Multiscale Saliency Detection for 3D Meshes Using Random Walk

Se-Won Jeong and Jae-Young Sim
School of Electrical and Computer Engineering, UNIST, Ulsan, Korea
E-mails: bee911017@unist.ac.kr, jysim@unist.ac.kr

***Abstract*—We propose a novel multiscale saliency detection algorithm for 3D meshes based on random walk framework. We construct a weighted undirected graph on an input 3D mesh model, by taking the vertices and edges in the mesh as the nodes and links of the graph. We compute a curvature value at each vertex using position and normal information, and assign a high weight to an edge connecting two vertices which have distinct curvature values each other. We perform random walk on the graph and find the stationary distribution of random walker, which is used as an initial saliency distribution. Moreover, in addition to local curvature characteristics, we also reflect global attributes of 3D geometry for saliency detection. We employ the saliency distributions at coarser scale meshes as restarting distributions of the random walker at finer scale meshes, based on random walk with restart framework. Experimental results show that the proposed algorithm detects the overall salient regions in 3D meshes as well as their local geometric details, faithfully.**

I. Introduction

The human visual system (HVS) understands scene contents in an image based on selective perception of interested regions, called salient regions. Inspired by the property of HVS, researches have been performed to automatically extract salient regions from a given image. Itti *et al.* [1] proposed a center-surround method, assuming that an image region is salient when its feature is dissimilar to those of surrounding regions. They adopted various features to compute saliency maps at different scales. Costa [2] regarded saliency as visiting frequencies of random walker to the nodes of a fully connected graph defined on an image. Kim *et al.* [3] also adopted a random walk framework to find saliency maps at multiple image scales. They employed the saliency map computed at a coarse scale as a restarting distribution for biased random walk to compute the saliency map at a fine scale. Saliency detection techniques have been actively applied for various applications of image processing and computer vision, such as image segmentation [4], object recognition [5], and image compression [6].

While the saliency detection for images has drawn much attention, relatively little work has been done to detect saliency for 3-dimensional (3D) meshes. 3D mesh saliency detection has versatility in many 3D graphics applications. For example, mesh simplification can be performed efficiently by preserving important or salient geometric features. Also, optimal viewpoints can be automatically selected for fast rendering of visually salient regions of 3D meshes. There were several attempts to find saliency for 3D meshes. Lee *et al.* [7] computed saliency maps using the difference of Gaussian-weighted curvature values based on center-surround concept. Leifman *et al.* [8] computed a saliency map using a patch association method which considers the geodesic distances of a vertex from the vertices with high feature values. Song *et al.* [9] used spectral decomposition of Laplacian matrix to find saliency of 3D meshes.

However, The previous methods mainly use local geometric features to extract saliency of 3D meshes, and thus may not capture rough distribution of saliency. In this paper, we propose a novel graph-based saliency detection algorithm for irregular 3D meshes using random walk with restart. We first compute a geometric curvature feature value at each vertex in an input 3D mesh model. Then we construct an undirected weighted graph on the 3D mesh structure, where the weight is defined by using the curvature feature. Based on the graph, we obtain the steady-state distribution of random walker as a saliency distribution. Moreover, we also develop a hierarchical saliency detection scheme by taking the saliency distribution computed at a simplified mesh as a restarting distribution for random walker at a fine mesh, based on the framework of random walk with restart (RWR) [3] [10]. Experimental results show that the proposed algorithm finds overall salient regions as well as locally detailed geometric features, faithfully.

The rest of this paper is organized as follows. Section II describes the saliency detection algorithm of 3D meshes using random walk. Section III explains the multiscale saliency detection algorithm using random walk with restart. Section IV presents experimental results. Finally, Section V concludes the paper.

II. Random Walk Based Saliency Detection

Random walk is a mathematical modeling of probabilities of random movement [11]. In this section, we adopt the random walk framework to compute saliency for 3D meshes. We compute a curvature value as a geometric feature at each vertex. Then we construct a graph derived from the connectivity structure of an input 3D mesh model, and find the steady-state distribution of random walk on the graph which describes the saliency distribution on the 3D mesh model.

A. Feature Extraction

Curvature is one of the widely used geometric features in 3D mesh processing, which measures the deviation of 3D surface regions from flat surfaces. Fig. 1 shows an example

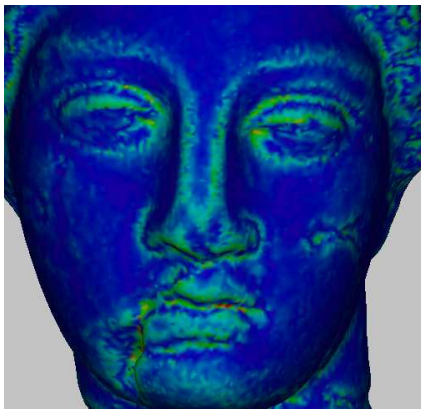


Fig. 1. Distribution of mean curvature values on the 'Venus' model.

