Multiscale Saliency Detection for Colored 3D Point Clouds Based on Random Walk

Se-Won Jeong*, Jae-Seong Yun*, and Jae-Young Sim*†
* Department of Electrical Engineering, UNIST, Ulsan, Korea
† Graduate School of Artificial Intelligence, UNIST, Ulsan, Korea
E-mails: {swjeong, jsyun, jysim}@unist.ac.kr

***Abstract*—Saliency detection for 3D visual data has been actively studied, but relatively little effort has been made to detect both of the geometric and photometric saliency for large-scale colored 3D point clouds (LSC3DPCs). We propose a random walk based multiscale saliency detection algorithm for LSC3DPCs acquired by terrestrial light detection and ranging devices. We employ Fast Point Feature Histogram descriptor and Lab colors to estimate the geometric and photometric features of points, respectively. We partition an input LSC3DPC model into supervoxel clusters at three different scales of octree. Then we build a fully-connected graph of clusters at each scale such that an edge connecting two clusters with more dissimilar features to each other is assigned a higher weight. We perform random walk simulation on the graphs at multiple scales to yield multiscale saliency maps, respectively, which are then averaged together to generate a final saliency map. Experimental results show that the proposed method estimates the global and local saliency of LSC3DPCs more faithfully compared with the existing method.**

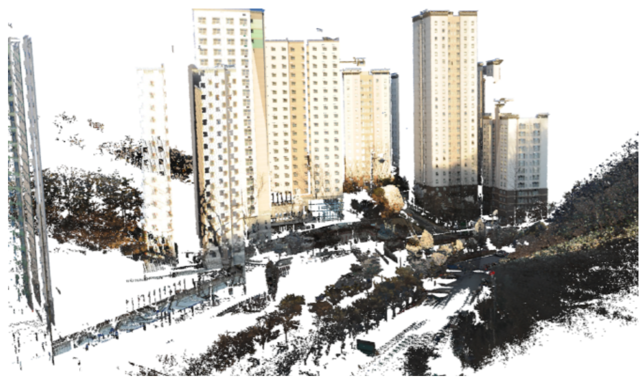


Fig. 1: Large-scale colored 3D point cloud.

I. Introduction

Human Visual System (HVS) can easily recognize the visual contents from images by focusing on visually prominent regions selectively. Saliency detection is a technique that automatically detects visually important or meaningful regions of input images, and it has been applied to many applications of image processing and computer vision such as recognition [1], retrieval [2], and segmentation [3]. In recent years, 3D geometric saliency detection, which extracts geometrically distinct regions from 3D visual data, has been also studied for various types of 3D visual data including RGB-D images [4]–[9], polygonal meshes [10]–[14] and point clouds [15]–[19].

In particular, light detection and ranging (LiDAR) based terrestrial scanning devices can capture 3D real-world environmental scenes, providing highly detailed large-scale 3D point clouds with color information, as shown in Figure 1. Such large-scale colored 3D point clouds (LSC3DPCs) can facilitate various immersive visual applications, for example, 3D environmental map generation for autonomous vehicles and 3D contents generation for virtual reality and mixed reality. Note that a typical LSC3DPC model captures 360° environmental scene and is usually composed of several millions of points. Therefore, automatic detection of visually salient regions from a LSC3DPC model is critical to alleviate the computational complexity for geometric processing of LSC3DPC data. Attempt has been made for automatic saliency detection of large-scale 3D point clouds [17], [19]. However, they considered 3D geometric features only without color information, or did not consider global characteristics of 3D model. In this paper, we introduce a novel saliency detection method for LSC3DPCs which exploits geometry and color features together to capture both of the local and global characteristics. We first employ a supervoxel hierarchy where each cluster contains points with similar normal directions and colors. We compute the global features for each cluster by averaging the local geometry and color features of the points in the cluster. At each scale of supervoxel clusters, we construct a fully-connected graph where we obtain saliency distribution of clusters by performing random walk (RW) simulation on the graph. Then we obtain a final saliency map by averaging the multiscale saliency maps. Experimental results show that the proposed algorithm detects global and local saliency of LSC3DPCs more faithfully compared with the existing method [19].

The rest of the paper is organized as follows: Section II reviews the related work of saliency detection for 3D visual data. Section III explains the proposed algorithm and Section IV represents the experimental results. Section V concludes the paper.

II. Related Work

Saliency detection for RGB-D images: Lang *et al.* [4] estimated a depth image from a pair of stereoscopic images, and obtained a human fixation map on the depth image. They computed depth prior from the fixation map which is then used to improve the performance of existing saliency maps of 2D images. Ju *et al.* [6] partitioned a color image into superpixels using SLIC [20], and estimated the saliency at each superpixel as the sum of anisotropic center-surround difference of depth values between a superpixel and its neighboring superpixels. They obtained a final saliency map by refining the coarse scale saliency map using depth prior and Gaussian prior. Peng *et al.* [7] first obtained a patch-wise low-level saliency map using center-surround difference in terms of depth and color features. Then they generated spanning trees at several highly salient patches, where mid-level saliency are estimated as the occupation frequency of spanning trees. A final saliency map is given by fusing the low and mid-level saliency maps with Gaussian prior.

