



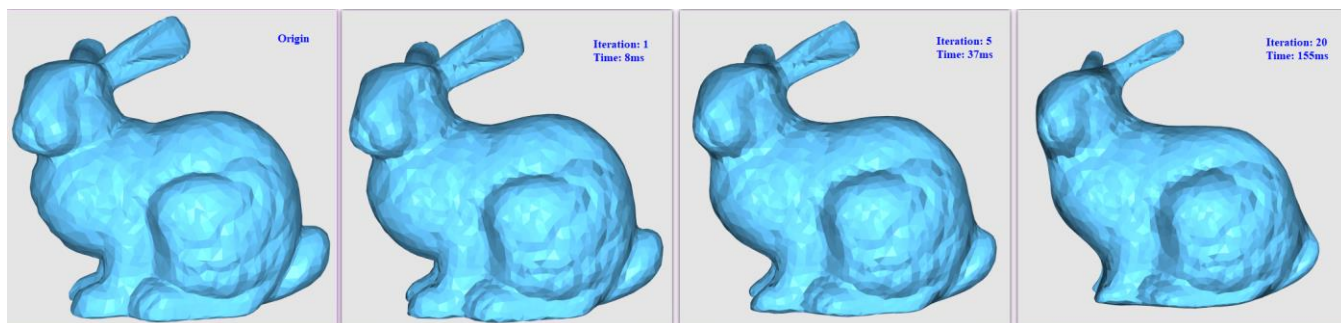
Assignment 1 Laplacian smoothing

Experiment Results:

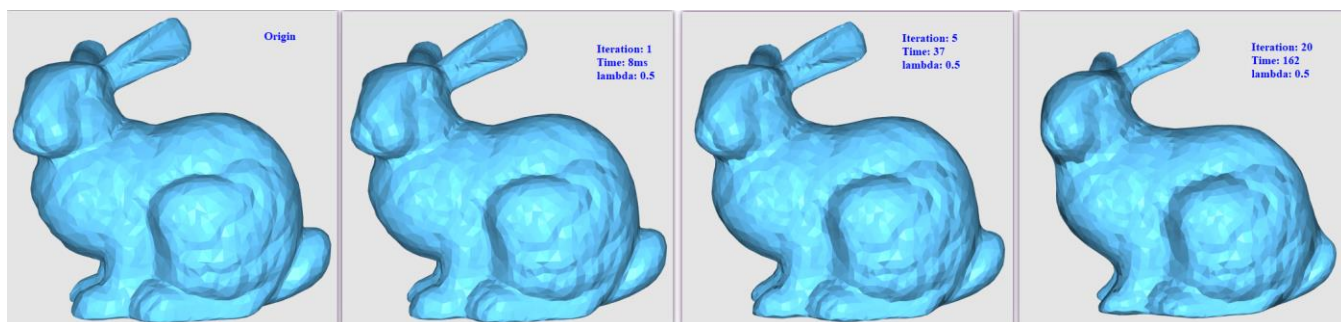
Time shows on the below figures is milliseconds.