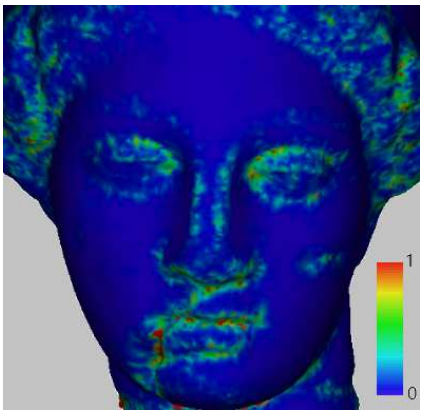


Fig. 2. Results of saliency detection using random walk.

of mean curvature values on the 'Venus' model, where the green color represents large curvature values. We can see that locally curved surface regions are assigned as relatively large curvature values. In this paper, we compute the Gaussian curvature [12] at each vertex as a geometric feature.

B. Saliency Computation

Let $\mathcal{M}(V, E)$ be a given 3D mesh model, where V and E denote the sets of vertices and edges, respectively. We define an undirected graph on $\mathcal{M}(V, E)$ by taking each vertex as a node. For each vertex $v_i \in V$, we compute a Gaussian curvature value κ_i. Then, we assign a weight w_{ij} to each edge e_{ij} connecting two vertices v_i and v_j, which is given by

$$w_{ij} = \begin{cases} |\kappa_i - \kappa_j| e^{-\alpha \frac{d_{ij}}{D}}, & \text{if } e_{ij} \in E, \\ 0 & \text{otherwise}, \end{cases} \tag{1}$$

where D denotes the diagonal length of the bounding box including the 3D mesh model, and d_{ij} is the Euclidean distance between v_i and v_j. α is empirically set to 100. Note that the edge weight w_{ij} becomes large when v_i and v_j yield distinct curvature feature values each other.

We apply the conventional random walk based saliency detection algorithm for images to compute the saliency of 3D meshes. We model an $|V| \times |V|$ transition probability matrix $\mathbf{P}$. Specifically, the (i, j)th element $\mathbf{P}(i, j)$ is the transition probability that the random walker moves from v_j to v_i. For saliency computation, $\mathbf{P}(i, j)$ should be proportional to the edge weight w_{ji}, so that the random walker at v_j is highly probable to jump to v_i which has a distinct curvature feature value from v_j. Hence we set

$$\mathbf{P}(i, j) = \frac{w_{ji}}{\sum_k w_{jk}} \tag{2}$$

to make the sum of all the outgoing probabilities from the vertex v_j becomes 1.

Note that, for each vertex v in a 3D mesh, only its 1-ring neighbor vertices are connected to v, which causes the most of the elements $\mathbf{P}(i, j)$ to become zero. However, the transition matrix $\mathbf{P}$ is irreducible if and only if the associated graph is strongly connected, that is, every vertices are reachable from any vertex [13]. When $\mathbf{P}$ is irreducible, we can find a unique steady-state probability distribution $\boldsymbol{\pi}$ which satisfies [11]

$$\boldsymbol{\pi}\mathbf{P} = \boldsymbol{\pi}. \tag{3}$$

Moreover, since $\mathbf{P}$ is symmetric, we can determine $\boldsymbol{\pi}$ such that the i-th element $\boldsymbol{\pi}_i$ is given by

$$\boldsymbol{\pi}_i = \frac{\sum_k w_{ik}}{\sum_j \sum_k w_{jk}}. \tag{4}$$

Finally, we compute a normalized saliency value s_i for each vertex v_i, which is proportional to the steady-state probability $\boldsymbol{\pi}_i$.

