**Instructions for Usage**

Kaimo Hu

[hukaimo02@gmail.com](mailto:hukaimo02@gmail.com)

This document specifies how to use the software “CGALRemeshing.exe”. This software demonstrates how to use the two remeshing algorithms that have been implemented in CGAL based on the Surface\_mesh data structure, namely *Isotropic remeshing* and *Min angle remeshing*. The two algorithms were first proposed in [1] and [2].

In the following, we give the detailed instructions for usage in each menu of the software.

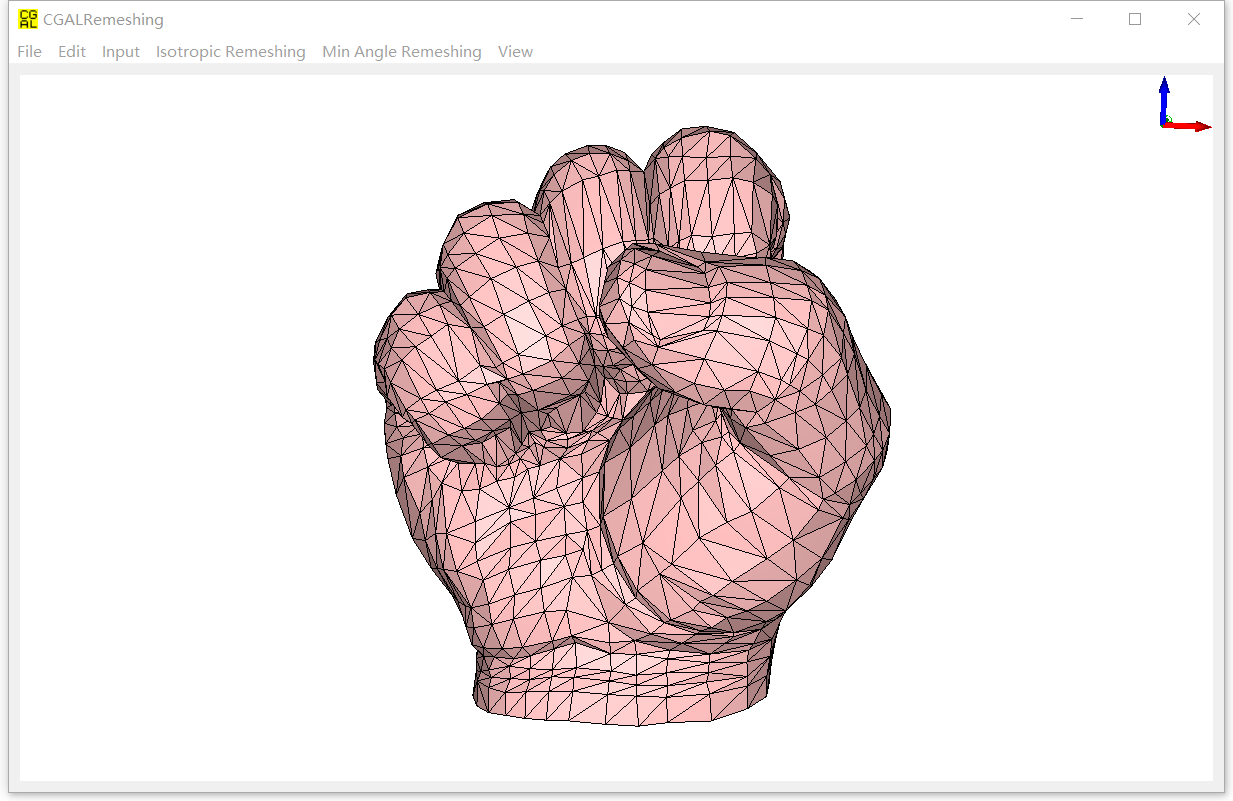


Figure 1: the main window of the software. Users can toggle between the input surface mesh (blue) and the remesh surface mesh (pink).

1. **File**:

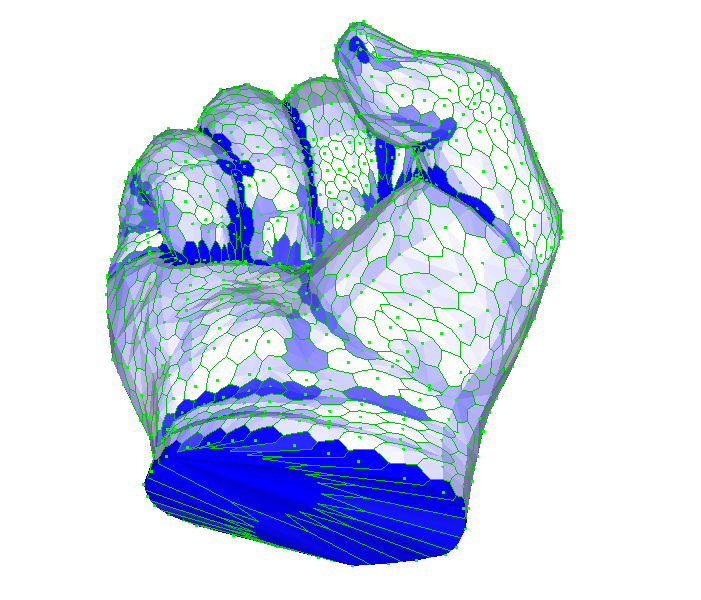
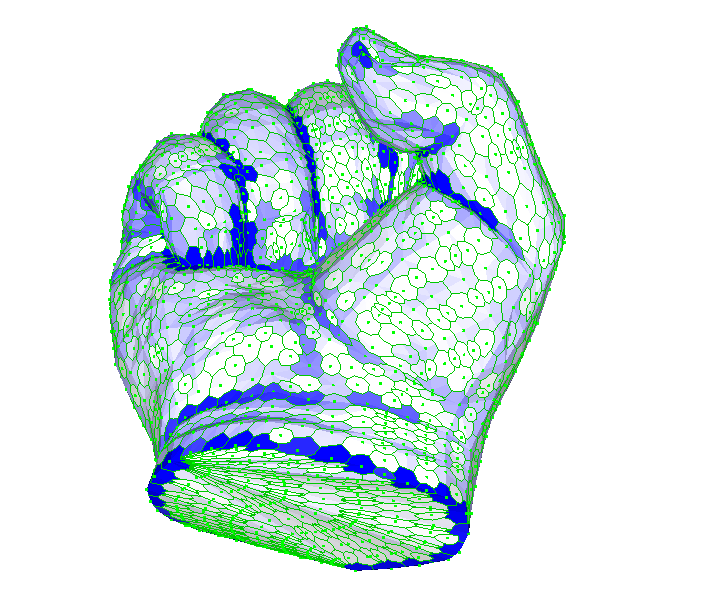
* “Open…”: Users can open an .off file that represent the input surface mesh. The system then initializes the remesh surface mesh as the copy of the input surface mesh.
* “Open input…”: When the remesh surface mesh is already there, users can use this item that only modify the input surface mesh. However, when the remesh surface mesh does not exist, then it will be initialized as the copy of the new opened input surface mesh.
* “Open remesh…”: Just performs in the similar way as “Open input…”.
* “Save remesh as…”: Users can save the remeshing results as an .off file.

1. **Edit**:

* “Copy snapshot”: Copy the snapshot to the clipboard.
* “Save snapshot…”: Save the snapshot as an image.

1. **Input**

* “Eliminate degenerations”: If the input surface mesh has some degenerated facets, users are supposed to eliminate the degenerations in advance.
* “Split long edges”: Some input surface meshes might have feature vertices or creases that are connected by long edges. This makes the visualizations a bit confusing (Although it does not affect the remeshing results). By splitting long edges on the input surface mesh, the visualization can be improved. See Figure 2 for comparison.
* “Properties…”: This item shows the properties of input surface mesh. See Figure 3.

(a) (b)

Figure 2: (a) The visualization comparison of vertex maximal normal dihedrals: (a): before split long edges; (b) after split long edges.

