

LARGE SCALE 3D POINT CLOUD MODELING FROM CAD DATABASE IN COMPLEX INDUSTRIAL ENVIRONMENTS

Mathias Corsia,* Théodore Chabardès, Hassan Bouchiba, Andrés Serna

Terra3D Research, 36 boulevard de la Bastille, 75012 Paris, France.

www.terra3d.xyz

{mathias.corsia, theodore.chabardes, hassan.bouchiba, andres.serna}@terra3d.fr

Commission II, WG II/4

KEY WORDS: Point Cloud, Modeling, Segmentation, CAD, PCA, 3D Features, 3D Keypoints, Registration

ABSTRACT:

In this paper, we present a method to build Computer Aided Design (CAD) representations of dense 3D point cloud scenes by queries in a large CAD model database. This method is applied to real world industrial scenes for infrastructure modeling. The proposed method firstly relies on a region growing algorithm based on novel edge detection method. This algorithm is able to produce geometrically coherent regions which can be agglomerated in order to extract the objects of interest of an industrial environment. Each segment is then processed to compute relevant keypoints and multi-scale features in order to be compared to all CAD models from the database. The best fitting model is estimated together with the rigid six degree of freedom (6 DOF) transformation for positioning the CAD model on the 3D scene. The proposed novel keypoints extractor achieves robust and repeatable results that captures both thin geometrical details and global shape of objects. Our new multi-scale descriptor stacks geometrical information around each keypoint at short and long range, allowing non-ambiguous matching for object recognition and positioning. We illustrate the efficiency of our method in a real-world application on 3D segmentation and modeling of electrical substations.

1. INTRODUCTION

Nowadays, 3D LiDAR technology is widely used for mapping complex industrial environments such as power stations, electrical substations, manufacturing plants, pipeline networks, among others. LiDAR produces massive data with high precision which can be exploited for offsite inspection, measurement, assets management and virtual visits. LiDAR acquisition systems are constantly evolving, allowing faster, more accurate and more dense scans at increasingly lower prices. However, the processing of such massive stream of data is the bottleneck that prevents the expand of this technology. Automatic processing of 3D data is still an open problem in the scientific community. In fact, several current industrial applications are based on manual or semi-automatic processing. A typical application consists in creating a 3D model of an industrial plant from 3D LiDAR point clouds (Sternberg, Kersten, 2007, Gonzalez-Aguilera et al., 2012).

3D models of complex industrial environments are great information sources to improve site security and optimize industrial functioning. The creation of 3D models is generally carried out by specialized designers based on 2D plans, photos and 3D measurements of the industrial environment. In the case of electrical substations, access is dangerous due to electrical shock risk. Thus, a remote sensing technology such as LiDAR is an appropriate candidate to perform those 3D measurements. Moreover, several manufacturers provide precise 3D CAD models of their industrial equipment (insulators, electrical transformers, etc.), then the modeling processing can be defined as the process of creating a 3D virtual environment where CAD objects from a catalogue are positioned exactly as in the real world. In this context, 3D data offers great precision landmarks where modeling processing can be based on.

* Corresponding author

The main drawback of this pipeline is that 3D data manipulation and interpretation may not be trivial. Visual object identification and manual positioning may be painfully time-consuming and expensive. The use of automatic methods to process 3D point clouds will make possible the creation of 3D models in large scale applications.

This paper focuses on the automatic modeling of electrical substations based on a twofold framework: i) electrical elements are automatically segmented from the 3D point cloud (see Figure 1); ii) each segmented object is compared with every CAD model in a catalogue in order to determine its corresponding 3D model and its location in the environment (See Figure 2).

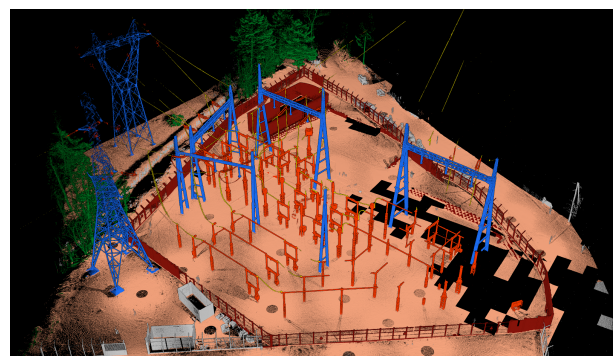


Figure 1. Classified 3D point cloud of an electrical substation. Each colour represents a different type of object.

The automatic processing of 3D data from electrical substations has been previously studied in the literature (Arastounia, Lichti, 2013, Arastounia, Lichti, 2014, Arastounia, Lichti, 2015, Wu et al., 2018). These approaches are based on segmentation followed by model retrieval, use prior knowledge and exploit the repetitive patterns of the environment.

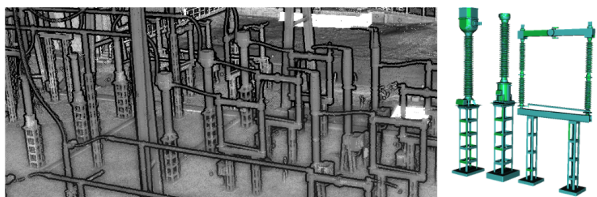


Figure 2. On left, a 3D point cloud of an electrical substation; on right, three CAD models expected to be found in the scene.

Modeling this kind of industrial environments is very interesting from both the application and the technical point of view. On one hand, from an application point of view, 3D modeling of electrical substations is fundamental to perform tasks such as asset management, inventory, virtual training, emergency planning, preventive maintenance, renovation planning, among others. On the other hand, from a technical point of view, this use case presents very interesting challenges such as: i) contrary to simulated environments, 3D point clouds from complex industrial environments present irregular local shapes, anisotropic density, occlusions and noise (Nan et al., 2012); ii) since electrical substations are composed of a few number of element categories (e.g. disconnectors, transformers, circuit breakers, insulators and wires) with low intra-category variability, object matching may be ambiguous; iii) since many electrical elements are nearly symmetric with respect to a vertical plane, correct positioning is a challenging task.

