The value of integrating Scan-to-BIM and Scan-vs-BIM techniques for construction monitoring using laser scanning and BIM: The case of cylindrical MEP components

Frédéric Bosché[a,*], Mahmoud Ahmed[b], Yelda Turkan[c], Carl T. Haas[b], Ralph Haas[b]

[a]*School of the Built Environment, Heriot-Watt University, Edinburgh, Scotland*

[b]*Department of Civil Engineering, University of Waterloo, Ontario, N2L 3G1, Canada*

[c]*Civil, Construction and Environmental Engineering, Iowa State University, Ames, IA 50010, USA*

Abstract

There is a growing need for tools automating the processing of as-built 3D laser scanned data, and more particularly the comparison of this as-built data with planned works. This paper particularly considers the case of tracking MEP components with circular cross-sections, which essentially include pipes, and some conduits and ducts. Discrepancies between the as-built and as-planned status of pipes, conduit and ductwork result from changes that occur in the field and that are either unnoticed (human error) or not reflected in the 3D model. Previous research has shown that the Hough transform, with judiciously applied domain constraints, is a practical and cost-effective approach to find, recognize and reconstruct cylindrical MEP works within point clouds automatically. Previous research has also shown that "Scan-vs-BIM" systems that are based on the geometric alignment and comparison of as-built laser scans with as-designed BIM models can effectively recognize and identify MEP components as long as they are constructed near their as-planned locations. The research presented in this paper combines the two techniques in a unified approach for more robust automated comparison of as-built and as-planned cylindrical MEP works, thereby providing the basis for automated earned value tracking, automated percent-built-as-planned measures, and assistance for the delivery of as-built BIM models from as-designed ones. The proposed approach and its improved performance are validated using data acquired from an actual construction site. The results are very encouraging and demonstrate the added value of the proposed integrated approach over the rather simpler Scan-vs-BIM system. The two main areas of improved performance are: (1) the enabled recognition and identification of objects that are not built at their as-planned locations; and (2) the consideration for pipe completeness in the pipe recognition and identification metric.

Keywords: MEP; 3D laser scanning; BIM; Scan-vs-BIM; Scan-to-BIM; Hough transform; progress tracking, percent built as planned.

* Corresponding author. Tel.: +44 131 451 4659; *E-mail addresses:* f.n.bosche@hw.ac.uk (F. Bosche) mfouad@uwaterloo.ca (M. Ahmed), yturkan@iastate.edu (Y. Turkan), chaas@uwaterloo.ca (C.T. Haas), haas@engmail.uwaterloo.ca (R. Haas).

1 Introduction

Traditional progress tracking practice depends on visual inspections, and daily or weekly reports created based on those inspections. The inspectors' duty is to ensure that work meets contract specifications and schedule. They use checklists during inspections and logs to report deficiencies that are discussed at follow-up weekly meetings [1]. This traditional practice relies heavily on the inspectors' personal judgment, observational skills, and experience which come with a high probability of incomplete and inaccurate reports. In the early 2000's, the Architectural-Engineering-Construction/Facility Management (AEC/FM) industry realized the urgent need for fast and accurate project progress tracking.

In response to this need, researchers have studied several emerging technologies for automating project inspection. These include Radio Frequency Identification (RFID) [2][3][4][5][6][7], Ultra-Wide Band (UWB) [8][9][10][11], Global Navigation Satellite System (GNSS) [6][12], 2D imaging [13][14][15][16][17][18][19], Photogrammetry [20][21][22][23][24][25][29], and three-dimensional (3D) Terrestrial Laser Scanning (TLS) [22][26-54]. All these approaches hold much promise for automated progress tracking, however they have so far only focused on a few areas of application: progress in the supply chain (prefabrication and laydown yards), workers' productivity (through location and action tracking), and tracking structural work progress and quality. One of the important areas where tracking could provide significant value is the tracking of Mechanical, Electrical and Plumbing (MEP) components, which includes piping installation. The benefits of efficient tracking of MEP components' installation include:

1) Early identification of deviations between the as-built and as-design situations, so that required remedial actions can be taken before high rework costs are experienced;

2) Faster acceptance of work by the main contractor, so that sub-contractors can be paid on time and even earlier than is common practice; and
3) Assistance through automation of some of the steps involved in updating Building Information Modeling (BIM) models to reflect as-built works that deviate or add to original BIM models, but will not require rework. Indeed, in many cases liquidated damages and an updated as-built BIM model may be preferable to rework.

However, tracking of MEP works is made difficult by significant discrepancies between the as-built and as-planned status of MEP components that result from changes that occur in the field that are either unnoticed (human error) or not reflected in the design documents. These unreported discrepancies also challenge the delivery of reliable as-built design documents (e.g. as-built BIM model) to clients.

Among the technologies discussed earlier, 3D TLS has been considered by many as the best available technology to capture 3D information on a project with high accuracy and speed. It holds much promise in a variety of applications in the AEC/FM industry [26][27][28][29][30]. For example, it has already been proven to be valuable for construction managers to help them track progress, control quality, monitor health, as well as create as-built 3D models of facilities [31-54]. The best demonstration of this value has been the exponential growth of the laser scanning hardware and software market in the last decade. Much of this growth is now focusing on the interface between laser scanned data and BIM models. Nonetheless, the recognition (and identification) of objects in 3D TLS data remains an open challenge with marketed software offering only semi-automated, and often limited solutions. This is the case of MEP components, including pipes. Robust automated recognition and tracking of cylindrical MEP components would enable:

1. Improved identifications of discrepancies between as-planned and as-built conditions of MEP components, so that corrective actions can be taken in a timely manner. This is particularly important for mechanical contractors, since an increasing number of them are using BIM models for fabricating pipes and ductwork.
2. Having accurate as-built conditions of MEP components, so that mechanical remodelings can be planned confidently from the BIM model, and thus help prevent material wastes and rework, hereby saving cost and time. Furthermore, there is a growing interest and demand from industry for implementing BIM models for Facilities Management. Having accurate as-built conditions of MEP components included in BIM models would allow facility managers to integrate their building operation and maintenance schedules with BIM models, which would allow them to locate and maintain these components efficiently.

Recent research in the recognition of MEP works in 3D TLS data has shown that the Hough transform, with judiciously applied domain constraints, is a practical approach to automatically find, recognize and reconstruct cylindrical objects (e.g. pipes) from point clouds [48][49]. However, this approach is not sufficient on its own to identify objects to support reliable progress tracking and quality control. In parallel, previous research has also shown that “Scan-vs-BIM” systems, that are based on the geometric alignment and comparison of as-built laser scans with as-designed BIM models, can effectively recognize and identify in point clouds 3D objects contained in the BIM models [31][32][33] – as long as they are constructed near their as-planned locations. The research reported here combines these two approaches in a single framework to better meet the need for automated comparison of built and planned cylindrical MEP components, hereby providing the basis for automated earned value tracking, automated discrepancy identification and calculation of “*percent built as-planned*”, and assistance for the generation as-built BIM models.

This paper is organized as follows. Section 2 first reviews significant research and developments in the area of object recognition in 3D point clouds. Our novel approach for the recognition and identification of cylindrical objects in 3D point clouds is described in Section 3. Experimental results are reported in Section 4 and the performance of the new approach discussed in Section 5.

2 Background

2.1 3D point cloud data processing

Using 3D point clouds produced by laser scanners for generating as-built information is becoming a standard practice in construction, rehabilitation and facilities maintenance in areas ranging from process plants to historical preservation. Building on basic research in robotics and machine vision, research on automated as-built generation goes back over twenty years (e.g. [13]).

Acquisition of 3D information with laser-scanning (but also structured lighting and photogrammetry) has led to significant research on developing processes and algorithms for processing the 3D point cloud data, with focus on different applications. These include: as-built modelling [29][34][36] [40][41][42][43][44] [48][49][50][51], quality assessment of existing infrastructure and construction sites [25][35][37][45][54], progress tracking [20][21][22][23][24] [31][32][33][46][47][52][53], and structural health monitoring [38] [39]. Some of the knowledge thereby created has influenced or been adopted by practitioners. Yet, in the commercial sphere, the level of automation of current software solutions for processing TLS data, and in particular for recognizing objects in TLS data, remains limited.

With the advent of 3D BIM, many of the newer approaches actively use the (3D) information contained in BIM models to develop *supervised* object detection and recognition algorithms that more effectively process the point cloud data [20][21][31][32][27][33][35][46][47] [52][53][54]. Reliance of these approaches on prior BIM information certainly imposes limitations; but BIM is very rapidly being adopted across the industry for building design, construction and asset management, so that these limitations will diminish over time.

Focusing specifically on cylindrical MEP components, despite some significant effort in the processing of point clouds generated by TLS [48][49][50] or low-cost photogrammetry [23][24], progress remains limited. In particular, the automatic detection of occlusions of pipes (so that a pipe is not recognized as two different ones) remains an issue that needs to be investigated. Additionally, the automatic recognition of elbows and T-connections between pipe segments (so that pipes are recognized as a continuous pipe spools or networks as opposed to a set of disconnected pipe segments) needs further investigation. Effective detection of occlusions and connecting components would significantly improve the speed of generating accurate pipe network models.

Before getting into more details with specific techniques, it is worth pointing that the terms “detection”, “recognition” and “identification” are commonly used, but their use is not always consistent across the literature. In this manuscript, we use them as follows:

- *Detection*: an object is present. More specifically here, this means that some specific features are found in the data (e.g. circular cross-sections).
- *Recognition*: the type of object can be discerned. More specifically here, this means that the analysis of the features enables discerning objects of a specific type (e.g. pipes with circular cross-sections).

- *Identification*: a specific object can be discerned. More specifically here, this means that each recognized object can be matched to a specific object in a known list (e.g. a recognized pipe is discerned as being a certain pipe present in the project BIM model).