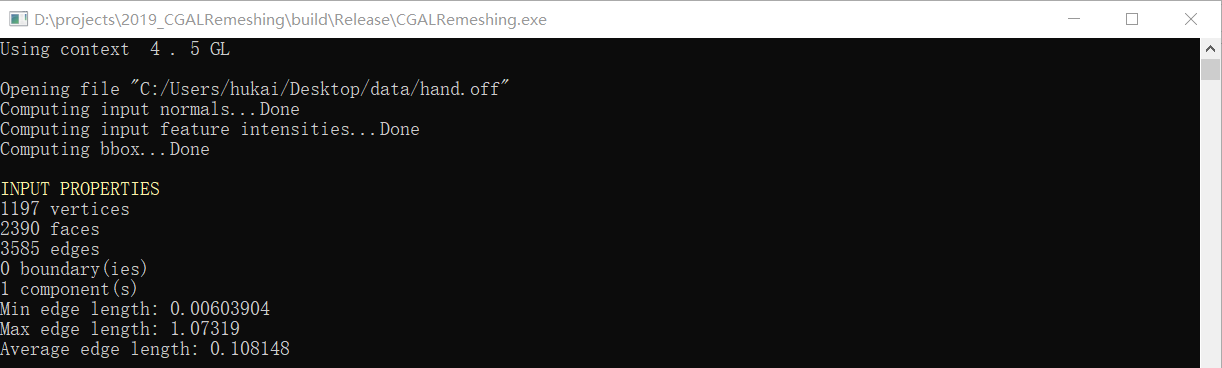


Figure 3: The properties of the input surface mesh.

1. **Isotropic Remeshing**

Under this menu, users can perform remeshing using the isotropic remeshing algorithm [1].

* “Split borders”: Split the long edges in the borders of the surface mesh. This will help the isotropic remeshing performs better.
* “Isotropic remeshing”: Perform the remeshing based on the specified parameters below.
* “Parameter settings…”: users can set the parameters that are used in the isotropic remeshing process. Figure 4 shows the parameter names.
  + *Target edge length:* The target edge length of the isotropic remeshing. The default value is set as the average length of the input surface mesh.
  + *Smooth iteration count*: The smooth iteration count of the isotropic remeshing. The default value is 3.

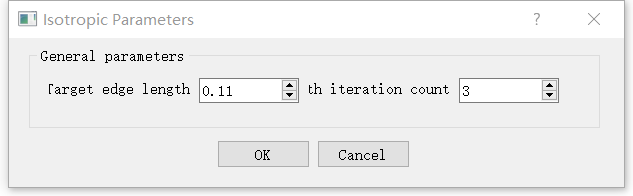


Figure 4: Parameter setting dialog of the Isotropic Remeshing.’

1. **Min Angle Remeshing**:

This menu runs the isotropic remeshing algorithm. Users are supposed to read Sec. 3 of [2] for more details. Here we only explain what each item does.

* “Reset from input”: Reset the remesh surface mesh as the copy of the input surface mesh.
* “Generate links”: Generate the samples on input surface mesh and remesh surface mesh, and construct the links of these samples. The samples include facet samples, edge samples and vertex samples. Users can view the samples and links through “View -> Surface mesh properties -> Samples and links”. Please see Figure 5 for more details.
* “Min Angle Remeshing”: Performs the total pipeline of the isotropic remeshing, include “Initial mesh simplification”, “Greedy angle improvement” and “Final vertex relocation”. The “initial mesh simplification” and “Final vertex relocation” are optional, according to the parameter settings (See Figure 8).
* “Initial mesh simplification”: Performs the initial mesh simplification before minimal angle improvement. After this, the mesh complexity can be reduced significantly while the approximation error is strictly bounded. Figure 6 shows the simplification results.
* “Greedy angle improvement”: This menu performs the minimal angle improvement. In order to make it easier for user interaction, we have designed the following three items:
  + Split local longest edge: Compulsively split the local longest edge in order to avoid dead loops when improving the minimal angle step by step. See line 15 in Alg. 3 in the paper for more details.
  + Increase minimal angle: Improve the minimal angle in one step. The minimal angle is indicated in red line (See the red lines in Figure 7). See Alg. 3 in the paper for more details.
  + Maximize minimal angle: Improve the minimal angle up to the user specified minimal angle threshold. See Figure 7 for the remeshing result of Figure 6.

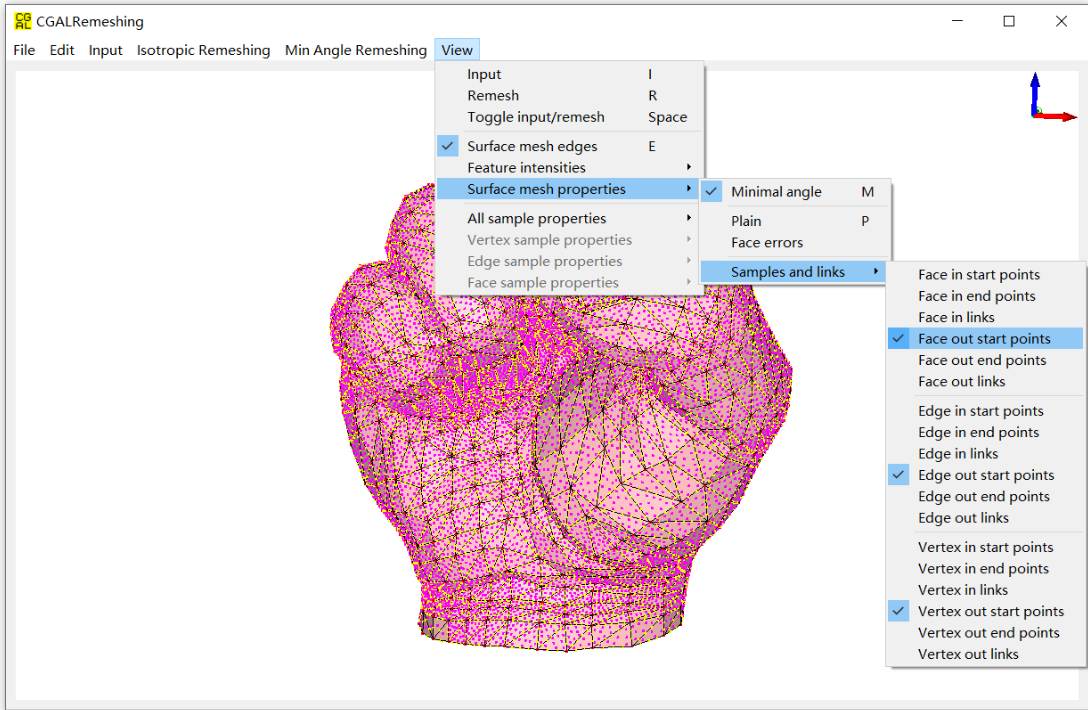


Figure 5: The facet, edge and vertex samples on remesh surface mesh.

* “Final vertex relocation”: The final vertex relocation for improving the overall triangle quality. See Sec. 3.3 in the paper for more details.
* “Parameter settings…”: users can set all the parameters that are used in the remeshing process. Figure 8 shows all the parameter names. According to extensive experiments, we set the default values of parameters as their optimal ones.
  + *Max error threshold*: The approximation error, expressed as the percentage of diagonal length of the input surface mesh.
  + *Min angle threshold*: The minimal angle that the remesh surface mesh should achieve.
  + *Max mesh complexity*: The maximal mesh complexity that the remesh surface mesh should maintain, expressed as the number of vertices of the remesh surface mesh.

