

VIRTUALLY ENHANCED SCULPTURE

Blaž Meden, Blaž Jeršan, Gorazd Rajar, Franc Solina

**Computer vision laboratory,
Faculty of Computer and Information Science, University of Ljubljana,
Večna pot 113, 1000 Ljubljana, Slovenia
E-pošta: franc.solina@fri.uni-lj.si,
{blaz.meden, blaz.jersan, gorazd.rajar}@gmail.com**

ABSTRACT: *The article describes how a simulation environment for projecting virtual water droplets trickling over slanted surfaces was implemented. The motivation for this work was to enrich a stone sculpture by introducing a dynamic dimension into the usual static environment. The 3D shape of the projection surface on the sculpture was captured by the Kinect range scanner.*

1. INTRODUCTION

An age old way of making stone sculpture more dynamic and energetic is to combine stone with running water. This is manifested by the abundance of fountains made out of stone throughout history and in all parts of the world. We decided to animate a stone sculpture with virtual water droplets using modern computer technology. In recent years methods for capturing the 3D shapes have become much faster and easier to use. Sensors such as the Microsoft Kinect have opened the door to new applications where the 3D shapes of objects and scenes have to be known.

In this article we describe how we used the Kinect range sensor to capture the 3D shape of a stone sculpture to be able to project on the sculpture virtual water droplets that are running over the sculpture obeying the physical law of gravity. Projects which are somewhat related to our project are interactive sandboxes [1,2,3] where the users can shape the sand in the sandbox into new forms which are continuously captured by a Kinect. A video projector is also used to color the sand according to some rules. These boxes can be educational, demonstrating geographic, geologic, and hydrologic concepts or a platform for interactive games. The virtual space projected onto a physical 3D landscape becomes alive with all forms ranging from spiders, beetles and snails through ammonites and trilobites to sharks and dinosaurs, all inhabiting a Mixed Reality ecosystem [4].

To simulate the trickling of water droplets over a slanted surface we have to achieve the following goals:

1. capture a cloud of 3D points on the surface on which we want to project the animated droplets,
2. create a 3D model from the cloud of 3D points,
3. create a dynamic model for water,
4. project the liquid droplets on the 3D model and simulate their movement according to the laws of gravity,
5. adjust the projection of animated droplets to the 3D surface.

For testing of the system we used instead of the envisioned stone sculpture a white sheet stretched over support. The actual stone sculpture is still in the process of carving.

The rest of the article is divided as follows, Section 2 describes the equipment and tools that we used for implementation of our goals, Section 3 is on the installation and operation of the whole system, Section 4 shows the results of simulation, and Section 5 concludes the article.

2. EQUIPMENT AND TOOLS

2.1 Hardware

Range sensor—Kinect Kinect is a Microsoft product that captures range images on the principle of using multiple cameras. Kinect appeared together with the game console Xbox 360 primarily to capture the movement of players of computer games. But since Kinect is easy to use and is reasonably priced it became popular with a large user community that applies Kinect to a large variety of applications that need 3D shape capture. The output of Kinect is a cloud of 3D points in a given coordinate system.

Video projector For the projection of animated droplets on the 3D surface any digital video projector could be used that has a wide projection angle so that the projector needs not to be elevated too high above the projected surface. The elevation depends of course also on how long and wide is the sculpture.

Computer The processing power requirements for the envisioned system are quite high. The system must support the processing of the input image and the simulation of water droplets which is computationally very demanding and proportionally dependent on the number of simulated water droplets. We used a laptop computer with two graphical cards which made possible a simultaneous simulation of 15.000 water droplets at the speed of 30 frames per second. This gave a reasonably good approximation of liquid simulation.

2.1 Software

OpenFrameworks OpenFrameworks is an open source framework for integration of various technologies which are used in computer graphics, computer vision and interactivity in general. The framework is written in C++ and represents a high level abstraction of OpenGL, GLEW, Glut, FreeType, OpenCV and other libraries. The framework enables the use of different libraries which are developed by an active

community of developers. Beside the OpenFrameworks we used in the project also ofxKinect (which supports the communication with the range sensor Kinect) and ofxGui (which offers elements for adjusting the user interface parameters).