Surface feature detection, and in particular smooth curved surface detection, are topics of fundamental importance to 3D point cloud processing and have been widely studied. For detecting specified simple parametric surfaces, such as planes, cylinders, spheres and tori in point clouds, transform approaches have been considered, in particular the Hough Transform [55][56][57] that is used here (See Section 2.2 for details). Other types of transforms have been investigated for object shape detection, such as the Radon transform. For example, van Ginkel et al. [58] investigated the generalised Radon transform to detect curves. However, the Radon transform has several drawbacks that make it unsuitable for the investigated point clouds. Its brute-force approach demands extensive computational resources; and its restriction to line drawings or sketch-like formats mandate an additional edge detection step. Van Ginkel et al. [59] studied the Hough transform, the Radon transform, and the mathematical relationship between them.

Alternatively, curved surfaces can also be searched for directly in noisy point clouds, without employing any transform. Such approaches have been widely studied and typically consist in first capturing local surface curvature at each point using neighboring points, and then segmenting the point cloud using some region growing and clustering methods [60][61][62][63][64][65][66][67][68]. For example, Besl and Jain [60] proposed an approach that estimates local curvature using the mean and Gaussian curvature and then applies a region growing algorithm employing the fitting of quadratic surfaces. Methods proposed by Hoppe et al. [61] and Shaffer et al. [62] estimate local surface properties by analyzing the eigenvalues and eigenvectors of the covariance matrix of point neighbourhood clusters. Pauly et al. [65] presented a multi-scale technique that works across multiple resolutions of the point cloud to extract the line features of 3D object surfaces. Rabbani et al. [66] presented a curved surface region growing method based on surface normal and local smoothness constraints. Klasing et al. [68] presented a review and experimental comparison of surface normal estimation methods. The challenges in surface growing (curved or planar) are over-segmentation (which is typically addressed through a post-processing step) and noise handling. The latter is a key issue which has been addressed by many researchers including Carr et al. [69] in a fundamental sense and Xiong et al. [70] as applied to building construction. Future research is desirable to compare the use of the Hough transform as described in this paper with curvature based surface growing approaches. However, this is outside the scope of the research reported here.

In the following two sections, we focus on the Hough transform for the detection of simple parametric surfaces, in particular cylindrical surfaces. Then, the employed Scan-vs-BIM technique for object recognition is reviewed.

2.2 Hough Transform

The Hough transform is a technique that can be utilized to detect parametric features within noisy data. It is usually carried out in three steps. The first step is concerned with creating and quantizing a parameter space, which is followed by the application of a voting rule in that parameter space [55][56]. The shape parameters within the accumulated array of votes are extracted during the final step. The technique was first introduced to detect straight lines using a parametric representation of the line in an image. In this case, the Hough transform requires two parameters: the slope and intercept [55], or the length and orientation of the normal vector to the line from the image origin [56]. Modified versions of the technique were developed by Duda and Hart [56] for extracting 2D curved shapes and by Cheng and Liu [57] for extracting ellipses. A comprehensive review of basic and probabilistic Hough based methods can be found in [71].

In construction engineering, Haas [13] implemented a 2D Hough transform for underground pipe detection. Vosselman et al. [51] investigated using a 3D Hough transform to extract planar surfaces from point-clouds. Newman et al. [72] proposed a method that combines the Hough transform and a regression procedure to recognize 3D shapes such as cylinders, spheres and cones. Rabbani et al. [44] have investigated a 5D Hough transform approach to extract cylindrical objects from point clouds. While that work was seminal research, its application was severely limited by the computational complexity resulting from the dimensionality of the Hough space. In general, high-dimensional Hough spaces are not practical. Working in Hough-space with more than two dimensions requires simplifications through judicious use of domain constraints, as described by Rabbani et al. themselves.

To address the memory and computational complexity constraints of the Hough transform, Borrmann et al. [73] proposed the Hough space accumulator structure, while Pietrowcew [74] presented a Fuzzy Hough methodology that adjusts the votes in the parameter space to extract special shapes.

Ahmed et al. [48][49] demonstrate the application of judicious use of domain constraints to efficiently detect circular-cross-sections in orthogonal directions (XYZ) of 3D TLS data, and consequently recognize objects with cylindrical shapes. In their approach, it is assumed that most cylindrical MEP components are built in orthogonal directions along the main axes of a facility. Circular cross-sections should then be identifiable in 3D point cloud data slices taken along those three directions. The recognition of cylindrical pipes could then be inferred from the set of circular cross-sections detected in slices along each of the directions. In summary, the technique implements the following steps:

1) Resample the original point-cloud to a number of thin slices. Slices are defined at a pre-determined interval along the X, Y and Z directions (e.g. 10cm);
2) For each slice, apply the Hough transform to find circles of expected diameters;
3) Connect centers of collinear detected circles (using rules described in Ahmed et al. [48][49]), then fit straight lines through the circles' centers,
4) Filter out the systematic errors due to slicing tilt,
5) Reconstruct the 3D pipes using the computed centerlines and their respective radii,

Applications of the Hough transform to laser scanned data have focused on *detection* of simple geometric features (e.g. straight lines, circular sections) and subsequent *recognition* of objects having those features; but these steps alone do not enable the *identification* of those objects – which is necessary for robust progress tracking. For example, the Hough transform can be used to detect all pipes with a pre-defined radius within a scanned point cloud, but it is just a first step in their identification, i.e. the mapping between the detected pipes and those defined in the designed 3D BIM model of the facility. Further steps are required for recognition and identification, including: (1) registration of sets of detected cylindrical objects and sets of cylindrical objects from the BIM model, (2) application of reasoning based on cylindrical object characteristics such as diameter, direction and proximity, (3) application of reasoning based on object connectivity, and (4) recognition and identification decision making based on these preceding steps.

2.3 Scan-vs-BIM Method

In the case that an as-designed BIM model of the works to be tracked is available, the prior information contained in the model can be leveraged to not only detect and recognize the objects contained in the model, but also identify them [31][32][33]. Bosché and Haas [31][32] proposed such an approach and refer to it as "*Scan-vs-BIM*" [53]. In the Scan-vs-BIM approach, 3D laser scanned point clouds are first aligned in the coordinate system of the 3D model. This can be done using site benchmarks or using automated or semi-automated registration techniques [75][76]. Once the registration is completed for all available scans, objects contained in the as-designed BIM model are recognized and identified in the combined point cloud using the following four-step process:

1 – *Matching/Recognized Point Clouds*: For each scan, each point is matched with a 3D model object. Matching is done by projecting the point orthogonally on the surfaces of all N_{Obj} objects of the 3D BIM model. Then, the object with (1) the closest surface to the point, but with distance not larger than a threshold δ_{max} (we use δ_{max}=50mm), and (2) a surface normal vector not further than α_{max} (we use α_{max}=45°) from that at the as-built TLS point is considered matching object. This process effectively segments each initial scan into N_{Obj}+1 point clouds; one per object that includes all the points matched to that object and another one containing all the points not matched to any model object. We call the latter the "*NonModel*" point cloud.

2 - *Occluding* Point Clouds (i.e. point clouds acquired from objects that do not seem to correspond to any object in the BIM model but that are occluding objects that are contained in the BIM model): For each as-built scan, the *NonModel* point cloud is further processed to identify the *NonModel* points that lay between the scanner and 3D model objects. The result of this process is not just an overall *Occlusion point cloud*, but also its segmentation into N_{Obj} point clouds; one per object that includes all the points occluding that object.

3 - *As-planned* Point Clouds: For each scan, a corresponding *virtual* as-planned scan is calculated. This is done using the 3D model and the same scanner's location and scan resolution as those of the actual (as-built) scan obtained from the registration process. Each as-planned point is calculated by

projecting a ray from the scanner onto the 3D model. The result of this process is not just an as-planned scan, but also its segmentation into N_{Obj} point clouds; one per object that includes all the points matched to that object. Note that we do not retain any *NonModel* as-planned point cloud.

<u>4 - Object Recognition</u>: The results of the first three steps are finally aggregated. Each model object then has:

- A matched/recognized surface area, $S_{recognized}$ (derived from the points contained in the matching Point Cloud).
- An occlusion surface area, $S_{occluded}$.
- An as-planned surface area, $S_{planned}$.

These surface areas allow the calculation of two metrics used for inferring the recognition of the object:

$$\%_{\mathrm{recognized}} = \frac{\mathrm{S_{recognized}}}{\mathrm{S_{recognizable}}} = \frac{\mathrm{S_{recognized}}}{\mathrm{S_{planned}\text{-}S_{occluded}}}$$

$$\%_{\mathrm{confidence}} = \frac{\mathrm{S^{w}_{recognized}}}{\mathrm{S_{recognizable}}} = \frac{\mathrm{S^{w}_{recognized}}}{\mathrm{S_{planned}\text{-}S_{occluded}}}$$

where $S^{w}_{recognized} = \sum_{i=1}^{n}\left(\left(1 - \left|\frac{\delta_i}{\delta_{max}}\right|\right) S_i\right)$

$\%_{\mathrm{recognized}}$ estimates the level of recognition by calculating the percentage of surface expected to be recognized that is actually recognized. $S^{w}_{recognized}$ is a weighted recognized surface where the contribution of each point to the recognized surface (i.e. the surface it covers, S_i) is weighted based on the quality of its matching (i.e. the distance δ_i from the as-built point to the matching surface).

$\%_{\mathrm{confidence}}$ thus extends $\%_{\mathrm{recognized}}$ by taking account for the deviation between the as-built and designed positioned of objects. $\%_{\mathrm{confidence}}$ can be used as a measure of the level of confidence in the recognition of each object, or the level to which the object can be considered *built as planned*. We refer the reader to [52][53] for details.

It has been shown through experiments with real-life data that the Scan-vs-BIM approach performs extremely well for structural works tracking. Furthermore, this approach directly enables the identification of objects. However, the features used by the approach (surface orientation and point proximity) work only for objects with minor geometrical discrepancy between the as-built and as-planned states. For example, any object built at a location further away than δ_{max} (50mm) cannot be recognized and identified; in fact, it was shown in [53] that the performance of this approach can drop significantly in the case of MEP works.

