A Concise and Provably Informative Multi-Scale Signature Based on Heat Diffusion

April 17, 2024

1 General Overview

In this research paper [1], it focuses on searching the shape signature. There are certainly several approaches to the idea, structure detection, partial matching, but these methods still lack of effective processing, and can perform with specific application. So, the properties of this research should be multi-scale and informative, including effective.

Heat Kernel Signature (HKS) computes point signature by analysing from the neighborhood of a point on a discrete surface to compare the matching shape. As the time progresses, heat will diffuse to the larger neighborhoods, which means heat diffusion can represent the local shape features. In other words, the matching shape have a same heat progression, or signature in a long period of time. It can be used in many applications, for example, shape registration, partial matching.

This project is inspired from various past researches. Global Point Signature on a certain point relies on eigenvectors and eigenvalues of Laplace-Beltrami operator [2], which is sensitive to the noise. Another example is Diffusion map and distance [3], which is similar to this paper.

2 Heat Operator and Heat Kernel

The heat diffusion is analysed on Riemannian manifolds M, which is significant to Heat Kernel Signature. This operator is defined by the heat equation below (Eq. 1):

$$\Delta_M u(x,t) = -\frac{\partial u(x,t)}{\partial t} \tag{1}$$

where Δ_M means the Laplace-Beltrami operator on M. $u(x,t)$ is the Dirichlet boundary condition function that satisfied the equation $u(x,t) = 0$ with any vertex $x \in \partial M$ and time t. With initialized value of heat distribution f, the heat operator $H_t(f)$ stand for heat distribution at the specific time t, which can be defined by equation:

$$H_t f(x) = \int_M k_t(x,y) f(y)\, dy \tag{2}$$

Upon the equation 2, $k_t(x,y)$ is the minimum function that satisfied an equation, called heat kernel. The heat kernel on compact M can be calculated by eigenfunction decomposition of Laplacian-Beltrami operator. In this experiment, heat diffusion is applied on 2 sample meshes, *cow.obj* and *human.obj* [4].

$$k_t(x,y) = \sum_{i=0}^{\infty} e^{-\lambda_i t}\phi_i(x)\phi_i(y) \tag{3}$$

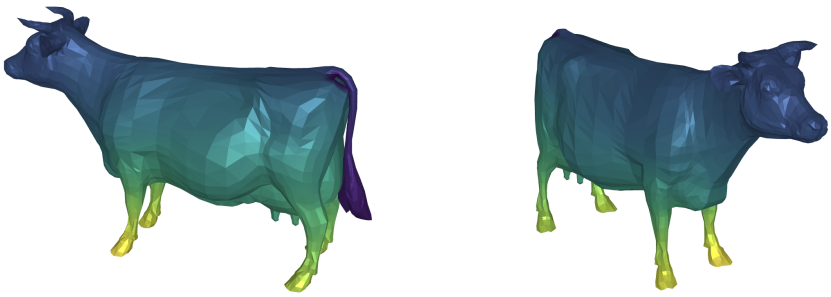


Figure 1: Heat Operator on *cow.obj* at time of 0.5 with 300 eigenvectors

Figure 2: Heat Operator on *human.obj* at time of 3 with 300 eigenvectors

3 Heat Kernel Signature (HKS)

According to the paper [1], heat kernel is informative which contains the information of geometry of the shape. This is because it is operated from heat equation $\Delta_M u(x,t) = -\frac{\partial u(x,t)}{\partial t}$, which is consist of spatial data changing over a period of time. Given a point $x \in M$, Heat Kernel Signature is defined by the equation:

$$HKS(x,t) = k_t(x,x)$$

Functionally, if the eigenvectors, calculated from Equation 3, are not identical, the equation can be converted to analyse a point x over an function of time t.

$$k_t(x,x) = \sum_{i=0}^{\infty} e^{-\lambda_i t}\phi_i(x)^2 \tag{4}$$

Figure 3: Heat Kernel Signature on *cow.obj* at time of 0.5 with 300 eigenvectors

Figure 4: Heat Kernel Signature on *human.obj* at time of 3 with 300 eigenvectors

With heat kernel signature, the different between heat operator (Figure 1 and Figure 2) and heat kernel signature (Figure 3 and Figure 4) can be seen that the latter one have the heat moving faster than the first one. On the other hands, the graphs, which is plotted between time interval of $[t_1, t_2]$, visualize the approach of heat throughout the mesh. The time interval is set where $t_1 = 4ln10/\lambda_{300}$ and $t_2 = 4ln10/\lambda_2$, where λ_2 and λ_{300} are 2nd and 300th eigenvalues respectively. The progresses of heat between two methods are same development with different value.

