A ROTATION INVARIANT 3D INDOOR SCENE LABELING APPROACH BASED ON CONDITIONAL RANDOM FIELDS

Yankun Lang, Haiyuan Wu, Qian Chen

Wakayama University, Japan

ABSTRACT

In this paper, we present an efficient framework for 3D indoor scene labeling based on a Conditional Random Field model. To make this framework invariant to camera rotation, a novel feature vector is developed, which is simple but discriminative. Meanwhile, we re-define the pairwise potential to improve the performance. A method for learning the labeling compatibility is proposed to exploit the strong contextual relations between class labels. We evaluate our approach on three datasets and the experimental results show that it achieves higher accuracy compared with several state of the art researches.

Index Terms— Indoor scene labeling, rotation invariance, CRF, 3D point cloud, contextual relations

1. INTRODUCTION

3D Scene labeling has been an important step towards pedestrian tracking/detection researches. Generally, for an indoor scene, after removing the "background", person object can be more easily detected by using a binary classifier from the rest objects. It significantly saves lots of resources, and simplifies the detection problem as well. The chosen of the feature descriptors decides the quality of the labeling results. Besides that, exploiting the contextual relations between neighboring labels helps to improve the performance especially when the features are not sufficiently discriminative. 3D spatial or structural features developed from 3D positions, normals of surface are very discriminative when the scene is fixed, and some works either learn the model of context relations from a large dataset or by hand-designing. However, such features and models are very sensitive to camera rotation. The neglect of rotation invariance makes scene labeling unstable in some practical applications where using a moving camera.

To solve these problems, in this paper, we focus on developing a 3D indoor scene labeling framework that is robust against camera rotation. Similar to most works, each 3D scene (point cloud) is oversegmented into supervoxels, then a graphical model is trained to produce label for each supervoxel. Differently, since our labeling framework is designed as a "preprocessing" for person detection, labels in our work only contain Background (Wall, Roof and Ground) and Person Candidates. Meanwhile, to evaluate the efficiency of our approach, the dataset is created by a fixed camera, and only the scene in the testing set will be rotated by a random angle. Besides that, contributions in this paper include: 1), we develop a set of feature vectors which is discriminative and rotation invariant based on the histogram of normal vectors. 2), we propose an approach of learning the contextual relations between neighboring labels to make it invariant to camera rotation. 3), we develop a new pairwise potential model to improve the labeling results.

2. RELATED WORK

Many 2D scene labelling studies have been reported in literatue[1, 2, 3, 4, 5, 6, 7], in which a lot of valuable information cannot be extracted from 2D image. Recent advances in 3D sensing technology make it possible to exploit 3D geometric and structural features. Many existing 3D features such as: spherical harmonic invariants[8], curvature maps[9], and conformal factors[10] have been proven to be unapplicated. Point Feature Histogram (PFH) proposed in [11] encodes a point's k-neighborhood geometrical properties. Fast Point Feature Histograms (FPFH) proposed in [12] retains most of the discriminative power of the PFH but has an improved calculation speed. Arbeiter G's work [13] proved that FPFH had a better performance of classifying primitive 3D local surfaces by comparing with Radius-Based Surface Descriptor (RSD) and Principal Curvatures (PC), but it is weak for distinguishing the objects having the similar shapes. To solve this problem, Viewpoint Feature Histogram (VFH)[14] encodes important statistics between the viewpoint and the surface normals to make it more discriminative.

Graphical model (Markov Random fields, Conditional Random fields) has been widely used for solving the labeling problems. In M.Najafi's work [15], a non-associative higher-order CRF was proposed, making the model invariant to the size or number of the segments. Mayank B's work [16] proposed a spatial 3D feature called Vertical Support Histogram (VSH), and manually designed a context model. However, the context relation model defined in these works was hand-designed thus unstable. In Daniel Wolf's work [17], for the purpose of rotation invariance, they proposed a robust feature and the method of learning the label relationships. This work

achieved a high accuracy but the feature was learned from an enlarged training set by rotating each scene for ten times.