Saliency detection for 3D meshes: Lee *et al.* [10] first proposed a saliency detection method for 3D meshes, which estimates the saliency at each vertex by computing the difference of curvature compared to neighboring vertices. Moreover, multiscale saliency maps are also obtained by changing the scope of neighboring vertices. Leifman *et al.* [12] employed a spin image as a descriptor of each vertex, and computed a vertex distinctness using the geodesic feature distance from a vertex to its neighbors. The highly distinct vertices are regarded as focus points and the saliency values of the other vertices are additionally weighted according to their geodesic feature distances to the closest focus points. Song *et al.* [13] applied spectral processing to saliency detection for 3D meshes. They computed mesh spectrum and estimated the saliency as the spectral deviation from locally averaged spectrum, and generated a final saliency map by combining the saliency maps at different scales. Jeong and Sim [14] adopted 3D semi-regular meshes to compute mesh saliency. They computed angular deviation of normal vectors between neighboring faces and constructed a fully connected graph at each scale of semi-regular mesh. A final saliency distribution is obtained by taking the maximum among the normalized stationary distributions of random walks on the graphs at multiple scales.

Saliency detection for 3D point clouds: Kim *et al.* [15] first clustered an input 3D point cloud using information-theoretic clustering [21], and evaluated cluster-wise volume after compression (VAC) values. They also projected the clusters to the corresponding 2D image and computed three different cluster-wise color features. Then they obtained a final saliency distribution by combining cluster-wise VAC values and color features together. Akman and Jonker [16] detected saliency on 3D point clouds obtained by using time-of-flight camera. They computed irregularities of surface normals at each point in multiple scales which are then averaged together to compute saliency. Shtrom *et al.* [17] computed geometric saliency of large-scale 3D point clouds using a geometric feature descriptor of Fast Point Feature Histogram (FPFH) [22]. They computed low level and high level saliency maps based on the dissimilarity of the feature descriptors derived from different sets of neighboring points, and generated a final saliency map by taking their weighted summation. Leroy *et al.* [18] constructed supervoxel clusters at two different scales

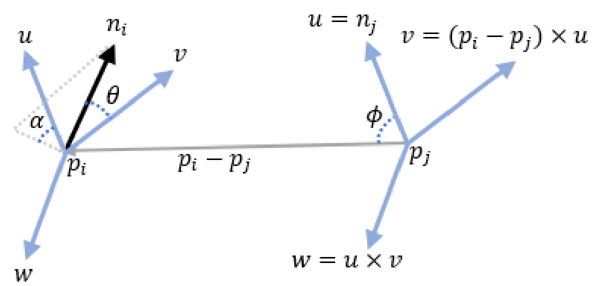


Fig. 2: Relative angular variation associated with p_i and p_j with respect to a local coordinate asystem.

of a 3D point cloud model, and then computed saliency at each scale using color rarity in six different color spaces. The final saliency map is obtained by applying Gaussian filtering to their weighted summation. Yun and Sim [19] computed saliency on LSC3DPCs. They constructed a supervoxel hierarchy and computed the geometric and color features for each supervoxel cluster using FPFH [22] and Lab color space, respectively. They computed a saliency map at each scale using the local contrast between neighboring clusters, and obtained a final saliency map by averaging the saliency maps at multiple scales.

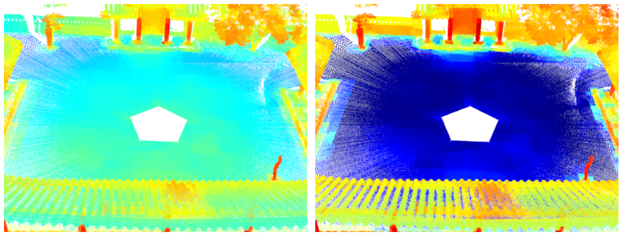
III. Proposed method

In general, the previous methods of saliency detection for 3D point clouds employed geometric features [16], [17] or color features [18] only. Yun and Sim [19] employed both of the geometric and color features together to estimate the saliency distribution of LSC3DPCs, however they considered local feature contrast mainly without considering global characteristics of 3D scenes.

Random walk (RW) is a stochastic modeling of random movement on a graph, which has been widely used for saliency detection of images [23], videos [24], and 3D meshes [14]. In this work, we devise a RW-based saliency detection method for LSC3DPCs. We compute local geometric and color features at each point using FPFH and Lab color representation, respectively. We build a hierarchy of multiscale supervoxel clusters using the octree structure, and we construct a fully-connected graph at each scale where a graph node corresponds to a supervoxel cluster and the edge weight is defined as the difference of geometric and color features between the connected clusters. We compute a cluster-wise saliency map at each scale through RW simulation on the associated graph, and then estimate a final saliency map by averaging the saliency maps obtained at three different scales.

A. Feature computation

We also adopt FPFH [22] to extract local geometric features, which is a 33-dimensional vector describing the angular difference of normal vectors associated with neighboring points. Let Ω_i be the set of neighboring points of a given point p_i located within the distance of d_{leaf} from p_i. To obtain FPFH,



(a) (b)

Fig. 3: Comparison of saliency distributions (a) with the center prior assumption and (b) without the center prior assumption, respectively.

three different angular variations of (α, θ, ϕ) associated with a pair of p_i and its neighbor p_j are described based on the local coordinate system as shown in Figure 2. Then a simplified point feature histogram $SPFH(p_i)$ is defined at p_i as the histogram of (α, θ, ϕ) computed over all the points belonging to Ω_i. Finally, FPFH $\mathbf{h}_i$ at point p_i is computed as

$$\mathbf{h}_i = SPFH(p_i) + \frac{1}{|\Omega_i|} \sum_{p_k \in \Omega_i} \frac{SPFH(p_k)}{|\psi(p_i) - \psi(p_k)|}, \quad (1)$$

where $\psi(p)$ is the position of the point p. We use Boulch *et al.* [25]'s method for normal vector estimation which detects corner regions of 3D model using the randomized Hough transform and computes the normal vectors of points around corner regions more precisely. We also normalize each feature vector such that the sum of the elements in each vector becomes 1.

