A short report on shape comparison metrics

Martin Metodiev

March 4, 2025

1 Introduction

Here we review some useful metrics for shape analysis. Ideally, a shape metric will be independent of the relative spatial location of two objects, and also rotational and scaling variations [1], [3]. An intuitive approach is to use the Hausdorff distance as a measure of similarity between two sets A and B:

$$d_H(A, B) = \max\left(\sup_{a \in A} \inf_{b \in B} \|a - b\|, \sup_{b \in B} \inf_{a \in A} \|a - b\|\right) \tag{1}$$

Unfortunately this approach does not take into account the connectivity of the mesh and only considers the vertices. This implies that physical information, such as roughness and deformation is neglected. Furthermore, it is not invariant to translations and rotations and therefore meshes should be aligned or coregistered prior to any comparisons. Another approach is to use the the spectrum of the Laplace-Beltrami operator, as it is translation and rotation invariant and in addition it can take into account the connectivity of the mesh [1], [2]. Here we briefly review the use of Hausdorff distance and the ShapeDNA approach [1] for the purpose of comparing 3D meshes.

2 Methods

Python bindings for the vtk library were used to generate various meshes. For simplicity, here we only consider triangular meshes, although the ShapeDNA method can be extended to quad-meshes.We investigate three simple geometrical shapes - spheres, cuboids and pyramids - across two categories: translated, scaled and deformed. The final dataset categories are as follows:

- cuboids (translated and scaled)
- cuboids (translated/scaled and deformed)
- spheres (translated)
- spheres (translated and deformed)
- pyramids (deformed)

For each of those categories we generate around 50 random shapes. Deformation was simulated in the following manner: for the cuboids, a surface deformation was implemented using a 3D sinusoidal noise pattern. Vertex positions were displaced along their surface-normal directions with a magnitude modulated by multiplicative sine/cosine functions of their Cartesian coordinates, creating organic surface undulations while preserving the fundamental cubic structure. Sphere

deformation was achieved via a radial scaling modulated by angular-frequency sinusoidal noise. Vertex positions were displaced along their radial vectors using a superposition of two sinusoidal patterns in spherical coordinates ($8\phi/4$ and $16\phi/8$ frequencies), scaled by noise scale. This created axis-asymmetric surface perturbations while preserving spherical topology. As for the pyramids, two concurrent deformations were applied: a helical transform along the z axis, and a surface perturbation via Gaussian white noise.

The Hausdorff distance was computed with a custom numpy implementation. The Laplace-Beltrami spectrum was also computed via a custom implementation, using scipy and inspired by the LaPy Python package, which contains a readily available implementation of the ShapeDNA algorithm. Eigenvalues were computed for all of the shapes outlined above and Euclidean distance was used to compare Eigenvalue spectra. Then, tSNE (scikit-learn) was performed on the Eigenvalue spectra for all shapes and a two timensional embedding was obtained. For clarity, all the code used to obtain these results is available here.

