges which resulted in polygons being created for each of the edges. This lead to higher processing times and visual distortions as demonstrated in Figure 5. Future works may benefit from investigating alternative approaches to vectorise edge images.

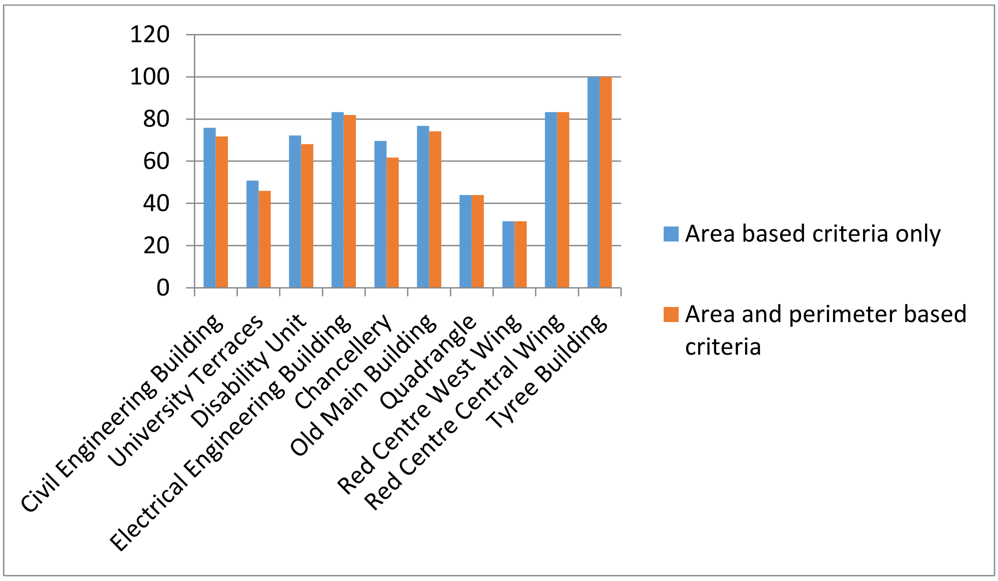
The authors attempted to identify areas of windows from the vector images by selecting polygons based on their area and perimeter. This involved creating queries to select polygons with areas that lay within a maximum and minimum value as demonstrated in Figure 6. These thresholds were selected for each window type in a façade using a trial and error approach which aimed to correctly identify the maximum number of windows while reduce the number of polygons which were incorrectly identified as windows. Results were refined by introducing a restriction on window perimeter to the original area query.

Figure 6: Window detection. a) edge image, b) polygons identified as true positives, c) polygons selected with area between 100 and 340 or 40 and 100 square units, d) polygons of these which are identified as true positives.



The aim of this research was to identify polygons which would correspond to areas of window panes in the actual building façade. For each image the following results were recorded: the number of polygons selected that correspond to areas of windows (true positives), polygons selected that did not correspond to areas of windows in the façade (false positives) and the number polygons that correspond to windows but were not selected (false negatives). Figure 7 displays the result for true positives when two different queries were used to identify windows, an area based criteria and a query that also imposes restrictions on maximum polygon perimeter.

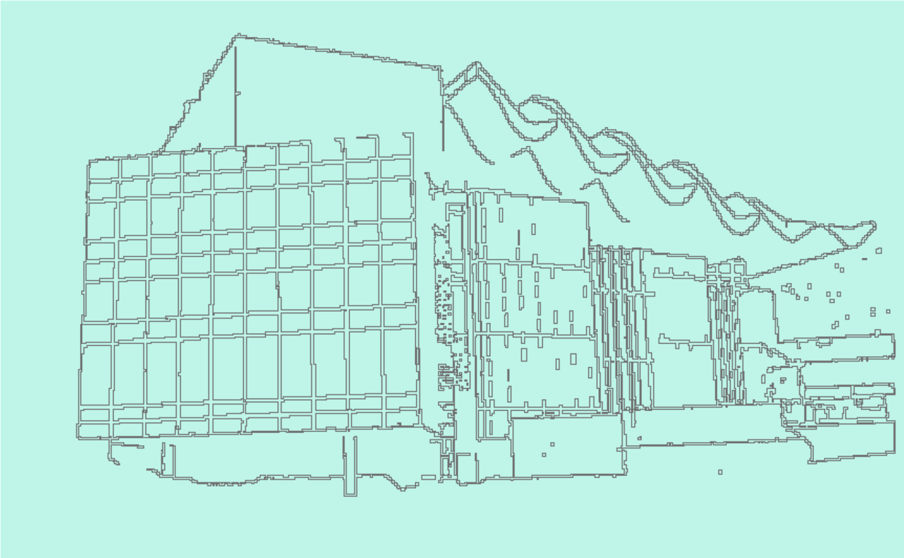
Figure 7: Polygons correctly identified as windows (true positives) as a percentage of total number of windows.