2.4 Contribution

The review of the Hough transform and Scan-vs-BIM techniques highlights a radical complementarity in terms of performance. While the Hough transform can robustly detect circular cross-sections in the presence of significant amounts of occlusions, and Mahmoud et al. [48][49] have shown that those detections can support the recognition of cylindrical objects, their method cannot be used on its own to infer their identification. Furthermore, the method of Mahmoud et al. can only recognize objects with cylindrical shape, i.e. circular cross-sections along a straight centerline; it cannot recognize objects with non-collinear circular cross-sections (e.g. curved pipes, elbows). On the other hand, the Scan-vs-BIM technique of [31][32][53] enables the recognition and identification of simple and complex objects, but its recognition metrics are not robust to recognize objects that are significantly displaced from their designed location. It also cannot recognize objects that are not contained in the BIM model.

Bosché et al. [53] have suggested that, given an as-designed BIM model, as-built 3D data could be more effectively processed by integrating Scan-vs-BIM with Scan-to-BIM techniques (such as Hough Transform – based techniques) (Figure 1). How to do so remains a significant gap in the knowledge base.

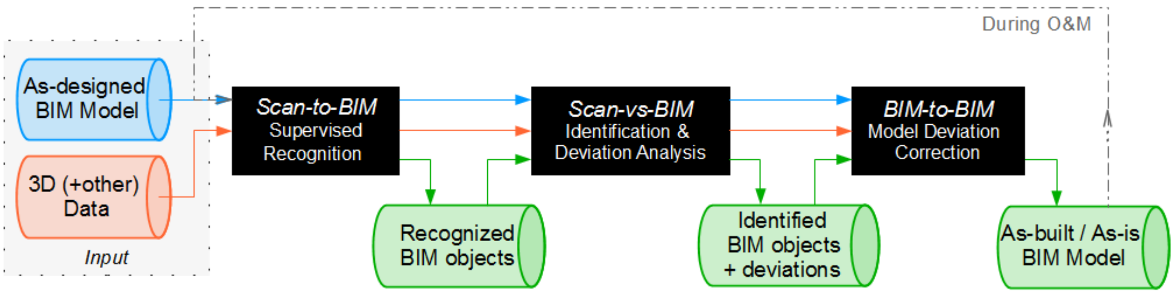


Figure 1: Data processing system for life-cycle BIM model dimensional information management proposed in Bosché et al. [53].

This paper presents an approach that uniquely attempts to achieve this. It integrates the Hough transform–based circular cross-section detection approach of Ahmed et al. [48][49] with the Scan-vs-BIM approach of Bosché et al. [31][32][53] to robustly and automatically recognize and identify all objects with circular cross-sections in as-built TLS point clouds. It is also able to detect cylindrical objects that are not contained in the BIM models – such as those that are "field run", which is an extremely common practice world-wide. It attempts to benefit from the strengths of both approaches while simultaneously elevating their respective limitations. The approach is detailed in Section 3 and validated with an experiment conducted with data acquired on a real-life project (Section 4). The performance is discussed in Section 5, which is followed with the conclusions and suggestions for future work (Section 6).

3 Proposed Approach

Our proposed approach integrates the Hough transform-based circular cross-section detection approach of Ahmed et al [48][49] within the Scan-vs-BIM system of Bosché et al. [31][32][53]. The process contains five steps (see also Figure 2):

1. **Register as-built point cloud with the (as-planned) BIM model**. The as-built point cloud data is registered in the coordinate system of the (as-planned) BIM model. This is the same procedure as the step 1 of the Scan-vs-BIM approach described in Section 2.3. We refer the reader to [32][53][54] for details.
2. **Generate "virtual" as-planned point cloud**. From Step (1), the locations of the scanners (when acquiring the as-built data) are now known in the coordinate system of the BIM model. It is thus possible to generate a "virtual" as-planned point cloud where the BIM model acts as the scanned scene. This is the same procedure as the step 3 of the Scan-vs-BIM approach described in Section 2.3. We refer the reader to [32][53] for details.
3. **Extract circular cross-sections from the as-built and as-planned point clouds;** see Section 3.1.
4. **Match the cross-sections extracted from the as-built point cloud to the cross-sections extracted from the as-planned point cloud;** see Section 3.2.
5. **For each (as-planned) object contained in the BIM model and with circular cross-section (e.g. pipe), infer its recognition/identification, and to which extent it can be considered "built as planned";** see Section 3.3.

Steps 3 to 5 are detailed in sub-sections 3.1 to 3.3 respectively.

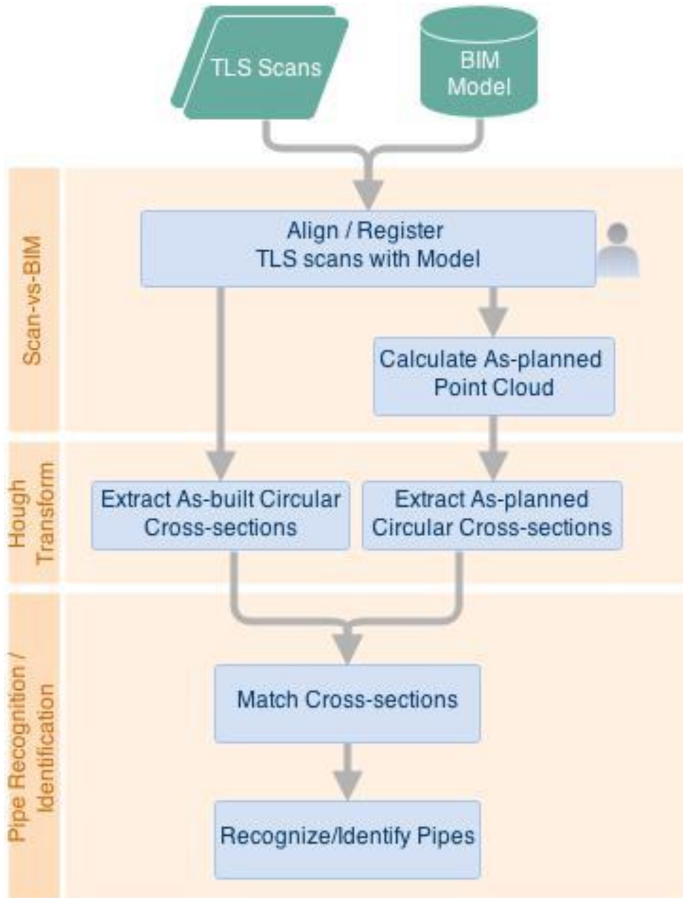


Figure 2: Summary of the proposed novel approach to automatically recognize and identify in TLS data objects with circular cross-sections (e.g. pipes) contained in a project's as-designed BIM model.

3.1 Circular Cross-Section Detection

The application of the Step 1 and 2 of the proposed method produces an as-planned 3D point cloud, with the same characteristics as the as-built point cloud (field of view and point density), and in the same coordinate system as the as-built point cloud.

The Hough transform -based circular cross-section detection method of Ahmed et al. [48][49] is then applied to both point clouds. Very importantly, this is done using this exact same slicing of the data (in three orthogonal directions and at constant intervals along those directions) for both point clouds.

The result of this process is a set of circular cross-sections detected within the as-built point cloud, and another set of circular cross-sections detected within the as-planned point cloud. Furthermore, each data slice is associated with a set of as-built and as-planned cross-sections.

3.2 Circular Cross-Section Matching

Once circular cross-sections have been extracted from both the as-built and as-planned point clouds, the goal is to find, for each as-built cross-section, its best matching as-planned cross-section, if any. For this, we use a cross-section similarity criterion that integrates three sub-criteria with respect to:

- *Location*: the similarity sub-criterion, S_L, is calculated based on the distance between the centers of the as-built and as-planned cross-sections relative to a maximum distance d_{max}:

  $$S_L = 1 - \frac{\|\boldsymbol{c}_{ap} - \boldsymbol{c}_{ab}\|}{d_{max}},$$

  where $\boldsymbol{c}_{ab}$ is the coordinate vector of the center of the as-built cross-section, $\boldsymbol{c}_{ap}$ is the coordinate vector of the center of the as-planned cross-section. We set $d_{max} = 2m$, but one could also consider setting d_{max} as a multiple of the as-planned radius of the object's cross-section. $S_L = 1$ when the centers are exactly the same; $S_L = 0$ when the distance between the centers is d_{max}. Furthermore, we discard any match between cross-sections that are further away than d_{max}, i.e. for which $S_L < 0$.
- *Radius*: the similarity sub-criterion, S_R, is calculated based on the difference between the radii of the as-built and as-planned circular cross-sections relative to a maximum value Δ_{max}:

  $$S_R = 1 - \frac{|r_{ap} - r_{ab}|}{\Delta_{max}},$$

where r_{ab} is the radius of the extracted as-built cross-section, r_{ap} is the designed radius of the as-planned cross-section, and $\Delta_{max}= \alpha r_{ap}$. We set $\alpha = 0.25$. $S_R = 1$ when the radii are exactly the same; $S_R = 0$ when they differ by Δ_{max}. Furthermore, we discard any match between cross-sections with differences in radii larger than Δ_{max}, i.e. for which $S_R < 0$.

- *Orientation*: the similarity sub-criterion, S_O, is calculated as the absolute value of the cosinus of the angle between the normal vectors to the as-built and as-planned cross-sections.

$$S_O = \left|\cos\left(\boldsymbol{n}_{ap} \cdot \boldsymbol{n}_{ab}\right)\right| ,$$

where $\boldsymbol{n}_{ab}$ and $\boldsymbol{n}_{ap}$ are the normal vectors of the extracted as-built and as-planned cross-sections, respectively. $S_O = 1$ when the normal vectors are collinear; $S_O = 0$ when they are orthogonal.