3 Discussion

Let us review the performance of the Hausdorff and the ShapeDNA methods. To start with, we we investigate the distance measures obtained from those methods for a small subset of our data, shown on the following figure
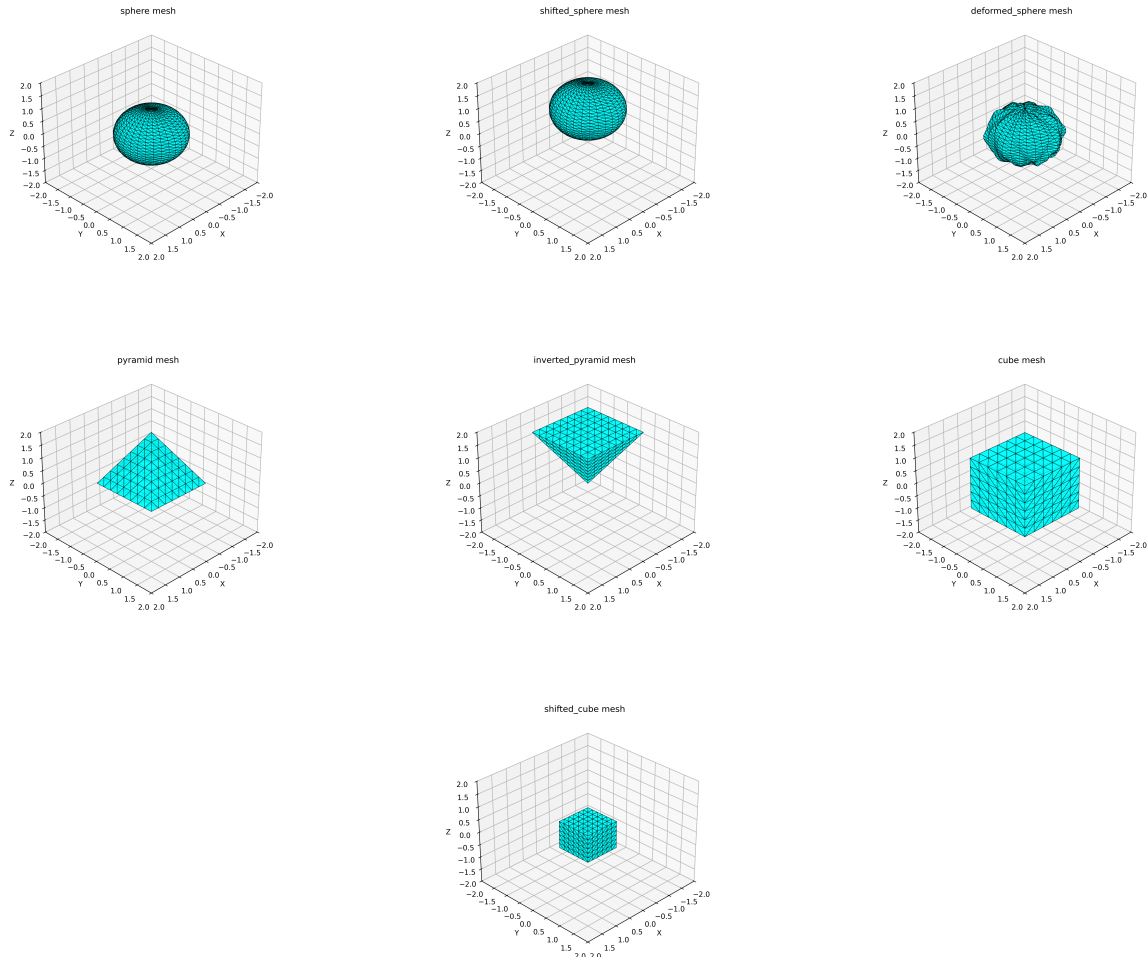

Figure 1: Some shapes.