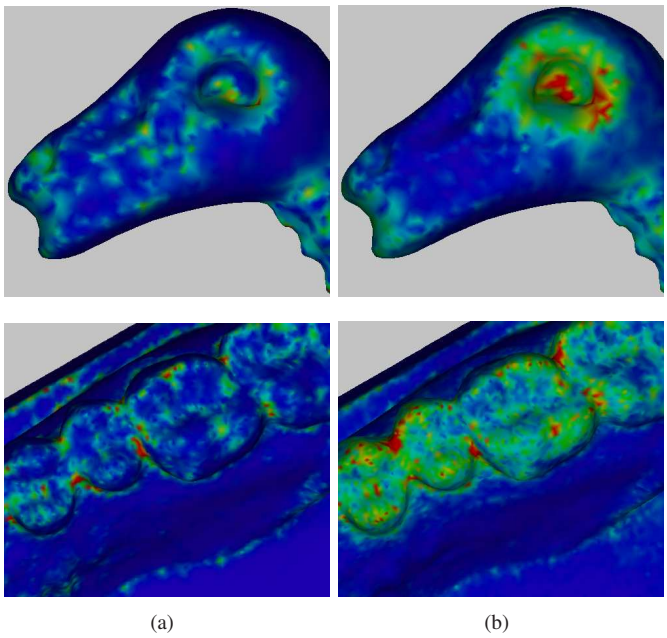
$$s_i = \frac{\boldsymbol{\pi}_i - \pi_{\min}}{\pi_{\max} - \pi_{\min}}, \tag{5}$$

where $\pi_{\min}$ and $\pi_{\max}$ denote the minimum and maximum values of $\boldsymbol{\pi}_i$. Fig. 2 illustrates the resulting saliency distribution on the 'Venus' model. It is observed that highly detailed or severely curved surface regions yield high saliency values, while smooth and flat surface regions have low saliency values.

III. Multiscale Saliency Detection

While the random walk technique detects geometrically salient features from 3D mesh models, as shown in Fig. 2, the saliency detection results tend to extract local features only, and often fail to capture global features. It is because the transition probability $\mathbf{P}(i, j)$ is non-zero only if e_{ij} is a valid edge in a 3D mesh model. Therefore, random transition is only occurred between v_i and v_j which are directly connected neighboring vertices each other.

Attempts have been made to overcome this drawback for 3D mesh saliency detection by averaging feature values of neighboring vertices [7] or by using patch-association method [8]. In this work, we propose a hierarchical saliency detection algorithm for 3D meshes which detects local and global features together based on the idea of multiscale saliency detection using RWR [3]. Fig. 3 compares the saliency detection results of random walk and RWR. We see that the overall salient regions, such as the eye region in the 'Dinosaur' model and the teeth region in the 'Teeth' model, are not detected in



(a) (b)

Fig. 3. Saliency detection results between ordinary random walk and RWR on the 'Dinosaur' model (top) and the 'Teeth' model (bottom). (a) Random walk and (b) RWR.

the ordinary RW, however, RWR extracts these overall salient regions as well as the locally detailed features faithfully.

A. RWR

RWR is a generalized version of random walk, which enforces the random walker to stay at nodes according to a given restarting distribution [10]. RWR is modelled by

$$(1-\epsilon)\boldsymbol{\pi}\mathbf{P} + \epsilon\mathbf{r} = \boldsymbol{\pi} \tag{6}$$

where $\mathbf{r}$ is a restarting distribution vector and ϵ is a restarting probability. In [3], the steady-state distribution obtained in a coarser scale image is used as a restarting vector for the random walk at the higher scale image.

Note that multiple scales of an image are easily obtained by simple up-down sampling. In order to generate different scales of 3D meshes, we employ mesh simplification as a corresponding tool to down-sampling of image. Then we find the steady-state distribution at a coarser scale mesh, and use it as a restarting distribution for RWR at the finer scale mesh. Consequently, RWR equation for 3D meshes is given by

$$\boldsymbol{\pi}^l = (1-\epsilon)\mathbf{P}^l\boldsymbol{\pi}^l + \epsilon\mathbf{r}^l, \tag{7}$$

where $\boldsymbol{\pi}^l$, $\mathbf{r}^l$, and $\mathbf{P}^l$ are the steady-state distribution, the restarting distribution, and the probability transition matrix, respectively, at the l-th scale mesh.

B. Restarting Distribution

We design the restarting distribution as

$$\mathbf{r}^l = U(\boldsymbol{\pi}^{l-1}) \tag{8}$$

where $U(\cdot)$ is an interpolating operator which maps the set of vertices V^{l-1} at $(l-1)$th scale to the set of vertices V^l at l-th scale. Note that, during the mesh simplification, the vertices in a simplified mesh are also maintained in the finer meshes, *i.e.,* $V^{l-1} \subset V^l$. Therefore, we implement $U(\cdot)$ by considering a curvature value of each vertex in a fine mesh as the averaged curvature value of the K-nearest neighboring vertices in the simplified mesh. Specifically, we determine the restarting distribution value $\mathbf{r}_i^l$ at the i-th vertex $v_i \in V^l$ as

$$\mathbf{r}_i^l = \begin{cases} \boldsymbol{\pi}^{l-1}_{f(v_i)}, & \text{if } v_i \in V^{l-1}, \\ \frac{1}{K}\sum_{k=1}^{K} \boldsymbol{\pi}^{l-1}_{f(v_{i,k})}, & \text{otherwise,} \end{cases} \tag{9}$$

where $f(v)$ is the index of v in V^{l-1}, and $v_{i,k}$ means the v_i's k-th nearest neighboring vertex in V^{l-1}. Fig. 4 shows the RWR results on a coarse scale mesh and a fine scale mesh on the 'Teeth' model. We see that the saliency detection result on the simplified mesh captures global features, and that on the fine mesh refines highly detailed local features.

