



Title:

A plug-in for Local Mesh Modeling to Extend VTK Library: An Application for the Medical Sector

Authors:

Andrea Vitali, andrea.vitali1@unibg.it, Università degli studi di Bergamo
 Daniele Regazzoni, daniele.regazzoni@unibg.it, Università degli studi di Bergamo
 Caterina Rizzi, caterina.rizzi@unibg.it, Università degli studi di Bergamo
 Giorgio Colombo, giorgio.colombo@polimi.it, Politecnico di Milano

Keywords:

Polygonal 3D modeling, medical devices, VTK, prosthesis design

DOI: 10.14733/cadconfP.2017.435-439

Introduction:

In the last years, rising technologies, such as mixed reality and additive manufacturing, have always required to improve the quality of interaction with 3D models based on polygon meshes. The use of polygon meshes is also due to the use of reverse engineering techniques [3], [5]. The demand of reverse engineering is rising constantly in several industrial fields (e.g., medical fields and fashion industry) and the request of getting high quality polygonal mesh is growing. The most important features of a high-quality mesh are [4], [9], [14]:

- **The initial point cloud has to be ordered as much as possible in the 3D space.** This means the points have to be equidistant among them and the faces have a good allocation/distribution along the surfaces of the object.
- **The polygonal mesh has to be manifold.** The Manifoldness is very important to allow the correct creation of objects by using additive manufacturing technology.

Many algorithms have been developed to manage polygonal meshes usable for industrial design [9], [12], [14]. Available operations on polygonal meshes can be subdivided in two main categories:

- **Full mesh modelling.** There exist a set of algorithms allowing improving the quality of the whole 3D mesh. They permit to reposition, decimate and subdivide all faces of the 3D polygonal model in order to get a high-quality polygonal model. These algorithms are usually applied before starting the modelling phase.
- **Local mesh modelling.** It permits to locally model the initial 3D polygon mesh to design requested final object (Fig. 1.(a). and Fig. 1.(b).). These algorithms allow in real-time adding/removing points and faces in specific zones of the 3D model according to the user's needs.

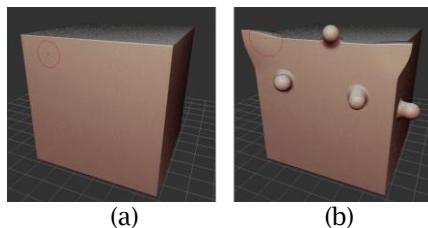


Fig. 1: A polygonal cube (a) modified in several zones by using both dragging and inflating virtual tools (b) of SceptGL application.

Summarizing, several software applications have been developed to test potentiality and performances of these algorithms applied on polygonal meshes [2], [13]. However, each one is based on its own data structure and only few can be easily re-used to develop customized CAD applications requiring mesh modeling. In fact, in several contexts, from industrial to medical ones, modeling tool should gather all useful algorithms for mesh modeling.

Research aim:

This paper presents a set of software modules for using a set of existent mesh modelling algorithms as part of an open-source software library widely used to develop scientific virtual environments.

The main aim has been the software development of local mesh modelling algorithms to shape triangle meshes as an extension of VTK (Visualization ToolKit) library [15] (an open-source software development kit -SDK) usually exploited to develop scientific applications [1], [11]). The implemented algorithms permit to create virtual modelling tools that emulate the tasks executed during the traditional manufacturing process by using hands or tools (e.g., rasp). They have been applied to the design of a medical device that requires a high level of customization, i.e. the socket of lower limb prosthesis.

Software development:

At present, there exist important applications developed by starting from VTK in relevant contexts, such as medical field and industrial 3D simulations. VTK makes available file reader in order to load the mesh topology inside a memory structure which can be used to get needed information (e.g., points, faces and connected edges) through a single programming class named `vtkPolyData`.

The developed modules encapsulate known algorithms presented in [9], [14]. They allow to locally model the mesh in real-time by starting from both operations of decimation and subdivision. Algorithms of local decimation allow to reduce the number of triangles in the selected zone of the 3D mesh; a part of points is deleted and remained points are repositioned in order to get new triangles with same size without losing the original shape. The algorithms of subdivision permit to generate smaller triangles by starting from the selected triangles. Also in this case, new triangles are repositioned with same size along the surface. The repositioning of triangles is based on the mesh Laplacian smoothing [10] calculating the curvature along the selected surface through the Laplace operator.