The shapes presented in Figure 1 are a sphere, a translated sphere, a deformed sphere, a pyramid, an inverted pyramid, a cube and a translatedi and scaled down cube. We then compute the Hausdorff distance and the Euclidean distance between the Eigenvalues for all pair combinations. The results of this calculation are presented in the following distance matrices:
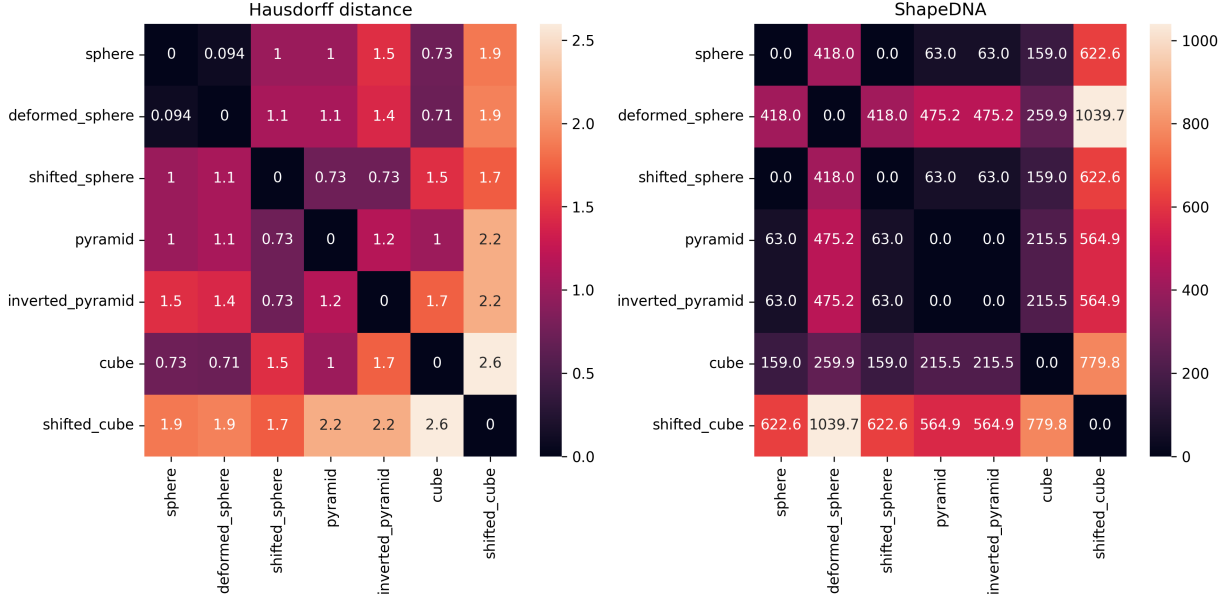
Figure 2: Left: Hausdorff distance matrix. Right: Euclidean distance between Laplace-Beltrami Eigenvalues distance matrix.

We can see in Figure 2 that the Hausdorff metric fails to take into account translation and rotation (see sphere vs shifted sphere and pyramid vs inverted pyramid). For example, distance between a sphere and a shifted sphere is the same as that between a sphere and a pyramid, and interestingly even greater between a sphere and an inverted pyramid. More notably, the the Hausdorff distance between the sphere and the deformed sphere is small - this is because most of the deformation information is encoded in the connectivity of the mesh, rather than the point cloud data.