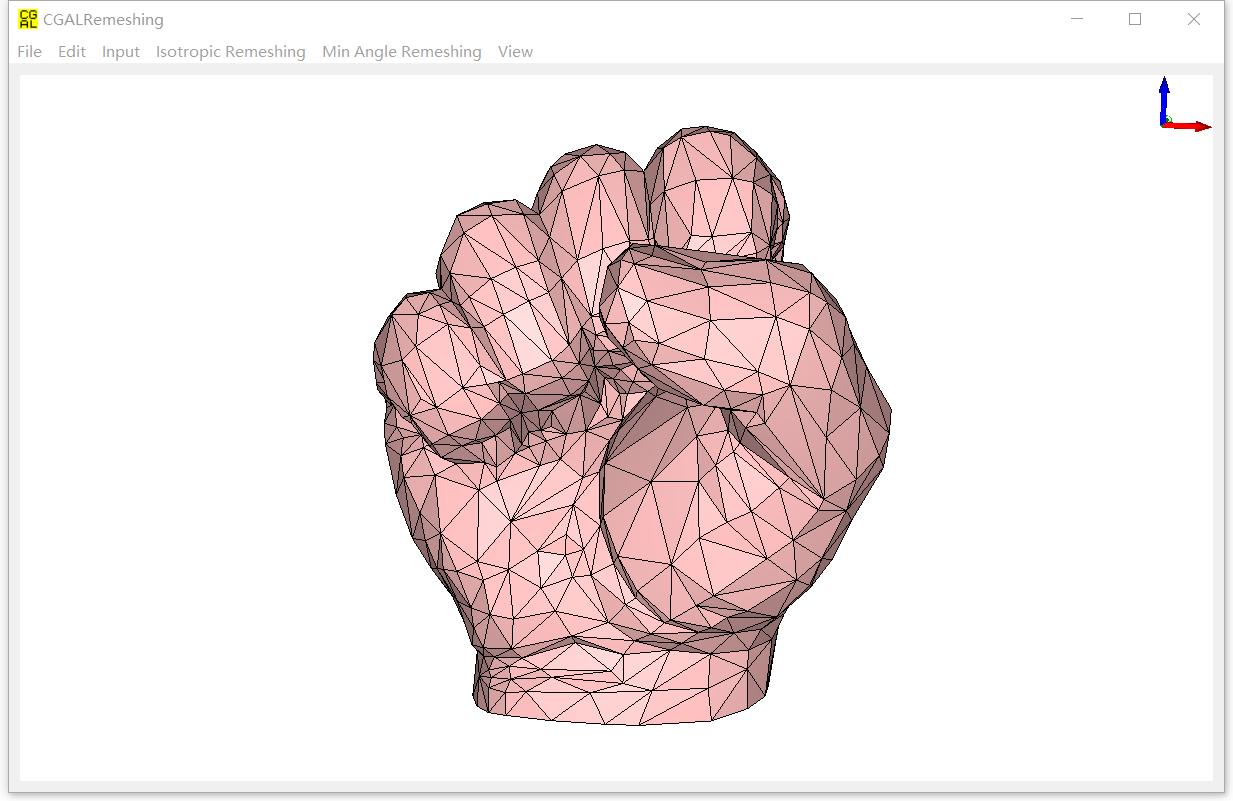


Figure 6: The result of initial mesh simplification results of Figure 1.

* + *Smooth angle delta*: The minimal step for angle improvement. If a local operator improves the minimal angle less than this value, then it is considered the minimal angle has not been improved.
  + *Apply edge flip*: Indicates whether we apply the local operator “Edge flip” when improving the minimal angle. Experiments show that enable the edge flipping achieves much better results.
  + *Edge flip strategy*: Indicates whether we want to improve the minimal angle, or the vertex valences when flipping an edge. Experiments show that “Improve the minimal angle” is better.
  + *Flip after split and collapse*: Indicates whether we perform “Edge flip” after applying edge split or edge collapse, such that the local valance of the new generated vertex can be improved. Experiments show that enabling this is better.
  + *Relocate after local operations*: Indicates whether we perform “Vertex relocate” after applying other local operators, such as edge split, edge collapse. Note we do not perform “Vertex relocate” after edge flip in order to perverse sharp features. Experiments show that enabling this is better than not.
  + *Relocate strategy*: The options for “Vertex relocation”, we now have barycenter or CVT barycenter. Barycenter is simply the average of the one-ring vertices, which is also called Laplacian-operation. Instead, the CVT barycenter is the Centroid Voronoi Tessellation center of the one-ring vertices. Experiments show that the CVT barycenter is better.

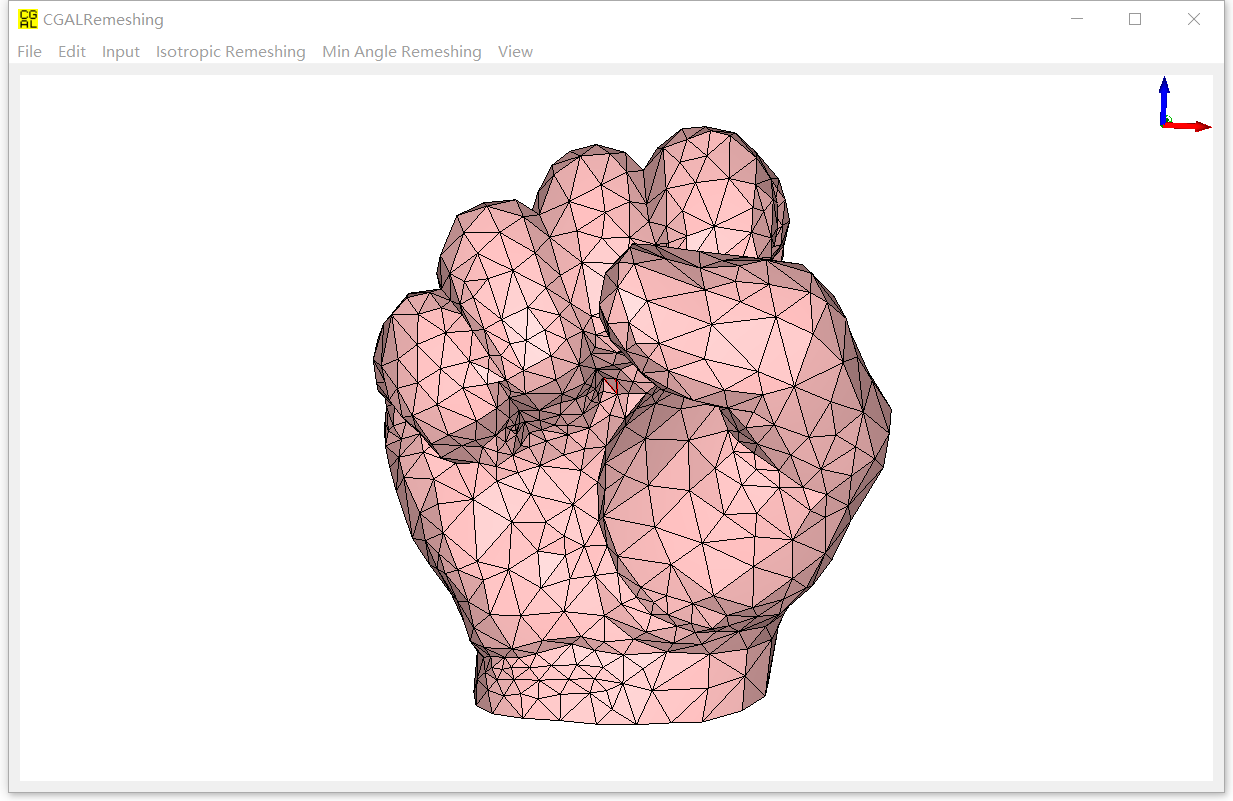


Figure 7: The remeshing result of Figure 8 by maximizing the minimal angle.

* + *Keep vertex in one ring*: Indicate whether we want to keep the vertex in one-ring region when performing “Edge collapse” or “Vertex relocate”. This was indicated in Fig. 4 of our paper. Experiments show that disabling this is much better. However, we may have some folder-overs in some results.
  + *Use local AABB tree*: Indicates whether we construct the local AABB tree or not when simulating the edge collapse operator. If it is true, we use the constructed local AABB tree to update the links from input surface mesh to remesh surface mesh; Otherwise, we update the links directly. Experiments show that the efficiencies for both cases are almost the same.
  + *Collapse list size*: In some rare cases, the software traps into local dead loops (e.g., edge split -> edge collapse -> edge split ->edge collapse…). Hence, we maintain a global collapse list, such that if an edge is collapsed not long ago (recorded in the list), we deny the edge collapse when it can be in later steps. This parameter indicates the maximum size of the collapse list. The larger this value is, the less possibility that the software traps into local dead loops, but higher computational and memory cost. Experiments show that this strategy is really effective.