Subdivision and decimation are always applied before each operation of modelling according to final behavior of the emulated operation, such as inflate/deflate the mesh, flattening a selected part and so on. All developed algorithms make the mesh of high-quality and the modelling operations keep the triangle mesh manifold.

Software architecture

The software platform has been developed in C++ by following the object-oriented paradigm (Fig. 2.). The system links a polygonal mesh to a set of virtual tools that can be applied to design the product 3D model by local mesh modeling. Furthermore, an event listener has been developed to exploit the execution pipeline of VTK, which automatically detects interaction events, such as event of mouse and keyboard. Thus, the software developer can associate an interaction event for each modelling operation the final application requires.

An acquired 3D triangle model can be loaded into a `vtkPolyData` object using either `vtkObjReader` or `vtkStlReader`, which can respectively read OBJ and STL files in application based on VTK. The class `vtkPolyData` contains the topology information of the associated 3D mesh previously loaded. This class makes available a set of methods allowing to dynamically manage the lists of points and triangles and the relations between them. Namely, the modules use these methods to add/remove/modify both points and faces and their relationships. After the execution of the algorithm relative to either subdivision or decimation, these methods update the topology of the triangle mesh. Once the algorithm has been executed and the topology modified, the method `Modify()` automatically updates the OpenGL Vertex Buffer Objects for rendering the mesh according to the new topology.

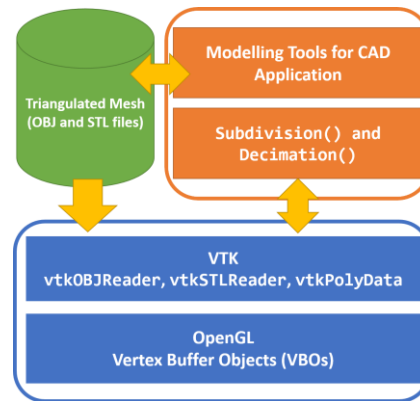


Fig. 2: Software architecture: orange rectangle contains the developed modules of the new platform; blue rectangle shows the basic SW library used.

Application of mesh modelling tools to a medical case study:

The system has been adopted to develop a knowledge-guided CAD system, named Socket Modelling Assistant (SMA). SMA is part of a virtual design environment for lower limb prosthesis design and permits to create the 3D model of the socket according to patient's characteristics (e.g., lifestyle) and residual limb morphology [6], [7].

The 3D socket model is created by applying the virtual modeling tools on a polygonal mesh acquired either using Magnetic Resonance Imaging (MRI) technique or laser scanning. SMA is also interfaced with a Finite Elements Analysis (FEA) system to analyze the pressure distribution at the socket-residual limb interface and, once reached the final shape, exports the polygonal mesh for 3D printing with data-driven multi-material option [8].

In this application, the socket is designed around the polygonal model of the residual limb acquired by MRI. Traditionally the prosthetist modifies locally the shape of a positive chalk; therefore, we had to develop virtual modeling tools that emulate the traditional design tasks interacting with dynamic meshes (the geometrical models of the residual limb and the socket) and locally modify the mesh either in automatic or interactive mode.

Two virtual tools, named *sculpting* tool and *lower zone definition* tool, have been developed, which exploit developed platform for local mesh modelling. Using the *sculpting* tool, the user can proceed with an interactive shape manipulation on both socket and amputated lower limb. A set of virtual operations have been developed to emulate the usual operations done by the orthopedic technicians by using tools and hands to shape the positive chalk model of the socket.

The operations allowed on the mesh are subdivided as in the following:

- **Inflate/Deflate:** the user can add and remove material along the normal of the selected surface. If the user select the inflate operation the algorithm adds virtual material towards the outside of the virtual model. Deflate operation execute the same operations toward the interior of the virtual object (Fig. 3(a)).
- **Flattening:** this operation allows flattening the selected surface. Flattening operation can be applied to the initial model of virtual residual limb in order remove a particular deformation relative to the acquired residual lower limb Fig. 3(b)).
- **Dragging:** this tool allows adding new mesh portions by dragging the cursor in any direction. Inside SMA, dragging operation is used to add a new part of the mesh along the ischiatic area in order to prepare the internal side of the final socket (Fig. 4(a)).
- **Smoothing:** this operation smooths and brushes the 3D mesh to remove details relative to the initial 3D model of the amputated lower limb, such as scars (Fig. 4(b)).