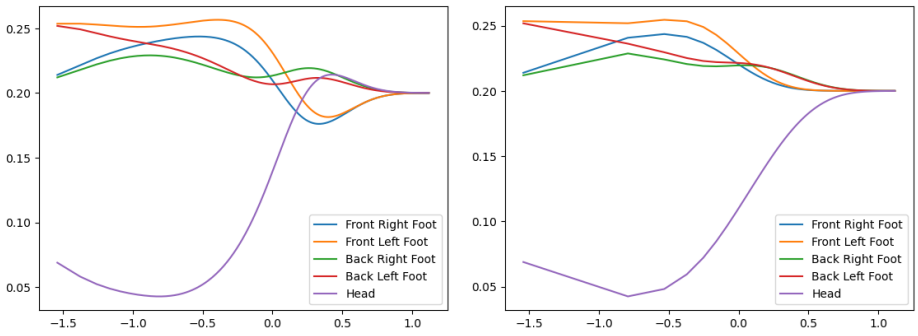


Figure 5: Left: Heat Kernel Signature on *cow.obj* Right: Heat Operator on *cow.obj*

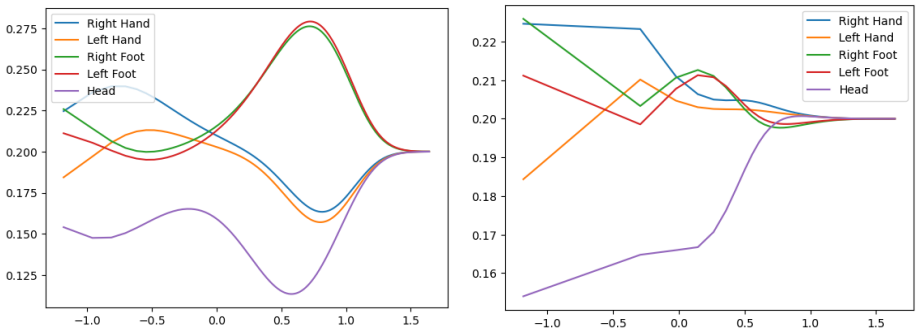


Figure 6: Left: Heat Kernel Signature on *human.obj* Right: Heat Operator on *human.obj*

4 Discrete Setting

In this section, it explain how to implement mesh Laplace operator, proposed by Mikhail Belkin [5]. It is mentioned that cotangent Laplace operator is not good for convergence [1]. Instead, it uses mesh Laplace operator to estimate heat, which is robust and stable. The mesh Laplace operator L is a matrix that is defined by inverse of diagonal matrix of areas at the vertices A, and weight matrix of neighborhoods W $[L = A^{-1}W]$. From the equation above $\Delta_M u(x,t) = -\frac{\partial u(x,t)}{\partial t}$, it can be converted to the form $u_t = e^{-tL}u_0$, where u_0 is an initial heat distribution, and e^{-tL} is a matrix exponential.

$$e^{-tL} = \sum_{i=0}^{\inf} \frac{(-tL)^i}{i!}$$

A matrix exponential can be viewed as heat operator, so combining with Equation 2 and Equation 3, it can clarify that we can use eigenvectors and eigenvalues to determine heat kernel that is stable, or non-sensitive data. Eventually, the mesh Laplace operator is defined by the equation:

$$L_K^h f(w) = \frac{1}{4\pi h^2} \sum_{t \in K} \frac{\text{Area}(t)}{\#t} \sum_{p \in V(t)} e^{\frac{-\|p-w\|^2}{4h}} (f(p) - f(w)) \tag{5}$$

5 Multi-Scale Matching

The purpose of this paper is to capture the heat kernel signature of meshes at different scales. The paper shows 2 heuristic to observe the trajectory of heat distribution over a function of time. The first heuristic is finding the difference between $k_t(x', x')$, where x' is a point that matches at other scale, and $k_t(x, x)$ during the interval of time $[t_1, t_2]$. As the time increase, the difference of larger scale is small compared to smaller scale. To reduce the error, it is suggested to normalize the heat $k_t(x, x)$ over the manifold M, $\int_M k_t(x, x)\, dx$ to contribute the shape of signature uniformly, called heat trace. Next, the heat variation at small t has more visible effect on large t. This is due to the heat distribution's behavior is defined by the average of heat dissipation in neighborhood. Thus, at the smaller scale, the signature can change logarithmically and can represent HKS well.