- **Explict Uniform Weight**

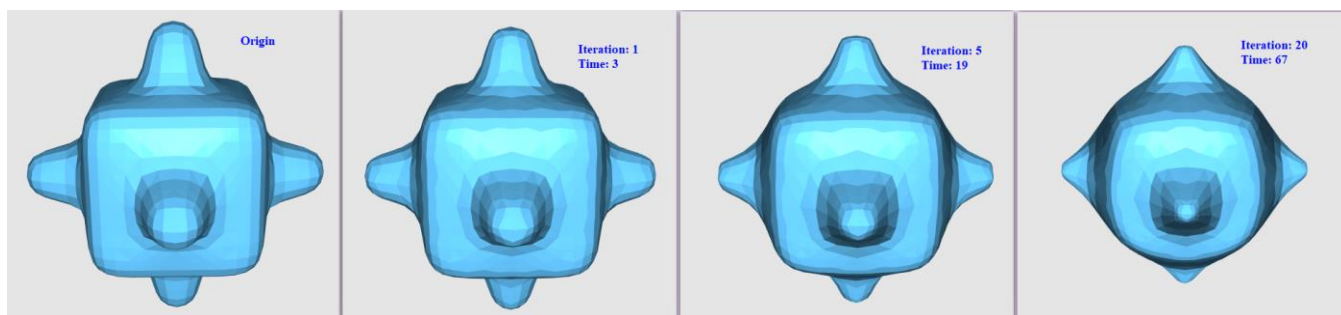
$$\lambda = 1$$



$$\lambda = 0.5$$



$$\lambda = 1$$

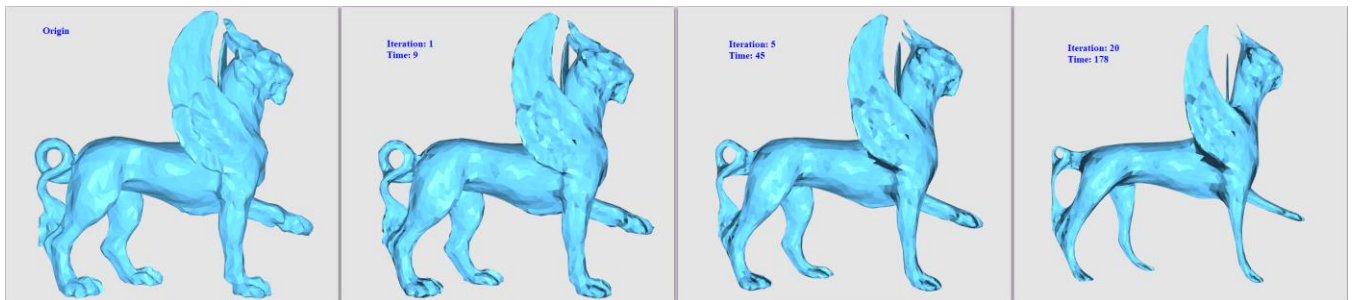




Hong Kong University of Science & Technology

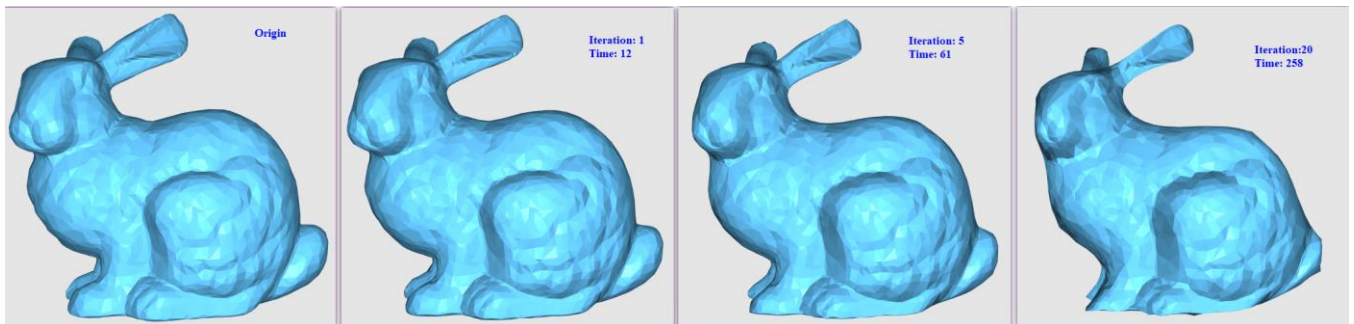
COMP5411 Advanced Computer Graphics

$\lambda = 1$

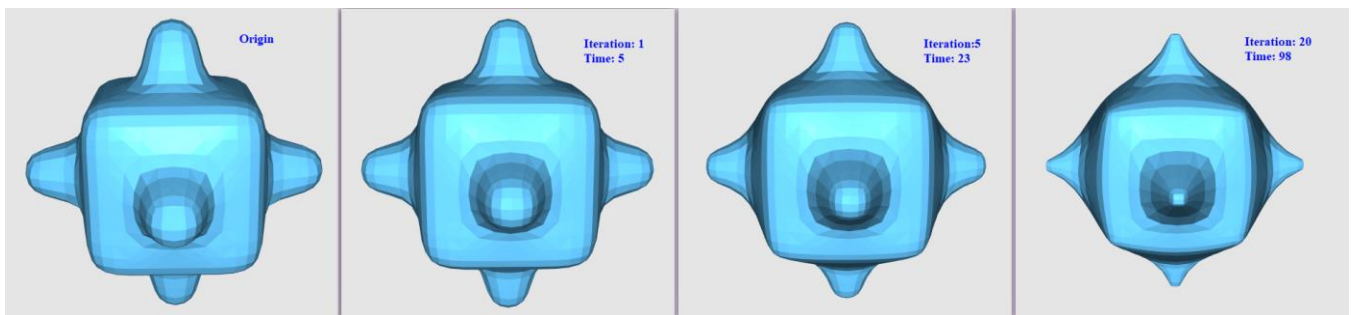


- Explicit Cotangent Weight

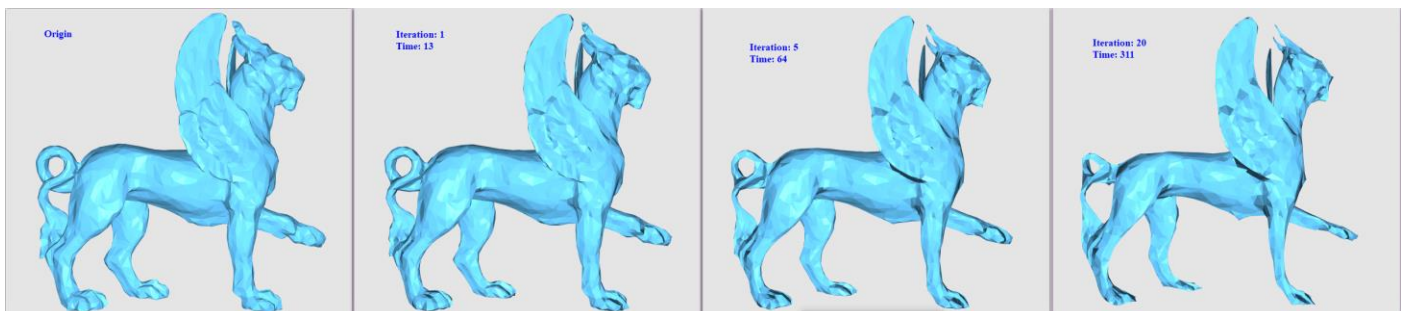
$\lambda = 1$



$\lambda = 1$



$\lambda = 1$



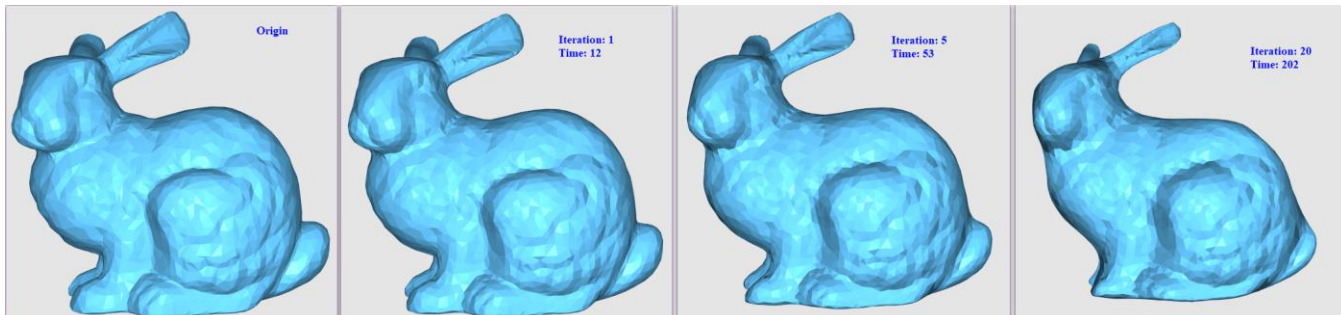


Hong Kong University of Science & Technology