We use the Lab color representation as local color features, which reflects the perception characteristics of the HVS faithfully.

B. Supervoxel clustering

A single LSC3DPC model is usually composed of several millions of points, and hence point-wise saliency computation often fails to capture global saliency of 3D scene while yielding huge computational complexity. We partition an input LS3DPC model into local supervoxels and estimate saliency at each supervoxel cluster. We first construct an octree [26] associated with an input LS3DPC model. In order to adaptively divide an LSC3DPC model into octree nodes according to the geometry of a captured target scene, we set the side length of the leaf node, d_{leaf}, to 8cm for indoor scenes and to 2.5% of the height of the tallest structure, e.g., building, for outdoor scenes, respectively. Then we select the three consecutive levels of the octree, where the side lengths of nodes are 2, 4, and 8 times larger than that of the leaf node. At each selected scale, we perform the supervoxel clustering [27].

C. Graph Construction

At each scale, we construct the fully-connected graph $G(\mathcal{N}, \mathcal{E})$ where each node $n_i \in \mathcal{N}$ corresponds to each supervoxel cluster $\mathbf{c}_i$, and each edge $e_{ij} \in \mathcal{E}$ connects two nodes n_i and n_j. We define the features at each supervoxel cluster as the average FPFH and the average color of all the points within the cluster. We determine the weight w_{ij} for each edge e_{ij} in $G(\mathcal{N}, \mathcal{E})$ such that w_{ij} becomes large when n_i is highly distinct from n_j based on the center-surround contrast. Specifically, we estimate the geometry dissimilarity $\delta_{\text{geo}}(\mathbf{c}_i, \mathbf{c}_j)$ between two clusters $\mathbf{c}_i$ and $\mathbf{c}_j$ by computing the dissimilarity of the corresponding FPFHs using histogram intersection kernel [28].

$$\delta_{\text{geo}}(\mathbf{c}_i, \mathbf{c}_j) = 1 - \sum_{k=1}^{33} \min\{\mathbf{H}_i(k), \mathbf{H}_j(k)\} \quad (2)$$

where $\mathbf{H}_i$ and $\mathbf{H}_j$ are the FPFHs defined at $\mathbf{c}_i$ and $\mathbf{c}_j$, respectively, and $\mathbf{H}(k)$ denotes the k-th element of $\mathbf{H}$. Note that $\delta_{\text{geo}}(\mathbf{c}_i, \mathbf{c}_j)$ has a high value when two histograms are dissimilar to each other. We also define the color dissimilarity $\delta_{\text{col}}(\mathbf{c}_i, \mathbf{c}_j)$ between $\mathbf{c}_i$ and $\mathbf{c}_j$ as

$$\delta_{\text{col}}(\mathbf{c}_i, \mathbf{c}_j) = \|\mathbf{I}_i - \mathbf{I}_j\| \quad (3)$$

where $\mathbf{I}_i$ and $\mathbf{I}_j$ are the Lab colors of $\mathbf{c}_i$ and $\mathbf{c}_j$, respectively.

Considering both of the geometry and color dissimilarities, we design an edge weight as

$$w_{ij} = 1 - \exp\left(-\rho \frac{\delta_{\text{geo}}(\mathbf{c}_i, \mathbf{c}_j)}{\beta \cdot \delta_{\text{geo, max}}} - (1-\rho) \frac{\delta_{\text{col}}(\mathbf{c}_i, \mathbf{c}_j)}{\beta \cdot \delta_{\text{col, max}}}\right) \quad (4)$$

where $\delta_{\text{geo, max}}$ and $\delta_{\text{col, max}}$ denote the maximum values of $\delta_{\text{geo}}(\mathbf{c}_i, \mathbf{c}_j)$ and $\delta_{\text{col}}(\mathbf{c}_i, \mathbf{c}_j)$, respectively. We empirically set $\beta = 0.11$ and $\rho = 0.5$. Note that, as did in [14], the edge weight w_{ij} does not consider the Euclidean distance between two clusters $\mathbf{c}_i$ and $\mathbf{c}_j$, since the center-prior assumption that has been widely used in many graph-based saliency detection techniques for 2D images [29], [30] does not hold for 3D models. Figure 3 compares the saliency maps of 'Korean house' model obtained with and without the center-prior concept, respectively. We see that the saliency map with the center-prior assumption in Figure 3(a) highlights non-salient floor regions as salient. On the contrary, the resulting saliency distribution in Figure 3(b) shows that the proposed weight design without center prior assumption captures visually salient regions faithfully.