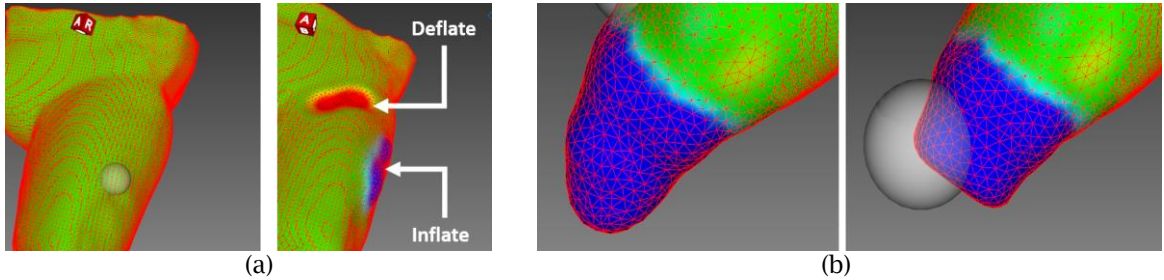


Fig. 3: The initial 3D model of the residual lower limb is shaped by using deflating and inflating operation in two different zones (a). Flattening operation exploited inside SMA (b).

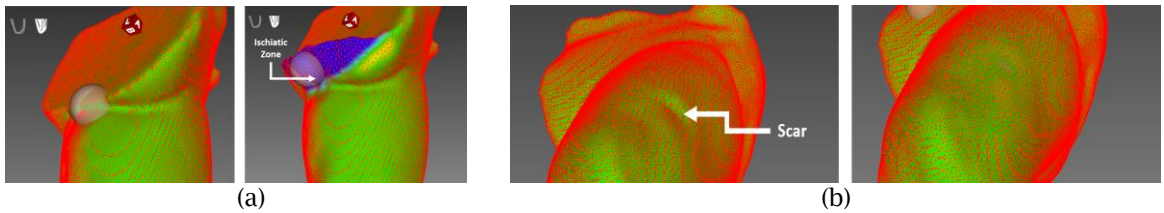


Fig. 4: Dragging operation applied on a 3D scanned model of a residual lower limb (a). Using the virtual smoothing tool on the virtual residual limb to remove a scar (b).

Developed platform has also been exploited to automatically design the lower part of the socket. The *lower zone definition* tool creates the final shape of the lower zone by starting for the 3D model of the residual limb. The user must only define the final distance between the lowest point of the residual lower limb and the lowest point of designed socket using the proper slider (Fig. 5.a.). According to this distance SMA automatically applies the dragging operation until the mesh has reached the required shape. Finally, flattening is applied in order to create a planar surface in order to simplify the adding of the link with the 3D socket model (Fig. 5.b.).

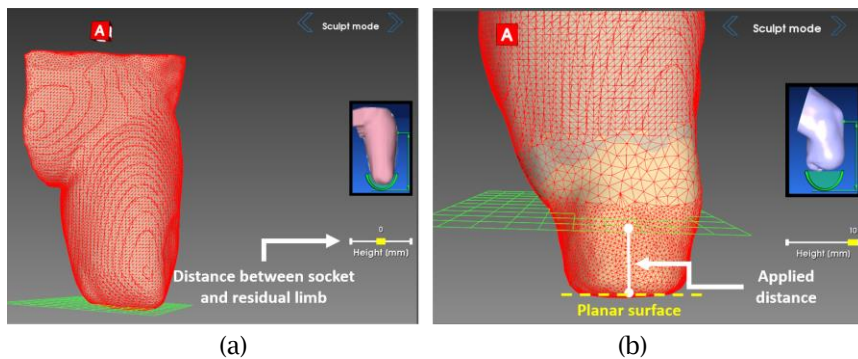


Fig. 5: Definition of the lowest part by using local modelling operations in automatic way.

Conclusions:

The paper describes a software platform for local mesh modeling based on VTK. This research work aims to go beyond the limits of other software solutions for mesh modeling, which are not easily reusable to create CAD application in industrial and medical fields.

The developed platform makes available several software modules for creating virtual modelling tools based on both subdivision and decimation of a triangulated mesh. The structures of the topology of the mesh are totally based on those defined in VTK and thus, this approach allows the developers to exploit all other functionalities that are available in VTK.