Our proposed method handles all these issues. Even if we chose to focus on the specific use case of electrical substations, the proposed method is highly generic and could be applied to a wide range of applications. Our contribution is two-fold:

- We design a novel segmentation algorithm based on a region growing process and edge detection. This method is proven to be robust to 3D point clouds of complex environments;
- We propose a matching method for recognizing and positioning CAD models onto 3D point clouds. This method is based on 3D features estimation on both segmented objects and CAD models.

The pipeline of our method is illustrated in Figure 3.

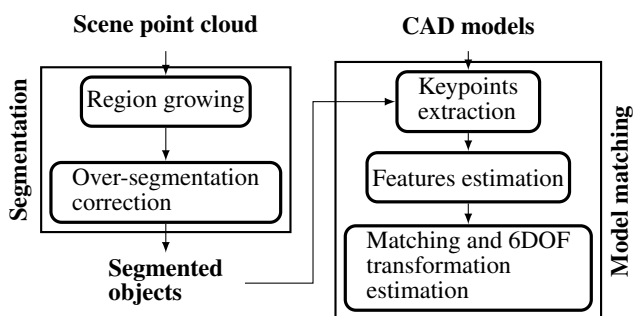


Figure 3. Overview of the proposed method

The rest of the paper is organized as follows. Section 2 presents a revision of the state-of-the art. Section 3 explains our two-steps method for modeling an industrial environment from a 3D point cloud and a catalogue of CAD models. Section 4 shows our experiments and results. Finally, Section 5 presents conclusions and perspectives.

2. RELATED WORK

Manual segmentation of 3D LiDAR data is not only expensive and time-consuming but also non repeatable in terms of quality since it depends on the fatigue of operators. Therefore, developing automatic segmentation methods is an active field in the scientific community. Several segmentation methods have been proposed using graph-cuts (Golovinskiy, Funk, 2009), image processing techniques (Serna, Marcotegui, 2014), multi-scale and dimensionality features (Hackel et al., 2016, Weinmann et al., 2015), deep learning (Thomas et al., 2019), among others. In the case of electrical substations, (Arastounia, Lichti, 2013) proposed a segmentation method based on the principal direction of points' distribution. This is done by forming and analyzing 9 different directions in 3D space. The assumption is that electrical components from a substation environment are made of vertical or horizontal parts and are mostly straight. While this observation can be made for most electrical components, it does not hold for cables (catenary shape) and very small parts (insulator rings). (Wu et al., 2018) proposed a reconstruction method of a sub-electrical station based on the detection of regularities of the scene. An electrical substation can indeed be considered as a collection of multiple instances of a few classes of vertical objects on a regular grid. While demonstrating promising results on the reconstruction of these environments, this methods can fail when disparities appears, such occlusions or some unexpected variations (additional cables, boxes, open/closed insulators). It also fails at objects of arbitrary orientations. In this work, we make the simpler assumption that the objects are made of geometrically simple parts, and disregards the orientation of those parts in space. We believe that this approach will not only be more robust to variation of the entities of the electrical substation, but also it could be more easily adapted to a variety of industrial environments. The goal of this proposed context-agnostic method is providing a collection of geometrically simple objects which can be easily clustered with some prior knowledge of the nature of the scene.

In the literature we find keypoint extractors that assume implicitly that the processed object is generally smooth and presents some protuberant areas, characteristic of the object that should be detected as key zones. Such processes are efficient for most common objects but fail as soon as the query object is everywhere irregular and salient, such as industrial pieces. (Gelfand et al., 2005) and (Rusu, 2009) introduced novel local descriptors that are used both for matching purposes and feature point extraction. In these methods, feature points are selected as points whose feature is rare with respect to all the features computed on the object. Consequently, they are not necessarily repeatable over various partial views of the same object, neither robust to segmentation errors or incomplete data. Besides, the extraction method of (Gelfand et al., 2005) keeps only a handful of keypoints, which shrinks ambiguity for the matching step but is not suitable for detecting keypoints on irregular and noisy point clouds from real industrial acquisitions. (Sipiran, Bustos, 2011) proposes a generalization of the Harris corner detection in 3D which can perform a relevant keypoint extraction on objects whose angles are rare, but is not adapted for crenelated objects such as ours. In (Zhong, 2009), authors extract keypoints in salient zones based on a criteria of maximum ratio between the eigen values of the scatter matrix on spherical neighbourhoods around each point. This method is efficient for extracting feature points on objects that are almost everywhere smooth, but is not reliable for objects that are volumetrical at every scale such as complex objects in industrial environments.

Feature-based matching relies often on local surface description of keypoints. The main assumption is that keypoints are located on regions which are locally unique over the objects. In the industrial context, key zones are rarely unique since industrial object have often planar or axial symmetries. (Tombari, 2010) proposes the SHOT descriptor based on histogram features for local surface description. (Rusu, 2009) introduced the FPFH descriptor for gathering angular information within a spherical neighbourhood, but even if keypoints are well positioned and descriptor are powerful, a local description would not solve reliably the orientation problems inherent to the geometry of industrial pieces. A multi-scale solution is considered by (Lu, 2014) for object recognition but is still limited to local description without sufficient contextual information.

All these considerations led us to present two contributions: first, a keypoint extractor able to capture key areas on industrial objects that are everywhere detailed; second, a multi-scale feature for describing these key zones both locally and globally, as chunks of an object rather than local surfaces without context.

3. METHODOLOGY

3.1 Segmentation

In this section, we propose a generic segmentation approach based on shape analysis and region growing. Since this segmentation process produces geometrically coherent regions, usually corresponding to over-segmented objects, an agglomeration process is also proposed in order to get objects of interest in 3D point clouds of electrical substations. This agglomeration process relies on context-based knowledge.