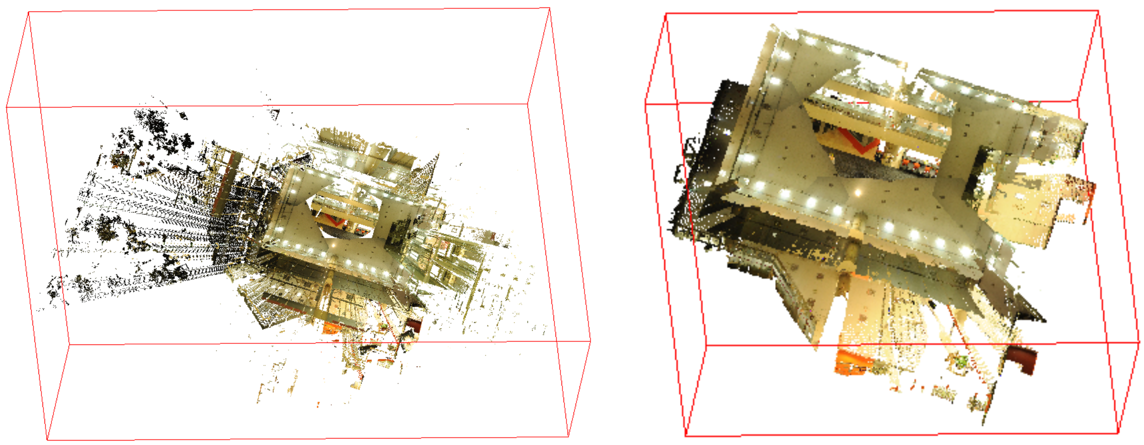
D. Saliency detection

In RW simulation, we regard a graph node is more salient where a random walker visits the node more frequently. Therefore, we design a transition matrix $\mathbf{P}$ of the Markov chain such that each (j,i)th element $\mathbf{P}(j,i)$ is given by

$$\mathbf{P}(j,i) = \frac{w_{ji}}{\sum_j w_{ji}}. \quad (5)$$

Note that $\mathbf{P}(j,i)$ represents the transition probability that the random walker moves from n_i to n_j, and thus the sum of all outgoing probabilities from a certain node becomes 1. G is a fully-connected graph, and thus $\mathbf{P}$ is irreducible, which results in a unique steady-state distribution $\boldsymbol{\pi}$ satisfying

$$\boldsymbol{\pi} = \mathbf{P}\boldsymbol{\pi}. \quad (6)$$



(a) (b)

Fig. 4: Pre-processing of false point removal. (a) The raw 'Library' model with noisy falsely detected points which was used in the existing method [19]. (b) The pre-processed 'Library' model by removing false points which was used in this paper. The bounding boxes are shown in red.

In addition, since the transition matrix $\mathbf{P}$ is derived from an undirected graph G, $\boldsymbol{\pi}$ satisfies a detailed balance with $\mathbf{P}$ and we get a closed-form solution for $\boldsymbol{\pi}(i)$, the i-th element of $\boldsymbol{\pi}$ [31] given by

$$\boldsymbol{\pi}(i) = \frac{\sum_j w_{ji}}{\sum_j \sum_k w_{jk}}. \tag{7}$$

Then the normalized $\boldsymbol{\pi}(i)$ is served as a saliency value s_i for the cluster $\mathbf{c}_i$.

$$s_i = \frac{\boldsymbol{\pi}(i) - \pi_{\min}}{\pi_{\max} - \pi_{\min}} \tag{8}$$

where $\pi_{\max}$ and $\pi_{\min}$ denote the maximum and the minimum values of $\boldsymbol{\pi}(i)$'s, respectively. We obtain three saliency distributions of supervoxel clusters at three scales, respectively, and estimate the final saliency map by taking the average saliency value of the three saliency maps.

IV. Experimental Results

We evaluate the performance of the proposed saliency detection method on six LSC3DPC models acquired by using a 3D terrestrial LiDAR scanner. We also provide the resulting saliency distributions of the proposed method compared with that of the existing method [19].

A. Data acquisition

We generate six LSC3DPC models captured by a 3D terrestrial LiDAR scanner, RIEGL VZ-400 [32], with angular resolutions from $0.06°$ to $0.08°$. In general, raw point clouds acquired by LiDAR usually contain lots of noisy outlier points. Figure 4 (a) shows the raw data of 'Library' model which was used in the previous method [19]. We see that falsely generated noisy points are distributed in empty space along the radial directions from the location of scanner, which cause an inefficient and unbalanced bounding box to include the raw LSC3DPC model as depicted in red. The false points distributed over unnecessarily large space are also regarded as true points in saliency detection, and therefore we often get unreliable saliency distributions. To alleviate this drawback, we remove such noisy outlier points from the raw point clouds as shown in Figure 4(b), and use the pre-processed LSC3DPC models as our experimental dataset in this work. We see that the resulting bounding box in Figure 4(b) effectively includes valid points, which means that the size of the bounding box can be served as a good reference to figure out the effective range of captured target 3D scene.

B. Multiscale saliency estimation

Figure 5 shows the saliency maps estimated at three different scales. The saliency maps at coarse scales capture relatively large local areas of salient objects such as the black carrier in 'Room,' the red wall and the book-return machine in 'Library,' the dark floors in 'Parking lot,', and the whole area of the door in 'Korean house.' In contrary, the saliency maps obtained at fine scales detect locally detailed features, e.g., the door lock of the door in 'Room,' the speaker in 'Library,' and the roof in 'Korean house.' The final saliency maps shown in the last column are obtained by averaging the three hierarchical saliency maps obtained at different scales, where we see that both of the global saliency and the local details in LSC3DPCs are detected faithfully.