The resulting cross-section similarity criterion, integrating the three sub-criteria above, is then calculated as:

$$S_O = w_L S_L + w_R S_R + w_O S_O ,$$

where w_L, w_R, w_O and are three weights adding up to 1. $S = 1$ when the cross-sections have the same center, radius and orientation.

With a view on speeding up the matching process, as well as ensuring meaningful and consistent matches, we search for matches only within each data slice. In other words, for each as-built cross-section, we search for matching as-planned cross-sections only within the same TLS data slice. This implies that all considered matches are between cross-sections having the same orientation; or, for all considered matches $S_O = 1$. The orientation criterion can thus be discarded from the overall matching criterion, which becomes:

$$S = w_L S_L + w_R S_R \,,$$

where w_L and w_R are two weights adding up to 1.

Because S_L and S_R are both designed to take values in the range $[0; 1]$ and our discarding strategy leads to a situation where there is no obvious reason to advantage one of the criteria over the other, we propose to set the weights as: $w_L = w_R = 0.5$.

3.3 Object Recognition/Identification

For each (as-planned) object with circular cross-section contained in the BIM model, we analyze the cross-section matching results to: (1) infer whether it can be considered recognized/identified; and (2) to which extent it can be considered "built as planned". We propose to calculate the corresponding two metrics: $\%_{matched}$, that can be used to infer recognition and identification, and $\bar{S}$, that estimates the extent to which each object is geometrically "built as planned", as:

$$\%_{matched} = \frac{N_{matched}}{N_{planned}}$$

$$\bar{S} = \frac{\sum_{i=1}^{N_{matched}} (S_i)}{N_{matched}}$$

where $N_{planned}$ is the number of as-planned cross-sections for the given object; $N_{matched}$ is the number of those cross-sections that have been matched to as-built cross-sections; and S_i is the similarity measure for the ith match.

$\%_{matched} = 1$ when all as-planned cross-sections have been matched, which implies that the object is most likely recognized and identified. In contrast, $\%_{matched} = 0$ when none of the cross-sections are matched, implying that the object is most likely not recognized.

$\bar{S} = 1$ when all the matches between as-planned and as-built cross-sections are exact; i.e. the recognized/identified part of the object (whether complete or incomplete) is “built as planned”. In contrast, $\bar{S} < 1$ implies that the recognized/identified part of the object is not built exactly as planned. Figure 3 qualitatively summarizes how these two metrics can be collectively analyzed to interpret the results.

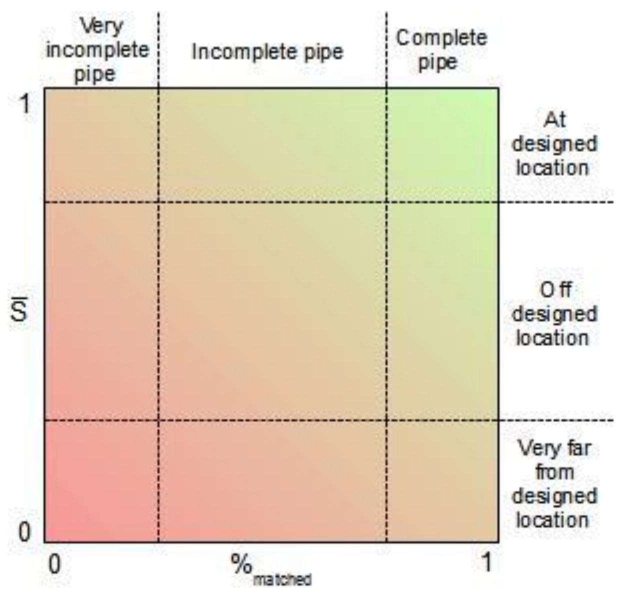


Figure 3: Possible interpretation of the combined values of $\%_{matched}$ and $\bar{S}$.

It is also possible to integrate the two metrics above into a single one, $\bar{S}'$:

$$\bar{S}' = \frac{\sum_{i=1}^{N_{matched}}(S_i)}{N_{planned}}$$

$\bar{S}'$ can be interpreted as a measure of the level to which each entire object is "built as planned" (not just the detected parts, i.e. cross-sections). $\bar{S}' = 1$ when all the planned cross-sections are matched to as-built cross-sections and these matches are exact; i.e. the object is "built as planned". In contrast, $\bar{S}' < 1$ implies that the object is not complete, not built as planned, or a combination of those two cases. For example, $\bar{S}' = 0.5$ could result from half the as-planned cross-sections being perfectly matched but the other half being not matched at all (which could mean that only a section of the object is fully installed); alternatively, it could result from all the as-planned cross-sections being matched, but the matching similarities are on average only 0.5, which means that the object is built, but not as planned.

It is interesting to note that the individual object $\bar{S}'$ values can be aggregated to derive measures of the level to which overall systems or areas are "built as planned". The following formula, implementing a weighted average of the objects' $\bar{S}'$ values, can be used:

$$\bar{S}'_{system} = \frac{\sum_{j=1}^{M_{objects}} \left(N_{planned,j} \bar{S}'_j \right)}{M_{objects}} = \frac{\sum_{j=1}^{M_{objects}} \left(\sum_{i=1}^{N_{matched,j}} \left(S_{j,i} \right) \right)}{M_{objects}}$$

where $M_{objects}$ is the number of objects in the considered system (or area), and $\bar{S}'_j$ is the estimation of the extent to which the j[th] object can be considered "built as planned".

It is important to note that, in contrast with the original Scan-vs-BIM technique that takes occlusions from other objects into account in the object recognition and identification metric (see definitions of $\%_{recognized}$ and $\%_{confidence}$ in Section 2.3), the effect of occlusions is not considered in the metric $\%_{matched}$. This could be considered in future work. We point out however that $\bar{S}$ and $\bar{S}'$ directly work with the matched cross-sections and therefore are not impacted by occlusions.

3.4 As-built Modelling

Once the as-planned pipes have been recognized, it is possible to conduct their as-built modelling by generating pipes along the cross-sections matched to each as-planned pipe. In this paper, we simply propose to split the cross-sections into groups of collinear cross-sections (across several layers), and then apply the method proposed by Ahmed et al. [48][49]. This method generates the best fitting centerline (filtering out any false cross-sections) from the group of cross-sections, and then uses this centerline along with the cross-sections radius to generate cylinders representing the straight pipe.

4 Experiments

4.1 Data

We conducted an experiment with data collected during the construction of the Engineering VI Building at the University of Waterloo that is designed to shelter the Chemical Engineering Department of the university (a five-storey, 100,000-square-foot building). The data collected include 2D drawings and a set of field laser scans. The authors created a 3D CAD/BIM model of the 5th floor based on the information provided on 2D drawings.

This project was chosen for the study as the building includes numerous pipes and ducts, to provide water and gas to different laboratories and to collect and evacuate chemical fumes from them. This study focused specifically on the service corridor of the fifth floor (31m x 3.4m) as it contains all the pipes coming from the lower levels and going all the way up to the penthouse. Figure 4 shows the service corridor section of the 3D CAD/BIM model.

Laser scans were acquired from the corridor using the FARO LS 880 HE laser scanner, which employs phase-based technology (see Table 1 for the technical characteristics of the scanner). Six scans were acquired along the corridor because of the density of the pipes and ducts and the narrowness of the corridor (Figure 5).

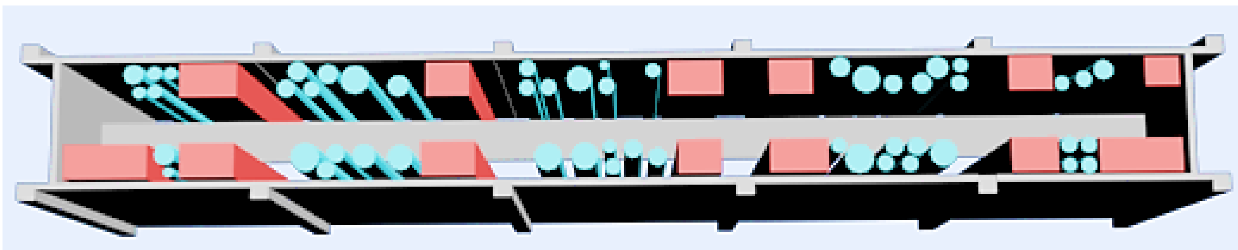


Figure 4: 3D model of the 5th floor corridor of Engineering VI.

Table 1: Characteristics of the FARO LS 880 HE scanner

Laser Type		*785nm; near infrared*
Distance	Range	*0.6 m to 76 m.*
	Accuracy	*±3 mm @ 25 m.*
Angle	Range	*Hor: 360°; Vert: 320°*
	Accuracy	*Hor: 16 µrad; Vert: 16 µrad*
Maximum Resolution		*Hor: 13 µrad; Vert: 157 µrad*
Acquisition Speed		*up to 120,000 pts/s*

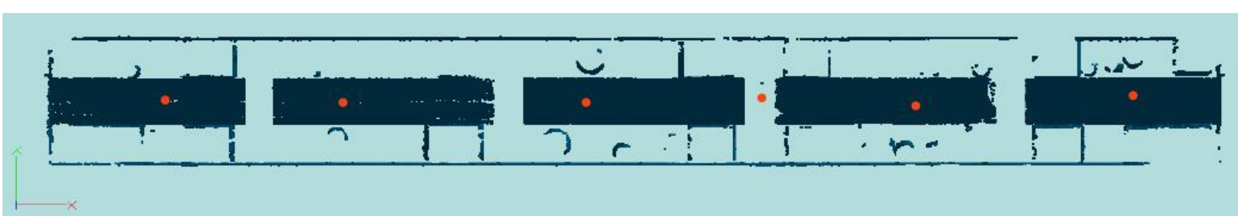


Figure 5: Combined six laser scans of the 5th floor corridor Engineering VI; the dots show the scanning locations.