3.1.1 Shape analysis in the eigenvalues domain PCA (principal component analysis) aims at transforming a data set of a given dimension to a set of smaller dimension. PCA can be computed by the covariance method. Eigenvectors are extracted from the matrix \mathbf{V} that diagonalizes the covariance matrix \mathbf{C} , so that $\mathbf{V}^{-1}\mathbf{C}\mathbf{V} = \mathbf{D}$, \mathbf{D} being a diagonal matrix that encodes the eigen values.

While eigen vectors give us information about the spatial orientation of a set of points (tangents and normals), eigen values describes the overall shape (length, width and thickness). In this work, we propose edge detection and segmentation methods based on the analysis of eigenvalues. Our main assumption is that the intersection between two surfaces produces an edge which can be located by analyzing the eigenvalues variation at a local scale.

Let us denote by \mathcal{P} a set of 3D points and $\lambda_1, \lambda_2, \lambda_3$ the eigen values of the covariance matrix of \mathcal{P} , sorted in ascending order. Let us consider the normalized vector, denoted by $\hat{\lambda}(\mathcal{P})$, of the following vector $(\lambda_1, \lambda_2, \lambda_3)$. The magnitude of each eigenvalues depends on the scale. Thus, the normalization of this vector gives overall proportions in each dimension.

As $\hat{\lambda}(\mathcal{P})$ is normalized, the point $(\lambda_1, \lambda_2, \lambda_3)$ belongs to the unit sphere. It can be transformed in polar coordinates (ϕ, θ, r) which can be projected into a 2D space since $r = 1$. Thus, ϕ and θ for a given set \mathcal{P} of eigenvalues $(\lambda_1, \lambda_2, \lambda_3)$ are:

$$\phi = \arctan \frac{\lambda_2}{\lambda_1}, \quad \theta = \arctan \frac{\sqrt{\lambda_1^2 + \lambda_2^2}}{\lambda_3} \quad (1)$$

The 2D domain \mathcal{G} of $\hat{\lambda}$ is bounded by the triangle given by the following three top points L, P, V , as shown in Figure

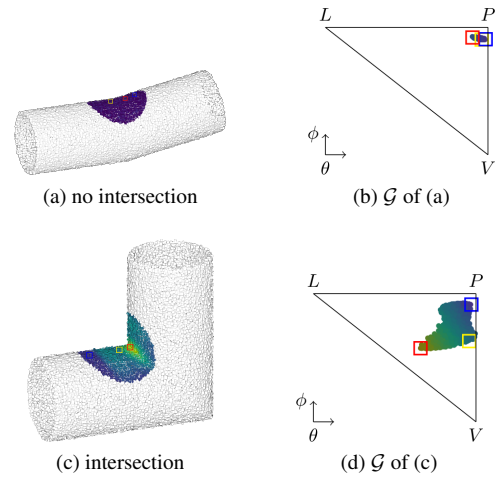


Figure 4. In (a), a cylinder slightly bent at its middle at a 10° angle, generated with an added Gaussian noise. In (b), the projection of the coloured points of (a) in \mathcal{G} . In (c), two cylinders intersecting at a 90° angle, generated with an added Gaussian noise. In (d), the projection of the coloured point of (c) in \mathcal{G} . In (a) and (c), coloured points are weighted by their δ value from purple to yellow. In all figures, the position of the point of interest p is shown in a yellow square, and the least and most stable points are shown in red and blue respectively.

4b. One can notice that $\hat{\lambda}(\mathcal{P}) = L$ would be obtained if \mathcal{P} is a set of points evenly distributed along a line. Respectively, P would be obtained from a set of points evenly distributed along a plane and V from an evenly distributed set of 3D points. As $\lambda_1, \lambda_2, \lambda_3$ are sorted and their vector is normalized, the extremities of the triangle can intuitively be assessed as $(1, 0, 0)$, $(1/\sqrt{2}, 1/\sqrt{2}, 0)$ and $(1/\sqrt{3}, 1/\sqrt{3}, 1/\sqrt{3})$. Thus, their respective projection in \mathcal{G} are $(0, \pi/2)$, $(\pi/4, \pi/2)$ and $(\pi/4, \arctan \sqrt{2})$.

In this context, we can define the *shape instability* of a point with respect to its location inside the triangle LPV . A point with low stability will be located far from vertices L, P or V , the most unstable point being located at the circumcenter of the triangle. Those points are very interesting since they represent transition zones between different surfaces.

Let us denote N_p the set of points neighbouring the point p given some neighbourhood relationship. In this work, we only consider the case of radial neighbourhoods. For each point p , we compute $\hat{\lambda}(N_p)$.

We define the *shape instability* $\delta(p)$ as the minimal distance in \mathcal{G} from $\hat{\lambda}(N_p)$ to L, P or V :

$$\delta(p) = \min_{s \in \{L, P, V\}} \|\hat{\lambda}(N_p) - s\| \quad (2)$$

Now, let define the most stable point s_p and the less stable point i_p in neighbourhood N_p respectively as:

$$s_p = \arg \min_{x \in N_p} \delta(x), \quad i_p = \arg \max_{x \in N_p} \delta(x) \quad (3)$$

The span $\rho(p)$ of a point p is then defined as the difference between the instability of s_p and i_p for a given p :

$$\rho(p) = \delta(s_p) - \delta(i_p) \quad (4)$$

The span is correlated to the presence of a transition zone in

the 3D space. On the one hand, Figure 4a illustrates a cylinder slightly bent at its middle at a 10° angle. This slight transition is confirmed by a low ρ value (low dispersion of blue points) in Figure 4b. On the other hand, Figure 4c presents two cylinders intersecting at a 70° angle. This transition is observed in the \mathcal{G} domain in Figure 4d where ρ value is higher.

In Figure 5, the span ρ of a point cloud with various objects (steel beams, cables, boxes, ground) is shown. We observe a high value of ρ when different surfaces intersect and a low value otherwise.

