Using Context to Create Semantic 3D Models of Indoor Environments

Xuehan Xiong
xxiong@andrew.cmu.edu
Daniel Huber
dhuber@cs.cmu.edu

The Robotics Institute,
Carnegie Mellon University

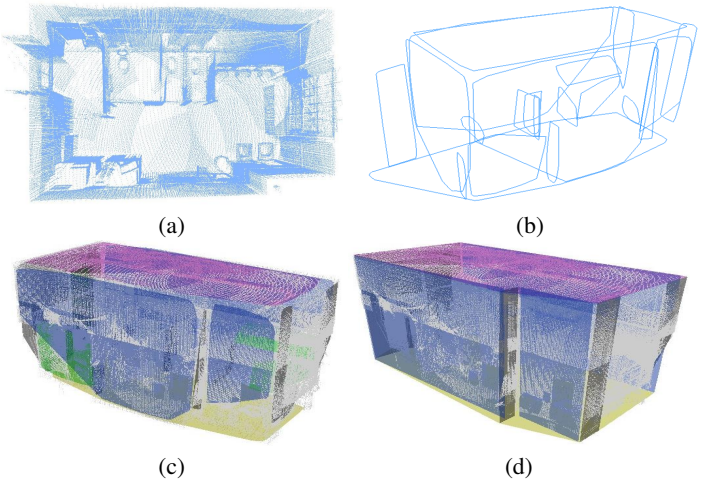

Figure 1: Algorithm overview. (a) The input point cloud is encoded in a voxel data structure; (b) planar patches are detected and modeled; (c) planar patches are classified using local features as well as contextual relationships between patches (magenta = ceilings, yellow = floors, blue = walls, green = clutter); and (d) patch boundaries are re-estimated and clutter surfaces are removed.

In this paper, we present a novel approach to transform the raw 3D point cloud meansured from laser scanners into a semantic building model. In industry practice, this transformation is currently conducted manually – a laborious, time-consuming, and error-prone process. Methods to reliably automate this transformation would be a tremendous benefit to the field and would likely speed the adoption of laser scanning methods for surveying the as-built conditions of facilities.

The primary challenge in the classification of structural components is distinguishing relevant objects from clutter. The distinction can be difficult or impossible if objects are considered in isolation, especially for highly occluded data sets. In our situation, contextual information could help to recognize objects of interest and to distinguish them from clutter through the relationships between the target surface and other nearby surfaces. For example, if a surface is bounded on the sides by walls and is adjacent to a floor on the bottom and a ceiling on the top, it is more likely to be a wall than clutter, independently of the shape or size of that surface. In this way, the interpretation of multiple surfaces can mutually support one another to create a globally consistent labeling.

The concept of using context to model building interiors has been studied by other researchers [3, 4]. Most previous approaches rely on hand-coded rules, such as "walls are vertical and meet at 90° angles with floors. Such rules are usually brittle and break down when faced with noisy measurements or new environments. Our algorithm automatically learns which rules are important based on training data so that more discriminative features are weighted more during the classification.

Fig. 1 shows the major steps of our approach. We begin with input data of a point cloud representing a room or a collection of rooms. First, we encode the point cloud into a voxel structure to minimize the variation in point density throughout the data (fig. 1 (a)). Next, we detect planar patches by grouping neighboring points together using a region-growing method similar to [5]. We model the patch boundaries using its convex hull (fig. 1 (b)). We use these planar patches as the input to our classification algorithm. The algorithm uses contextual relationships as well as local features to label the patches according to functional categories, such as *wall*, *floor*, *ceiling*, and *clutter* (fig. 1 (c)). Finally, we remove clutter patches from the scene and re-estimate the patch boundaries by intersecting adjacent components (fig. 1 (d)).