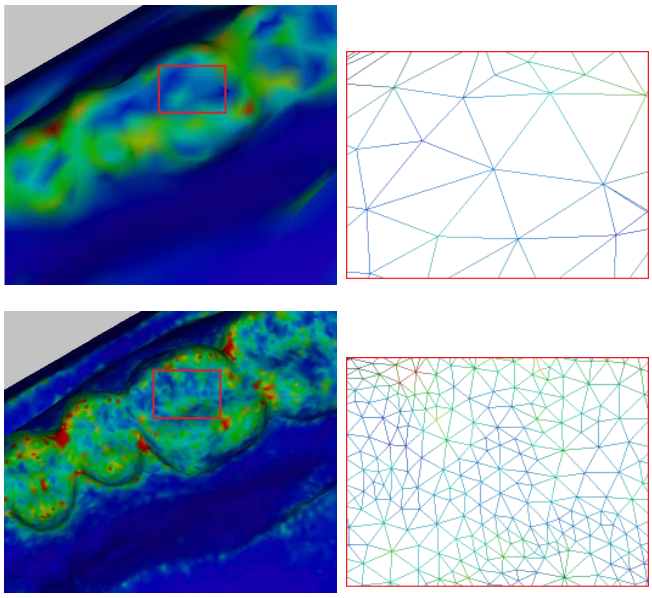


Fig. 4. Saliency detection results using RWR at a coarse scale mesh (top) and a fine scale mesh (bottom).

TABLE I
PROPERTIES OF TEST MESH MODELS.

Model	Number of Vertices	Number of Faces
Dinosaur	28,098	56,192
Teeth	58,303	116,602
Venus	67,173	134,342

IV. EXPERIMENTAL RESULTS

We test the performance of the proposed algorithm on the three mesh models: 'Dinosaur,' 'Teeth,' and 'Venus.' Table I provides the properties of test mesh models. We generate four different scales of meshes, 100%, 25%, 12.5%, and 3%, for each test model, by simplifying the original mesh using Qslim method [14].

Fig. 5 compares the saliency detection results of the proposed algorithm with that of the Lee *et al.*'s method [7]. We