C. Comparison with existing method

We compare the performance of the proposed method with that of Yun and Sim's method [19] which is the only saliency detection method for LSC3DPCs using geometry and color features together. Note that, as we explained in Section IV-A, we regenerate the test dataset of LSC3DPCs by removing false points from the LSC3DPC dataset used in [19]. Therefore, when using the same parameters reported in the paper [19],

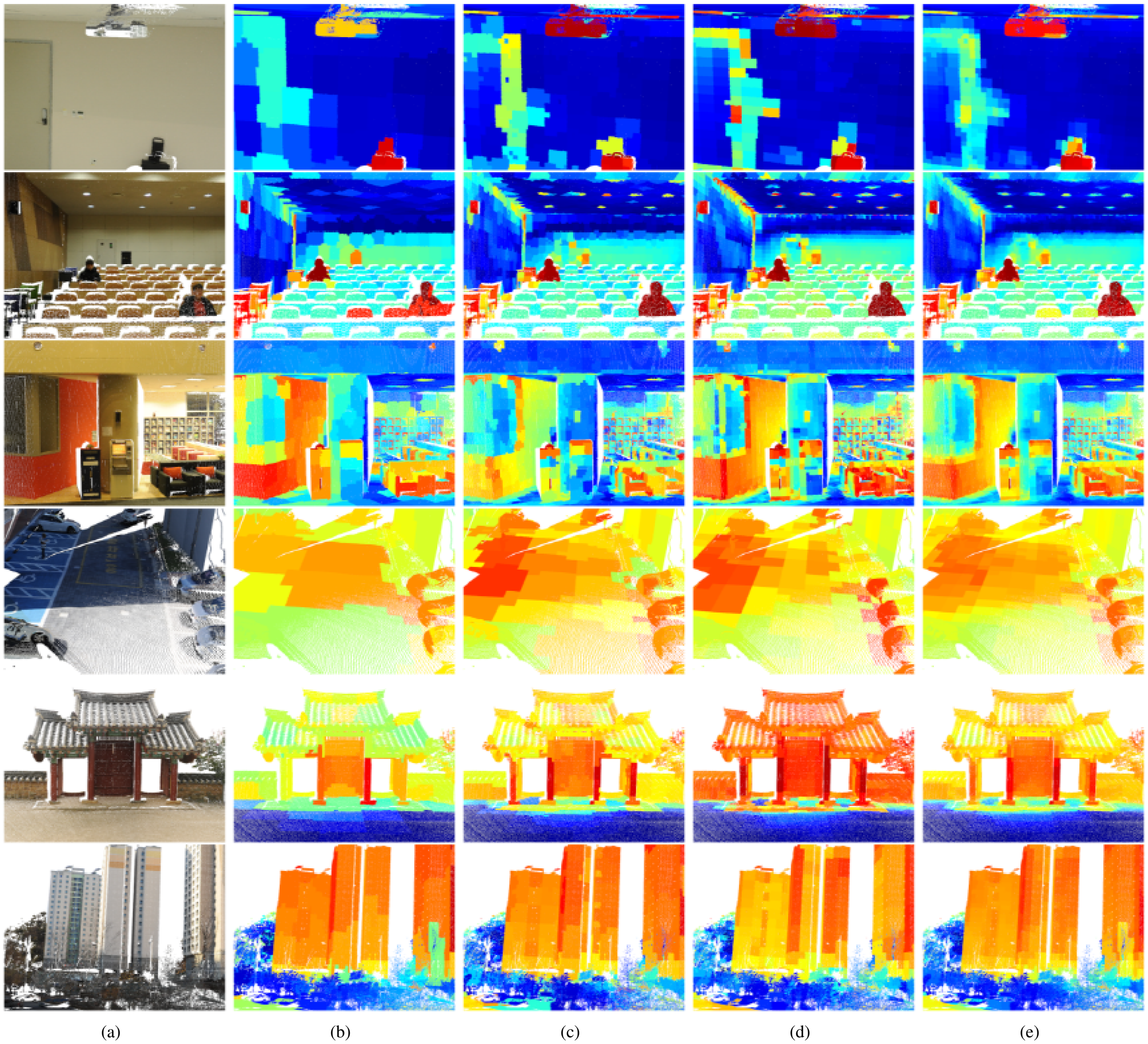


Fig. 5: Multiscale saliency distributions obtained by the proposed method. (a) Input LSC3DPC models. (b) Coarse-scale saliency maps. (c) Medium-scale saliency maps. (d) Fine-scale saliency maps. (e) Final saliency maps. From top to bottom, 'Room,' 'Classroom,' 'Library,' 'Parking lot,' 'Korean house,' and 'Dormitory.'

the resulting saliency maps may exhibit undesirable results which are different from the results in their paper. For fair comparison, we changed several parameters of the existing method [19] to make the resulting saliency maps look similar or better to that shown in their paper.