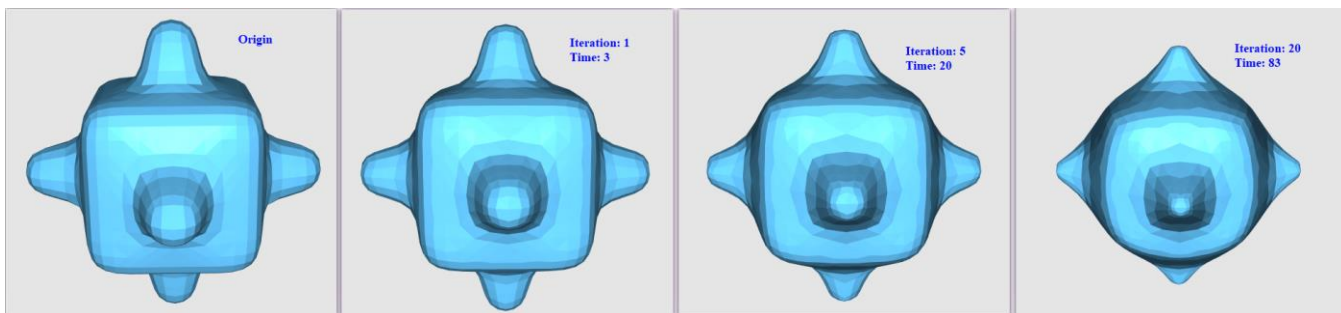
COMP5411 Advanced Computer Graphics

- Implicit Uniform Weight

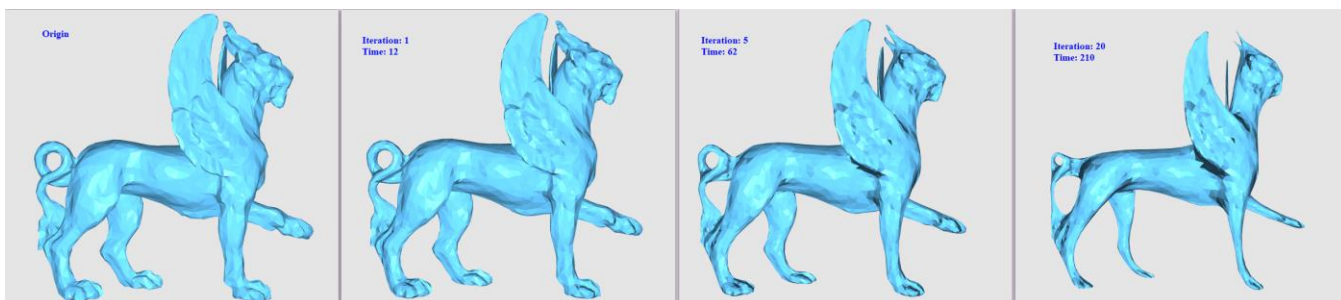
$\lambda = 1$



$\lambda = 1$

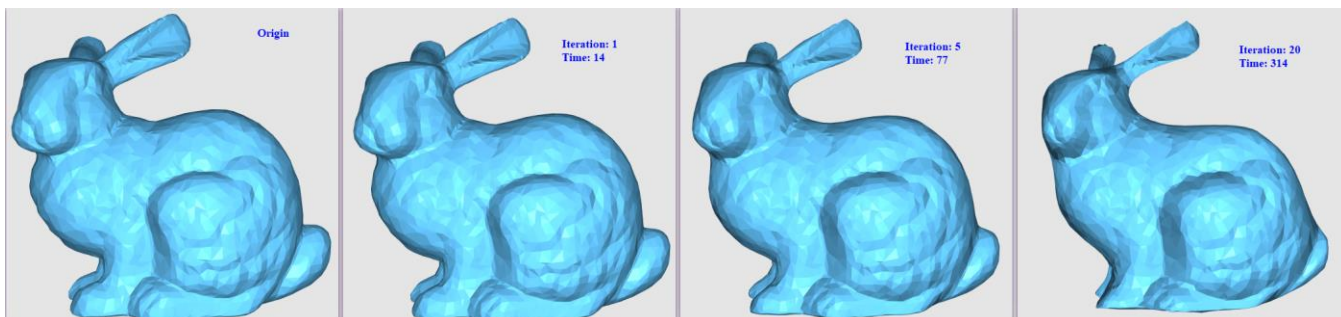


$\lambda = 1$



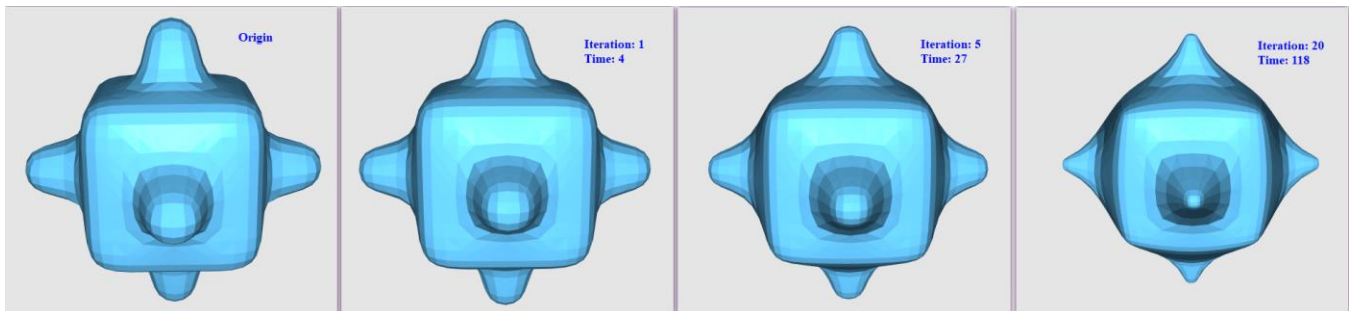
- Implicit Cotangent Weight

$\lambda = 1$

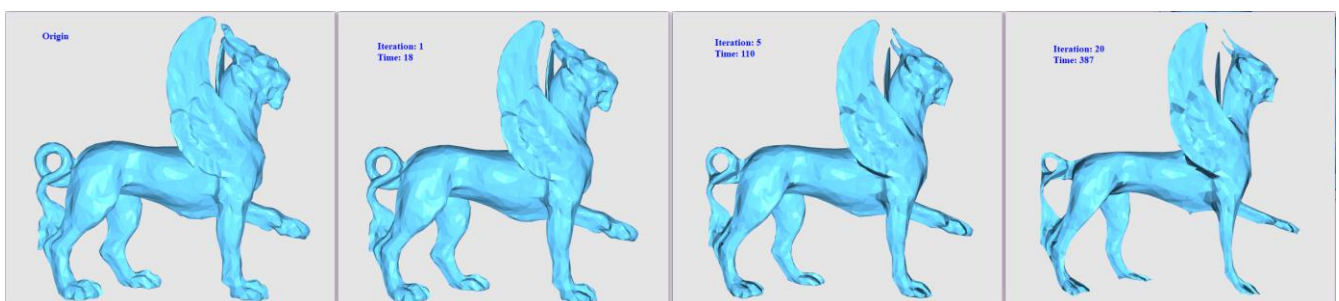




$\lambda = 1$



$\lambda = 1$



Discussion:

Weight method (Unifrom vs Cotangent):

Compared with uniform weight method, the cotangent weight can preserve the shape and details of the objects better. In the iteration of uniform weight smoothing, the mesh of the object get smoothed nearly in the same degree. While, as for the cotangent weight smoothing, the meshes not smooth equally.

Lambda:

A larger λ makes the smoothing processing more quickly. And it will takes more iterations for a smoothing processing with a small λ to get the same degree smoothing results as the one using a large λ .

Time Consuming:

The number of vertexes is the major aspect that influence the processing time. Time increases with the number of vertexes the mesh has.

Number of Iterations:

The mesh will slightly shrunk as the iterations increase until it become disappear. The deformation will become obviously with the iteration increase.

Library and Reference

Library:

Use Eigen BiCG to solve the sparse linear system.

Reference:



Hong Kong University of Science & Technology

COMP5411 Advanced Computer Graphics

- [1] http://eigen.tuxfamily.org/dox/classEigen_1_1BiCGSTAB.html
- [2] http://eigen.tuxfamily.org/dox/group__TutorialSparse.html#title3
- [3] <https://www.geeksforgeeks.org/measure-execution-time-function-cpp/>