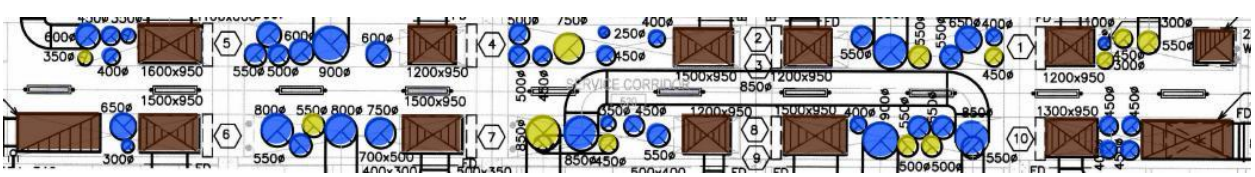
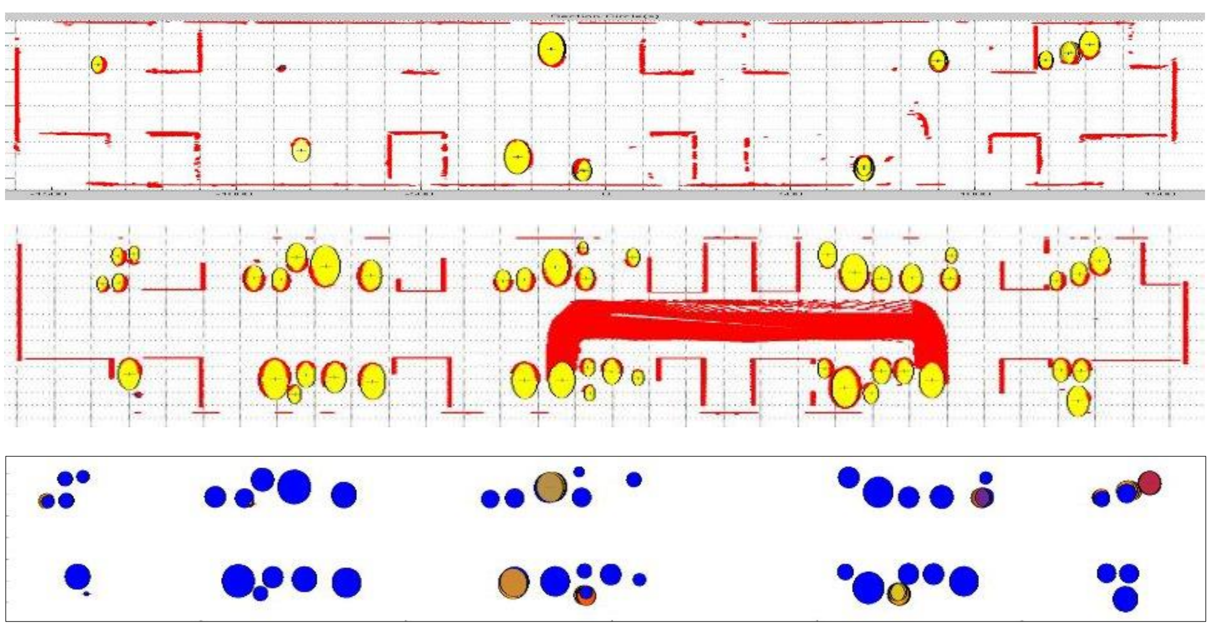


Figure 6: Top view of the corridor highlighting the pipes visually identified as present (at least partially) in the corridor at the time of scanning. The pipes present are shown in yellow, those absent are in blue. In brown are ducts that were also present.

4.2 Results

4.2.1 Cross-section Detection

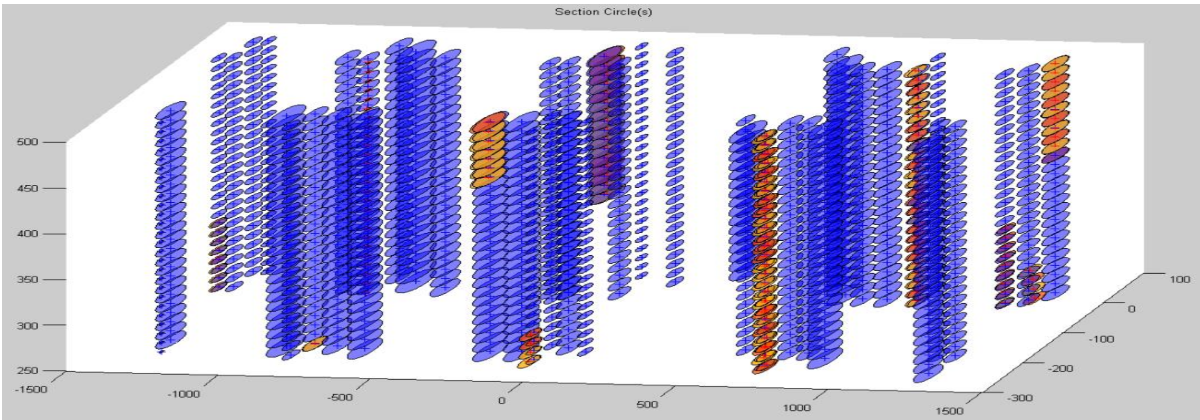
After aligning the point cloud of the six scans in the coordinate system of the project 3D CAD/BIM model, the as-planned point cloud is automatically calculated and the circular cross-sections automatically extracted from the as-planned and as-built point clouds. Because the pipes contained in the corridor are essentially all vertical, we focus on those only, and apply the Hough transform -based method of Ahmed et al. [49] solely with slices along the vertical (Z) axis. Twenty six slices are automatically generated with 10 cm intervals. From this, the system automatically detects 1176 as-planned circular cross-sections and 164 as-built circular cross-sections (see Figure 7).



(a)

(b)

(c)



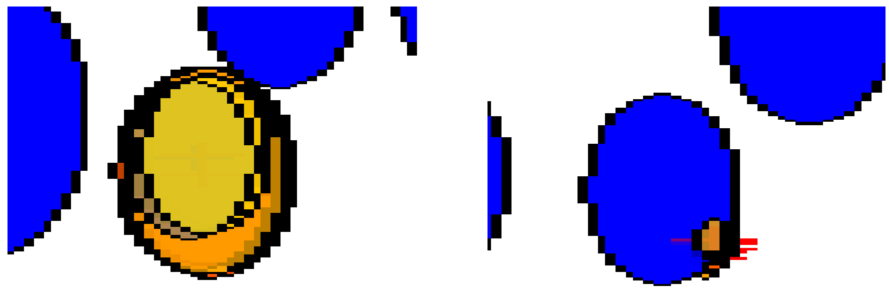
(d)

Figure 7: Extracted cross-sections detected in the as-built (a) and as-planned (b) point clouds. (c) and (d) show the as-built (orange) and as-planned (blue) cross-sections altogether.

After applying the circular cross-section matching approach described in Section 3.1, 112 of the 164 as-built cross-sections are matched to as-planned cross-sections, and all with similarity levels > 0.95. Looking at the 52 as-built cross-sections that are not matched, these come from two sets of 26 cross-sections:

- The first 26 cross-sections were detected at the same location as another set of 26 as-built cross-sections but for a different radius (see Figure 8(a)). The system matched the latter set to the locally corresponding as-planned cross-sections because they had the exact same radius; the other set was thus correctly rejected.
- The second set of 26 cross-sections comes from a very small pipe present in the corridor at the time of scanning but that did not correspond to any pipe in the 3D model (see Figure 8(b)). These cross-sections were thus correctly rejected by the system. Note that, using the same dataset, the original Scan-vs-BIM approach of Bosché et al. had wrongly suggested that this pipe was present in the scene (albeit with some low level of confidence) [53].

In conclusion, the 52 cross-sections that are not matched to any as-planned cross-section, are actually correctly not matched by the system. Note, however, that the non-matched detected cross-sections could still be used to inform and partially automate a manual update of the BIM model. For example, the pipe with small diameter found by the system could be added directly to the BIM model.



(a) (b)

Figure 8: The two cases where as-built cross-sections are (correctly) not matched to any as-planned one. (a) two sets of cross-sections are extracted at the same location; the system rejects the set with the largest radius because it is too dissimilar to the locally corresponding as-planned cross-sections; (b) small temporary pipe clearly not corresponding to the local as-planned pipe.

4.2.2 Pipe Recognition and Identification

After aggregating the results for each pipe actually present in the corridor (i.e. the yellow pipes in Figure 6), the pipe recognition/identification metrics described in Section 3.3, namely $\%_{matched}$, $\bar{S}$ and $\bar{S}'$, are calculated and summarized in Table 2 and Figure 9. The results highlight a few points:

- For two of the pipes that can be visually recognized in the data, the system fails to detect any circular cross-section. This is due to the fact that too few points were actually scanned from those pipes to enable the confident detection of cross-sections.
- In this particular experimental dataset, all the matched as-built cross-sections are very close to their matching as-planned ones ($\bar{S} \geq 0.95$), which indicates that pipes, or at least partial sections of pipes, are recognized at their expected locations.
- For six pipes, fewer than half the as-planned cross-sections are recognized. As summarized earlier in Figure 3, this and the corresponding high $\bar{S}$ values for those objects indicate that they

are likely identified at their as-built locations, but are incomplete (which is confirmed by a visual analysis of the data; see also Figure 11).

- For three pipes (09, 20 and 26), all as-planned cross-sections are recognized, and are found very close to their designed locations and with the same radius. These pipes would thus be correctly considered fully identified.

Table 2: Recognition results ($\%_{matched}$, $\overline{S}$, $\overline{S}'$) for each of the pipes actually present (at least partially) in the as-built point cloud.