Figures 6 and 7 compare the resulting saliency maps of the proposed method and [19]. We see that [19] assigns relatively high saliency values to lots of non-salient regions such as the chairs in 'Classroom,' the floors in 'Korean house,' and the forests in 'Parking lot' and 'Dormitory.' In particular, [19] often fails to capture the relative importance of salient scene structures yielding similar geometric features to the neighbors but different color features from the neighbors, for example, the red wall in 'Library' and the white boards in 'Classroom.' Moreover, [19] exhibits limited performance to highlight globally salient structures completely such as the people in 'Room' and 'Classroom,' the red wall and the stair in 'Library,' the roof in 'Korean house,' and the buildings in 'Dormitory.' On the other hand, the proposed method faithfully detects the globally salient regions, e.g., the people in 'Room,' 'Classroom,' and 'Parkinglot', the cars and the white storage tanks in 'Parking lot,' the red wall in 'Library,' the houses in

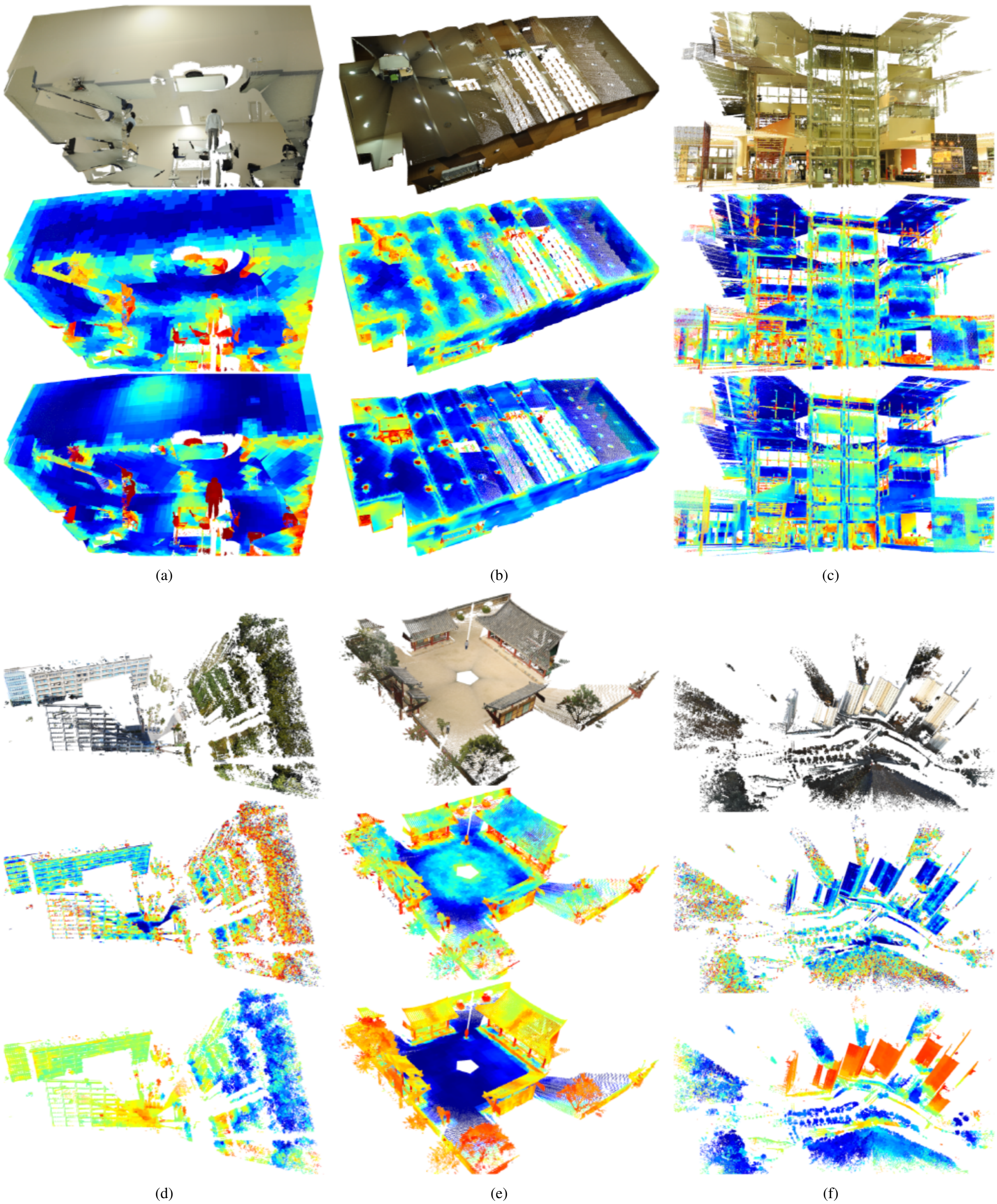


Fig. 6: Comparison of resulting saliency maps obtained by using Yun and Sim's method [19] and the proposed method, visualized in bird-eye view. Input LSC3DPC models (top), the saliency maps obtained by using Yun and Sim's method (middle), and the saliency maps obtained by using the proposed method (bottom). (a) 'Room,' (b) 'Classroom,' (c) 'Library,' (d) 'Parking lot,' (e) 'Korean house,' and (f) 'Dormitory.'