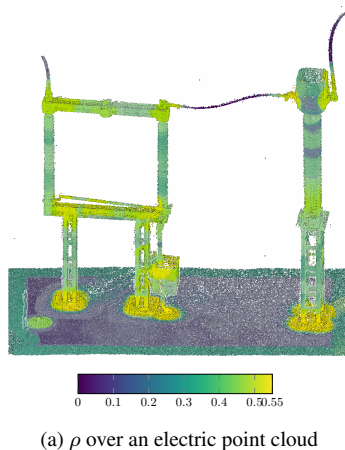


Figure 5. 3D point cloud from an electrical substation.

Instability values are shown on the given colour scale. We observe high values of ρ on edges and low values on flat zones.

3.1.2 Region growing We propose a region growing process based on the previously introduced definitions of shape instability and span. Region growing process starts from carefully selected seeds and propagates to adjacent points given some region membership criterion.

In our work, those seeds are points whose shape instabilities are low. The membership of two adjacent points to the same region is constrained to the presence of instability in their neighbourhood. We will define this region membership criterion based on a set of points called edges, whose shape instabilities are high. Thus, we give the following definitions of seeds and edges in our region growing process. We define the set of seeds S as the set of stable points whose span is lower than a given threshold t_0 :

$$S = \{s, \rho(s) < t_0\} \quad (5)$$

Then, we define the set of edges E as the set of unstable points whose shape instability is greater than a given threshold t_1 :

$$E = \{e, \delta(e) > t_1\} \quad (6)$$

Our region membership criterion is based on the presence of edges. We will say that two adjacent points are from the same region if both points are closer to each others than to any edge point. Our proposed method is similar to image segmentation methods based on region growing. It is an iterative process and the goal is to partition a point cloud into regions $\{R\}$. At a given iteration, a seed s is randomly selected and a new region $R = \{s\}$ is grown. R is grown from each neighbour of points from R that satisfy the proposed region membership criterion. A new seed is selected and a new region is grown when no more points are added to R .

The results of this region growing method are illustrated in Figure 6. In Figure 6a, one can observe that our process selects

seeds from regions with no prominent features. Oppositely, edge points are all picked from regions of intersecting objects. In Figure 6b, the result of the region growing is shown and one can see the resulting regions are geometrically coherent, despite having a slow curve (as cables) or small shape disturbances (due to miscalibration of the acquisition).

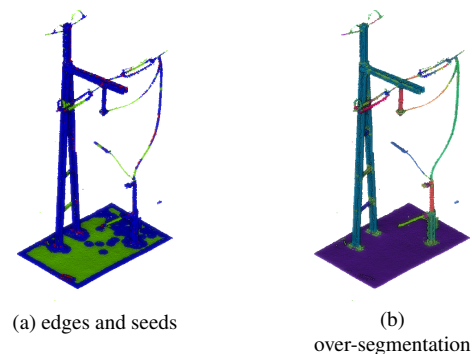


Figure 6. Point selected from the process described in section 3.1.2 in (a). Seeds are shown in green, while edges are shown in red. Over-segmentation generated by the proposed region growing algorithm in (b), colored with random colours.

Objects correspond to geometrically coherent regions. One can see that cables, ground patches, poles and objects of primitive forms are correctly labelled. More complex structures such as steel beam are regrouped into one object.

3.1.3 Over-segmentation correction The previous region growing process produces geometrically coherent regions. In general, these regions correspond to over-segmented objects since it is only based on shape criteria. Depending on the application, an agglomeration process is needed in order to get objects of interest. In the case of electrical substations, this agglomeration process is straightforward.

First, the ground is usually flat and can be extracted as the largest connected component from the region growing result. Second, cables are the most elongated objects of the over-segmentation. They can be detected by the averaged local linearity measure over the object. A linearity threshold, set by visual inspection, is then used. Finally, we suppose that the remaining points are from electrical elements, as substations are rarely occupied with any other type of object. In a substation environment, electrical elements are always a few meters apart in order to prevent short circuit. Once ground and cables are removed, a simple neighbourhood search with a carefully selected radius is able to reconstruct whole electric elements.

3.2 Feature based matching

The results of the latter segmentation method provides object hypothesis that may have a CAD counterpart within a catalogue of modeled object expected to be found in the scene. Our work brings a twofold contribution for achieving feature-based matching between these segments extracted from real acquisition data and reconstructed models. We firstly designed a generic keypoint extractor that is suited for dealing with shape auto-similarity and captures precisely the structural details that are characteristic of the object's orientation. Secondly, we introduce a novel multi-scale descriptor, aimed at describing the local neighbourhood of each keypoint as well as gathering information about its global localization over the whole object in an efficient way. In order to have a unified method for extracting keypoints and features on both CAD models and 3D point clouds, we work on point cloud versions of each CAD model in

the catalogue obtained by uniform sub-sampling over the faces of the mesh. Sampling resolution is roughly the same as the 3D scene. Once features are computed on both CAD models and 3D scene segments, we perform a brute-force matching and estimate the 6 DOF transformation that best fits the CAD models on each segment and compute a score for identifying the best corresponding CAD model.

3.2.1 Keypoints extraction For our use case, keypoints must be located in structural zones of the object that are uncommon and salient. Industrial pieces have details everywhere, thus we designed a keypoint extractor based on a filtered dimensionality analysis. We compute the PCA of a spherical neighbourhood \mathcal{N}_i around each point of the query object at a characteristic radius r_σ of the object's geometry, and compute the a_{3D} coefficient introduced in (Demantké, 2012), that measures the saliency σ_i of the query point p_i

$$\sigma_i = \frac{\sqrt{\lambda_3}}{\sqrt{\lambda_1} + \sqrt{\lambda_2} + \sqrt{\lambda_3}} \quad (7)$$

where $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$ are the eigen values of the covariance matrix. Note that the eigen values of the neighbourhood characterize the expansion of the point cloud along its principal axis; thus, the greater the σ_i , the more volumetric the neighbourhood. Since key zones on objects are mainly volumetric (such as corners, junctions and protuberances), relevant keypoints rely in zones of high saliency. On Figure 7 we show a specific part of an industrial piece with the saliency as scalar field. On that Figure, areas 1 and 2 would be relevant to extract as they are structural and characteristic of the object's orientation, whereas area 3 is fully auto-similar and would lead to ambiguity in the object's orientation in later processes. On such a complex object with milling and bars, the more salient points are not necessarily on key areas, thus applying a rigid threshold for extracting relevant zones would fail. This is shown on Figure 7 where no trade-off is affordable for keeping 1 and 2 only.