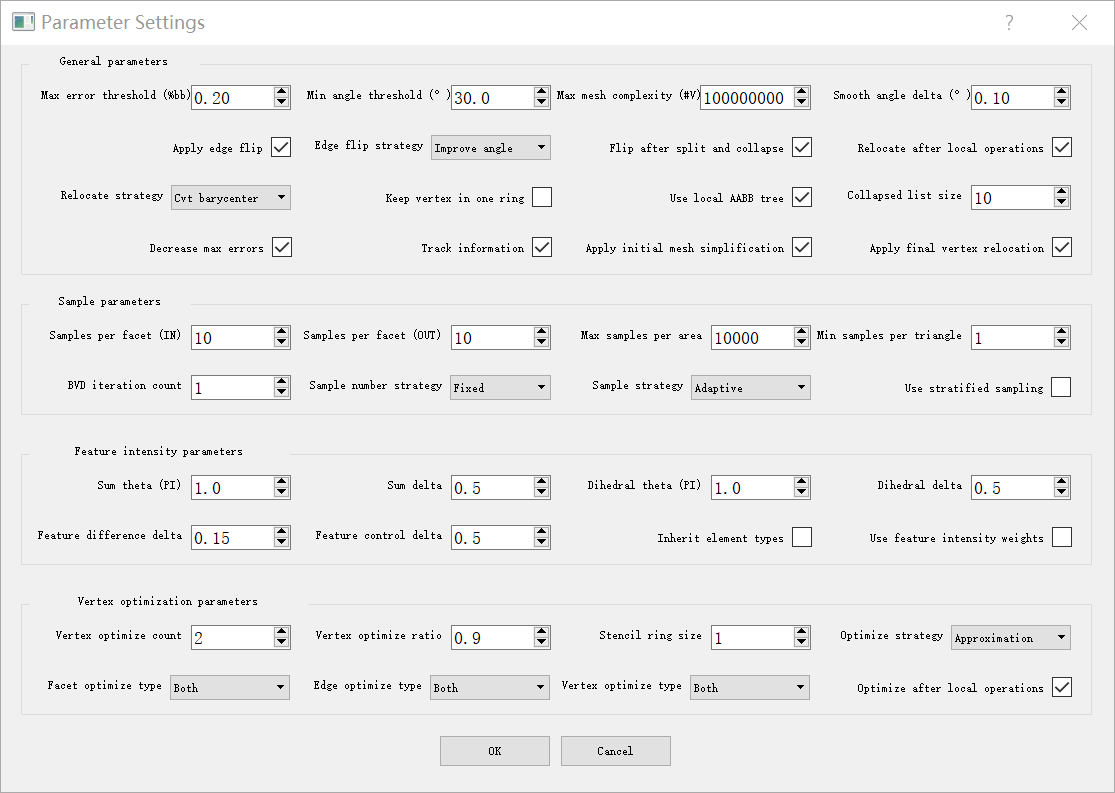


Figure 8: Parameter setting dialog of the Min Angle Remeshing.

* + *Decrease max errors*: Due to the randomness of sampling, sometimes the max error threshold cannot be strictly bounded to the threshold. If this parameter is enabled, we explicitly decrease the max error as long as we find the Hausdorff distance between the input and the remesh exceeds the threshold. This will make sure the approximation error is strictly bounded regardless of the sampling randomness.
  + *Track information*: If this parameter is enabled, we output the detailed information when each local operator is applied. The detailed information may include current maximal error, minimal angle, or the size of the dynamic priority queue.
  + *Apply initial mesh simplification*: If this parameter is enabled, we apply the initial mesh simplification before improving the minimal angles. This reduces the mesh complexity around 20-30% on average.
  + *Apply final vertex relocation*: If this parameter is enabled, we apply the final vertex relocation after improving the minimal angles to the min angle threshold. This will improve the average quality of the triangles with respect to the parameter “Smooth angle delta”. The smaller “Smooth angle delta” is set, the better results we get along with the higher computational cost.
  + *Samples per facet (IN)*: The number of samples per facet on input surface mesh.
  + *Samples per facet (OUT)*: The number of samples per facet on remesh surface mesh.
  + *Max samples per area*: The maximal number of samplers per unit area. This parameter is used to avoid too dense sampling.
  + *Min samples per triangle*: The minimal number of samples per triangle. This parameter is used to guarantee that each triangle, no matter how small it is, has certain number of samples on it.
  + *BVD iteration count*: The large this parameter is, the more uniform the samples in the facets are. However, the computational cost will be dramatically higher as well.
  + *Sample number strategy*: The option of sample number strategy on remesh surface mesh. If it is “Fixed”, then the number of samples per facet is roughly the same as the one on the input surface mesh; If it is “Variable”, then the number of samples per facet is variable with respect to the size of facets of the remesh surface mesh: the more facets the remesh surface mesh has, the smaller number of samples we generate on each facet of the remesh surface mesh. The “Variable” option makes the total samples on the input surface mesh and remesh surface mesh roughly the same.
  + *Sample strategy*: The option of sample strategy. If it is “uniform”, then the number of samples per facet is proportional to its area; If it is “adaptive”, then the number of samples per facet is roughly the same.
  + *Use stratified sampling*: If this parameter is enabled, the vertex samples, edge samples and the facet samples partition the area of the surface mesh, respectively (we call the partition area as the capacity of each sample); If it is disabled, all the three types of samples will partition the area of the surface mesh together. In the former case, the feature samples (include the vertex samples and the edge samples) own higher area weights; while in the latter case, all the samples have the same area weights. Enabling this parameter preserves features better, but sacrifices the overall approximation error, because the facet samples dominates all the samples.
  + *Sum theta (PI)*: The maximal value of Gaussian curvature, expressed as the times of PI. If a Gaussian curvature exceeds some value, it will be clamped to this value.
  + *Sum delta*: The scale of the Gaussian curvature. If dividing a Gaussian curvature with this scale and the result exceeds Sum theta (PI), then it is clamped to Sum theta (PI).
  + *Dihedral theta (PI)*: The maximal value of large dihedral angle value, expressed as the times of PI. If the large dihedral angle exceeds some value, it will be clamped to this value.
  + *Dihedral delta*: The scale of the large dihedral angle. If dividing the large dihedral angle with this scale and the result exceeds Dihedral theta (PI), then it is clamped to Dihedral theta (PI).
  + *Feature difference delta*: This parameter is only used when getting the initial position of the vertex after applying edge collapse. If the feature intensity difference between the two end points is smaller than their maximal value multiplied with this parameter, then the midpoint will be selected as the initial position; otherwise, the end point with higher feature intensity will be selected.
  + *Feature control delta*: This parameter is used for vertex classification. Please refer to Fig. 10 in the paper for more detailed explanation.
  + *Inherit element types*: If this parameter is enabled, we calculate the edge types (crease edge or non-crease edge) of the remesh surface mesh in advance, and then maintain the edge types explicitly during the whole remeshing process. Otherwise, we calculate the edge types according to the given parameters each time when a local operator is applied.
  + *Use feature intensity weights*: If this parameter is enabled, the weight for each sample is its according area multiplied by its feature intensity; Otherwise, the weight is its according area. Enabling this parameter keeps the features better.
  + *Vertex optimize count*: The number of iterations we perform when optimizing a vertex position. Please refer to the first row of Fig. 15 in the paper for more details.
  + *Vertex optimize ratio*: The ratio to get to the optimal position when optimizing a vertex position. Refer to the second row of Fig. 15 in the paper for more details.
  + *Stencil ring size*: The stencil ring size when collecting the samples from input surface mesh to remesh surface mesh. Please refer to the gray region of Fig. 3 as well as the discussion in Sec. 6.1 in the paper for more details.
  + *Optimize strategy*: Options for vertex position optimization. If it is “Approximation”, then the vertex position is just the optimal position we calculated; otherwise, the vertex position is the projection of the optimal position to the input surface mesh. Please refer to Sec. 6.2 in the paper for more details.
  + *Facet optimize type*: The facet sample types used for optimizing a vertex position.
  + *Edge optimize type*: The edge sample types used for optimizing a vertex position.
  + *Vertex optimize type*: The vertex sample types used for optimizing a vertex position.\*Optimize after local operations*: If this parameter is enabled, we perform the vertex position optimization after each local operator is applied.
* “Properties…”: This item shows the properties of remesh surface mesh. Besides the basic ones that are the same as the input surface mesh, it shows the additional properties that the remesh surface mesh has. See Figure 9 below.

1. **View**:

We provide plenty of visualizations for the input surface mesh and remesh surface mesh.