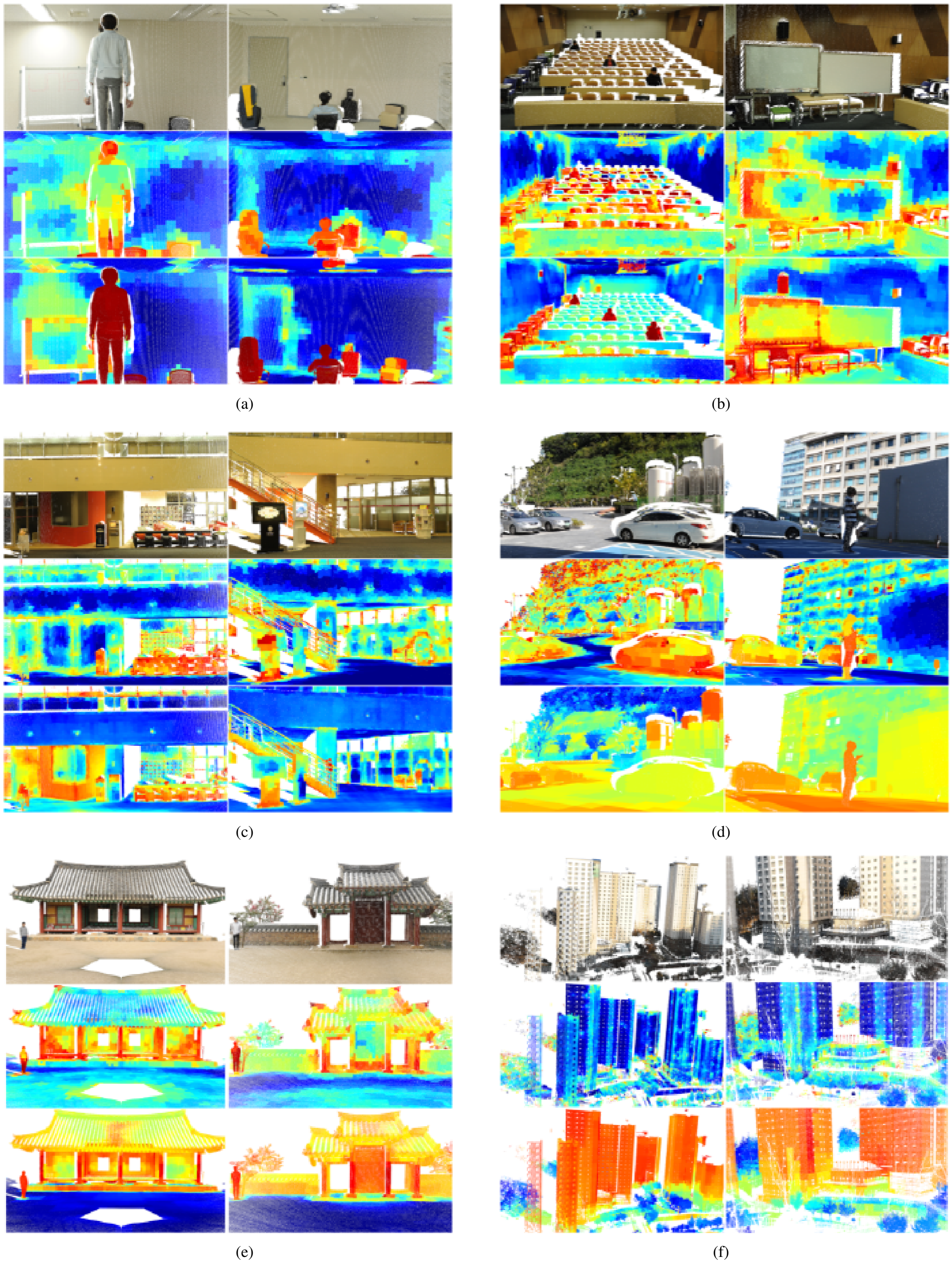


Fig. 7: Detailed comparison of saliency distributions. Input LSC3DPC models (top), the saliency maps obtained by using Yun and Sim's method [19] (middle), and the saliency maps obtained by using the proposed method (bottom). (a) 'Room,' (b) 'Classroom,' (c) 'Library,' (d) 'Parking lot,' (e) 'Korean house,' and (f) 'Dormitory.'

'Korean house,' and the buildings in 'Dormitory,' as well as the local details such as the projector and the boundaries of the white boards in 'Room' and 'Classroom,' and the roof patterns and the pillars of the house in 'Korean house.' In addition, the proposed method provides more smooth and natural saliency distributions, since we evaluate the cluster-wise saliency values using RW simulation where the representative features of a cluster are computed as the average features of all the points in the cluster.

V. Conclusions

In this paper, we proposed a RW-based multiscale saliency detection algorithm for LSC3DPCs. We first clustered an input point cloud at three different scales, and extracted geometric features of FPFH and Lab color features for each cluster. We constructed a fully-connected graph of clusters at each scale, respectively, where the edge weights are defined by evaluating the dissimilarities of geometry and color features between the clusters. We performed RW simulation on the graphs at three different scales, respectively, which are then averaged together to yield a final saliency distribution. Experimental results demonstrated that the proposed method successfully detects globally salient scene structures as well as locally detailed geometric and photometric features in LSC3DPCs. Our future work includes the generation of ground truth saliency for LSC3DPCs based on the perception characteristic of HVS.

VI. Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) within the Ministry of Science and ICT (MSIT) under Grant 2020R1A2B5B01002725.

References

[1] Y. Li, Q. Miao, K. Tian, Y. Fan, X. Xu, R. Li, and J. Song, "Large-scale gesture recognition with a fusion of rgb-d data based on saliency theory and c3d model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 10, pp. 2956–2964, 2018.

[2] H. Zhou, A. Liu, W. Nie, and J. Nie, "Multi-view saliency guided deep neural network for 3-d object retrieval and classification," *IEEE Trans. Multimedia*, vol. 22, no. 6, pp. 1496–1506, 2020.

[3] Z. Tao, H. Liu, H. Fu, and Y. Fu, "Multi-view saliency-guided clustering for image cosegmentation," *IEEE Trans. Image Process.*, vol. 28, no. 9, pp. 4634–4645, 2019.