All the works above lack the stability against camera rotation, which is the problem supposed to be solved in our work. Details of our research will be given in the next chapters.

3. 3D INDOOR SCENE LABELLING

This section discusses about the approach of 3D indoor scene labeling, including the graphical model, features and inference algorithm. Each scene is captured by a Kinect RGB-D camera, then organized as a point cloud. Then each point cloud is oversegmented into supervoxels with the approach in [18] to ensure that points in each supervoxel have same properties. These supervoxels are the atomic units in our graphical model.

Given a segmented point cloud consisting of N supervoxels, we aim to predict a labeling $\mathbf{y} = \{y_1, ..., y_N\}$ for it. Each label y_i can take any value from a pre-defined set of values $L = \{l_1, \ldots, l_L\}$. With the global observation $\mathbf{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_N\}$, $(\mathbf{y}, \mathbf{V})$ can be modeled as a Conditional Random Fields model characterized by a Gibbs energy:

$$E(\mathbf{y}|\mathbf{V}) = \sum_i \Psi_i(y_i|\mathbf{v}_i) + \sum_{i \neq j} \Psi_{ij}(y_i, y_j|\mathbf{v}_i, \mathbf{v}_j) \quad (1)$$

where Ψ_i is the unary potential calculated independently for each supervoxel, Ψ_{ij} is the pairwise potential. The optimal label assignment $\mathbf{y}^*$ can be solved by minimizing the Gibbs energy as:

$$\mathbf{y}^* = \arg\min_{\mathbf{y}} E(\mathbf{y}|\mathbf{V}) \quad (2)$$

Details of Ψ_i and Ψ_{ij} will be given in the following subsections.

3.1. Unary Potential

After the whole 3D point cloud is clustered into supervoxels, we aim to use a set of features which are invariant to camera rotation to calculate the unary potential. Details of the feature used in our work is shown in table 3.1. The 33-dimention FPFH is used to capture the appearance of the voxel. Apparently, coordinate information is inadequate to be used as position feature but the relative position in the whole scene is unchanged. For each voxel, the distance from its centroid point to that of the whole scene is used as a feature to capture the spatial location information. In addition, another strong feature is developed based on the histogram of the normal vector. For each point, after rotation, its normal vector will be changed with the same variable, while in a supervoxel, the scale of the normal vector histogram will remain similar. We use deviation of the histogram to emphasize the invariance and describe the orientation of surface.

For each voxel, we use an efficient classifier to calculate the posterior probability $p(y_i|\mathbf{x}_i)$ for each label $y_i \in \mathbf{L} = \{l_1, l_2, \cdots, l_N\}$ conditioned on the extracted feature vector $\mathbf{x}_i$, then we use this probability to initialize the unary potential as:

$$\Psi_i(y_i|\mathbf{x}_i) = -\log\{p(y_i = l|\mathbf{x}_i)\} \quad (3)$$

Notice that for calculating the unary potential, we have changed $\mathbf{v}$ with the feature vector proposed in this subsection $\mathbf{x}$ since features we use for unary potential and pairwise potential are different. The classifier is chosen as Random Forest (RF) since it can handle high dimensional feature vectors as well as large number of training examples and output the probability $p(y_i|\mathbf{x}_i)$ directly. Meanwhile, it has a remarkable computation speed while maintaining high accuracy.

Since this probability is produced independently, the labeling result is generally noisy and inconsistent. We use pairwise potential to refine the performance by considering the contextual relations of labels.