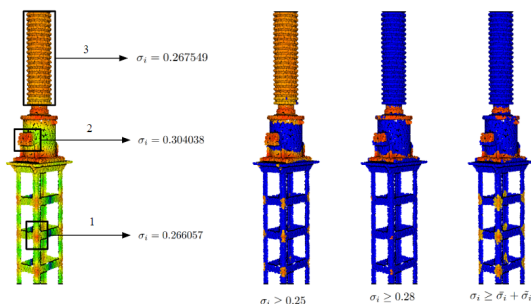


Figure 7. Structural chunk of an industrial object coloured by saliency and remaining points for different extraction methods.

However, points that are locally excessively salient compared to their neighbours are structural, even if they are not absolutely salient. Therefore, for each point p_i , we build a distribution on the saliency of neighbouring points and estimate the average saliency and standard deviation in \mathcal{N}_i

$$\bar{\sigma}_i = \frac{1}{|\mathcal{N}_i|} \sum_{k \in \mathcal{N}_i} \sigma_k, \quad \tilde{\sigma}_i = \sqrt{\frac{1}{|\mathcal{N}_i|} \sum_{k \in \mathcal{N}_i} (\sigma_k - \bar{\sigma}_i)^2} \quad (8)$$

We extract the query point p_i as a keypoint if $\frac{\tilde{\sigma}_i}{\bar{\sigma}_i} > \rho_\sigma$ where ρ_σ is a rigid threshold (typically 5% is sufficient) for avoiding

picking points on homogeneously salient zones such as zone 3 on Figure 7, and if $\sigma_i > \bar{\sigma}_i + \tilde{\sigma}_i$. Thus, points in structural zones of objects such as junctions, corners and protuberances are captured in patches.

Rather than searching for isolated and very accurately located keypoints such as (Gelfand et al., 2005), we keep dense patches of key zones, which is much more reliable for real and noisy data. A qualitative comparison with other keypoint extractors is presented in Section 4.2.

The second step of the keypoints extraction consists in selecting points sparsely and regularly on the shape. This step is not compulsory but it can help aligning coarsely the point clouds for the further RANSAC registration, while the key patches extracted with our method contribute to orienting correctly the point clouds.

3.2.2 Features extraction The next step of our method consists in describing the keypoints through 3D features. The SHOT and PFH descriptors introduced by (Tombari, 2010) and (Rusu, 2008) respectively are efficient for shape description, but they do not capture any contextual information, and therefore are not so reliable for registering auto-similar objects such as those we can find in industrial environments. We designed a hybrid descriptor inspired by the SPFH descriptor introduced in (Rusu, 2009) that gathers local and contextual information. To capture information present in neighbourhoods of radii $\{r_i\}_{1 \leq i \leq p}$ of a keypoint, we firstly generate p sub-sampled versions $\{\mathcal{P}_i\}_{1 \leq i \leq p}$ of the input cloud with spatial resolution $\beta_i \times r_i$ ($\beta_i \in [0, 1]$). The parameter β_i controls the density of points in a neighbourhood of radius r_i searched in \mathcal{P}_i . For instance, setting $\beta_i = 0.1$ for a searching radius $r_i = 0.5m$ will result in a neighbourhood where the distance between each point is at least $\beta_i \times r_i = 5cm$, which is sufficiently dense to describe local geometry in a neighbourhood of radius r_i . At scale $k \in \llbracket 1, p \rrbracket$, we compute for each keypoint the Local Reference Frame introduced in (Tombari, 2010) for circumventing the normal's orientation issue and having rotation invariant descriptors, and then compute the SPFH descriptor of a spherical neighbourhood of radius r_k searched in \mathcal{P}_k . Unlike the original authors of this feature, we do not decorrelate the three computed angles for simplifying the feature's expression, because it actually brings ambiguity and reduces significantly the descriptive power of the SPFH descriptor. On Figure 8, we have connected by coloured lines the points used to compute the pair features. The green lines for instance connect the query keypoint to points of its spherical neighbourhood of radius $r_3 = 2.50m$, showing that a third of the feature's information is exclusively dedicated to gathering contextual information. Repeating the operation at every desired scale, we finally stack the SPFH descriptors computed at each scale in one single feature vector.

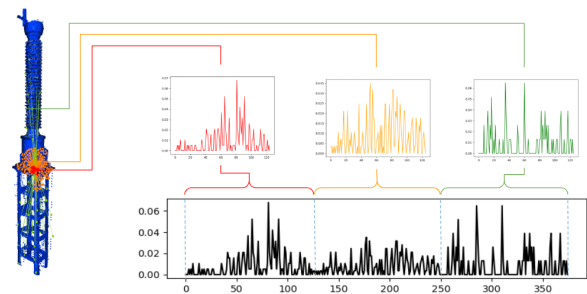


Figure 8. Multi scale neighbours for computing our feature on a keypoint with three radii and 5^3 bins per histogram.

The resulting features are descriptive of the local geometry around the keypoints thanks to SPFH with short radius and also contains contextual information gathered into the histograms of long searching radius, allowing highly discriminating power and non ambiguous matching: in our use case, a local description of the keypoints, as structural as they are, would not be completely reliable because of the auto-similarity aspect of the objects.

