Project ideas

Here is a preliminary list of possible project topics. For some of them there one or more
relevant chapters in [12]. If you decide on a project and can’t get the related papers on
the web (due to access priviledges), please let me know and I will send you the papers.
The references given here are indicative, you can find more (and perhaps better ones) if
you search around. (More information on each topic will keep on coming!)

• Porting Arrangements package demo (currently in Qt3) to Qt4. (This will require
building the demo with Qt3 so that we see what’s in it!)

• Random point generator on a circle. The generator curently in CGAL works with
Cartesian<double> kernel (without taking into account the actual kernel that is
used). This of course creates problems as we’ve seen in the 2d covex hull exercises.
It will be nice to have an exact generator.

• Convex hull (2d, 3d) in $O(n \log h)$ time. Here we would to implement Chan’s algo-
rithms [8]), which are not part of CGAL. There are several reports on the web about
implementations and experiments with these algorithms.

• Triangulating simple polygons (non-convex) in $O(n)$ time. Implement the algorithm
by Amato et al. [4]. For background, look at Ch. 3 in [12].

• Shape matching: minimizing the Hausdorff distance of two point sets under trans-
For several exact and approximation algorithms look at [11, 14, 10, 1, 2]

• Shape matching: minimizing the Earth Mover’s Distance (EMD) under rigid motions
(arrangements). EMD is a well known similarity measure commonly used in computer
vision and other areas. Here, we would like to implement several approximation
algorithms and heuristics. Look at [7] and references therein. For fixed positions of
the two patterns (that is, when transformations are not considered) there are several
implementations on the web for computing the EMD (something we’ll need to do),
see for example [17].
• Guarding art galleries, guarding terrains (visibility polygons, regions). For visibility polygons in the plane, look at Ch. 3 in [12], the chapter [16], and numerous references therein. Note: some of the early algorithms were wrong but corrected in subsequent papers. Find a reference for an algorithm that works also for non-simple polygons, i.e., with holes. For visibility regions in terrains, look at [13].

• Shortest paths among obstacles (visibility graphs). Look at Ch. 15 [12] for a description of some algorithms. We can later discuss which algorithm to implement.

• Shortest paths on terrains. Look at [9] and [15].

• Motion planning in robotics: can a robot reach its destination? (minkowski sums, visibility-voronoi complex). Look at Ch. 13 in [12].

• Curve reconstruction from points. Implement the Crust Algorithm by Amenta et al. [6, 5]. This is one of the first reconstruction algorithms that appeared in the literature.

• Surface reconstruction from point sets. I am still discussing with Dror on this one.

• Binding CGAL to other (scripting) languages.

• Look at the packages of CGAL to find something that you like.

References