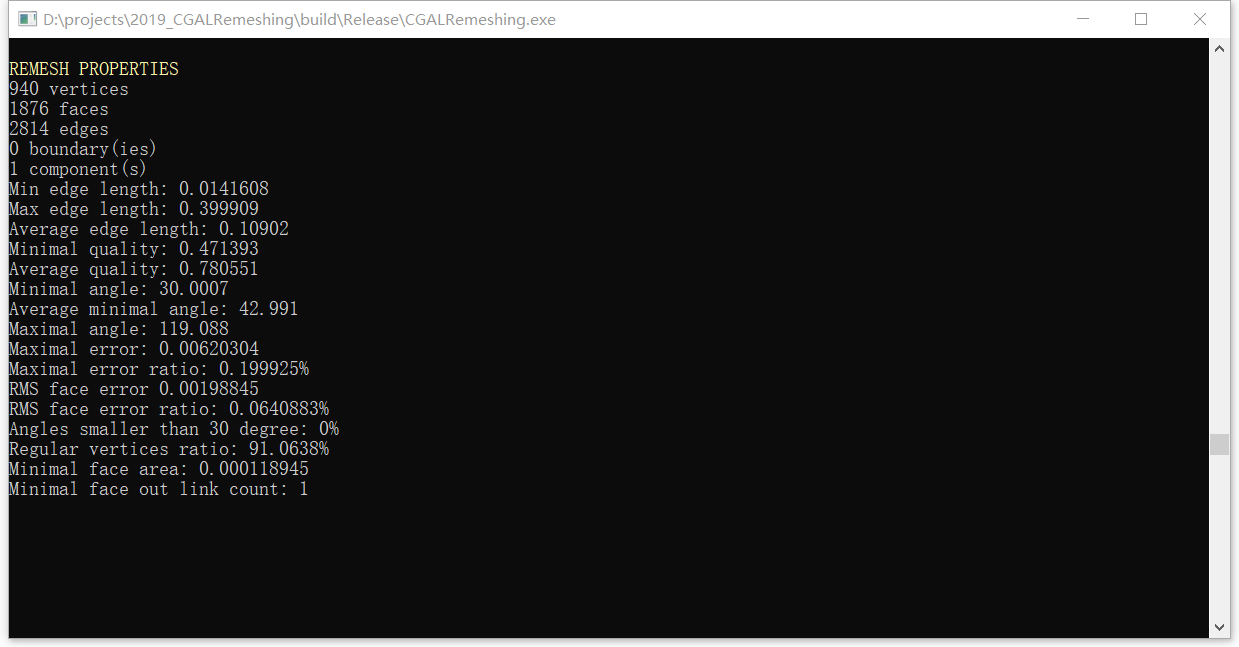


Figure 9: The properties of remesh surface mesh (after remeshing). We see that both the maximal approximation error and the minimal angle comply with the input parameters.

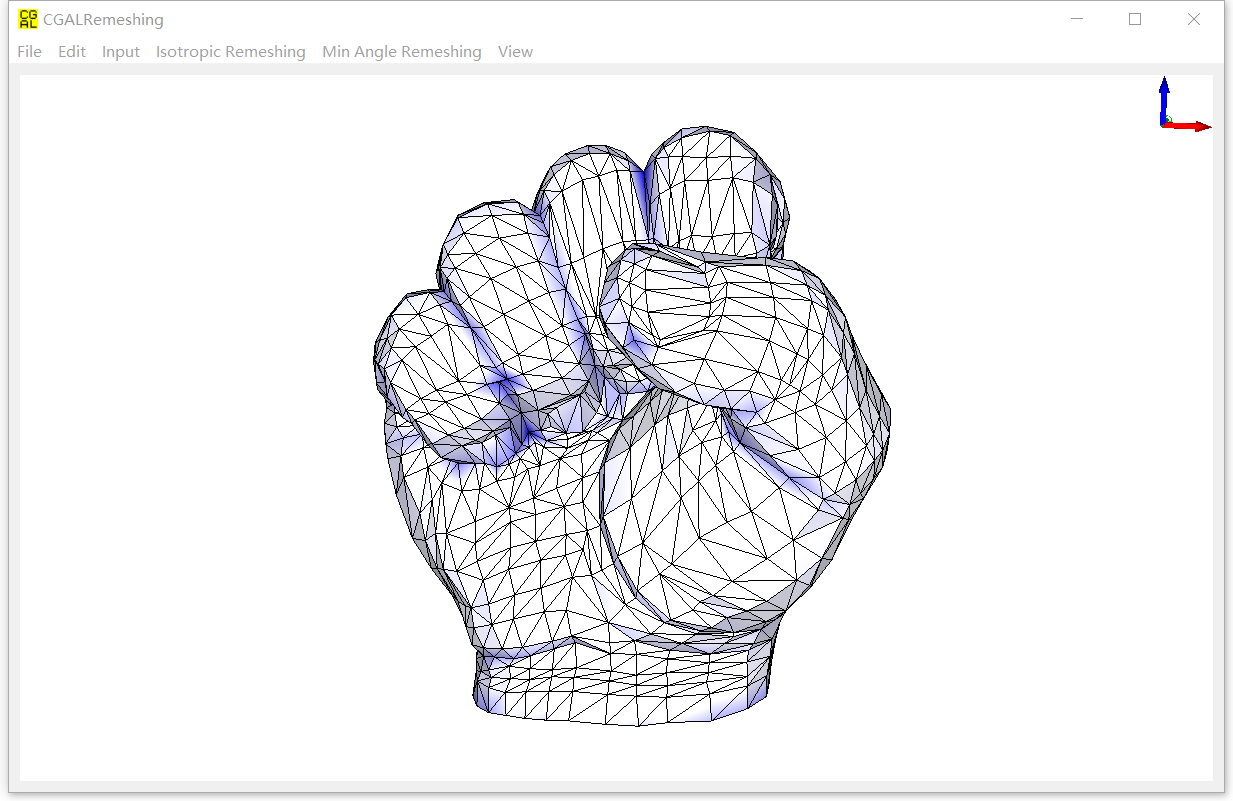


Figure 10: Visualization of interpolated feature intensities.

* “Input”: Enable or disable the visualization of the input surface mesh.
* “Remesh”: Enable or disable the visualization of the remesh surface mesh.

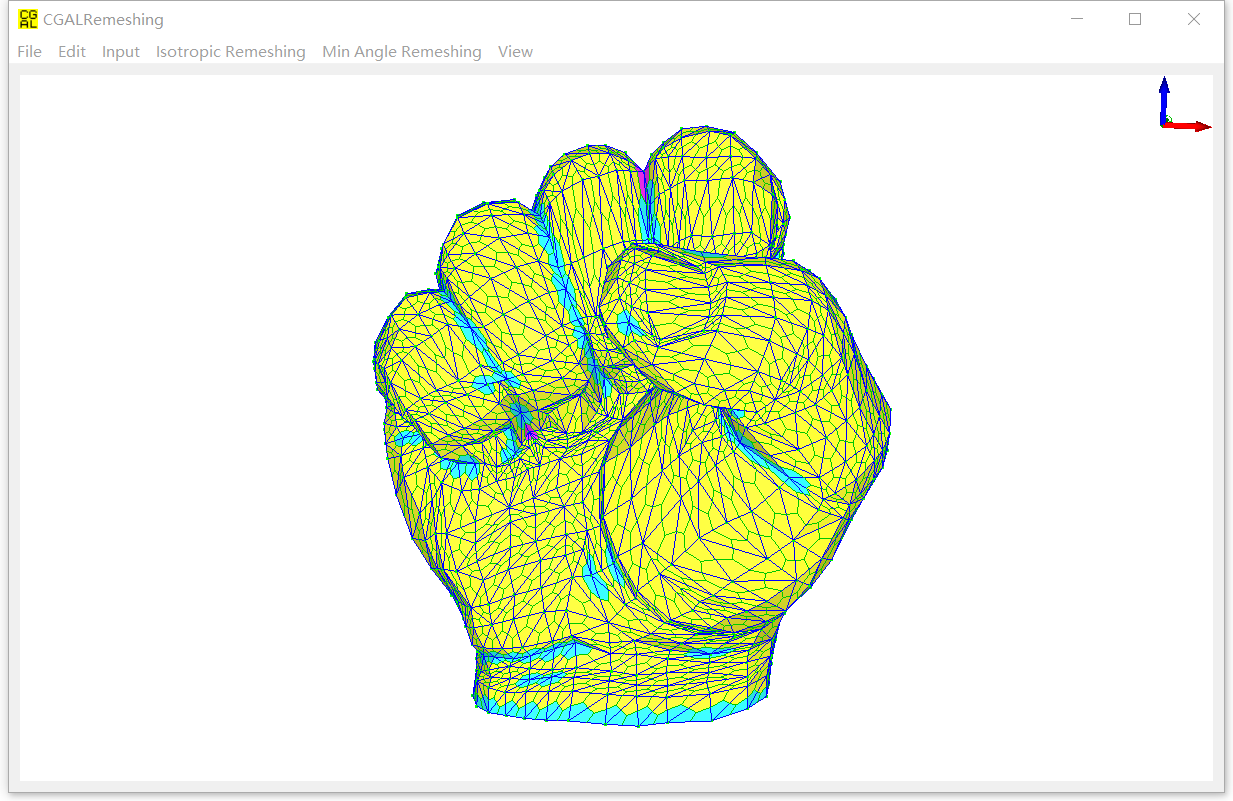


Figure 11: Visualization of element classifications.