Microsoft Visual Studio Microsoft Visual studio is an integrated development environment which supports the C++ programming language and is on the Windows operating system the recommended tool for application development under OpenFrameworks. The entire programming code of the project is written in C++ so that during the entire project development Visual Studio was used.

3. INSTALLATION AND OPERATION OF THE SYSTEM

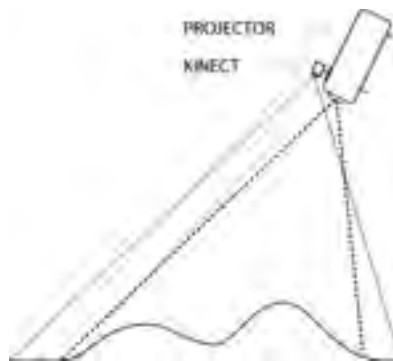


Figure 1: The setup of the equipment: the video projector and Kinect are aligned and close together above the sculpted surface.

The range sensor and the video projector must be mounted above the sculpture so that all parts of sculpture's surface are visible from their viewpoint. The field of view of the range sensor and the video projector should overlap as much as possible (Fig. 1) so that the alignment of the projection to the 3D model of the sculpture can be accurate enough and simple to achieve. Kinect has in our case a wider field of view as the video projector. Ideally, the projector and Kinect should be mounted directly above the sculpture so that the whole sculpture can be evenly illuminated by the projection and any shadows due to the configuration of the sculpture could be eliminated.

3.1 Range image capture

Kinect captures 3D points by using an infrared projector and infrared camera. The captured 3D points can be visualized as an intensity image of size 640×480 pixels or a cloud of 3D points. Our application takes the cloud of 3D points and aligns them with the global coordinate system of the space. Next, the 3D points are projected onto the horizontal plane. The projection is then saved as an intensity image. This gives us the perpendicular view of the scanned objects. From the projection we generate the 3D model of points which we color according to their height. The generated model serves as

a virtual representation of the scanned space and represents the supporting surface for liquid simulation.

Determination of the coordinate system of the 3D point cloud The 3D points obtained from Kinect are defined in a coordinate system which is rotated according to the slant of the Kinect sensor. The 3D points must therefore be transformed with an inverse rotation into a global coordinate system. The slant of the sensor is indicated by the gyroscope which is built into the Kinect. It is possible also to manually set the slant angle on the user interface of the system. The 3D points are after the transformation aligned with the global coordinates of the space. The global coordinates represent the following dimensions:

- z axis – vertical dimension or the direction of gravitation,
- x axis – first dimension of the horizontal plane,
- y axis – second dimension of the horizontal plane.

Conversion of the 3D point cloud into a 3D model The 3D model was constructed out of the generated intensity image which represents the vertical projection of the 3D point cloud. The 3D model is a grid of points where the coordinates of each 3D point are the x and y coordinate of the corresponding pixel in the intensity image and the z coordinate is the intensity (height) of the pixel. This 3D model represents the surface of the scanned sculpture. For debugging and easier visualization we color coded the height of the 3D model using the colors shown in Fig. 2.

3.2 Simulation of water droplets

Since we could not find an appropriate open source solution for the simulation of water droplets that move in the direction of the highest gravity gradient, we decided to write our own code. For the purpose of our visualization goal we decided to model the liquid with a high number of small particles in the form of spheres that move on the surface of our 3D model. Each particle or droplet is an independent unit that moves independently. The direction and the velocity of the movement of each droplet depend on:

- the gradient of the surface where the droplet currently is,
- the direction and velocity of the droplet in the previous step of the simulation,
- an adjustable chance factor.

A droplet therefore moves down the slope according to the slant of the surface. Since a droplet has a mass, the inertia from the previous step influences its movement in the next step of the simulation. Individual droplets however do not have any physical dimension and therefore no collisions between droplets can appear. The detection of collisions is computationally demanding and would reduce the maximum possible number of droplets that we can simulate on our hardware in real time. Additionally we do not use any shading and illumination of droplets which is also computationally demanding and does not add much to the realistic view of the liquid. A multitude of moving droplets create the appearance of running liquid. Since a large number of droplets reappear in the

same point of the 3D model and continue according to the surface gradient, we introduced a chance factor which takes care that the path of the water droplets originating from the same point is not identical. This chance factor greatly improves the appearance of the liquid and is computationally not demanding.