[4] C. Lang, T. V. Nguyen, H. Katti, K. Yadati, M. Kankanhalli, and S. Yan, "Depth matters: Influence of depth cues on visual saliency," in *Proc. IEEE ECCV*, 2012, pp. 101–115.

[5] J. Wang, M. P. Da Silva, P. Le Callet, and V. Ricordel, "Computational model of stereoscopic 3d visual saliency," *IEEE Trans. Image Process.*, vol. 22, no. 6, pp. 2151–2165, June 2013.

[6] R. Ju, L. Ge, W. Geng, T. Ren, and G. Wu, "Depth saliency based on anisotropic center-surround difference," in *Proc. IEEE ICIP*, Oct. 2014, pp. 1115–1119.

[7] H. Peng, B. Li, W. Xiong, W. Hu, and R. Ji, "Rgbd salient object detection: A benchmark and algorithms," in *Proc. IEEE ECCV*, Sept. 2014, pp. 92–109.

[8] H. Song, Z. Liu, H. Du, G. Sun, O. Le Meur, and T. Ren, "Depth-aware salient object detection and segmentation via multiscale discriminative saliency fusion and bootstrap learning," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4204–4216, Sept. 2017.

[9] H. Chen and Y. Li, "Progressively complementarity-aware fusion network for rgb-d salient object detection," in *Proc. IEEE CVPR*, June 2018.

[10] C. H. Lee, A. Varshney, and D. W. Jacobs, "Mesh saliency," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 659–666, July 2005.

[11] X. Chen, A. Saparov, B. Pang, and T. Funkhouser, "Schelling points on 3d surface meshes," *ACM Trans. Graph.*, vol. 31, no. 29, pp. 29:1–29:12, July 2012.

[12] G. Leifman, E. Shtrom, and A. Tal, "Surface regions of interest for viewpoint selection," in *Proc. IEEE CVPR*, June 2012, pp. 414–421.

[13] R. Song, Y. Liu, R. R. Martin, and P. L. Rosin, "Mesh saliency via spectral processing," *ACM Trans. Graph.*, vol. 33, no. 6, pp. 6:1–6:17, Feb. 2014.

[14] S.-W. Jeong and J.-Y. Sim, "Saliency detection for 3d surface geometry using semi-regular meshes," *IEEE Trans. Multimedia*, vol. 19, no. 12, pp. 2692–2705, Dec. 2017.

[15] G. Kim, D. Huber, and M. Hebert, "Segmentation of salient regions in outdoor scenes using imagery and 3-d data," in *IEEE Winter Conf. on Appl. Comput. Vis.*, Jan. 2008, pp. 1–8.

[16] O. Akman and P. Jonker, "Computing saliency map from spatial information in point cloud data," in *Advanced Concepts for Intelligent Vision Systems*, 2010, pp. 290–299.

[17] E. Shtrom, G. Leifman, and A. Tal, "Saliency detection in large point sets," in *Proc. IEEE ICCV*, Dec. 2013.

[18] J. Leroy, N. Riche, and M. Mancas, "3d saliency based on supervoxels rarity in point clouds," in *Proc. IEEE ICCV*, Oct. 2015.

[19] J. S. Yun and J. Y. Sim, "Supervoxel-based saliency detection for large-scale colored 3d point clouds," in *Proc. IEEE ICIP*, Sept. 2016, pp. 4062–4066.

[20] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Susstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.

[21] Christian Böhm, Christos Faloutsos, Jia-Yu Pan, and Claudia Plant, "Robust information-theoretic clustering," in *Proc. Int. Conf. on Knowledge Discovery and Data Mining*, Aug. 2006, pp. 65–75.

[22] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (fpfh) for 3d registration," in *IEEE ICRA*, May 2009, pp. 3212–3217.

[23] V. Gopalakrishnan, Yiqun Hu, and D. Rajan, "Random walks on graphs for salient object detection in images," *IEEE Trans. Image Process.*, vol. 19, no. 12, pp. 3232–3242, Dec. 2010.

[24] H. Kim, Y. Kim, J.-Y. Sim, and C.-S. Kim, "Spatiotemporal saliency detection for video sequences based on random walk with restart," *IEEE Trans. Image Process.*, vol. 24, pp. 2552–2564, Aug. 2015.

[25] A. Boulch and R. Marlet, "Fast Normal Estimation for Point Clouds with Sharp Features using a Robust Randomized Hough Transform ," *Comput. Graph. Forum*, vol. 31, no. 5, pp. 1765–1774, Aug. 2012.

[26] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *Proc. IEEE ICRA* , May 2011.

[27] J. Papon, A. Abramov, M. Schoeler, and F. Wrgtter, "Voxel cloud connectivity segmentation - supervoxels for point clouds," in *Proc. IEEE CVPR*, June 2013, pp. 2027–2034.

[28] A. Barla, F. Odone, and A. Verri, "Histogram intersection kernel for image classification," in *Proc. IEEE ICIP*, Sept. 2003, vol. 3, pp. III–513.

[29] T. Liu, J. Sun, N.-N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," in *Proc. IEEE CVPR*, June 2007.

[30] C. Yang, L. Zhang, and H. Lu, "Graph-regularized saliency detection with convex-hull-based center prior," *IEEE Signal Process. Letters*, vol. 20, no. 7, pp. 637–640, July 2013.

[31] J. R. Norris, *Markov chains*, Cambridge University Press, U.K., 1997.

[32] RIEGL, "Riegl vz-400 3d terrestrial laser scanner," .