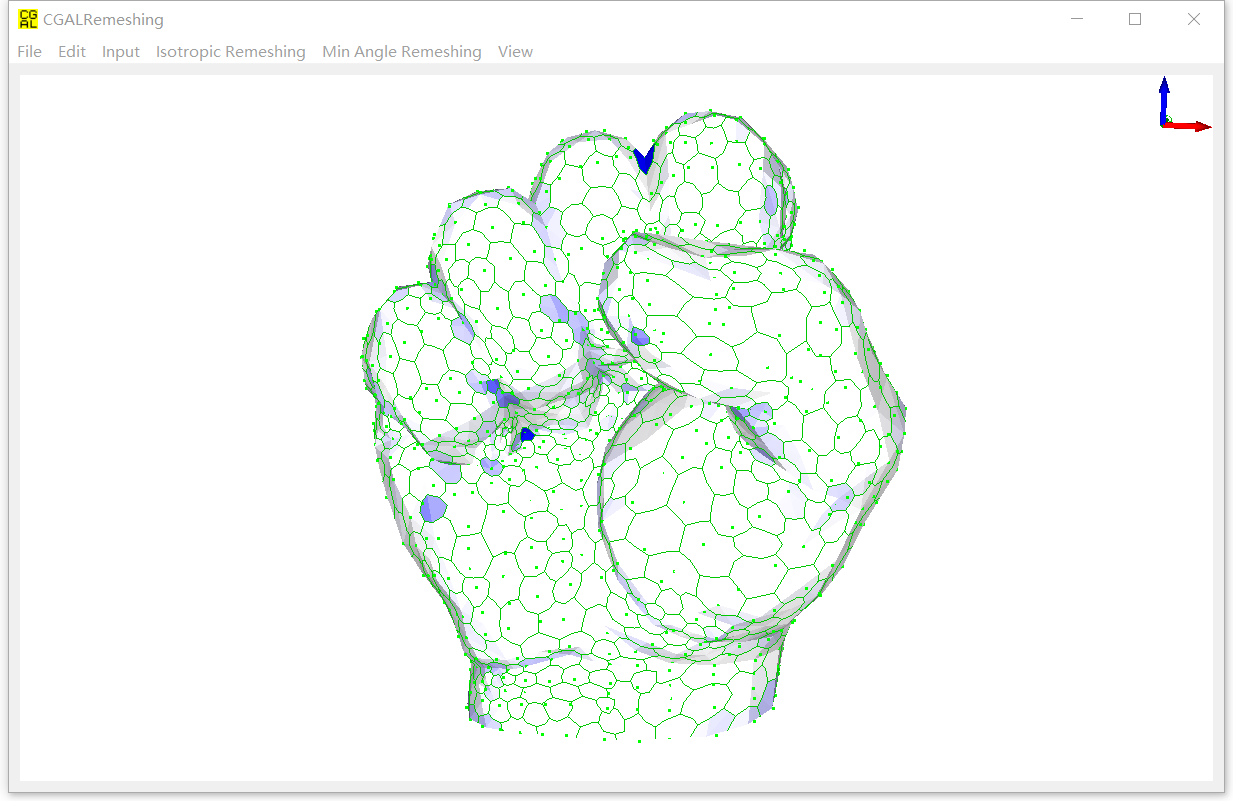


Figure 12: Visualization of Gaussian Curvatures for vertices.

* “Toggle input/remesh”: Toggle between the input surface mesh and remesh surface mesh.

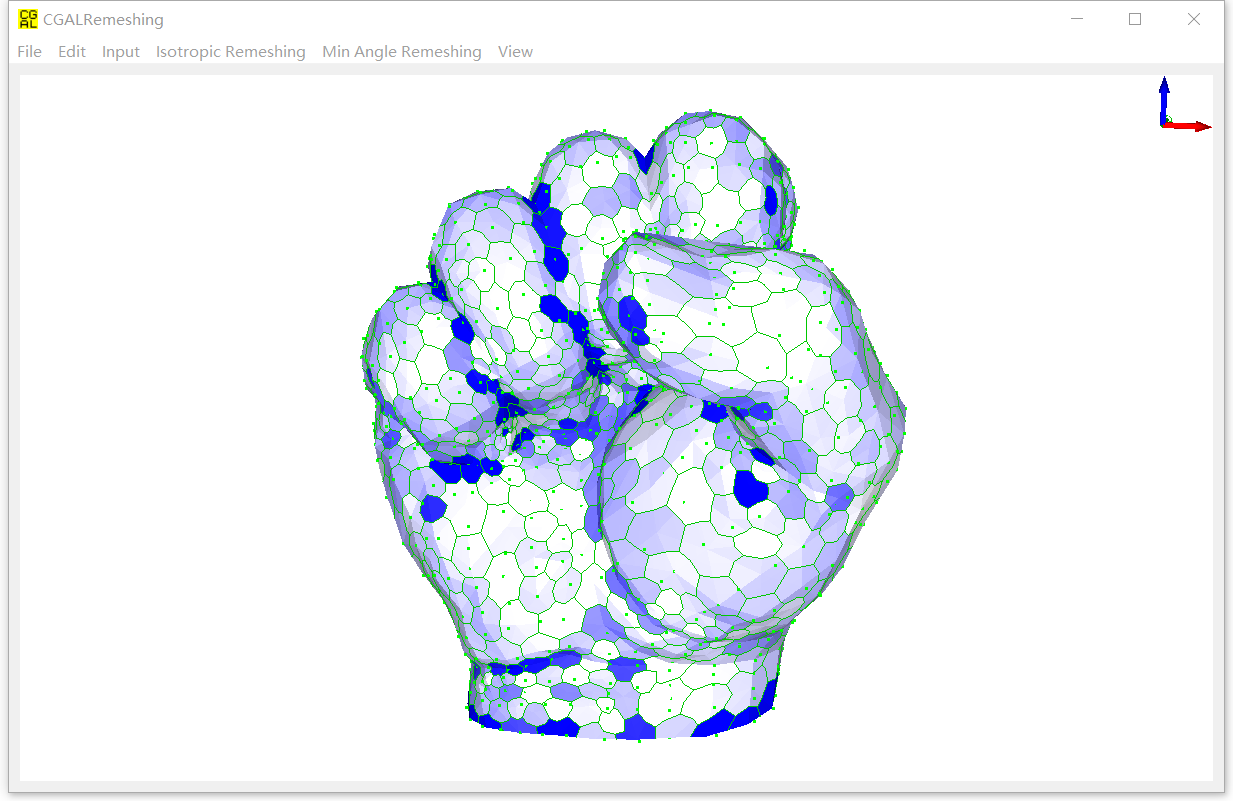


Figure 13: Visualization of maximal normal dihedral for vertices.

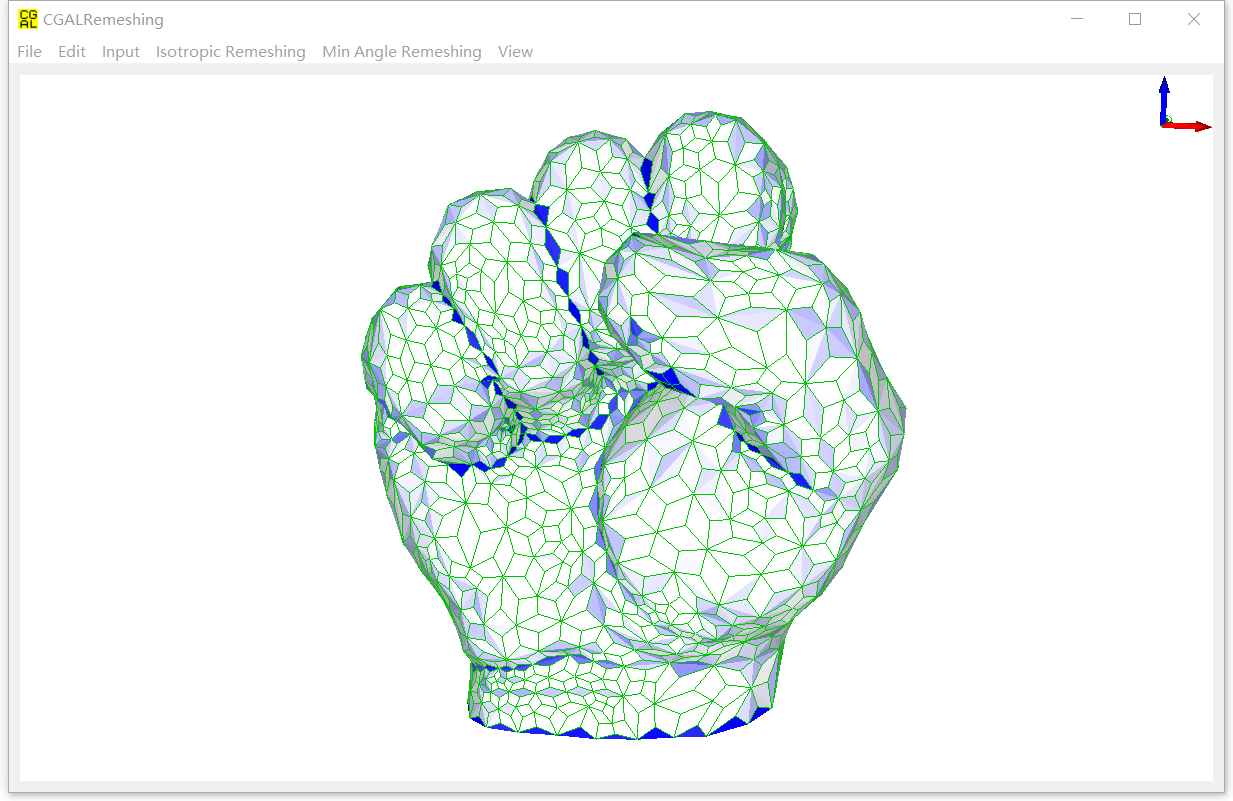


Figure 14: Visualization of normal dihedral for edges.