Ground truths were defined as enclosed polygons with areas greater than 10 square units which correspond to window panes in the edge image. Unenclosed polygons had areas too large to be included in statistically useful results while small polygons could be attributed to noise introduced to the image during edge extraction. Ground truths were defined in terms of the edge image rather than the original photograph due to discrepancies between the two images which are a result of edge extraction.

Good results were obtained for the Electrical Engineering building (Figure 6) and Tyree building façades (Figure 8) where windows were represented as polygons of a consistent size over the face. This occurred when windows on the façade were uniform in shape and size, when windows were not obstructed by objects in the foreground and when the photograph had been taken at an angle and position which minimised the affect of perspective distortions.

Figure 8: The regular shape of windows on the front face of the Tyree building lead to all windows being correctly identified.



A large numbers of false results were produced on highly textured façades such as that of the University Terraces (Figure 9) where polygons created within the façade texture were detected as windows. Another source of false positives was the polygons created around edge pixels. Polygons created around edge pixels and within façade textures often had much larger perimeters then window areas. Figure 9 demonstrates how introducing restrictions on maximum perimeter improved the results by reducing the number of false positives while only decreasing the number of true positives slightly.

Figure 9: Highly textured facades. a) Polygons defined as windows, b) polygons selected with areas between 150 and 750 square units and c) polygons selected when additional restriction is applied on maximum polygon area (shape length ≤ 250).