With these observations, the difference between 2 Heat Kernel Signature can be determined during $[t_1, t_2]$ by the equation:

$$d_{[t_1,t_2]}(x, x') = \left(\int_{t_1}^{t_2} \left(\frac{|k_t(x, x) - k_t(x', x')|}{\int_M k_t(x, x) dx} \right)^2 d \log t \right)^{1/2} \tag{6}$$

The images below visualize the difference of heat kernel signature between selected vertex and other vertices in the mesh between time interval of $[t_1, t_2]$. From Figure 7, the vertices, which are used on *cow.obj* and *human.obj*, are set on front right leg and right hand respectively. It can be seen that the shape that is similar to the selected area has a low difference. The difference can be seen in scale below.

Low **High**

Figure 7: Heat Kernel Signature distance at time interval of [1, 10] Left: Heat Kernel Signature distance on *cow.obj* Right: Heat Kernel Signature distance on *human.obj*

The images above (Figure 7) use the time interval between $t_1 = 1$ and $t_2 = 10$. On the other hands, if the time interval is set between $t_1 = 0.01$ and $t_2 = 1000000$, the gap will be too large and it cannot visualize the difference accurately (Figure 8) as the heat will have the same progress at the end, according to Figure 5 and Figure 6. In contrast, if the gap of time is small, like Figure 9 that uses time interval of [1, 2], it cannot evaluate the difference of heat diffusion too. Based on Figure 9, there is no difference in cow feet, which is contrasting to Figure 7, and human hands' heat should differ from human feet', which is not applied in the image.

Figure 8: Heat Kernel Signature distance at time interval of [0.01, 1000000] Left: Heat Kernel Signature distance on *cow.obj* Right: Heat Kernel Signature distance on *human.obj*

Figure 9: Heat Kernel Signature distance at time interval of [1, 2] Left: Heat Kernel Signature distance on *cow.obj* Right: Heat Kernel Signature distance on *human.obj*

6 Extension

To evaluate the quality of the implementation, the program will be applied in shape retrieval to classify meshes. Maks Ovsjanikov proposed a paper [6], which consist of multiple evaluation methods, and one of them is feature description and bag of features. In this paper, it uses these algorithms to approach a classification for shape retrieval.

6.1 Feature Description

The approach to observe the similarity is converting heat kernel signature to groups of feature-based vocabulary, similar to groups of words. For each vertex $x \in X$, where X is a set of mesh vertices, it can define heat kernel signature as a set of features $P = (p_1(x), p_2(x), ..., p_n(x))$ where element is defined as

$$p_i(x) = c(x)K_{\alpha^{i-1}t_0}(x, x) \tag{7}$$

In this experiment, $c(x)$ is a constraint to satisfy the condition $||p(x)||_2 = 1$. The initial timer t_0 is set to be 0.1, and alpha value α is 1.32.

6.2 Bag of Features

The vocabulary is a set of centers of clustering defined as $P = (p_1, p_2, ..., p_v)$, where v is number of groups. The method to define the centers P is an unsupervised learning, or K-Mean algorithm on feature description $p(x)$. Each group has its own feature descriptor p_v, and for every vertex x, feature distribution $\theta(x)$ is a set of probability $\theta_j(x)$ of the point x relating to the descriptor p_j, for j in range of [0, v], defined as:

$$\theta_j(x) = c(x)e^{-\frac{||p(x)-p_j||_2^2}{2\sigma^2}} \tag{8}$$

where each set have constraint $c(x)$ to meet the condition $||\theta(x)||_1 = 1$. The sigma value is found that it has the best value at twice of the median distance between the cluster centers $p_1, p_2, ..., p_v$. By integrating the cluster from equation 9, the bag of features is obtained.