* “Surface mesh edges”: Enable or disable the visualization of surface mesh edges.

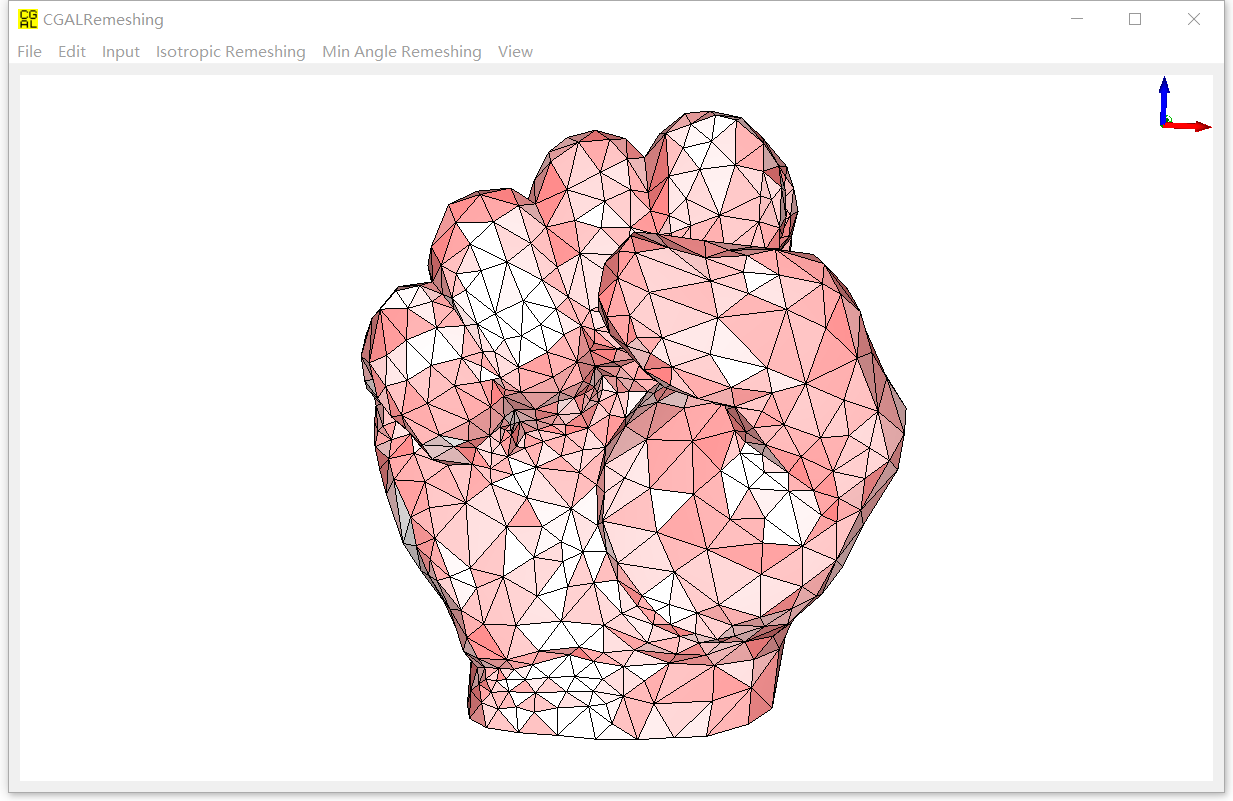


Figure 15: Visualization of approximation errors of facets on the remesh surface mesh.

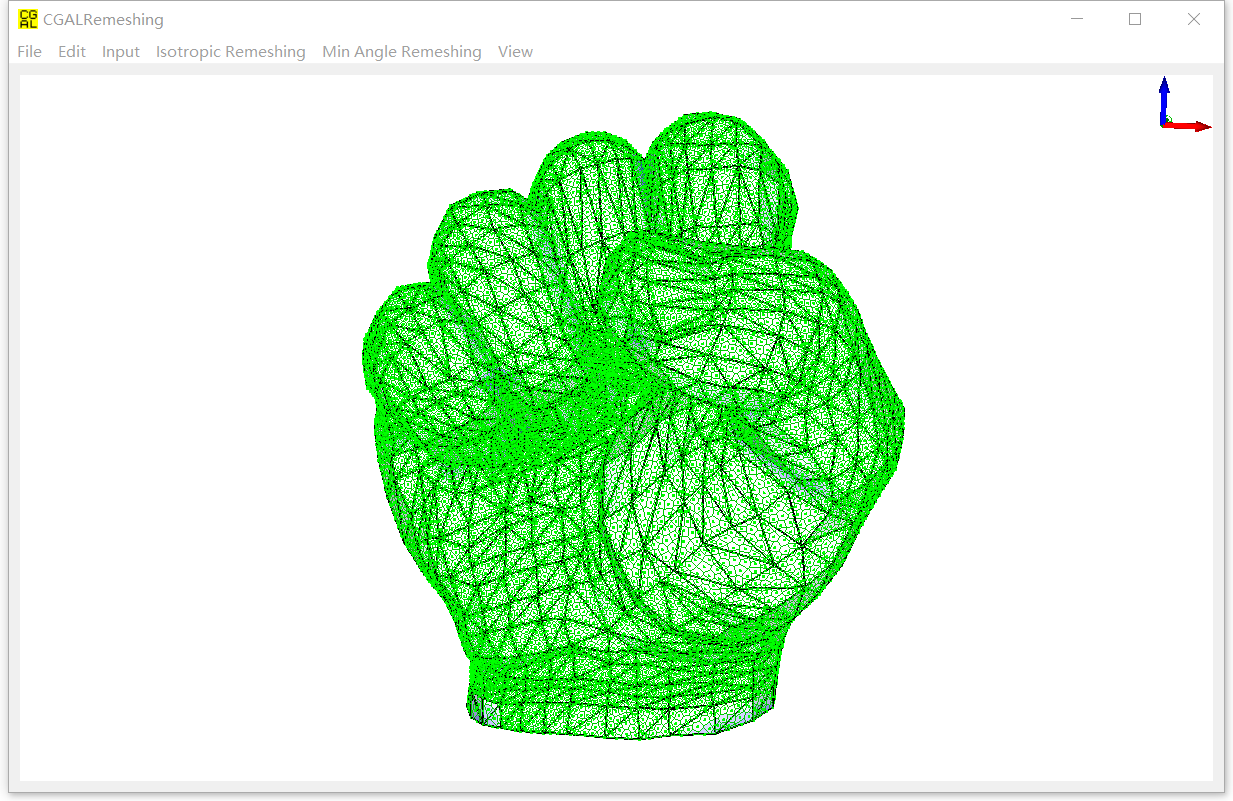


Figure 16: Visualization of samples’ feature intensities.