3.2.3 Feature matching and pose estimation

Going through each segment, we match its features with those of each CAD model by brute-force matching using χ_2 distance, better fitted for histogram comparison than L_2 distance (see Figure 9a), using a weighting on the different scales. If the feature $f_i = (f_i^1, \dots, f_i^N)$ was computed over N scales, we compute the weighted distance with f_j as follow :

$$d(f_i, f_j) = \sum_{k=1}^N \frac{1}{k} d_{\chi_2}(f_i^k, f_j^k) \quad (9)$$

for giving priority to local description while adding long range matching constraints. As we kept patches of keypoints, we cannot filter the matches as in (Lowe, 2004) which excludes matches that are not significantly better than the second best match in the feature space, under the assumption that features are unique. Instead, we only keep matches with distance to the nearest neighbour lower than $\mu_{nn} + \sigma_{nn}$ where μ_{nn} and σ_{nn} are the average distance to the nearest neighbour and standard deviation on the object. We estimate the 6 DOF transformation that best fits the query CAD model on the segment by RANSAC (Figure 9b) on the pairs from the matching step and finally apply a point-to-plane ICP registration (Figure 9c). The overall rigid transformation T_{CAD} is applied to the CAD object. The object shown is fully symmetric except on the area zoomed on Figure 9d : our keypoint extractor caught this very specific zone allowing the right pose estimation. We compute the intersection

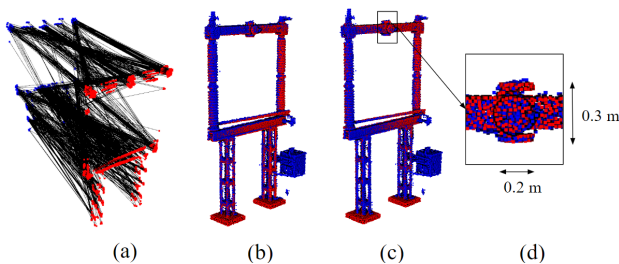


Figure 9. Steps for estimation the rigid transformation between a CAD's point cloud (red) and a 3D scene segment (blue) by feature matching and RANSAC. On this example the keypoints were extracted only with our saliency analysis.

over union (IoU) as our overlap measure : for each point of the 3D segment \mathcal{P}_{TLS} , we search for its nearest neighbour in the CAD cloud \mathcal{P}_{CAD} within a distance $r_{overlap}$. The intersection I is defined as the number of point having at least one neighbour, the union U is defined by $U = |\mathcal{P}_{CAD}| + |\mathcal{P}_{TLS}| - I$, and the overlap is given by $IoU = \frac{I}{U}$. The parameter $r_{overlap}$ defines the tolerance on the IoU . We set a rigid threshold on the minimum IoU for being a good match, because some segments may not have their CAD counterpart in the catalogue and we do not want to assign them to any 3D model. The CAD model that maximizes the IoU is considered as the right model and is positioned in the scene by applying the corresponding T_{CAD} .

4. EXPERIMENTS

4.1 Evaluation of the segmentation

We compare our method with two region growing algorithms. We aim at demonstrating the advantages of our approach based on eigenvalues compared to approaches based on eigenvectors and dimensionality features as defined in (Demantké, 2012). In both cases a PCA is computed at a local scale on each point using a radial neighbourhood.

The first region growing method exploits the tangent, the main component of the eigen vectors of this PCA. Seeds are selected from the most linear points. Two adjacent points are from the same region if the angle between their corresponding tangents are smaller than a given threshold.

The second region growing method is similar to the first but the linearity is replaced by the planarity for the seeds selection, and the tangent by the normal. We try by visual inspection of the results, to set the threshold of the previous region growing methods as to obtain the best possible examples. As for our method, we have chosen the thresholds values for the seeds and edges as defined in section 3.1.2 to $t_0 = 0.25$ and $t_1 = 0.48$. We obtain the results of Figure 10. The linear region growing in Figure 10a is able to extract every cable and two insulators chains but performs poorly on electrical elements. It is also unable to extract the ground. The planar region-growing in Figure 10b performs poorly overall, as there is no plane except for the ground. The ground and the bigger structure are not split as expected, as the edge between the two features a low slope. Selecting a different radius for the radial neighbourhoods may hold different results. Only our method in Figure 10c is able to extract all cables. Electrical structures are over-segmented, but their parts are geometrically coherent as expected. The chains of rings at the top are slightly over-segmented, but it is a complex structure and the two top chains are badly acquired in this particular case.

In Figure 10d, the result of the over-segmentation correction is shown, each object colourized with a random colour. It can be compared to the ground truth in Figure 10e. The ground is correctly extracted, along with all the cables. Each electrical object is correctly extracted along with the three bigger structures. The scanner positions have been extracted as circular planar regions on ground. While this might be problematic in other applications, it does not disrupt the object matching process as it does not resemble to any other entity in the catalogue.

4.2 Qualitative keypoint extractors comparison

For comparing our keypoint extraction method to well known methods, we used the (Rusu, Cousins, 2011) PCL implementation of the (Zhong, 2009) ISS and (Sipiran, Bustos, 2011) Harris3D keypoint extractors with the most adapted parameters, found qualitatively by trial and error. We also implemented the persistent feature point extraction of (Rusu, 2008) based on prior FPFH estimation. Results are shown on Figure 11. For each experiment, the same parameters were applied for extracting keypoints on all point clouds.