Table 1. Details of the feature vector used in unary potential

feature	dim
Normal-deviation	3
Color in CIELAB space	3
Distance from voxel centroid to cloud centroid	1
FPFH	33
Total number of features	40

3.2. Pairwise Potential

Pairwise potential denotes the cost for assigning each pair of labels to the neighboring voxels, which makes it enable to smooth the noise caused by independency. In this paper, we use a linear combination of m kernel functions to model the pairwise potential, which is given by:

$$\Psi_{ij}(y_i, y_j|\mathbf{v}_i, \mathbf{v}_j) = \sum_m \mu^m(y_i, y_j) w^{(m)} k^{(m)}(\mathbf{f}_i, \mathbf{f}_j) \quad (4)$$

where $w^{(m)}$ is the linear combination weights for each kernel. $\mu^m(\cdot)$ is a function that examines the compatibility of a pair of labels. Each kernel $k^{(m)}(\mathbf{f}_i, \mathbf{f}_j)$ is define as a Gaussian kernel:

$$k^{(m)}(\mathbf{f}_i, \mathbf{f}_j) = \exp\left\{-\frac{1}{2}(\mathbf{f}_i - \mathbf{f}_j)^T \Lambda^{(m)} (\mathbf{f}_i - \mathbf{f}_j)\right\} \quad (5)$$

where $\mathbf{f}_i$ and $\mathbf{f}_j$ are feature vectors for voxel $\mathbf{v}_i$ and $\mathbf{v}_j$ in an arbitrary feature space. Note that this feature vector is not same to the one introduced in subsection Unary Potential. $\Lambda^{(m)}$ is a symmetric, positive-definite precision matrix, which defines the shape of Each kernel $k^{(m)}$.

Hermans in [19] proposed a two-kernel potential consisting of appearance potential and smoothness potential. The appearance potential is given by:

$$k^{(1)} = \exp\left\{-\frac{|\mathbf{p}_i - \mathbf{p}_j|}{2\theta_\alpha^2} - \frac{|\mathbf{c}_i - \mathbf{c}_j|}{2\theta_\beta^2}\right\} \quad (6)$$

where **p** is the 3D positions of point and **c** is the color vectors of the voxel in CIELAB space. This model is used to specify the connections between voxels with similar appearance and color in a large range. θ_α and θ_β controls the influence range of the kernel and defines the degrees of similarity of the colors, respectively.

The second potential called smoothness potential is given by:

$$k^{(2)} = \exp\left\{-\frac{|\mathbf{p}_i - \mathbf{p}_j|}{2\theta_\gamma^2} - \frac{|\mathbf{n}_i - \mathbf{n}_j|}{2\theta_\delta^2}\right\} \quad (7)$$

where **n** is the surface normal vector of each voxel. This potential operates in a small range to examine the compatibility between labels according to the fact that close voxels with similar surface orientations are more likely to belong to the same label. θ_γ specifies the range which is much smaller than θ_α, and θ_δ defines the degree of similarity of the normals.

One disadvantage of this smoothness potential is that it only applies to the objects with regular shapes. For a pair of neighboring voxels belonging to a same irregular object, since the surface of each supervoxel are approximated as planes, even though the normals of their surfaces are different, variance of the normals of each point is very small. For this reason, we add another smoothness potential expressed as:

$$k^{(3)} = \exp\left\{-\frac{|\mathbf{p}_i - \mathbf{p}_j|}{2\theta_\gamma^2} - \frac{|\mathbf{d}_i - \mathbf{d}_j|}{2\theta_\epsilon^2}\right\} \quad (8)$$

where **d** is the normal-deviation same to the one in table 1. θ_ϵ defines the degree of their similarity.