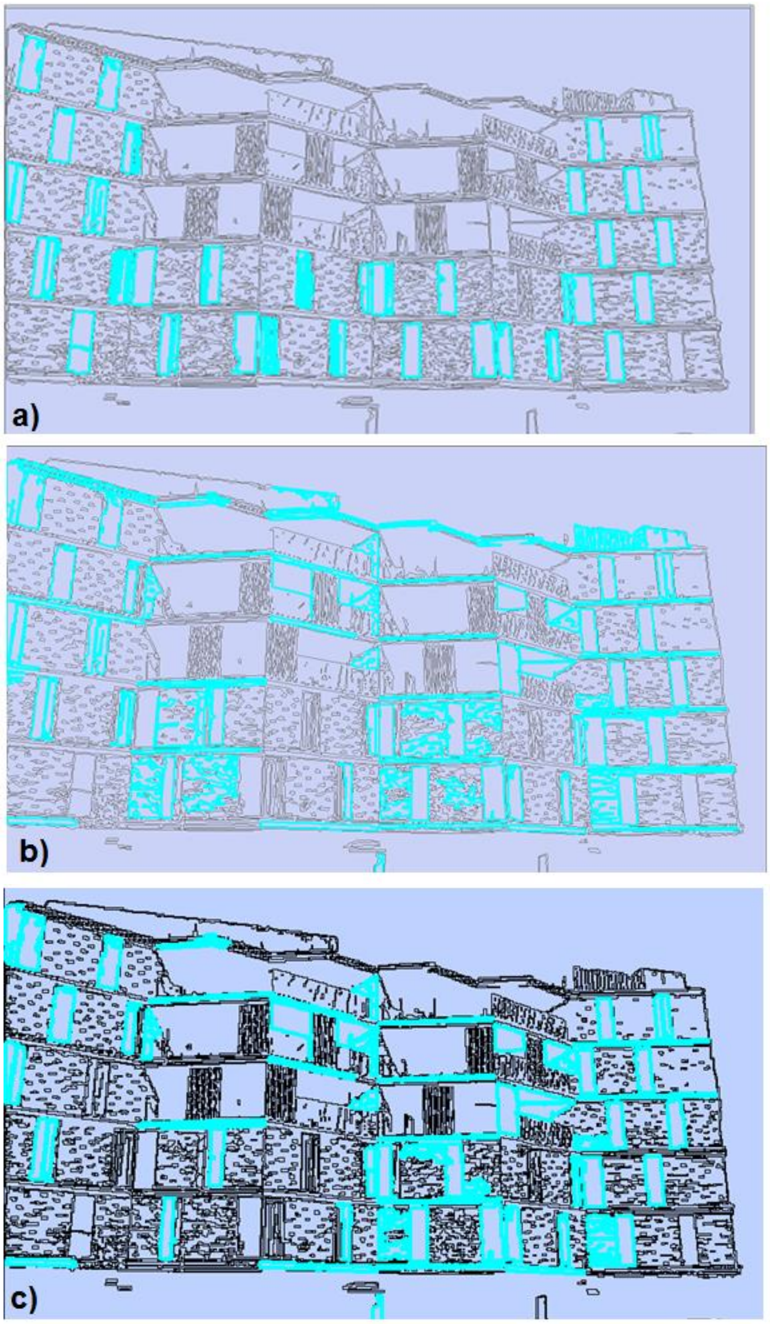
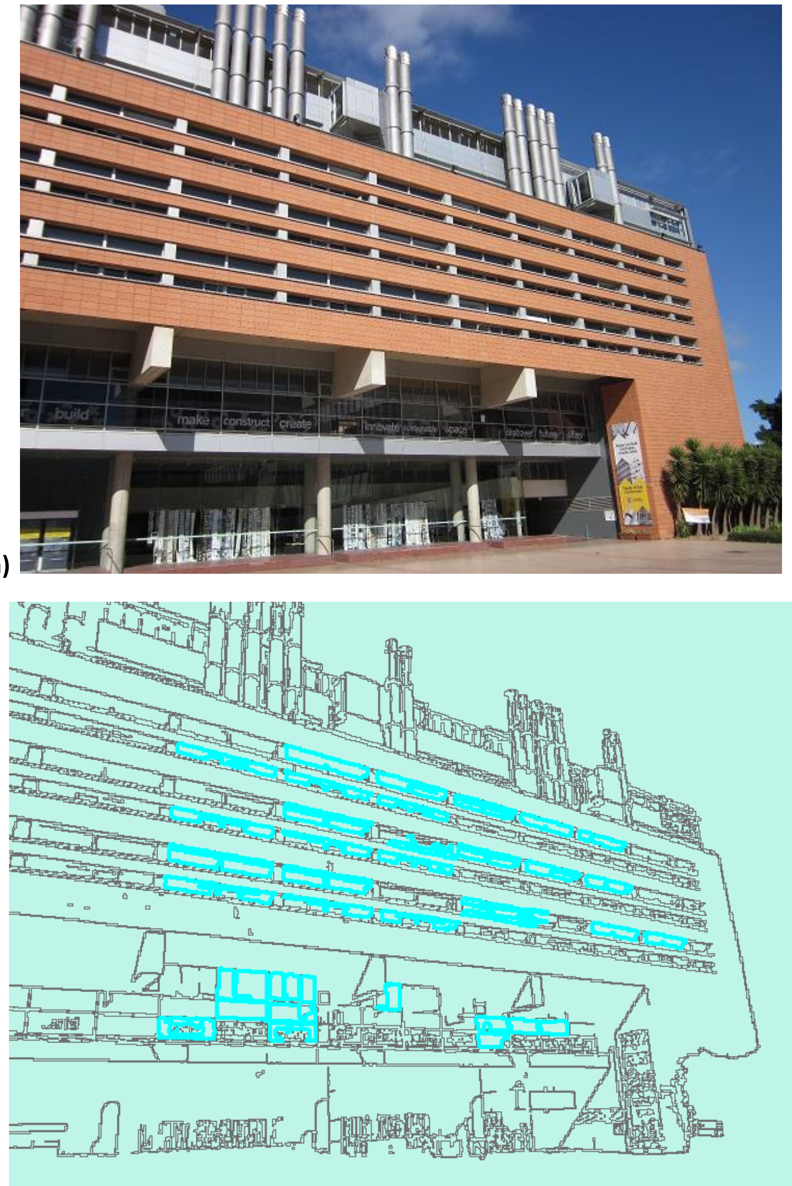


Figure 10: Red Centre West Wing. a) digital image, b) polygons selected with areas between 200 and 600 square units and perimeters less than 380 units.



a)

b)

Less effective window detection was observed when window polygons in an image were of varying sizes and shapes. This could be attributed to perspective distortions or distortions in the edge image due to lighting conditions such as for the image of the Quadrangle façade. The lowest number of true positives was observed on the façade of Red Centre West Wing (Figure 10) where window polygons varied from 13-1,200 square units due to lighting (shading around windows) and perspective distortions. The high variation in window area and perimeter made it difficult to select a threshold that maximised the number of true positives and minimised the number of false negatives.