In our use case, the Harris3D extractor detects keypoints on different zones, depending on the noise level and of the cloud density. The ISS extractor catches salient points everywhere on the structure, so it is nearly equivalent to a uniform sampling for such objects. The persistent FPFH points are relevant on the CAD object but are much disturbed by real measurement noise such as on object TLS 1, and the keypoints are not repeatable on the "crowned" object which was sub-segmented in

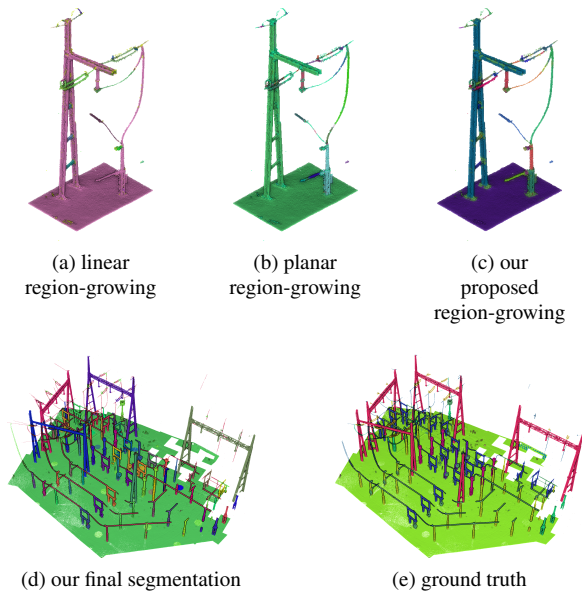


Figure 10. Comparison of two methods of region growing based respectively on the tangents (a) and on the normals (b) of the PCA and of the proposed region growing (c). Only our method is able to correctly differentiate cables, ground and steel beams. Comparison of the proposed segmentation (d) and of the ground truth (e).

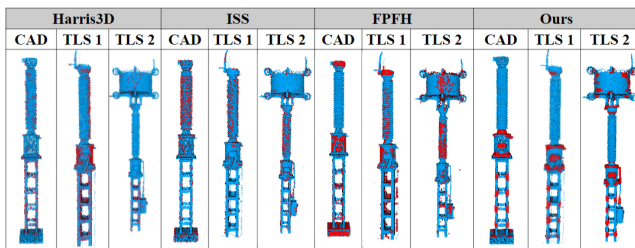


Figure 11. Qualitative comparison of keypoint extractors against ours on industrial pieces. Keypoints are represented in red.

TLS 2. Consequently they are not necessarily located on same areas on TLS segments and CAD objects, leading to inaccurate matching. Finally, our keypoint extractor is robust to noise, under- and over-segmentation, and offers the best located key-points reliably: we capture each area that is relevant for feature matching with high repeatability whether the piece is extracted from the 3D scene or the CAD model.

4.3 Feature-based matching evaluation

We dispose of a ground truth on an electrical substation near to Castellane in France. Each CAD model from a catalogue was manually positioned on the 3D scene. Thus we can evaluate the recognition rate as well as the accuracy of the positioning on the 6 DOF transformation. For these experimentation, we chose to extract salient keypoints with a radius $r_s = 0.30m$, completed with a coarse sampling of radius 15 cm. We describe keypoints with two radii $r \in \{0.30m, 2.0m\}$ for covering short and long range, given that the object's dimensions are roughly $1m \times 1m \times 5m$. Each SPFH descriptor is built with 5^3 bins, resulting in a 250 dimensional stacked feature vector. In our experiments, the catalogue includes 3 CAD objects, and 81 segments were identified on the scene. The threshold on IoU is set to 40%. For evaluating quantitatively the accuracy of our positioning method, we compare the computed

transformations $\{T_i\}$ to the ground truth $\{\tilde{T}_i\}$ by evaluating $\log(T_i^{-1}\tilde{T}_i) = (d\omega, du)^T \in \mathbb{R}^6$ for each object, where $d\omega$ and du are respectively the errors vectors in rotation and translation, and we compute the translation error $\zeta_t = \|du\|_2$ and rotation error $\zeta_\omega = \|d\omega\|_2$. For dealing with axial symmetry, we withdraw π to angular error greater than 90° but we count the number of flipped pieces. We have confronted our method for this use case to the well-known local methods FPFH (Rusu, 2009), SHOT (Tombari, 2010), and to a single scale SPFH feature. We show the betterment of our keypoint extraction method by carrying the matching experiments with both uniform sampling (denoted by u on Table 1) of radius $r = 15$ cm, and with our keypoints, for the same description methods. We compare the recognition rate τ , maximum errors in translation ζ_t^m and rotation ζ_ω^m and the percentage of flipped pieces $\tau_{flipped}$. Runtimes are between 10 and 20 minutes for all experiments, which is not limiting for this use case. In particular, the multi-scaling method does not induce significant runtime increase.

Method	τ	ζ_t^m (cm)	ζ_ω^m ($^\circ$)	$\tau_{flipped}$
SPFH _u	80%	947.1	87	38%
SPFH	90%	29.7	8	21%
FPFH _u	83%	27.2	79	30%
FPFH	77%	30.4	84	17%
SHOT _u	96%	30.9	89	30%
SHOT	96%	21.8	84	10%
Ours _u	100%	45.6	77	39%
Ours	100%	11.6	4.0	7%

Table 1. Quantitative comparison of our descriptor against state of the art descriptors.

As presented in Table 1, our keypoint selection method increases with no ambiguity the number of rightly oriented pieces. Moreover, our method increases for most description method the recognition rate. We see that our SPFH implementation outperforms the PCL FPFH descriptor because, as mentioned in Section 3.2.2, we do not decorrelate the computed angles. Our method proves to be the most reliable for object recognition, as we reach 100% recognition rate, and have the lowest errors in translation. Indeed, our features are more efficient for describing keypoints as being part of an object rather than the local descriptor chosen, thus global shapes are better matched. The maximum errors in rotation for SHOT and FPFH are due to the fact that the overlap reached was sufficient for matching segments that were incomplete because of data inherent issues, but local description failed at managing nearly axial symmetry. With our method, incomplete pieces are at worst flipped on their quasi-symmetry plan, but no significant rotation error occurs. Our experiments show that allying a simple feature such as SPFH with a multi-scale method and an accurate keypoint extraction can compete state of the art description methods such as SHOT.