Label compatibility functions $\mu^{(m)}$ are defined separately. We use a simple Potts model to define $\mu^{(2)}$ and $\mu^{(3)}$ for the smoothness potentials as:

$$\mu^{(2)}(y_i, y_j) = \mu^{(3)}(y_i, y_j) = 1_{[y_i \neq y_j]} \quad (9)$$

For the appearance potential, $\mu^{(1)}$ should capture context relations between different classes across a larger range. In our paper, $\mu^{(1)}$ is learned from the training set and should be invariant to camera rotation. Method for learning this model is shown as follows: first, in each point cloud of the training set, for every supervoxel labeled as l_i, we define a sphere centered on this voxel with the radius $r = \theta_\alpha$. Then we calculate the number of the voxels within this sphere N_{l_i} labeled as $l_i = \{l_1, \ldots, l_M\}$ respectively, and generate a global histogram $h_{l_i} = \{N_{l_1}, N_{l_2}, \ldots, N_{l_M}\}$ over all the voxels. Each histogram is normalized and used to construct a $M \times M$ symmetric matrix $H = [h_{l_1}^T, \ldots, h_{l_M}^T]^T$. The final learned matrix is averaged from the whole training set. Elements in the matrix show the likelihood of assigning a pair of labels to two neighboring voxels within the range $r = \theta_\alpha$.

Inference is solved by Mean Filed Approximation[20], which has a linear complexity in the number of voxels thus converges very fast. Meanwhile, the weights $w^{(m)}$ can be learned during the inference.

(a) dataset 1 (b) dataset 2 (c) dataset 3

Fig. 1. Original point cloud taken from the testing set

Table 2. Parameters used in our research

Segmentation	r_v	r_s	w_c	w_n	w_s
	1.8cm	9cm	0.4	1	0.2
RF Classifier	N_t		m_d		m_s
	100		30		20
CRF	θ_α	θ_β	θ_γ	θ_δ	θ_ϵ
	1m	(12,3,3)	15cm	0.05rad	0.1rad

4. EXPERIMENTAL RESULTS

4.1. Dataset and Parameters

Our experiment was implemented on three datasets taken from our laboratory by a fixed Kinect v2 camera, each of them consists of 80 frames of point cloud containing moving person as shown in Figure 1. The first 30 frames of point cloud were used as the training set, and the last 50 frames were used for testing, where each frame was rotated with a random angle between $-20°$ - $20°$.

All the parameters were set empirically as shown in Table 2. For the oversegmentation, voxel resolution r_v and seed resolution r_s control the size of each supervoxel, and w_c, w_n, w_s are the weights controlling the influence of color, surface normals and spatial distances for each cluster[18]. N_t, m_d and m_s control the maximum number of trees, the maximum tree depth and the minimum samples required at each leaf node of the RF classifier respectively. Iteration for inference is set to be 5. CPU used in our work is Intel Core i7 with the frequency 3.5GHz.

Table 3. Labeling accuracy after RF with different features. Top half: no rotations, lower half:with an arbitrary rotation

Method (no rotations)	R	W	C	G	Avg.
feature in [17]	98%	92%	90%	90%	92%
proposed feature	95%	90%	87%	89%	90%
Method	**R**	**W**	**C**	**G**	**Avg.**
feature in [17]	75%	71%	52%	70%	67%
proposed feature	90%	85%	84%	85%	86%

Table 4. Labeling accuracy for the whole datasets. R: Roof, W: Wall, C: Person Candidates, G: Ground, Avg.: Average accuracy

Method	R	W	C	G	Avg.
method of [16]	75%	65%	60%	63%	66%
method of [19]	80%	75%	78%	73%	76%
method of [17]	84%	80%	79%	78%	80%
Ours (without eq.(8))	93%	92%	87%	91%	91%
Ours (with eq.(8))	93%	93%	90%	91%	92%