Pipe Name	$N_{planned}$	$N_{matched}$	$\%_{matched}$	$\overline{S}$	$\overline{S}'$
Pipe_01	26	11	0.42	0.99	0.42
Pipe_02	26	4	0.15	0.95	0.15
Pipe_03	26	9	0.35	0.97	0.34
Pipe_09	26	26	1.00	0.97	0.99
Pipe_12	26	0	0.00	0.00	0.00
Pipe_18	0	0	0.00	0.00	0.00
Pipe_20	26	26	1.00	0.97	0.98
Pipe_26	16	16	1.00	0.96	0.98
Pipe_32	16	4	0.25	0.96	0.25
Pipe_35	26	7	0.27	0.97	0.27
Pipe_44	26	1	0.04	0.99	0.04
Pipe_51	26	8	0.31	0.98	0.30

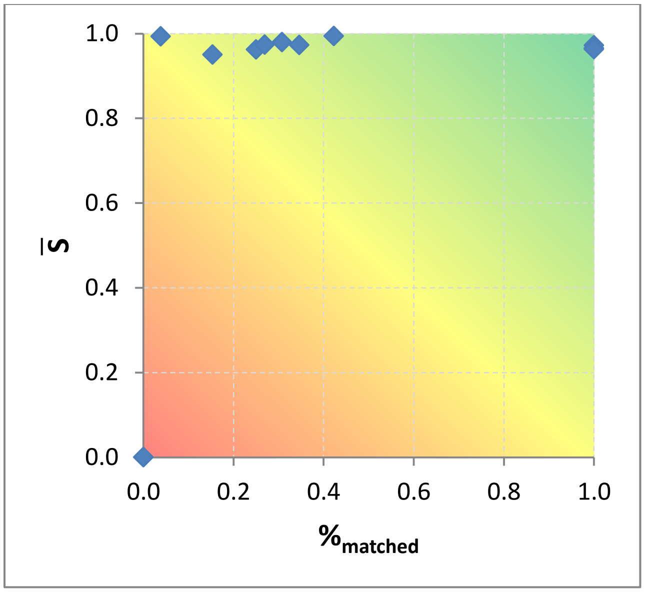


Figure 9: The recognition values $\%_{matched}$ and $\overline{S}$ for all the pipes present in the corridor. Figure 3 indicates how the results can be interpreted.

The results above indicate some level of robustness of our proposed approach, but it remains to be assessed how it compares against the original Scan-vs-BIM approach of Bosché et al.[53]. To conduct this comparison, we apply the original Scan-vs-BIM approach of Bosché et al. [53] to this dataset, and compare $\bar{S}'$ and $\%_{confidence}$ (the metric used in [53]) that both provide an estimation of the level of confidence in the matching of the as-planned objects to the as-built data. Table 3 and Figure 10 summarize the values obtained and their comparison. The results tend to demonstrate that the new approach is more robust, as illustrated with the following four examples (see Figure 11):

- *Pipe_20*: As can be seen in Figure 11(a), as-built points are found in large areas along the entire length of the pipe and these are at the same locations as the as-planned ones. For this reason, the two approaches both estimate high levels of confidence in the recognition/identification of the pipe ($\bar{S}' = 0.98$ and $\%_{confidence} = 0.81$).
- *Pipe_09*: As can be seen in Figure 11(b), as-built points are found in large parts along the entire length of the pipe. However, it appears that the pipe is not located exactly where it is planned to

be. Despite the fact that the out-of-place deviation is minor (~5cm), the original Scan-vs-BIM approach achieves a fairly low level of confidence in the recognition of the pipe ($\%_{confidence} = 0.49$). In contrast, the new approach correctly maintains a high level of confidence in the recognition ($\bar{S}' = 0.99$); it also provides information that can be readily used to automatically correct the as-built location of the pipe in the BIM model.

- *Pipe_32*: As can be seen in Figure 11(c), as-built points are found at the right location horizontally, but only the bottom section of the pipe is actually installed. But, because more points are recognized at the bottom of the pipe than planned, the original Scan-vs-BIM ends up reaching a level of confidence in the recognition of the entire pipe that is clearly over-estimated ($\%_{confidence} = 0.73$). In contrast, the new approach estimates a more appropriate level of confidence ($\bar{S}' = 0.25$).
- *Pipe*_02: As can be seen in Figure 11(e), as-built points are found at a horizontal location that is slightly different from the planned one, and only the bottom part of the pipe has actually been installed. The combined effect of the out-of-plane deviation (which is just ~6cm) leads the original Scan-vs-BIM approach to give a quasi-null level of confidence ($\%_{confidence} = 0.02$) – and actually reaches the conclusion that the pipe is not recognized. In contrast, the new approach once again estimates a higher, and generally more representative, level of confidence ($\bar{S}' = 0.15$).

Table 3: Comparison of the performance of the proposed approach ($\bar{S}'$) against the original Scan-vs-BIM approach of Bosché et al. [53] ($\%_{confidence}$) for recognizing each of the pipes actually present (at least partially) in the as-built point cloud.

Pipe Name	$\bar{S}'$	$\%_{confidence}$
Pipe_01	0.42	0.44

Pipe_02	0.15	0.02
Pipe_03	0.34	0.01
Pipe_09	0.99	0.49
Pipe_12	0.00	0.00
Pipe_18	0.00	0.55
Pipe_20	0.98	0.81
Pipe_26	0.98	0.46
Pipe_32	0.25	0.73
Pipe_35	0.27	0.32
Pipe_44	0.04	0.01
Pipe_51	0.30	0.33

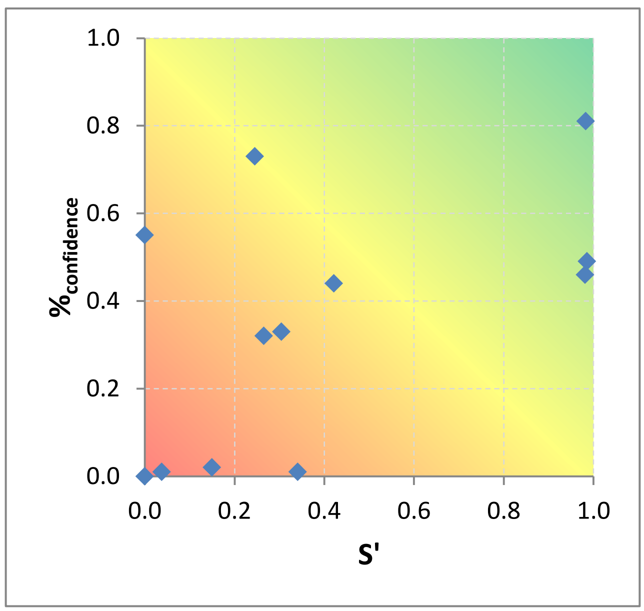


Figure 10: Graphical representation of the results summarized in Table 3.


(a)
(b)
(c)
(d)

Figure 11: The as-built and as-planned point-clouds for objects Pipe_20 (a), Pipe_09 (b), Pipe_32 (c), and Pipe_02 (d). From the left, the first column shows top views of the as-built point clouds, the second columns top views of the as-planned point clouds, the third column top views of both point clouds, and the last column side views of both point clouds.

Given all the $\bar{S}'$ values for all the pipes in the corridor, we can also calculate the overall level with which the corridor's piping is currently built as-planned (including whether objects are built or not), using the formula described in Section 3.3. We obtain: $\bar{S}'_{corridor\_piping}$=9%. This value is low essentially because many of the pipes are currently not installed. But, arguably, it provides a meaningful estimation of the level to which piping works in the corridor have progressed to date.

4.2.3 As-built Modelling

Once the cross-sections have been matched, the system not only calculates the $\bar{S}'$ value to infer the recognition/identification of each BIM model object (and infer whether it is built as planned), but it also generates the as-built model of each pipe. The result of this process with our experimental data is shown in Figure 12. In this figure, the pipes are labelled so that they can be related to the results reported in Table 2 and Table 3.

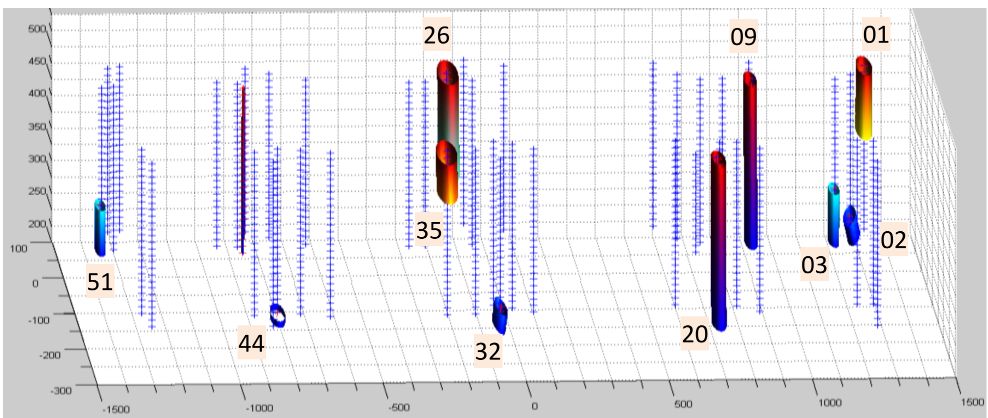


Figure 12: The as-built 3D models of the recognized/identified pipes, in comparison with the centerlines of the as-planned pipes.

5 Discussion

The experiment reported above, albeit arguably of a limited nature, does demonstrate the added value of the proposed new approach to detect, recognize and identify cylindrical MEP components, in comparison with the original Scan-vs-BIM approach of Bosché et al. [53]. The two main areas of improved performance are:

1. **Out-of-plane deviations (or, out-of-centerline deviations)**: The original approach can only recognize objects within 5cm or so from their as-planned locations. In contrast, the new approach is far less sensitive to such deviations, and maintains high levels of confidence up to and actually far beyond such distances.
2. **Pipe completeness recognition**: The original approach is not able to distinguish whether the recognized points are acquired at different locations along the pipes, and may consequently over-estimate its level of confidence. In contrast, the new approach, by matching cross-sections at regular intervals along the pipes, is able to take this factor into account when estimating its level of confidence.