A major factor that limited the accuracy of results was the lack of edge connectivity around window polygons. This resulted in window polygons which were not detected when they had areas and perimeters which lay outside the predicted area and perimeter ranges. Although this problem is a product of edge detection, results could be improved through methods which attempt to fill in holes in lines such as through the use of morphological filters.

Although area and perimeter thresholds would need to be selected individually for each façade, window identification could be simplified for geo-referenced images by exploiting properties of the regular repeating shape and size of windows in façades. Geometries for each window type could be obtained from direct measurement of one sample window (e.g. with a tape measure) or from architectural plans. This knowledge of actual area and perimeter could then be used to estimate thresholds to identify windows in the vector image referenced in real world coordinate systems.

It is recognised that in the proposed methodology, user error is not only introduced through threshold selection but through the subjective nature of identifying ground truths. More meaningful results could be achieved if window identification was performed on geo-referenced images and the total area of selected polygons was compared to known areas of windows present in the building façades.

5. DISCUSSION

5.1. Accuracy

Although a number of promising results were observed, an investigation into the accuracy of feature extraction would need to be performed before the results could be considered of practical use. Schmid et al. (2000) presented a review of the methods used to assess the performance of low level feature extraction algorithms. For the purpose of feature extraction for BIM, future works would benefit from a quantitative analysis of the accuracy of corner and edge detection. The accuracy of corner detection can be assessed by comparing the location of extracted corners with their known position in 3D coordinate systems. Thus a study into accuracy may need to be completed in conjunction with an investigation into the feasibility of geo-referencing the vertical images of façades. The effect of image scaling on the accuracy of corner accuracy would be a worthwhile area of investigation, particularly for the Harris corner detector which is known to perform poorly in the scaling transformation.

Due to their non discrete nature, edges can be hard to define, however, the accuracy of edge detectors could be evaluated using an approach similar to that suggested by Bhatla et al. (2012). In this method, lengths of edges extracted from the image are compared to actual lengths obtained from site conditions using distinct reference points (e.g. length of building edge, distance between artificial reference points) and accuracy is assed using percentage deviation. Alternatively, edge detection could be evaluated using known edge lengths from artificial images of building façades. However for this to be representative of actual conditions, artificial images would need to include features such as variations in illumination and variations in texture (Fram and Deutsch, 1975).

5.2. Window Identification

A number of false responses were obtained using an area-based criterion to identify windows from the edge image. This was due to polygons created within building textures and around edge pixels falling within the area range of windows in the façade. The number of false positives could be reduced in future works by imposing restrictions on the maximum number of vertices in addition to the area-based criteria. Due to their irregular shape, polygons created within building textures would be expected to have a large number of vertices. Alternatively, future works may wish to reduce the number of false positives using a centroid based criteria. For example by only selecting polygons who's centroid lies within its area. This could be expected to remove many false positives while retaining window polygons which are typically convex in shape.

Ideally, areas of windows could be identified using criteria which could be applied across all images. This could be achieved by identifying areas of windows based on properties of image texture. Image texture is a complex image to properly define, there is no standard definition however it can be described as 'the characteristic variation in intensity of a region of an image' (Davies, 2012, p.21). There are two main methods used to distinguish image textures: statistical based methods which attempt to characterise the distribution of grey values in an image while the structure based methods involve identifying areas composed of texture units within a certain displacement rule (Tuceryan and Jain, 1998). One approach to identify areas of windows would be to segment the image based on image texture and identify areas of windows as areas with common texture.

6. CONCLUDING REMARKS

As-built building information models are increasingly being looked to as an alternative to conventional paper documentation in construction processes. As well as facilitating communication between stake holders, they are used as a tool for improving construction safety and assessing the environmental performance of existing buildings. Subsequently, there is a growing interest in technologies which can be used to improve the efficiency of methods used to collect and process data for as-built building information models.