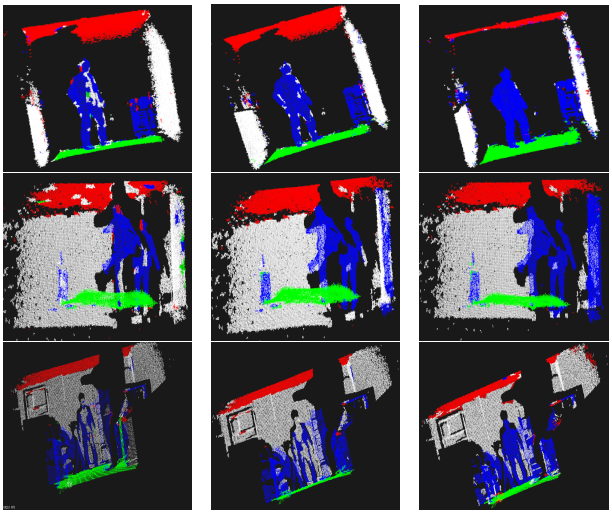
4.2. Evaluation and Comparison

To evaluate the rotation invariance of the proposed feature, we first compare the labeling result after the RF classifier with the feature vector proposed in[17]. The upper half of Table 3 shows the results on a testing set without any rotations, where using feature vector in[17] achieved a higher accuracy since it is more discriminative. However, in the case of rotating the testing set by a random angle, the proposed feature vector is more robust to camera rotation, thus gained a better result as shown in the lower half.

Figure 2 shows the labeling results on each dataset. Fig2.(a) shows the result of RF classifier, which is a coarse label prediction suffering from classification noise. We first used the CRF model without the 2nd-smoothness potential in eq.(8) to refine the labeling result. It successfully corrected the errors for the backgrounds (Wall, Roof, Ground), which are rigid objects, but the ambiguities still remained in the non-rigid objects as shown in Fig2.(b). The "second-smoothness" potential proposed in this work solved this problem and made our work obtain a high accuracy (Fig2.(c)). The process for each frame cost less than 400ms.

We compared our work with [16], [19] and [17], which is the most similar to ours. Similarly, only testing set was randomly rotated. Table 4 shows that the proposed method achieved the highest accuracy. Moreover, with the full CRF model including the "2nd-smoothness" potential the accuracy was improved by 4%. Both the methods of [16] and [19] gained a low accuracy since the label compatibility was not sufficiently learned. As for [17], the feature became weak in our experiment since it is sensitive to camera rotation.

At last, we used a single VFH and VFH combined with color and normal-deviation to detect person from "Person Candidates" results by a RF classifier. Detection results are shown in Figure 3. The average accuracy of using VFH achieved 92%, while the later improved the accuracy with 2%. More importantly, the labeling framework reduced the amount of the voxels need to be classified by 60% , which simplified the detection.



(a) RF results (b) without eq.(8) (c) full CRF

Fig. 2. Experimental results. Color for the label: Red: Roof. White: Wall. Blue: Person Candidates. Green: Ground.

5. CONCLUSION

In this paper, we have proposed an 3D indoor scene labeling framework based on a dense Conditional Random Field. A method of learning labeling compatibility has been proposed, which is used to depict the label compatibility. A robust new feature vector was developed for the purpose of rotation invariance. Experimental results showed that our framework achieved a better performance and was robust against camera rotation by comparing with state of the art works. Further research will focus on improving the feature vector and developing our framework for pedestrian detection and tracking.

Acknowledgment

This work is partially supported by JSPS KAKENHI Grant #26330195 to Q. Chen and JSPS KAKENHI Grant #15K01331 to H. Wu.

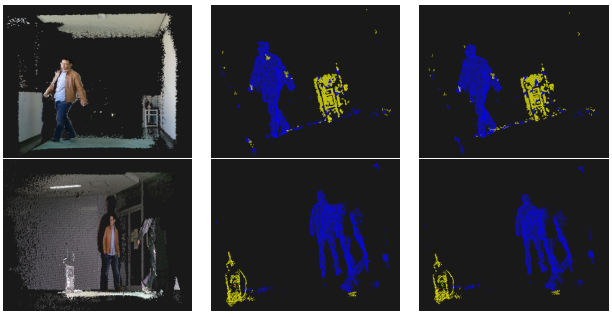


Fig. 3. Performance of person detection with VFH (middle) and the feature we proposed (right)

6. REFERENCES

[1] Peter Gehler and Sebastian Nowozin, "On feature combination for multiclass object classification," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 221–228.