The platform has been tested with SMA, a knowledge guided CAD application to design the socket of lower limb prosthesis. Two virtual tools have been developed that exploit new local mesh modelling

tools. The *sculpting tool* uses the local mesh modelling on the 3D mesh of the acquired residual lower limb to execute several operations for customizing the final socket shape. After the preliminary test phase performed by ICT engineering students, the use of the virtual tools was considered very easy and intuitive. Also the modelling operations allow executing detailed tasks on the mesh, which cannot be performed by the automatic procedure of SMA. The sculpting tool meets some limits relative the selection of the local zone the user needs to modify. In fact, the sphere selection does not permit to execute modeling operation such as corners and chamfers, for instance to link the socket with the rest of the prosthesis.

The *lower zone definition* tool exploits developed modules in automatic way. In fact, the user does not directly use the mesh modelling tools, but he/she exploits them through the slider that manage the distance between lower part of the residual limb and the lowest part of the designed socket. This approach totally hides the local mesh modelling to the user and the automatic procedure will be used also in other virtual tools of SMA in order to maintain the quality of the mesh very high. Furthermore, the automatic procedure of local mesh modelling will be proposed also in other medical contexts, such as the design of maxillo-facial and dental prosthesis.

Finally, future developments have been planned in order to test the developed modules in a mixed reality application to recreate the orthopedic laboratory where the orthopedic usually operate.

References:

- [1] 3DSlicer, <https://www.slicer.org/>.
- [2] Blender, <https://www.blender.org/>.
- [3] Bordegoni, M.; Covarrubias, M.; Caruso, G.; Cugini, U.: Freehand gesture and tactile interaction for Shape Design. *Journal of Computing and Information Science in Engineering*, 16(4), 2016. <https://doi.org/10.1115/1.4033230>
- [4] Calabrese, C.; Salvati, G.; Tarini, M; Pellacini, F.: cSculpt: a system for collaborative sculpting. *ACM Transactions on Graphics (TOG)*, 35(4), 2016. <https://doi.org/10.1145/2897824.2925956>
- [5] Colombo, G.; Comotti, C.; Redaelli, D. F.; Regazzoni, D.; Rizzi, C.; Vitali, A.: A Method to Improve Prosthesis Leg Design Based on Pressure Analysis at the Socket-Residual Limb Interface. *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers, 2016. <https://doi.org/10.1115/detc2016-60131>
- [6] Colombo, G.; Facoetti, G.; Rizzi, C: A digital patient for computer-aided prosthesis design. *Interface focus*, 3(2), 2013. <https://doi.org/10.1098/rsfs.2012.0082>
- [7] Colombo, G.; Facoetti, G.; Rizzi, C.; Vitali, A.: Mixed reality to design lower limb prosthesis. *Computer-Aided Design and Applications*, 13(6), 2016, 799-807. <https://doi.org/10.1080/16864360.2016.1168223>
- [8] Comotti, C.; Regazzoni, D.; Rizzi, C.; Vitali, A.: Additive Manufacturing to Advance Functional Design: an Application in the Medical Field, *Journal of Computing and Information Science in Engineering*, 2016. <https://doi.org/10.1115/1.4033994>
- [9] Dunyach, M.; Vanderhaeghe, D.; Barthe, L.; Botsch, M.: Adaptive Remeshing for Real-Time Mesh Deformation. In *Eurographics*, 2013, 29-32.
- [10] Herrmann, L. R.: Laplacian-isoparametric grid generation scheme. *Journal of the Engineering Mechanics Division*, 102(5), 1976, 749-907.
- [11] Paraview, <http://www.paraview.org/>.
- [12] Rajic, A.; Desnica, E.; Stojadinovic, S.; Lazic-Vulicevic, L.; Eric, S.: Reverse engineering and CAD inspection of knee implant using the Noomeo Optimum 3D Scanner. *Annals of the Faculty of Engineering Hunedoara*, 13(2), 2016, 39.
- [13] SculptGL, <https://stephaneginier.com/sculptgl/>.
- [14] Stanculescu, L.; Chaîne, R.; Cani, M. P.: Freestyle: Sculpting meshes with self-adaptive topology. *Computers & Graphics*, 35(3), 2011, 614-622. <https://doi.org/10.1016/j.cag.2011.03.033>
- [15] Visualization Toolkit, <http://www.vtk.org/>.