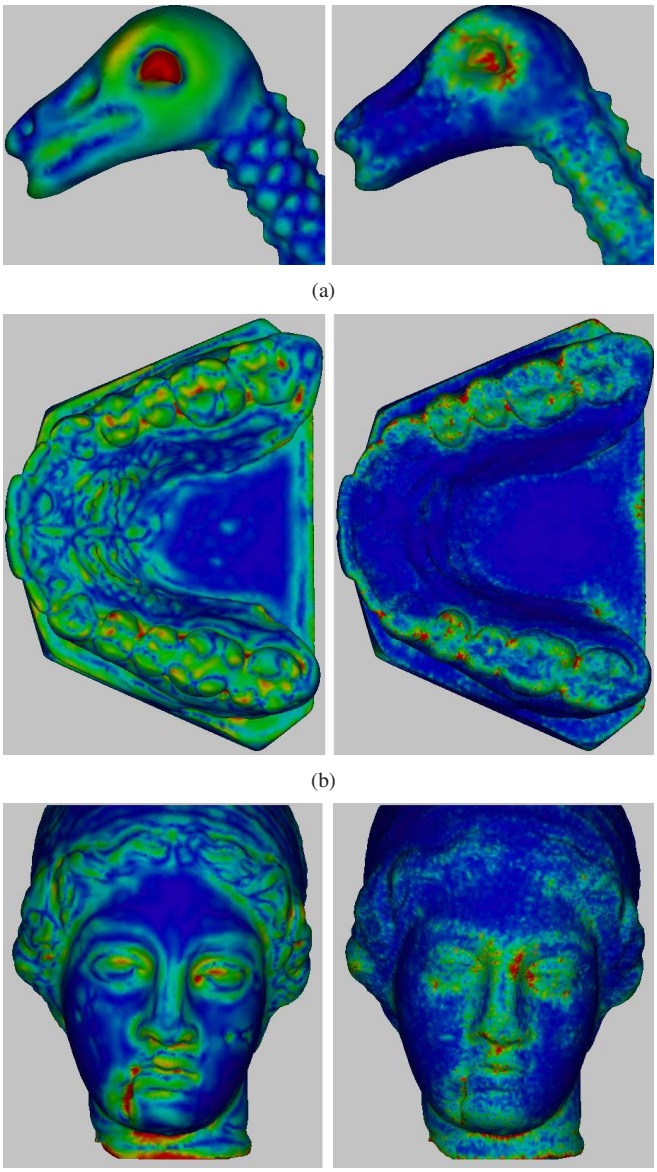


Fig. 5. Comparative results of saliency detection between the Lee *et al.*'s method [7] (left) and the proposed algorithm (right) on the (a) 'Dinosaur,' (b) 'Teeth,' and (c) 'Venus' models.

set the restarting probability ϵ in (7) as 0.3 in all experiments. As shown in Fig. 5(a), while the Lee *et al.*'s method highlights most of the head regions of the 'Dinosaur' as salient, the proposed algorithm only detects the interested region of eye. Also, in Fig. 5(b), the Lee *et al.*'s method extracts locally detailed regions of gum, however, the proposed algorithm only detects globally salient teeth regions. Similarly, in Fig. 5(c), the Lee *et al.*'s method detects most of the curved surface regions of the 'Venus' model including the hair part, but the proposed algorithm mainly focuses on the face region. Consequently, the proposed algorithm captures rough salient areas of an input 3D mesh model faithfully, while detects locally detailed geometric features accurately.

V. Conclusions

In this paper, we proposed a graph-based saliency detection algorithm for 3D meshes using RWR. We computed the Gaussian curvature at each vertex in an input 3D mesh model. We constructed a Markov chain on the graph of mesh structure, where each edge is assigned a weight which reflects the difference of curvature values between two connected vertices. We computed a saliency distribution of the 3D mesh by finding the steady-state distribution of random walk on the undirected graph. We also employ the saliency distribution at a coarse scale mesh to design a restarting distribution for RWR at a fine scale mesh. Experimental results demonstrated that the proposed multiscale saliency detection algorithm reliably captures rough salient regions of 3D mesh model as well as extracts local geometric details.

Future work may include to investigate a robust saliency detection algorithm for 3D meshes considering the effect of structural characteristics on the global features.

References

[1] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, Nov 1998.

[2] L. da Fontoura Costa, "Visual saliency and attention as random walks on complex networks," *ArXiv Physics e-prints*, Mar. 2006.

[3] J.-S. Kim, J.-Y. Sim, and C.-S. Kim, "Multiscale saliency detection using random walk with restart," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 24, no. 2, pp. 198–210, Feb 2014.

[4] J. Han, K. N. Ngan, M. Li, and H.-J. Zhang, "Unsupervised extraction of visual attention objects in color images," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 16, no. 1, pp. 141–145, Jan 2006.

[5] L. Wang, J. Xue, N. Zheng, and G. Hua, "Automatic salient object extraction with contextual cue," in *IEEE Int. Conf. Computer Vision (ICCV)*, Nov 2011, pp. 105–112.

[6] C. Guo and L. Zhang, "A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression," *IEEE Trans. Image Processing*, vol. 19, no. 1, pp. 185–198, Jan 2010.

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

[8] G. Leifman, E. Shtrom, and A. Tal, "Surface regions of interest for viewpoint selection," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, June 2012, pp. 414–421.

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

[10] J.-Y. Pan, H.-J. Yang, C. Faloutsos, and P. Duygulu, "Automatic multi-media cross-modal correlation discovery," in *Proc. Int. Conf. Knowledge Discovery and Data Mining*, ser. KDD '04. ACM, 2004, pp. 653–658.

[11] P. G. Doyle and J. L. Snell, *Random walks and electric networks*, ser. Carus mathematical monographs. Mathematical Association of America, 1984.

[12] G. Taubin, "Estimating the tensor of curvature of a surface from a polyhedral approximation," in *Proc. Int. Conf. Computer Vision*, Jun 1995, pp. 902–907.

[13] R. B. Bapat and T. E. S. Raghavan, *Nonnegative matrices and applications*. U.K.: Cambridge University Press, 1997.

[14] M. Garland and P. S. Heckbert, "Simplifying surfaces with color and texture using quadric error metrics," in *Proc. Conf. Visualization*, ser. VIS '98. Los Alamitos, CA, USA: IEEE Computer Society Press, 1998, pp. 263–269.