Software Engineering Project

Description of the project:

General overview:

In 2011, Ivan Sipiran and Benjamin Bustos published a paper entitled "Harris 3D: a robust extension of the Harris operator for interest point detection on 3D meshes" (see references and links), in which they present an elegant extension to 3D meshes of one of the most widely used operator in image processing, as illustrated below:

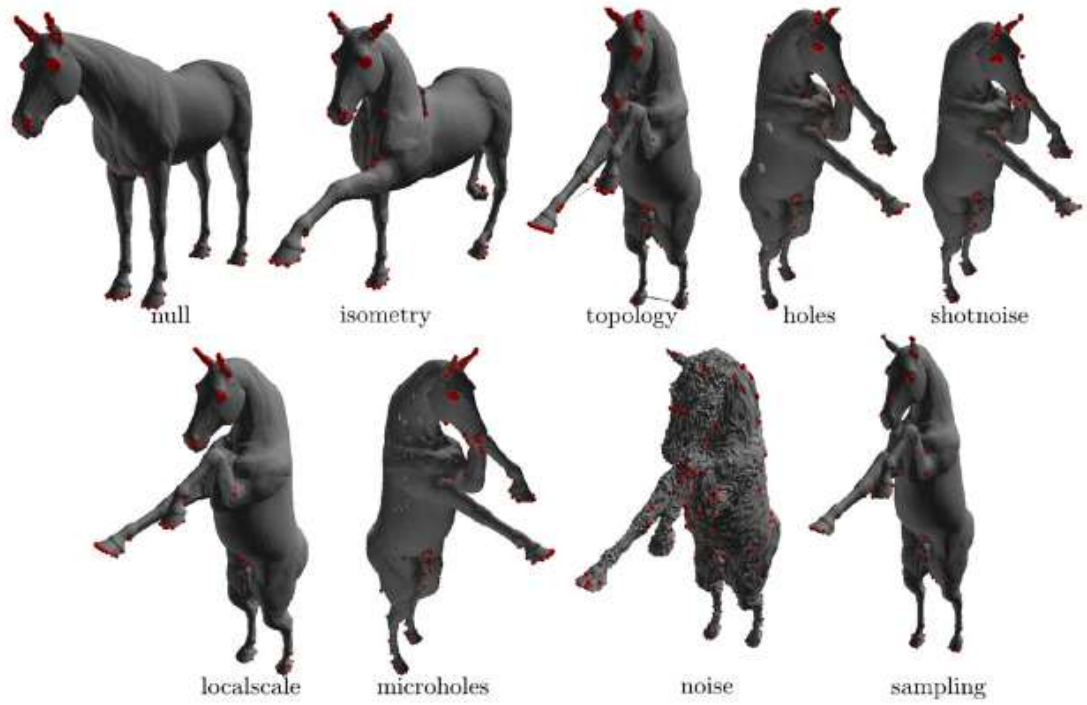

Figure 1: Examples of interest point detection for various poses, noises, and topological perturbations - Image taken from [1]

Objective

This project aims :

1. To (first understand, then) implement a research paper
2. To develop your programming skills in C++ (development of internal data structure, graphical user interface, 3D rendering, etc.)
3. To develop transversal skills (project management, team working, report writing, presentation, etc.)

Practical organization

Students can be spread within **groups of 3 students at most.**

The project is designed to be completed over 3 to 4 months and is supposed to start this week (end of September) to be finished during Christmas vacations. Final defense and presentation will be held in January.

Important milestones :

1. Mid term report due on Sunday November the 11th
2. Final report due Sunday January the 6th
3. Defenses will be held straight after the Christmas vacations. Precise time slots will be provided soon

Consider (and set) your professor as a member of your groups, so that he can follow your progresses and monitor your work throughout the semester. Some time slots for discussion will be arranged on a weekly (or bi-weekly) basis or upon groups' request. Use collaborative tools to work efficiently (github for code repository, trello (or anything else you like) for project management)

Evaluation and grading

All the students within a group shall be graded equally. If all the students in a group agree, some adjustments can be done to reward extra points for outstanding commitment (or inversely, some points might be substracted in case of lack of commitment).

The group will be evaluated as follows:

1. Technical quality of the project (50%). This includes the quality of the implementation (including a critical analysis of the provided sources, comments and explanations on the developed solutions), level of functionality achieved by the software, the easiness of use, the quality of the user interface, etc.
2. Project Management (25%). This includes the proper planning of the project, its realistic implementation and timely reach of the project milestones, the respect of all the deadlines, and the **equal share** of the workload between the students. In case of unsolvable conflicts that require the tutors to take actions (such as swapping students or rearranging groups), some points will be deduced from the final mark.
3. Reporting and Defense (25%). **Report must be written using LaTeX and present in detail all the technical aspects of the implementation.** The defense will be held on January. Each group will have an equal duration (to be adjusted depending on your needs) to present their software, implementations, and any other aspect deemed as important.

Required features and guidelines

Each group will have the opportunity to propose, then develop (if accepted), any functionalities deemed necessary to the software. You will not start your implementation from scratch since you will be provided all the sources developed by Sipiran and Bustos. However, as you will quickly see, the authors made the choice to rely on CGAL library which is an extremely powerful library, but rather delicate to install and manipulate if you are not an experienced user. Besides, the structure used to manipulate 3D meshes alone is rather... undeveloped, because CGAL does most of the work. Consequently, you are asked to implement this research paper under the following conditions :

- Do not use CGAL and rather **come up with your own class representation for 3D meshes**. Your class should allow to efficiently compute various neighbourhoods (points, faces). It should also allow for the representation of edges as well as any additional information you might consider important.
- The software must be equipped with an **interactive graphical user interface** in which the user can move the object, zoom in/out, and adjust the internal parameters of the provided algorithms to compare results
- The **3D rendering must be performed using Opengl**
- **Implementation must be in C++ only**. IDE can be Qt or Visual studio (whatever you choose, please make sure you can recompile your code interface in both IDEs). **OS must be Windows only.**
- The usage of extra libraries such as Point Cloud library, Eigen, etc. is allowed but it must be discussed with the tutors beforehand (to avoid exotic libraries, or libraries that do all the work for you).

Advices and other information

All the groups are allowed to collaborate, and some parts of the project can be developed jointly (once agreed with tutor). Collaborative work and active attitude toward the exchange of knowledge (seminars, tutorials, code examples, group discussion) will also be awarded extra points.

However, working together does not mean that all the groups shall have identical outputs…

Pay a special attention to the project management and the spread of the workload. Be sure that you are comfortable in your role within the group. Do not hesitate to seek for help. Pay also attention to over-commitment, exhaustion and health issues. Soon, you will have to work on several other projects, courses level will raise, weather will become cold, etc. Manage your time, spare your efforts, avoid coding late at night, etc

Never hesitate to ask for feedback and guidance, and avoid seeking for ready-to-go solutions. Please use our Edmodo platform for question so that all of us (including tutor) can benefit from the answers. Be sure you systematically cite your sources (papers, images, code) to avoid plagiarism.

At last, the sources, reports and documentation are available at :

References and useful links

1. Ivan Sipiran project page, including C++ and Matlab Code, original paper as well as several other excellent references: https://users.dcc.uchile.cl/~isipiran/harris3D.html
2. The point cloud library : http://pointclouds.org/
3. Eigen Library : http://eigen.tuxfamily.org/

Most importantly, enjoy!