3.2 User interface

During the simulation of the droplets the user can change different parameters of the simulation using a simple graphical user interface. When the whole installation is being set up, the first task is to align the picture projected with the video projector to the projection surface on the sculpture. The picture is initially approximately aligned based on the position of the Kinect and on the centroid of the 3D points. The virtual camera in the simulation model is positioned in the point where the Kinect is mounted and directed towards the centroid of all 3D points captured by Kinect (approximately toward the projection surface).

However Kinect and the video projector can not be exactly in the same position as can be seen in Fig. 1 and therefore they cannot have exactly the same viewing angle. This difference must be adjusted manually. This can be done on the user interface by changing the projection angle and the scaling of the 3D surface model.

User interface for droplet simulation On the user interface we can change the following parameters of the simulated droplets:

- the number of droplets in the simulation,
- the viscosity of the droplets,
- the value of the chance factor for the movement of the droplets,
- color of the droplets,
- transparency of the droplets.

Creation of water sources Droplets must come from somewhere in order to run down the sculpture. The user can define the position of water sources on the sculpture interactively by selecting the position on the surface model with a mouse click. One can select several sources, however, the number of water droplets is then divided among them so that the simulation always uses the same predefined number of water droplets. If the maximum possible number of droplets in the current implementation (15.000) is divided among ten sources one can still get a realistic visualization.

4. RESULTS

Fig. 2 shows a demonstration of the achieved visualization. For this visualization, the maximum number of 15.000 droplets was selected. The droplets are colored blue and when they reach the “water level”, also colored blue, they disappear and reappear in one of the sources. This visualization is only a proof of our concept for the simulation of water droplets since a white sheet supported by boxes was used as the projection surface.

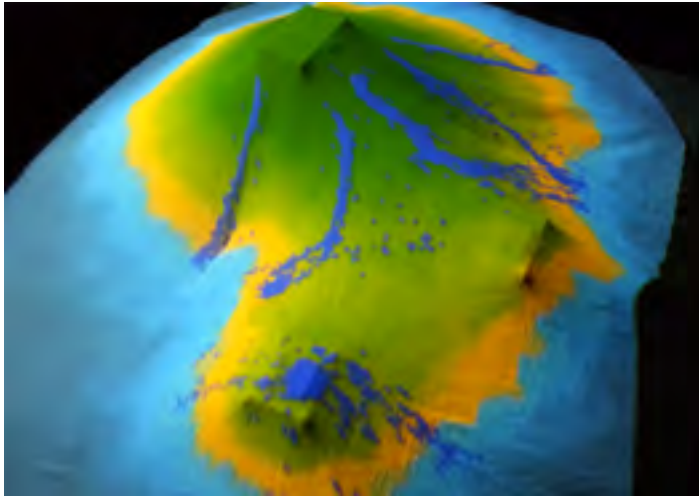


Figure 2: Final visualization of water droplets running down sloping surfaces.

A stone sculpture is being carved out of white marble specially for this project. The marble will not be polished to a high degree so that the light will be absorbed rather than specularly reflected.

5. CONCLUSIONS

We implemented a system for projecting dynamic content (trickling of water droplets) onto a 3D varying projection surface. The system was tested on a simulated surface since the actual stone sculpture which will be the centerpiece of this interactive installation is in the process of carving.

REFERENCES

1. Kreylos, O.: Augmented reality sandbox. <http://idav.ucdavis.edu/~okreylos/ResDev/SARndbox/>, online; accessed 14-November-2014
2. Project mimicry. <http://mimicry.monobanda.nl>, online; accessed 14-November-2014
3. Sandystation the interactive sandbox. <http://en.sandystation.cz/>, online; accessed 14-November-2014
4. St. Jean, J.: Augmented reality evolution timeline using open-frameworks. <http://developkinect.com/news/augmented-reality/augmented-reality-evolution-timeline-using-openframeworks> (2012), online; accessed 14-November-2014