As for the ShapeDNA method, the deformed sphere exhibits the largest distance to all other objects. As expected, the translated and rotated objects have the same Eigenvalues (and hence Euclidean distance of zero) - in particular the sphere and the shifted sphere and pyramid and the inverted pyramid. Interestingly, the deformed sphere is least similar to the shifted and scaled down cube and most similar to the zero-cented unit cube. Here we have used 75 Eigenvalues of the Laplace-Beltrami operator; it is important to point out that Euclidean distance may not be the best choice of a distance metric due to the high dimensionality; other strategies for comparing the spectra are discussed in the original paper [1]. Nevertheless, we still obtain intuitive and interpretable results. To understand the behaviour of this metric better, we next shift our attentiont to the entire dataset.
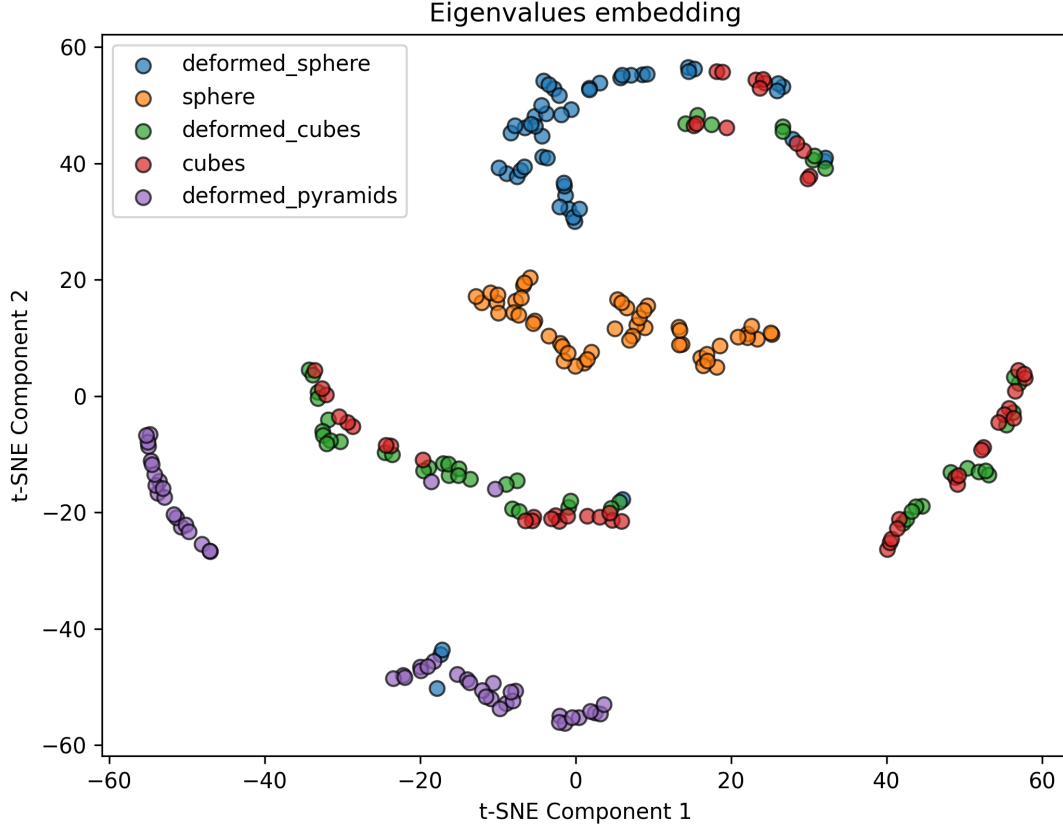
Figure 3: tSNE embedding of all 255 shape Eigenvalues.

The embedding in Figure 3 reveals that the ShapeDNA method successfully captures the similarity and differences between the five groups we used in this case study. The deformed pyramids segment nicely from the other groups. Cubes and deformed cubes overlap - it would be interesting to see what would happen if we increased the level of deformation. In contrast to this, spheres and deformed spheres seem to be separated in the embedding space, with some cubes and deformed cubes sneaking into the deformed spheres category. This result reveals that the Laplace-Beltrami Eigenvalues spectrum successfully captures the differences and similarities between our simple geometrical shapes. Here we use 75 Eigenvalues for each object, so it would be interesting what an optimal number of Eigenvalues to use would be.

4   Conclusions

We successfully implemented and tested a popular method for comparing 3D meshes. With the aid of dimensionality reduction we can see that the Eigenvalues spectrum holds valuable information about the spatial structure of such meshes. The Eigenvalues can also potentially be used as an input to a neural network. Another interesting approach for comparing shapes, which is not discussed here is with the use of the hyperbolic Wasserstein distance [3].

References

[1] Martin Reuter, Franz-Erich Wolter, and Niklas Peinecke. "Laplace–Beltrami spectra as 'Shape-DNA' of surfaces and solids". In: *Computer-Aided Design* 38.4 (2006). Symposium on Solid and Physical Modeling 2005, pp. 342–366. ISSN: 0010-4485. DOI: https://doi.org/10.1016/j.cad.2005.10.011. URL: https://www.sciencedirect.com/science/article/pii/S0010448505001867.

[2] Martin Reuter et al. "Discrete Laplace–Beltrami operators for shape analysis and segmentation". In: *Computers Graphics* 33.3 (2009). IEEE International Conference on Shape Modelling and Applications 2009, pp. 381–390. ISSN: 0097-8493. DOI: https://doi.org/10.1016/j.cag.2009.03.005. URL: https://www.sciencedirect.com/science/article/pii/S0097849309000272.

[3] Jie Shi, Wen Zhang, and Yalin Wang. "Shape Analysis with Hyperbolic Wasserstein Distance". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 5051–5061. DOI: 10.1109/CVPR.2016.546.