Geometrical Image Processing Laboratory
Computer Science Faculty
Technion

Make It Stand-Balancing Shapes For 3D Fabrication

By: Dvora Nagler
ID: 20046281-0
Supervisor: Yonathan Aflalo
Semester: Spring 201410/23/2014
Submission Data: 26/10/14

04-8294300 : פקס   04-8295535 : טל   yaronh@cs.technion.ac.il
http://gip.cs.technion.ac.il

Overview-Current problem and\or existing solutions

Artists, designers and architects use imbalance to their advantage to produce surprising and elegant designs.

The balancing process is challenging when manipulating geometry in a 3D modeling software.

There is usually no indication of gravity, support or weight.

In fact, volumes are only represented by their boundaries.

So it can be quite unintuitive to change the weight distribution.

With the advent of 3D printing technologies, we can produce physical realizations of 3D models.

If the shape is difficult to balance the printed object will often fail to stand.

In the recent years there has been a lot of research going on about 3D printing and fabrication in general.

In particular, advanced methods have been developed to make a model printable.

Currently, we don't find an articles focusing on shape balancing besides it.

The Solution:

We have an input surface representing a solid object.

Our goal is to modify this volume such that once printed, the model stands.

To do so, we manipulate the volume in two ways:

We carve inside the model to create inner voids.

But because this might not be enough we also consider deforming the model.

These two manipulations change the mass distribution and thus the center of mass position.

In our project, we focus only on inner carving.

The method generates two output meshes, MO and MI , defining the outer and inner surfaces.

MO is just a deformed version of the original mesh and MI is the inner surface representing the boundaries of interior voids.

When we have these two meshes we have all the information we need to 3D print the object.

We can formulate our objective as an energy to minimize in which the inner and outer surfaces become our variables.

The first one ECOM measures if the model stands.

The second one EM measures if the deformation of the model is acceptable.

We can pay attention that modifying the inside does not modify the visual appearance of the model.

The base of support is defined by all of the points of the mesh touching the ground, which we call the support polygon.

According physics laws, the object will stand if its center of mass c projects along the gravity direction into this support polygon.

Thus, we can define a target c* anywhere inside this polygon.

Means, we can find only x,y coordinates and can assume g=z while g is the gravity direction (otherwise, we can rotate the model).

Also it is sufficient to stand, we ensure the model will not trivially topple when pushed by small external forces by adding as stability offset from the support boundaries.

We simply use a squared distance between the current center of mass c and its target c* after projection on the ground plane.

If the center of mass projects onto this target then we know the model is standing and is stable enough.

Assuming a constant density, the center of mass can be computed by:

$$c(M_I, M_O) = (mean(x(in)), mean(y(in)), mean(z(in)))$$

Instead of :

$$c(M_I, M_O) = \frac{1}{mass} \int_{vol} (density \times r)dr$$

We are doing our calculations on a voxel grid. Each voxel in size of 1^3.

Each voxel is assigned a binary value saying if it is filled or empty.

In addition, we enforce some minimal thickness by simply disallowing to carve the voxels near the boundary, in order not to damage the outer surface.

We obtain the inner surface by taking the boundaries of the carved regions.

The volume discretization is computed once at the beginning and later deform together with the surface.

Firstly, we initialize all the voxels to be filled.

Given the current center of mass c and the target, we are looking on a plane perpendicular to the projected direction (c,c*).

Carving the voxels located on the right side of c* potentially brings the center of mass closer to the target and may decrease the energy.

While the filled voxels located on the left cannot help bringing the center of mass closer to the target, so we ignore them.

We remove voxels one by one from right to left (in order not to create floating components), and keep track of the energy.

When all the voxels have been processed, we pick the carve pattern that corresponded to the lowest energy value.
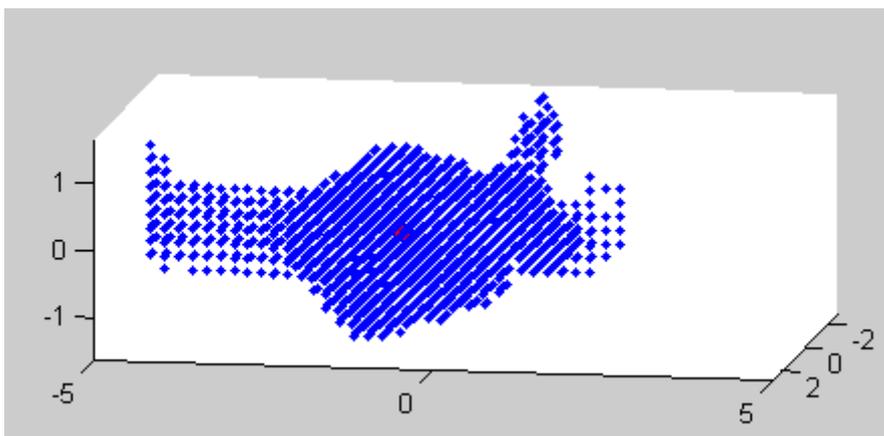
Inner surface of Homer after Algorithm

Algorithm:

1. Draw a line which go over c^* in the direction of g

2. Performing projection of point c on line obtained in step 1

3. Calculate the plane that goes over $(c^*)^{\perp g}$ and perpendicular to vector obtained in step 2

4. Removing voxel with maximal d_i
5. Return to step 2 until we carve all the voxels

6. Return the carve pattern α that corresponded to the lowest energy value

From complexity reasons, we can replace step 5 by return to step 2 until `norm(d)>1e-1` ,means, the size of distance vector bigger than threshold.

Summary & Conclusions:

Make it stand is an example of how 3D modeling can become an interactive process where the computer is taking care of a difficult specific task, in our case optimizing the balance, while the user can focus on the actual visual result.
The method presented in the article ensures stability of printed models so that they may be physically realized.
It combines stability objectives with shape preservation measures in order to respect the user's original design.
We can't print it using the printer in the lab (from technology reasons) and in order to print it, we need to use stratasys printer.
We can continue with implement shape deformation algorithm and to print it using suitable printer (maybe through smartphone application).

Bibliography:
1. Make It Stand: Balancing Shapes for 3D Fabrication
   Romain Prévost, Emily Whiting, Sylvain Lefebvre, Olga Sorkine-Hornung