Additionally, the proposed approach is capable of identifying objects (i.e. identify to which object each cross-section corresponds to). Therefore, it addresses the issue of "pipe occlusions" – i.e. ensuring that an occluded pipe is not recognized as two different ones.

Naturally, this performance needs to be confirmed with additional, more complex scenarios, in particular with pipes going in different directions (not just vertically). Yet, some limitations can already be pointed at that would require further investigation, in particular:

- The Hough transform -based approach for detecting circular cross-sections analyzes the data in pre-determined directions, in particular the main three orthogonal directions. While pipes and other cylindrical MEP objects tend to be run in these main, these three main directions could be complemented with at least 6 other ones to search for cross-sections oriented 45° from the main directions (this would also help in recognizing elbows). However, increasing the number of slicing directions proportionally increases the processing time. An alternative more general approach to extract cylindrical pipes, such as the one proposed by Son et al. [50], could be investigated.

- While the proposed new method to recognize and identify objects with circular cross-sections is more robust than the original approach employed by Bosché et al. [53], *false positive* and *false negative* recognitions could still occur. For example, the current approach cannot recognize a pipe that is further away than d_{max} from its planned location (false negative). Or, if a pipe is mis-located but happens to have an as-built location and radius that are the same as those of another pipe, then the system will wrongly recognize the pipe (false positive). Preventing such errors would require further prior information to be considered in the recognition and identification process, such as *component connectivity*.

6 Conclusions

This paper presented a novel approach to automatically recognize and identify objects with circular cross-sections (e.g. pipes) in 3D TLS data acquired from construction sites, and given the project's 3D design BIM model. This approach uniquely integrates an object detection and recognition technique (typically employed in Scan-to-BIM applications) with a Scan-vs-BIM approach inferring object recognition and identification from proximity analysis. Specifically, the approach integrates the efficient Hough transform -based circular cross-section detection approach of Ahmed et al. [48][49] within the Scan-vs-BIM object recognition and identification framework of Bosché et al. [31][32][53]. Objects are recognized based on the matching of as-built and as-planned cross-sections in terms of proximity, orientation and radius. The proposed object recognition metrics can be used not only to infer recognition, but also to estimate the extent to which each object is "built as planned". These individual estimations can also be aggregated to assess the extent to which a system, area or other grouping is built as planned, i.e. its “percentage built as planned”.

An experiment has been conducted using scans acquired from a utility corridor under construction. The results are very encouraging and already demonstrate the added value of the proposed integrated approach over the rather simpler Scan-vs-BIM approach of Bosché et al. [53]. While these results need to be confirmed with more complex scenarios, two main limitations are already identified that will require further investigations, namely: the search for pipes by the proposed Hough transform approach in pre-defined directions only; and the fact that false positive and false negatives may still occur (although the proposed approach already significantly reduces their chance of occurrence). Alternative approaches to the circular cross-section detection method employed here could be investigated that are more general and able to detect circular cross-sections, or more generally cylindrical pipes, in any direction. The metric used to recognize and identify the as-planned objects also presents some limitations that can only be addressed by applying higher-level reasoning, for example by analyzing object connectivity.

7 Acknowledgements

The authors would like to thank Gary Caldwell from Aecon Group Inc. for providing the 2D drawings of Engineering V Building, and for allowing us to take the scans of the construction. The authors would also like to thank to Arash Shahi and Yazan Chaban from the University of Waterloo for their help during this work.

This research is partially funded by NSERC CRD Grant, NSERC Discovery Grant, CII and SNC Lavalin.

8 References

[1] Schaufelberger, J.E., Holm, L. (2002). *Management of Construction Projects: A Constructor's Perspective*, Prentice Hall.

[2] Grau, D., Caldas, C. H., Haas, C. T., Goodrum, P. M., Gong, J. (2009). Assessing the impact of materials tracking technologies on construction craft productivity, *Automation in Construction*, 18, pp. 903-911.

[3] Ergen, E., Akinci, B., Sacks, R. (2007). Life-cycle data management of engineered-to-order components using radio frequency identification, Automation in Construction, 21, pp. 356-366.

[4] Li, N., Becerik-Gerber, B. (2011). Performance-based evaluation of RFID-based Indoor Location Sensing Solutions for the Built Environment, *Advanced Engineering Informatics*, 25 (3), pp. 535–546.

[5] Pradhan, A., Ergen, E., Akinci, B. (2009). Technological Assessment of Radio Frequency Identification Technology for Indoor Localization, *ASCE Journal of Computing in Civil Engineering*, 23 (4), pp. 230-238.

[6] Razavi, S. N., Haas, C. T., (2010). Multisensor data fusion for on-site materials tracking in construction, *Automation in Construction*, 19, pp. 1037-1046.

[7] Razavi, S.N., Moselhi, O. (2012). GPS-less indoor construction location sensing, *Automation in Construction*, 28, pp. 128-136.

[8] Teizer, J., Venugopal, M., Walia, A. (2008). Ultra-wide band for Automated Real-time Three-Dimensional Location Sensing for Workforce, Equipment, and Material Positioning and Tracking, *Transportation Research Record, Transportation Research Board of the National Academies*, Washington D.C, pp. 56–64.

[9] Cheng, T., Venugopal, M., Teizer, J., Vela, P.A. (2011). Performance evaluation of ultra-wideband technology for construction resource location tracking in harsh environments, *Automation in Construction*, 20, pp.1173-1184.

[10] Shahi, A., Aryan, A., West, J. S., Haas, C. T., Haas, R. G. (2012). Deterioration of UWB positioning during construction, *Automation in Construction*, 24, pp. 72-80.

[11] Saidi, K. S., Teizer, J., Franaszek, M., Lytle, A. M. (2011). Static and dynamic performance evaluation of a commercially-available ultra-wide band tracking system, *Automation in Construction*, 20, pp. 519-530.

[12] Grau, D., Caldas, C. H., Haas, C. T., Goodrum, P. M., Gong, J. (2009). Assessing the impact of materials tracking technologies on construction craft productivity, *Automation in Construction*, 18, pp. 903-911.

[13] Haas, C., (1986). *Algorithms to Map Subsurface Ferrous Conductors*. MSc Thesis, Carnegie Mellon University Department of Civil Engineering, Aug. 1986.

[14] Haas, C., Shen, H., Phang, W. A., & Haas, R. (1984). *Application of image analysis technology to automation of pavement condition surveys*. Publication of: Balkema (AA).

[15] Abeid, J., Allouche, E., Arditi, D., & Hayman, M. (2003). PHOTO-NET II: a computer-based monitoring system applied to project management. Automation in construction, 12(5), pp. 603-616.

[16] Ibrahim, Y. M., Lukins, T. C., Zhang, X., Trucco, E., & Kaka, A. P. (2009). Towards automated progress assessment of workpackage components in construction projects using computer vision. *Advanced Engineering Informatics*, 23(1), pp. 93-103.

[17] Chi, S., Caldas, C. H., & Kim, D. Y. (2009). A methodology for object identification and tracking in construction based on spatial modeling and image matching techniques. *Computer-Aided Civil and Infrastructure Engineering*, 24(3), pp. 199-211.

[18] Wu, Y., Kim, H., Kim, C., & Han, S. H. (2009). Object recognition in construction-site images using 3D CAD-based filtering. *Journal of Computing in Civil Engineering*, 24(1), pp. 56-64.

[19] Teizer, J., Caldas, C. H., & Haas, C. T. (2007). Real-time three-dimensional occupancy grid modeling for the detection and tracking of construction resources. Journal of Construction Engineering and Management, 133(11), pp. 880-888.

[20] Golparvar-Fard, M., Pena-Mora, F., Savarese, S. (2009). Application of D4AR – A 4-Dimensional augmented reality model for automating construction progress monitoring data collection, processing and communication, *Journal of Information Technology in Construction*, 14, pp. 129-153.

[21] Golparvar-Fard, M., Peña-Mora, F., Savarese, S. (2013). Automated progress monitoring using unordered daily construction photographs and IFC-based Building Information Models, ASCE *Journal of Computing in Civil Engineering*, (in press).

[22] El-Omari, S., & Moselhi, O. (2008). Integrating 3D laser scanning and photogrammetry for progress measurement of construction work. *Automation in construction*, 18(1), pp. 1-9.

[23] Ahmed, M., Haas, C., West, J., & Haas, R. (2011). Rapid Tracking of Pipe-Works Progress using Digital Photogrammetry. *Proceedings of the 9th Construction Specialty Conference*, Ottawa, Ontario, Canada, pp. 14-17.

[24] Ahmed, M., Haas, C., and Haas, R. (2012). "Using Digital Photogrammetry for Pipe-Works Progress Tracking" Canadian Journal for Civil Engineering, *CJCE Special Issue on Construction Engineering and Management*, 39(9) pp. 1062-1071.

[25] Ahmed, M., Haas, C. T., & Haas, R. (2011). Toward low-cost 3D automatic pavement distress surveying: the close range photogrammetry approach. *Canadian Journal of Civil Engineering*, 38(12), pp. 1301-1313.

[26] Cheok, G. S., Stone, W. C., Lipman, R. R., & Witzgall, C. (2000). Ladars for construction assessment and update. *Automation in Construction*, 9(5), pp. 463-477.

[27] Stone, W., Cheok, G. (2001). *LADAR sensing applications for construction*, Building and Fire Research, National Institute of Standards and Technology (NIST), Gaithersburg, MD.

[28] Jaselskis, E. J., Gao, Z., & Walters, R. C. (2005). Improving transportation projects using laser scanning. *Journal of Construction Engineering and Management*, 131(3), pp. 377-384.

[29] Brilakis, I., Lourakis, M., Sacks, R., Savarese, S., Christodoulou, S., Teizer, J., and Makhmalbaf, A. (2010). "Toward Automated Generation of Parametric BIMs based on Hybrid Video and Laser Scanning Data". *Advanced Engineering Informatics*, 24(4), pp. 456-465.