$$f(X) = \int_X \theta(x) da(x) \tag{9}$$

At the end, to define the similarity between two meshes, it can be calculated from the L1 distance of bag of features.

$$d_{BoF}(X, Y) = ||f(X) - f(Y)||_1 \tag{10}$$

7 Evaluation

The first evaluation is to observe the similarity of heat on corresponding vertices. Figure 11 visualizes the similarity by embedding Multi-Dimension Scale. The meshes that are used for observation are human meshes, imported from Articulated mesh animation [4] (Figure 10), and the vertices' heat are captured from Right hand, Left hand, Right foot, Left foot and Head. The graph demonstrates that different meshes can have similar heat diffusion at the same vertices.

Figure 10: Example of human meshes using for evaluation [4]

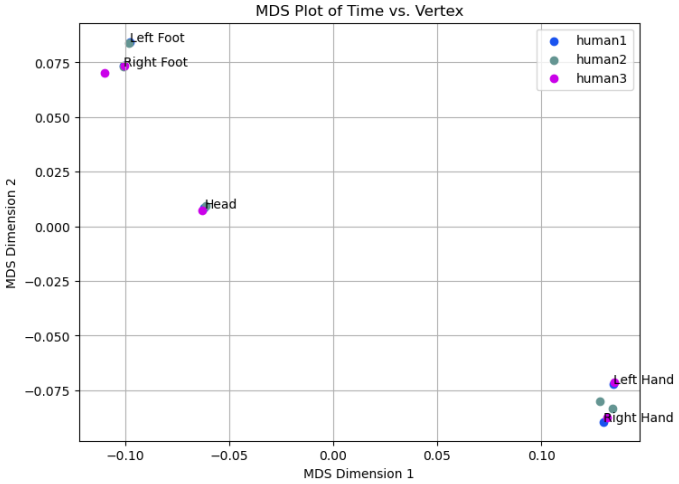


Figure 11: Multi-Dimension Scale of heat kernel signature on 3 human meshes

The experiment for K-Nearest Nearest uses dataset imported from SHREC'15. [7, 8] The dataset is consist of 7 classes, which are "santa", "horse", "dog", "bird", "laptop", "female" and "male" (Figure), and each class has 10 meshes. Each mesh is applied with bag of features algorithm as the features, and split into training set 80 percents and testing set 20 percents.

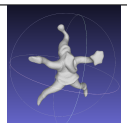
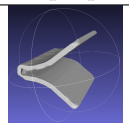
santa	horse	dog	bird	laptop	female	male
						

Table 1: The examples of meshes for each class [7, 8]

The task is to classify the meshes by using K-Nearest Neighbor. After training the model, the result will be as precision-recall curve to visualize the accuracy. As a result (Figure 12), the accuracy of the model is not high due to small amount of training dataset. To improve the model, it requires more mesh dataset.

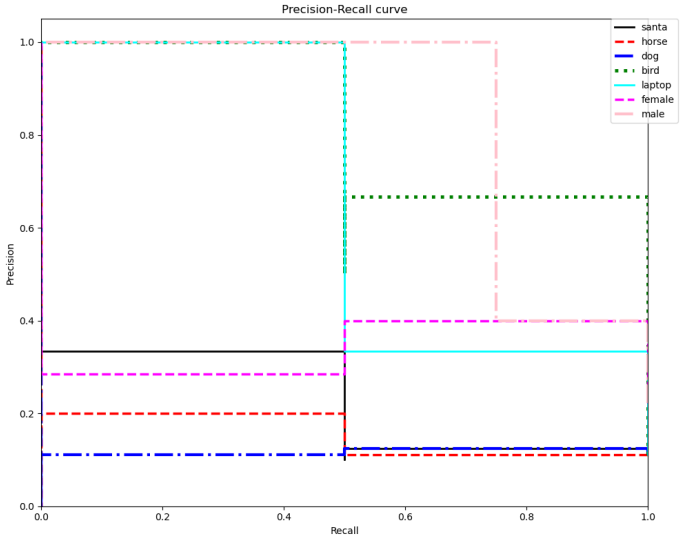


Figure 12: PR curve of K-Nearest Neighbor for shape retrieval

8 Discussion

According to the paper [1], there are still many limitations of heat kernel signature. The program needs to compute for a long time, so there should be an adjustment to reduce computational time which it suggested to use an equation $K_t = e^{-tL}A$. However, it is difficult to implement exponential matrix. The next problem is that it cannot compute with symmetric geometry shape, for instance, sphere, cube. Since it process a heat feature with eigenfunctions, the function cannot handle a