[2] Dharmendra Patidar, Nitin Jain, and Ashish Parikh, "Performance analysis of artificial neural network and k nearest neighbors image classification techniques with wavelet features," in *Computer Communication and Systems, 2014 International Conference on*. IEEE, 2014, pp. 191–194.

[3] Sudeep D Thepade and Madhura M Kalbhor, "Extended performance appraise of bayes, function, lazy, rule, tree data mining classifier in novel transformed fractional content based image classification," in *Pervasive Computing (ICPC), 2015 International Conference on*. IEEE, 2015, pp. 1–6.

[4] O.A. Vatamanu and M. Jivulescu, "Image classification using local binary pattern operators for static images," *SACI*, pp. 173–178, 2013.

[5] NST Sai and Ravindra C Patil, "Image retrieval using equalized histogram image bins moments," *International Journal on Signal & Image Processing*, vol. 2, no. 1, pp. 4, 2011.

[6] Bastian Leibe, Nico Cornelis, Kurt Cornelis, and Luc Van Gool, "Dynamic 3d scene analysis from a moving vehicle," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 2007, pp. 1–8.

[7] Yangqing Jia, Chang Huang, and Trevor Darrell, "Beyond spatial pyramids: Receptive field learning for pooled image features," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 3370–3377.

[8] Gilles Burel and Hugues Hénocq, "Three-dimensional invariants and their application to object recognition," *Signal Processing*, vol. 45, no. 1, pp. 1–22, 1995.

[9] Timothy Gatzke, Cindy Grimm, Michael Garland, and Steve Zelinka, "Curvature maps for local shape comparison," in *Shape Modeling and Applications, 2005 International Conference*. IEEE, 2005, pp. 244–253.

[10] Mirela Ben-Chen and Craig Gotsman, "Characterizing shape using conformal factors.," in *3DOR*, 2008, pp. 1–8.

[11] Radu Bogdan Rusu, Nico Blodow, Zoltan Csaba Marton, and Michael Beetz, "Aligning point cloud views using persistent feature histograms," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*. IEEE, 2008, pp. 3384–3391.

[12] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz, "Fast point feature histograms (fpfh) for 3d registration," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 3212–3217.

[13] Georg Arbeiter, Steffen Fuchs, Richard Bormann, Jan Fischer, and Alexander Verl, "Evaluation of 3d feature descriptors for classification of surface geometries in point clouds," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 1644–1650.

[14] Radu Bogdan Rusu, Gary Bradski, Romain Thibaux, and John Hsu, "Fast 3d recognition and pose using the viewpoint feature histogram," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE, 2010, pp. 2155–2162.

[15] Mohammad Najafi, Sarah Taghavi Namin, Mathieu Salzmann, and Lars Petersson, "Non-associative higher-order markov networks for point cloud classification," in *European Conference on Computer Vision*. Springer, 2014, pp. 500–515.

[16] Mayank Bansal, Bogdan Matei, Harpreet Sawhney, Sang-Hack Jung, and Jayan Eledath, "Pedestrian detection with depth-guided structure labeling," in *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 31–38.

[17] Daniel Wolf, Johann Prankl, and Markus Vincze, "Fast semantic segmentation of 3d point clouds using a dense crf with learned parameters," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4867–4873.

[18] Jeremie Papon, Alexey Abramov, Markus Schoeler, and Florentin Worgotter, "Voxel cloud connectivity segmentation-supervoxels for point clouds," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2027–2034.

[19] Alexander Hermans, Georgios Floros, and Bastian Leibe, "Dense 3d semantic mapping of indoor scenes from rgb-d images," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 2631–2638.

[20] Philipp Krähenbühl and Vladlen Koltun, "Parameter learning and convergent inference for dense random fields.," in *ICML (3)*, 2013, pp. 513–521.