Prof. Dr. Leif Kobbelt
Kersten Schuster, Philip Trettner

Visual Computing Institute | RWTH AACHEN UNIVERSITY

Real-time Graphics Assignment 4

Date Published: November 22nd 2017, Date Due: November 29th 2017, 11:44

- The assignments have to be done in groups of 2 students.
- Hand in the solutions to the exercises via L^{2}P.
- You are only allowed to change code inside the marked strips (STUDENT CODE BEGIN/END)!
- Any questions? → L^{2}P discussion forum or rtg@cs.rwth-aachen.de!

If not done yet, obtain the (publicly accessible) exercise framework and assignments from `https://www.graphics.rwth-aachen.de:9000/Teaching/rtg-ws17-assignments/`.
Use **git pull** to fetch the newest changes of the framework (including the code for this exercise).

The **only** files that you should modify and **upload**:

- Cloth.cc

In this assignment you will write a simple cloth simulation. The simulation is realized via a 2D grid of particles and springs connecting them. You can drag and drop the cloth with the left mouse button. W, A, S and D keys can be used for navigation while the right mouse button allows you to rotate the camera. Double-clicking the center mouse button resets the camera position.

Each subtask corresponds to a code strip with more detailed comments and hints.

Exercise 1 Simple Rendering [1 + 1 + 1 = 3 Points]

(a) In `createVAO(...)` (third strip), you should generate flat-shaded geometry for the particles by pushing back vertex data (position, normal, color) to the vector `vertices`. (Flat) face normals can be computed using the cross product of two vectors from that face/triangle. See comments in the code for more information.

(b) To get smooth (non-flat) particle normals, you should average normals from adjacent triangles in `createVAO(...)` (first strip).

(c) In the second strip in `createVAO(...)`, your job is to generate smooth-shaded geometry for the cloth mesh. For every quad (defined by four adjacent particles on its corners), four triangles have to be created (with one additional vertex in the quad center, see code comments for more information). Again, each triangle vertex consists of position, normal, and color.

Exercise 2 Simple Cloth Simulation [2 + 2 + 1 + 1 = 6 Points]

(a) Update the motion system (particle position and velocity) in `updateMotion(...)`. You cannot use acceleration directly, but you have access to the accumulated forces and the mass of particles.

(b) Apply Hooke's law in the second strip in In `updateForces(...)`. For that, you should compare the `restDistance` of the spring with current distance and add forces to the particles accordingly (`float accumulatedForces`). The spring constant (stiffness) is stored in `springK`. Remember that forces on the two particles have contrary signs. If you want (optional), you can accumulate the particle's stress variable for rendering the cloth in green-/reddish color (depending on stress).

(c) Without damping, the system will be highly unstable. In `updateForces(...)` (third strip) reduce forces according to velocity and damping factor `dampingD`.

(d) Apply gravity force to all particles. The acceleration constant is stored in `gravity`.

Exercise 3 Cloth - Sphere Collision [1 Point]

(a) In `addSphereCollision(...)`, detect whether a particle is inside the sphere and if so, project it to the (closest) sphere surface.
For stability reasons, you also have to project the particle velocity onto the tangent plane (i.e. there must be no velocity in normal direction anymore).