In classifying planar patches, we use a Conditional Random Field (CRF) model. A graph $G = (V, E)$ is produced by connecting each planar patch with its 4 nearest neighbors. Nearest neighbors are found by measuring the minimum Euclidean distance between patches. Our goal is to find a set of labels that maximizes conditional likelihood given in eq. (1).

$$P(\mathbf{y}|\mathbf{x},\theta,\omega) = \frac{1}{Z(\mathbf{x},\theta,\omega)} exp\{\sum_{i\in V} A(y_i,x_i) + \sum_{(i,j)\in E} I(y_i,y_j,x_i,x_j)\} \quad (1)$$

$Z(\mathbf{x},\theta,\omega)$ is known as the partition function. The local feature function, $A(y_i,x_i)$, encapsulates knowledge about the patches in isolation. The contextual feature function, $I(y_i,y_j,x_i,x_j)$, contains the information about a patch's neighborhood configuration.

The local feature function $A(y_i,x_i)$ is modeled as $\alpha\sum_{k=1}^{K}\delta(y_i=k)\theta_k^T g(x_i)$ where $g(x_i)$ is a vector of features derived from patch x_i (such as area and orientation) plus a bias term of 1. The parameter α controls the relative weight of local feature function versus the contextual feature function, and it is chosen during cross-validation. K is the number of class labels any node can take on, and δ is the Kronecker delta function. The parameter vector θ_k allows a class-specific weighting for each local feature.

For modeling contextual features, we define R pairwise relations between two planar patches. We encode a patch's neighborhood configuration with a matrix H, where the entry in the k^{th} row and r^{th} column, $h_{k,r}(y_i,x_i,x_j)$ is 1 if $y_i = k$ and r^{th} relationship is satisfied between x_i and x_j. Otherwise, $h_{k,r} = 0$, where $r \in \{1,...,R\}$. Then H is converted into a vector h by concatenating its columns. The contextual feature weighting vector ω_k is analogous to θ_k above. The contextual feature function $I(y_i,y_j,x_i,x_j)$ follows as $\sum_{k=1}^{K}\delta(y_i=k)\omega_k^T h(y_j,x_i,x_j) + \delta(y_j = k)\omega_k^T h(y_i,x_i,x_j)$

In our work, we choose to learn the parameters by maximizing the pseudo-likelihood [1] due to its simplicity and intractability of the original objective function (eq. 1). The log pseudo-likelihood l_{PL} for our model can be written as $\sum_i(-\ln Z_i + A(y_i,x_i) + \sum_{j\in N_i} I(y_i,y_j,x_i,x_j))$ where $Z_i = \sum_{k=1}^{K} exp\{A(k,x_i) + \sum_{j\in N_i} I(k,y_j,x_i,x_j)\}$, and N_i gives the indices of x_i's neighbors in G. From the properties of convex functions, we can derive that the log pseudo-likelihood function is concave, which indicates a global optimal solution. Optimal parameters θ and ω are derived from gradient ascent.

During inference, we use a local search algorithm, Iterated Conditional Modes (ICM) [2]. In each iteration, ICM maximizes the local conditional (posterior) likelihood. This procedure iterates until convergence. This is a reasonable alternate objective to optimize since the parameters θ and ω are learned from maximizing the product of local conditional likelihood.

We compare the results of our context-based CRF algorithm with a context-free method based on L_2 norm regularized Logistic Regression (RLR) performing on real world data. We find that using certain contextual information along with local features improves the classification rate from 84% to 90%.

[1] J. Besag. Efficiency of pseudo-likelihood estimation for simple Gaussian fields. *Biometrika*, 64(3):616–618, 1977.

[2] J. Besag. On the statistical analysis of dirty pictures. *Journal of Royal Statistical Society. Series B (Methodological)*, 48(3):259–302, 1986.

[3] A. Nüchter and J. Hertzberg. Towards semantic maps for mobile robots. *Robotics and Autonomous Systems*, 56(11):915–926, 2008.

[4] S. Pu and G. Vosselman. Automatic extraction of building features from terrestrial laser scanning. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36 (5), 2006.

[5] T. Rabbani, , F. A. van den Heuvel, and G. Vosselman. Segmentation of point clouds using smoothness constraint. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(5):248–253, 2006.