symmetric shape. The final one is to find the most suitable time interval of heat kernel signature at different time automatically. In the code, the interval is defined between $4ln10/\lambda_{300}$ and $4ln10/\lambda_2$, which is good for certain shapes but not all. To be applied in practical application, it needs to find any shape for user experience, so it is necessary to find a time interval based on the selected shape. This part should focus on the effect of shape character on heat progression over a time.

9 Conclusion

Heat kernel signature is a method to observe heat movement on the vertices of mesh to determine shape signature. It analyses heat progression based on vertex neighbors, where heat diffusion can represent local shape through time interval. The movement can be visualized with eigenvalues and eigenvectors, which are implemented in $k_t(x,x) = \sum_{i=0}^{\infty} e^{-\lambda_i t}\phi_i(x)^2$ (Equation 4). The difference of heat diffusion in a certain period of time can highlight the similarity of shapes well as its signature. It can be used for shape retrieval to observe the similar shape by using heat diffusion as features, and categorize them as bag of feature to reduce dimension. With this feature, it can find the class of mesh by K-Nearest Neighbor algorithm.

References

[1] J. Sun, M. Ovsjanikov, and L. Guibas, "A Concise and Provably Informative Multi-Scale Signature Based on Heat Diffusion," *Computer Graphics Forum*, 2009.

[2] R. M. Rustamov, "Laplace-Beltrami Eigenfunctions for Deformation Invariant Shape Representation," in *Geometry Processing*, A. Belyaev and M. Garland, Eds. The Eurographics Association, 2007.

[3] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker, "Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps," *Proceedings of the National Academy of Sciences*, vol. 102, no. 21, pp. 7426–7431, 2005. [Online]. Available: https://www.pnas.org/doi/abs/10.1073/pnas.0500334102

[4] D. Vlasic, I. Baran, W. Matusik, and J. Popović, "Articulated mesh animation from multi-view silhouettes," *ACM Trans. Graph.*, vol. 27, no. 3, p. 1–9, aug 2008. [Online]. Available: https://doi.org/10.1145/1360612.1360696

[5] M. Belkin, J. Sun, and Y. Wang, "Discrete laplace operator on meshed surfaces," in *Proceedings of the Twenty-Fourth Annual Symposium on Computational Geometry*, ser. SCG '08. New York, NY, USA: Association for Computing Machinery, 2008, p. 278–287. [Online]. Available: https://doi.org/10.1145/1377676.1377725

[6] M. Ovsjanikov, A. M. Bronstein, M. M. Bronstein, and L. J. Guibas, "Shape google: a computer vision approach to isometry invariant shape retrieval," in *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, 2009, pp. 320–327.

[7] Z. Lian, A. Godil, B. Bustos, M. Daoudi, J. Hermans, S. Kawamura, Y. Kurita, G. Lavoué, H. V. Nguyen, R. Ohbuchi, Y. Ohkita, Y. Ohishi, F. Porikli, M. Reuter, I. Sipiran, D. Smeets, P. Suetens, H. Tabia, and D. Vandermeulen, "SHREC'11 track: shape retrieval on non-rigid 3d watertight meshes," in *Proceedings of the 4th Eurographics conference on 3D Object Retrieval*, ser. EG 3DOR'11. Eurographics Association, 2011, pp. 79–88.

[8] D. Pickup, X. Sun, P. L. Rosin, R. R. Martin, Z. Cheng, Z. Lian, M. Aono, A. Ben Hamza, A. Bronstein, M. Bronstein, S. Bu, U. Castellani, S. Cheng, V. Garro, A. Giachetti, A. Godil, J. Han, H. Johan, L. Lai, B. Li, C. Li, H. Li, R. Litman, X. Liu, Z. Liu, Y. Lu, A. Tatsuma, and J. Ye, "SHREC'14 track: Shape retrieval of non-rigid 3d human models," in *Proceedings of the 7th Eurographics workshop on 3D Object Retrieval*, ser. EG 3DOR'14. Eurographics Association, 2014.