* “Feature intensities”: This menu visualizes the properties of feature intensities:

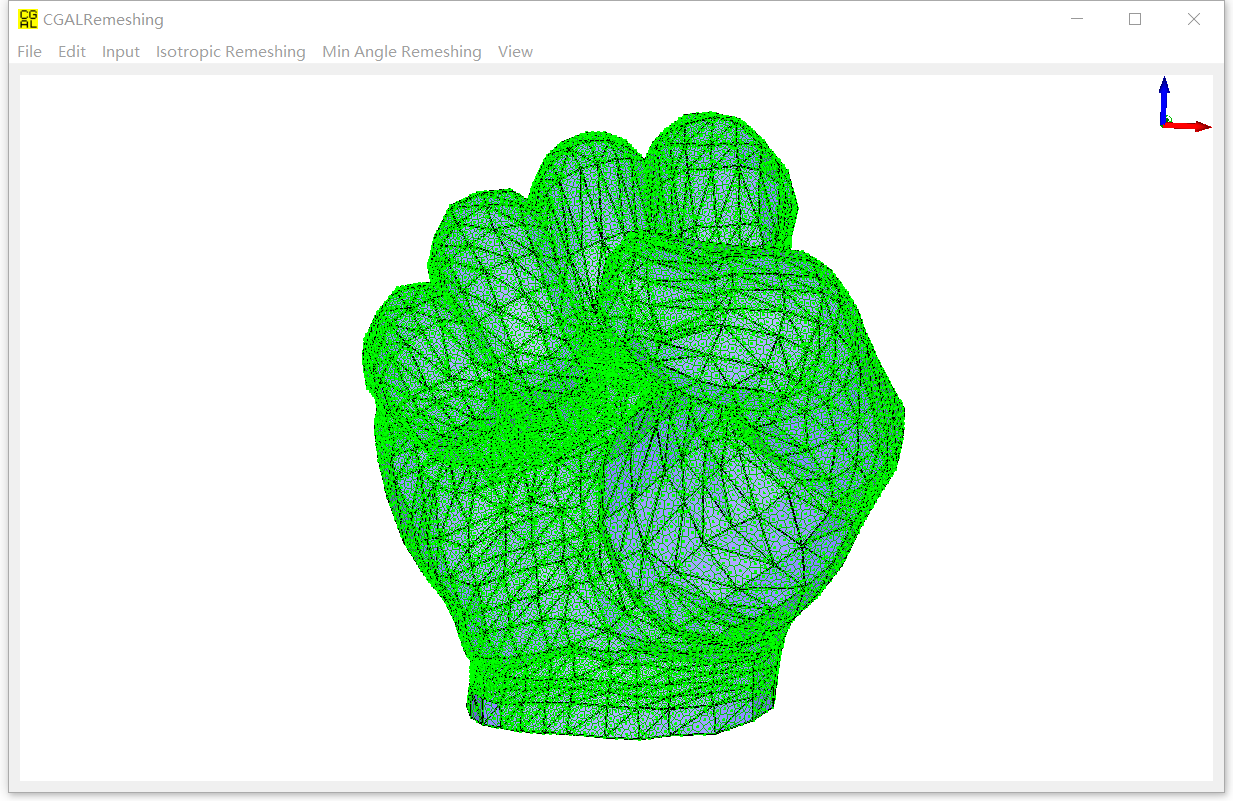


Figure 17: Visualization of samples’ capacities.

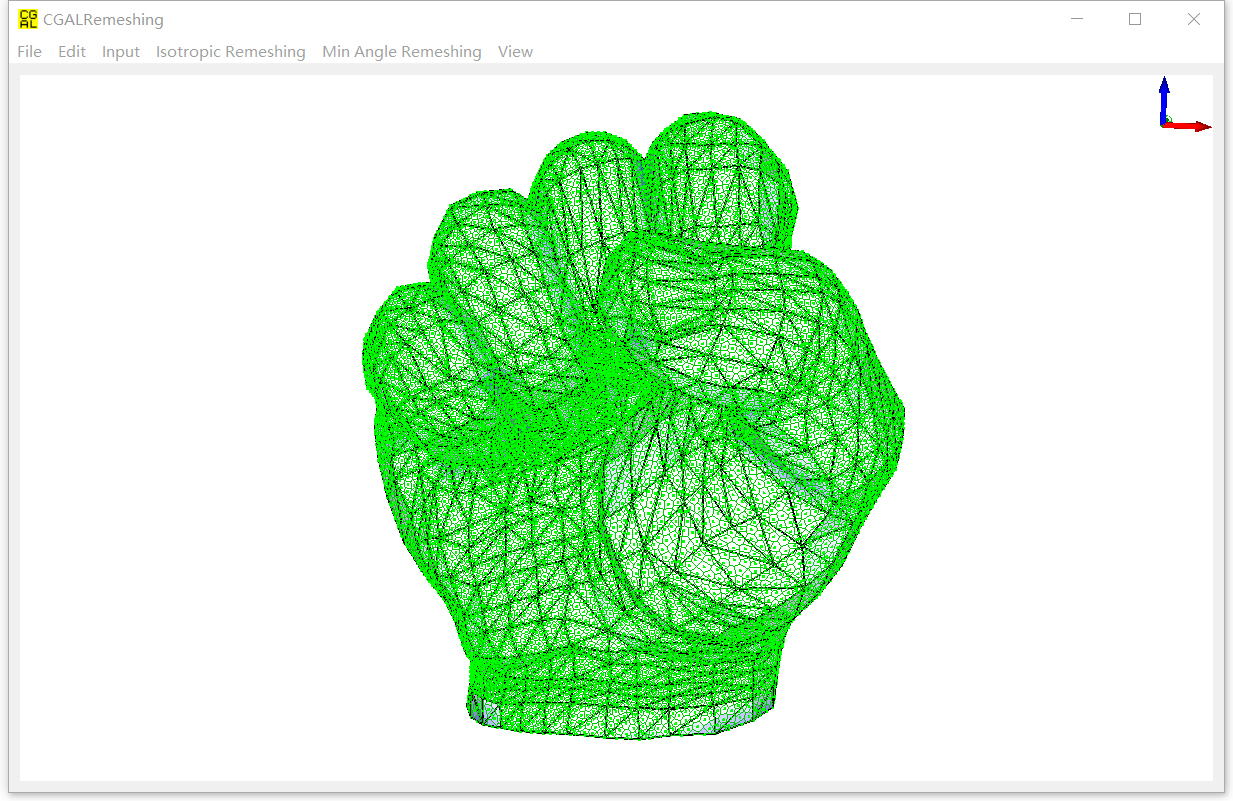


Figure 18: Visualization of samples’ weights.

* + “Interpolated feature intensities”: Areas with denser blue mean higher feature intensities. Feature intensities can visualize the features correctly. See Figure 10.
  + “Element classifications”: For vertices, feature vertices are in rendered in purple, crease vertices are in cyan, and smooth vertices are in yellow. For edges, the creases are rendered in red (if the parameter “inherit edge types” is enabled), while others non-crease edges are rendered in blue. Please refer to Figure 11 for intuition.
  + “Vertex Gaussian curvatures”: The Gaussian Curvatures for vertices. See Figure 12.
  + “Vertex maximal normal dihedrals”: The maximal normal dihedrals of vertices. Please see Figure 13 for intuition.
  + “Edge normal dihedrals”: The normal dihedrals for edges. See Figure 14.
* “Surface mesh properties”: This menu demonstrates the properties of surface meshes:
  + “Minimal angle”: Enable or disable the minimal angle of a surface mesh. See Figure 7.
  + “Plain”: Show the plain color of the surface meshes (input in blue, and remesh in pink).
  + “Facet errors”: Show the approximation error of each facets. Redder colors mean higher approximation errors. Please see Figure 15.
  + “Samples and links”: This menu shows the samples and links of input and remesh. Please see Figure 5 for the visualization of the samples and links.
* “All sample properties”: This menu visualizes the properties of samples when “Use stratified sampling” is disabled. Users can switch among the following three.
  + “Feature intensities”: Denser colors mean higher feature intensities of the samples. Please refer to Figure 16 for more details.
  + “Capacities”: Denser colors mean larger areas of the samples. See Figure 17.
  + “Weights”: Denser colors mean higher weights of the samples. Weights are the multiplications of feature intensities and capacities. See Figure 18.
* “Vertex sample properties”: Visualize the properties of vertex samples when “Use stratified sampling” is enabled. The properties are the same as “All sample properties”.
* “Edge sample properties”: Visualize the properties of edge samples when “Use stratified sampling” is enabled. The properties are the same as “All sample properties”.
* “Facet sample properties”: Visualize the properties of Facet samples when “Use stratified sampling” is enabled. The properties are the same as “All sample properties”.

Finally, we thank you for using our software, and sincerely hope that our work can benefit your work. If you have any questions, please do not hesitate to contact us. Enjoy!

**Reference**

[1] M. Botsch and L. Kobbelt. A remeshing approach to multiresolution modeling. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 185–192. ACM, 2004.

[2] Kaimo Hu, Dongming Yan, David Bommes, Pierre Alliez and Bedrich Benes. [Error-bounded and feature preserving surface remeshing with minimal angle elimination](http://www.google.com/url?q=http%3A%2F%2Fieeexplore.ieee.org%2Fabstract%2Fdocument%2F7756294%2F&sa=D&sntz=1&usg=AFQjCNEdASMBY_lAKpZUe5OUwOPAk1JLkQ). *Transactions on Visualization and Computer Graphics*, 2017, 23(12): 2560-2573.