The use of image processing to extract information about building geometries from digital pictures is an attractive alternative to laser scanners and total station theodolites as data can be collected quickly and does not require use of expensive, specialist equipment. The objective of this research was to investigate performance of feature extraction algorithms in obtaining information about edges and corners of buildings as well as areas of windows from digital images of façades for the purpose of building information modeling.

A number of promising results were produced, however, further work is required before feature extraction can be considered as a viable alternative for collecting information for as-built BIMs. Future investigations should focus on the use of automated methods to determine algorithm thresholds and parameters to account for differences in the architectural features of building facades as well as environmental factors such as shading and lighting.

7. REFERENCES

Becerik-Gerber, B., F. Jazizadeh, N. Li and G. Calis (2011). Application areas and data requirements for BIM-enabled facilities management, *Journal of Construction Engineering and Management*, 138(3): 431–442.

Bhatla, A., S.Y. Choe, O. Fierro and F. Leite (2012). Evaluation of accuracy of as-built 3D modeling from photos taken by handheld digital cameras, *Automation in Construction*, 28: 116–127.

Böhm, J. (2004). Multi-image fusion for occlusion-free façade texturing, *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 35(5): 867–872.

Boukamp, F. and B. Akinci (2007). Automated processing of construction specifications to support inspection and quality control, *Automation in Construction*, 17(1), at http://www.sciencedirect.com/science/article/pii/S092658050700043X.

Campbell, R. J. & Flynn, P. J. (2001). A survey of free-form object representation and recognition techniques. *Computer Vision and Image Understanding,* 81: 166-210.

Canny, J. (1986). A computational approach to edge detection, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6): 679–698.

Chuang, Y.-Y., B. Curless, D.H. Salesin and R. Szeliski (2001). "A bayesian approach to digital matting", *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, pp. II–264–II–271 vol. 2, IEEE.

Davies, E.R. (2012). *Computer and machine vision: theory, algorithms, practicalities*, Academic Press.

Davis, L.S. (1975). A survey of edge detection techniques, *Computer Graphics and Image Processing*, 4(3), at http://www.sciencedirect.com/science/article/pii/0146664X7590012X.

Delis, E.A. and A. Delis (1995). Automatic fire-code checking using expert-system technology, *Journal of computing in civil engineering*, 9(2): 141–156.

Deriche, R. and G. Giraudon (1993). A computational approach for corner and vertex detection, *International Journal of Computer Vision*, 10(2): 101–124.

Ernstrom, J.W. (2006). *The contractors' guide to BIM*, Associated General Contractors of America.

Fang, M., G. Yue and Q. Yu (2009). "The study on an application of otsu method in canny operator", *International Symposium on Information Processing (ISIP)*, pp. 109–112, Citeseer.

Fram, J.R. and E.S. Deutsch (1975). On the quantitative evaluation of edge detection schemes and their comparison with human performance, *Computers, IEEE Transactions on*, 100(6): 616–628.

Frueh, C., S. Jain and A. Zakhor (2005). Data processing algorithms for generating textured 3D building facade meshes from laser scans and camera images, *International Journal of Computer Vision*, 61(2): 159–184.

Ghisi, E. and J.A. Tinker (2005). An ideal window area concept for energy efficient integration of daylight and artificial light in buildings, *Building and Environment*, 40(1): 51–61.

Grady, L., T. Schiwietz, S. Aharon and R. Westermann (2005). "Random walks for interactive alpha-matting", *Proceedings of VIIP*, pp. 423–429.

Gregson, P.H. (1993). Using angular dispersion of gradient direction for detecting edge ribbons, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 15(7): 682–696.

Haala, N., Brenner, C. & Anders, K.-H. (1998). 3D urban GIS from laser altimeter and 2D map data. *International Archives of Photogrammetry and Remote Sensing,* 32**:** 339-346.

Hancock, E.R. and J. Kittler (1991). "Adaptive estimation of hysteresis thresholds", *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, pp. 196–201, IEEE.

Harris, C. and M. Stephens (1988). "A combined corner and edge detector", *Alvey vision conference*, p. 50, Manchester, UK.

Heath, M., S. Sarkar, T. Sanocki and K. Bowyer (1996). "Comparison of edge detectors: a methodology and initial study", *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR’96, 1996 IEEE Computer Society Conference on*, pp. 143–148, IEEE.

Khallil, M. and A. Aggoun (2006). "Edge detection using adaptive local histogram analysis", *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, pp. II–II, IEEE.