5. CONCLUSIONS

In this work, we proposed a general framework for modeling complex industrial environments from 3D point clouds and a catalogue of 3D CAD models. Our contribution is two-fold. The first step consists in a robust segmentation of all objects within the 3D point cloud. This segmentation method is based on a shape analysis in the eigenvalues domain. Segmentation result is a set of geometrically coherent regions which can be agglomerated according to some application-dependent prior knowledge. The second step queries the catalogue and evaluates the best correspondences between segmented objects and

CAD models. For this purpose, salient key areas are determined and described by a multi-scale feature on both CAD models and 3D segments. This multi-scale descriptor gathers local and contextual information, which allows managing the high degree of auto-similarity of industrial pieces. Extracted features are thus matched and the best fitting transformation is estimated for positioning automatically the corresponding CAD model directly on the 3D scene, as it is shown in Figure 12.

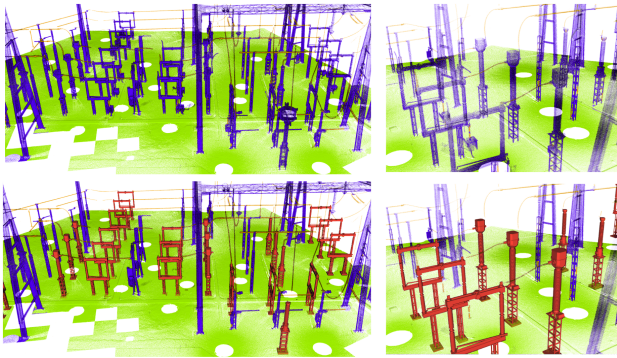


Figure 12. On top, the 3D scene where the ground (green), the wires (yellow) and other objects (purple) were segmented. On bottom, objects having their CAD counterpart in the catalogue were replaced by our modeling method.

Although we presented the specific use case of electrical substations, the proposed method is highly generic and could be applied to a wide range of applications. Our multi-scale method gathers information at short and long range from any geometric descriptor, insuring robust registration for complex objects. In the future, we plan to use more efficient local descriptors and decline them into multi-scale descriptors for reaching even more robust results on larger datasets. Other use cases such as pipelines and indoor industrial environments will be studied.

ACKNOWLEDGEMENTS

All data courtesy of RTE, the Electricity Transmission System Operator of France. In particular, the authors gratefully thank Hosni Bchir, David Couallier and their collaborators within the framework of RTE-JUMP project for providing data and sharing their knowledge on electrical substation environments.

REFERENCES

Arastounia, M., Lichti, D., 2013. Automatic extraction of insulators from 3D LiDAR data of an electrical substation. II-5/W2.

Arastounia, M., Lichti, D., 2014. Segmentation of planar surfaces in LiDAR point clouds of an electrical substation by exploring the structure of points neighbourhood. 40, 55–62.

Arastounia, M., Lichti, D., 2015. Automatic Object Extraction from Electrical Substation Point Clouds. *Remote Sensing*, 7(11), 15605–15629.

Demantké, J., 2012. Dimensionality based scale selection in 3D lidar point clouds. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 97–102.

Gelfand, N., Mitra, N., Guibas, L., Pottmann, H., 2005. Robust global registration. 197–207.

Golovinskiy, A., Funk, T., 2009. Min-cut based segmentation of point clouds. *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, IEEE, Kyoto, Japan, 39–46.

Gonzalez-Aguilera, D., Del Pozo, S., Lopez, G., Rodriguez-Gonzalvez, P., 2012. From point cloud to CAD models: Laser and optics geotechnology for the design of electrical substations. *Optics & Laser Technology*, 44(5), 1384–1392.

Hackel, T., Wegner, J. D., Schindler, K., 2016. Fast semantic segmentation of 3D point clouds with strongly varying density. *ISPRS annals of the photogrammetry, remote sensing and spatial information sciences*, 3(3), 177–184.

Lowe, D., 2004. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2), 3212–3217.

Lu, M., 2014. Recognizing Objects in 3D Point Clouds with Multi-Scale Local Features. *Sensors*, 14(12).

Nan, L., Xie, K., Sharf, A., 2012. A search-classify approach for cluttered indoor scene understanding. *ACM Transactions on Graphics*, 31(6), 137–147.

Rusu, R., 2008. Aligning point cloud views using persistent feature histograms. *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3212–3217.

Rusu, R., 2009. Fast point feature histograms (FPFH) for 3D registration. *Robotics and Automation, 2009. ICRA'09*, 3212–3217.

Rusu, R., Cousins, S., 2011. 3D is here: Point Cloud Library (PCL). *IEEE International Conference on Robotics and Automation (ICRA)*.

Serna, A., Marcotegui, B., 2014. Detection, segmentation and classification of 3D urban objects using mathematical morphology and supervised learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 93, 243–255.

Sipiran, I., Bustos, B., 2011. Harris 3D: a robust extension of the Harris operator for interest point detection on 3D meshes. *The Visual Computer*, 27(11), 963–976.

Sternberg, H., Kersten, T., 2007. Comparison of terrestrial laser scanning systems in industrial as-built-documentation applications. *Optical 3D Measurement Techniques VIII*, 1, 389–397.

Thomas, H., Qi, C. R., Deschaud, J.-E., Marcotegui, B., Goulette, F., Guibas, L. J., 2019. KPConv: Flexible and Deformable Convolution for Point Clouds. *Proceedings of the IEEE International Conference on Computer Vision*.

Tombari, F., 2010. Unique signatures of histograms for local surface description. *ECCV*, 356–369.

Weinmann, M., Jutzi, B., Hinz, S., Mallet, C., 2015. Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. *ISPRS Journal of Photogrammetry and Remote Sensing*, 105, 286–304.

Wu, Q., Yang, H., Wei, M., Remil, O., Wang, B., Wang, J., 2018. Automatic 3D reconstruction of electrical substation scene from LiDAR point cloud. *ISPRS Journal of Photogrammetry and Remote Sensing*, 143, 57–71.

Zhong, Y., 2009. Intrinsic shape signatures: A shape descriptor for 3D object recognition. *IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, 689–696.