[30] Jacobs G. (2008). 3D scanning: Using multiple laser scanners on projects, *Professional Surveyor Magazine*, 28.

[31] Bosché, F., Haas, C.T. (2008). Automated retrieval of 3D CAD model objects in construction range images, *Automation in Construction*, 17, pp. 499-512.

[32] Bosché, F. (2010). Automated recognition of 3D CAD model objects in laser scans and calculation of as-built dimensions for dimensional compliance control in construction, *Advanced Engineering Informatics*, 24(1), pp. 107-118.

[33] Kim, C., Son, H., Kim, C. (2013). Automated construction progress measurement using a 4D building information model and 3D data, *Automation in Construction*, 31, pp. 75-82.

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

[35] Tang, P., Anil, E., Akinci, B., Huber, D. (2011). Efficient and Effective Quality Assessment of As-Is Building Information Models and 3D Laser-Scanned Data, *ASCE International Workshop on Computing in Civil Engineering*, Miami, FL, USA.

[36] Lee, J., Son, H., Kim, C., Kim, C. (2013). Skeleton-based 3D reconstruction of as-built pipelines from laser-scan data, Automation in Construction 35, pp. 199-207..

[37] Lijing, B., Zhengpeng, Z. (2008). Application of point clouds from terrestrial 3D laser scanner for deformation measurements, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 37.

[38] Park, H.S., Lee, H.M., Adeli, H., Lee, I. (2007). A new approach for health monitoring of structures: terrestrial laser scanning, *Computer Aided Civil and Infrastructure Engineering*, 22, pp.19-30.

[39] Qui, D. W., Wu, J. G. (2008). Terrestrial laser scanning for deformation monitoring of the thermal pipeline traversed subway tunnel engineering, *XXIst ISPRS Congress: Commission V, WG 3*, Beijing, pp. 491-494.

[40] Valero, E.,Adán, A. and Cerrada, C. (2012). Automatic method for building indoor boundary models from dense point clouds collected by laser scanners, *Sensors*, 12, pp. 16099-16115.

[41] Xiong, X., Adán, A., Akinci, B., Huber, D. (2013). Automatic creation of semantically rich 3D building models from laser scanner data, *Automation in Construction*, 31, pp. 325-337.

[42] Kwon, S.-W., Bosche, F., Kim,C., Haas, C. T. andLiapi, K. A.(2003). Fitting Range Data to Primitives for Rapid Local 3D Modeling Using Sparse Range Point-clouds. *Automation in Construction*, 13(1), pp. 67-81.

[43] McLaughlin, J., Sreenivasan, S.V., Haas, C.T. and Liapi, K.A. (2004). Rapid Human-Assisted Creation of Bounding Models for Obstacle Avoidance in Construction. *Computer-Aided Civil and Infrastructure Engineering*, Vol. 19, pp. 3-15.

[44] Rabbani T., Heuvel F. van den, (2005). Efficient Hough transform for automatic detection of cylinders in point clouds. *ISPRS WG III/3, III/4, V/3 Workshop on Laser scanning*, Enschede, NL.

[45] Shih, N. J., & Huang, S. T. (2006). 3D scan information management system for construction management. *Journal of construction engineering and management*, 132(2), 134-142.

[46] Turkan, Y., Bosche, F., Haas, C. T., & Haas, R. (2012). Automated progress tracking using 4D schedule and 3D sensing technologies. *Automation in Construction*, 22, 414-421.

[47] Turkan, Y., Bosché, F., Haas, C. T., & Haas, R. (2012). Toward Automated Earned Value Tracking Using 3D Imaging Tools. *Journal of Construction Engineering and Management*, 139(4), 423-433.

[48] Ahmed M., Haas C.T., Haas R. (2013). Autonomous modeling of pipes within point clouds. *Proceedings of the 30th ISARC*, Montréal, Canada, pp. 1093-1100.

[49] Ahmed M., Haas C.T., Haas R. (2014) Automatic Detection of Cylindrical Objects in Built Facilities. *ASCE Journal of Computing in Civil Engineering* (in press).

[50] Son H., Kim C., Kim C. (2013). Knowledge-based approach for 3d reconstruction of as-built industrial plant models from laser-scan data, *Proceedings of the 30th ISARC*, Montréal, Canada, pp. 885-893.

[51] Vosselman, G. and Dijkman, S. (2001). 3D Building Model Reconstruction from Point Clouds and Ground Plans. *International Archives of Photogrammetry and Remote Sensing*, Volume XXXIV-3/W4 Annapolis, MD, 22-24 Oct. 2001.

[52] Turkan, Y., Bosché, F., Haas, C.T., Haas, R.G., (2013) Tracking of Secondary and Temporary Objects in Structural Concrete Work, Emerald, *Construction Innovation: Information, Process and Management*, (accepted).

[53] Bosché, F., Guillemet, A., Turkan, Y., Haas, C.T., Haas, R.G., (2013) Assessing the value of a Scan-vs-BIM framework for tracking the built status of MEP works, *ASCE Journal of Computing in Civil Engineering*, (in press).

[54] Bosche, F., Haas, C., and Akinci, B. (2009). ”Automated Recognition of 3D CAD Objects in Site Laser Scans for Project 3D Status Visualization and Performance Control.” *J. Comput. Civ. Eng.* 23, SPECIAL ISSUE: Graphical 3D Visualization in Architecture, Engineering, and Construction, 311–318.

[55] Hough, P. V. C., 1962. *Method, and means for recognizing complex patterns*, U. S. Patent 3069654.

[56] Duda R. O. and Hart P. E. (1972). Use of the Hough Transformation to Detect Lines and Curves in Pictures, *Communications ACM*, Vol. 15, pp. 11-15.

[57] Cheng Z. and Liu Y. (2004). Efficient Technique for Ellipse Detection using Restricted Randomized Hough Transform. *Proceedings of the International Conference on Information Technology (ITCC'04)*, Vol.2, pp.714-718.

[58] van Ginkel, M., Kraaijveld, M.A., van Vliet, L.J., Reding, E.P., Verbeek, P.W., and Lammers, H.J. (2003). Robust Curve Detection using a Radon Transform in Orientation Space. *Proceedings of the 13th Scandinavian Conference on Image Analysis*, pp. 125-132.

[59] van Ginkel, M., Luengo Hendriks, C.L., and van Vliet, L.J. (2004). *A short introduction to the Radon and Hough transforms and how they relate to each other*. Technical Report No. QI-2004-01. Quantitative Imaging Group, Imaging Science & Technology Department, Faculty of Applied Science, Delft University of Technology.`

[60] Besl, P. J. and Jain, R. C., (1988), Segmentation through variable-order surface fitting. IEEE Transactions on Pattern Analysis and Machine Intelligence, 10(2), pp. 167-192.

[61] Hoppe, H., DeRose, T., Duchamp, T., McDonald, J. and Stuetzle, W. (1992). Surface Reconstruction from Unorganized Points. Proceedings of SIGGRAPH '92, Chicago, Illinois.

[62] Koenderink, J. and Doorn, A. v. (1992). Surface Shape and Curvature Scales. Image and Vision Computing, pp. 557-565.

[63] Dorai, C. and Jain, A. (1997). "COSMOS-A Representation Scheme for 3D Free-Form Objects." IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(10), pp.1115-1130.

[64] Shaffer, E. and Garland, M. (2001). Efficient Adaptive Simplification of Massive Meshes. IEEE Visualization.

[65] Pauly, M., Keiser, R. and Gross, M. (2003). Multi-scale Feature Extraction on Point-Sampled Surfaces. Computer Graphics Forum, 22(3), pp. 281-289.

[66] Rabbani T., van den Heuvel, F. A., Vosselman, G., (2006). Segmentation of Point Clouds Using Smoothness Constraint. APRS Volume XXXVI, Part 5, Dresden 25-27 September.

[67] Jagannathan, A. and Miller, E. (2007). Three-dimensional Surface Mesh Segmentation using Curvedness-based Region Growing Approach. IEEE Trans. Pattern Analysis and Machine Intelligence, 29(12): 2195-2204.

[68] Klasing, K., Althoff, D., Wollherr, D., Buss, M. (2009), Comparison of surface normal estimation methods for range sensing applications. IEEE International Conference on Robotics and Automation, pp. 3206-3211.

[69] Carr, J.C., Beatson, R.K., McCallum, B.C., Fright, W.R., McLennan, T.J., Mitchell, T.J. (2003). Smooth surface reconstruction from noisy range data. International Conference on Computer Graphics and Interactive Techniques, pp. 119-126

[70] Xiong, X., Adan, A., Akinci, B., Huber, D. (2012), Automatic creation of semantically rich 3D building models from laser scanner data. Automation in Construction, pp. 325-337.

[71] Leavers, V.F. (1993). Which Hough Transform? *CVGIP: Image Understanding*, 58(2), pp. 250-264.

[72] Newman, T.S., Flynn, P.J., Jain, A.K. (1993). Model-Based Classification of Quadric Surfaces. *CVGIP: Image Understanding*, 58(2), pp. 235-249.

[73] Borrmann, D., Elseberg, J., Lingemann, K., Nüchter, A. (2011). The 3D Hough Transform for plane detection in point clouds: A review and a new accumulator design, *3D Research*, 2(2).

[74] Pietrowcew, A. (2003). Face detection in colour images using fuzzy Hough transform. *Opto-electronics review*, 11(3), pp. 247-251.

[75] Bosché, F. (2011). Plane-based Coarse Registration of 3D Laser Scans with 4D Models, *Advanced Engineering Informatics*, 26, pp. 90-102.

[76] Kim, C., Son, H., and Kim, C. (2013). Fully automated registration of 3D data to a 3D CAD model for project progress monitoring, *Automation in Construction*, 35, pp. 587-594.