Kim, H., R. Sakamoto, I. Kitahara, T. Toriyama and K. Kogure (2007). "Robust silhouette extraction technique using background subtraction", *10th Meeting on Image Recognition and Understand, MIRU*.

Kitchen, L. and A. Rosenfeld (1980). *Gray-level corner detection*, , DTIC Document.

Lee, S.C. and R. Nevatia (2004). "Extraction and integration of window in a 3D building model from ground view images", *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, pp. II–113–II–120 Vol. 2, IEEE.

Medina-Carnicer, R., F.J. Madrid-Cuevas, A. Carmona-Poyato and R. Muñoz-Salinas (2009). On candidates selection for hysteresis thresholds in edge detection, *Pattern Recognition*, 42(7): 1284–1296.

Melzner, J., S. Zhang, J. Teizer and H.-J. Bargstädt (2013). A case study on automated safety compliance checking to assist fall protection design and planning in building information models, *Construction Management and Economics*, 31(6): 661–674.

Mokhtarian, F. and F. Mohanna (2006). Performance evaluation of corner detectors using consistency and accuracy measures, *Computer Vision and Image Understanding*, 102(1), at http://www.sciencedirect.com/science/article/pii/S1077314205001852.

Moravec, H.P. (1980). *Obstacle avoidance and navigation in the real world by a seeing robot rover*, , DTIC Document.

Mortensen, E.N. and W.A. Barrett (1995). "Intelligent scissors for image composition", *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pp. 191–198, ACM.

Persson, M.-L., A. Roos and M. Wall (2006). Influence of window size on the energy balance of low energy houses, *Energy and buildings*, 38(3): 181–188.

Ramesh, T., R. Prakash and K.K. Shukla (2010). Life cycle energy analysis of buildings: An overview, *Energy and buildings*, 42(10): 1592–1600.

Rother, C., V. Kolmogorov and A. Blake (2004). "Grabcut: Interactive foreground extraction using iterated graph cuts", *ACM Transactions on Graphics (TOG)*, pp.

309–314, ACM.

Schmid, C., R. Mohr and C. Bauckhage (2000). Evaluation of interest point detectors, *International Journal of Computer Vision*, 37(2): 151–172.

Shaw, D. and N. Barnes (2006). "Perspective rectangle detection", *Proceedings of the Workshop of the Application of Computer Vision, in conjunction with ECCV 2006*, pp. 119–127, Citeseer.

Sirmacek, B., L. Hoegner and U. Stilla (2011). "Detection of windows and doors from thermal images by grouping geometrical features", *Urban Remote Sensing Event (JURSE), 2011 Joint*, pp. 133–136, IEEE.

Sobel, I. and G. Feldman (1968). A 3x3 isotropic gradient operator for image processing.

Stauffer, C. and W.E.L. Grimson (1999). "Adaptive background mixture models for real-time tracking", *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, IEEE.

Sun, J., J. Jia, C.-K. Tang and H.-Y. Shum (2004). "Poisson matting", *ACM Transactions on Graphics (ToG)*, pp. 315–321, ACM.

Tang, P., D. Huber, B. Akinci, R. Lipman and A. Lytle (2010). Automatic reconstruction of as-built building information models from laser-scanned point clouds: A review of related techniques, *Automation in Construction*, 19(7): 829–843.

Toyama, K., J. Krumm, B. Brumitt and B. Meyers (1999). "Wallflower: Principles and practice of background maintenance", *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, pp. 255–261, IEEE.

Tuceryan, M. and A.K. Jain (1998). Texture analysis, *The handbook of pattern recognition and computer vision*, 2: 207–248.

Wang, J. and M.F. Cohen (2008). *Image and video matting: a survey*, Now Publishers Inc.

Wenzel, S., M. Drauschke and W. Förstner (2008). Detection of repeated structures in facade images, *Pattern Recognition and Image Analysis*, 18(3): 406–411.

Zheng, Z., H. Wang and E. Khwang Teoh (1999). Analysis of gray level corner detection1, *Pattern Recognition Letters*, 20(2), at http://www.sciencedirect.com/science/article/pii/S0167865598001342.

Ziou, D. and S. Tabbone (1998). Edge detection techniques-an overview, *Pattern Recognition And Image Analysis C/C Of Raspoznavaniye Obrazov I Analiz Izobrazhenii*, 8: 537–559.