Vector Field Based Shape Deformations Note

ShengweiZHANG

May 27, 2015

1 Abstract

The paper construct a C^1 continuous, divergence-free vector field to deform a 3d model.

2 The Vector Field

In 3D, a divergence-free vector field v can be constructed from the gradients of two scalar fields $p(x, y, z), q(x, y, z)$ as

$$v(x, y, z) = \nabla p \times \nabla q \tag{1}$$

which in other words, v is constructed by the cross product of two divergence-free vector field ∇p and ∇q. In this paper we seprate the model into three parts: *inner region*, *outer region* and *blended region* and using a piecewise linear vector field to deform the 3d model. The points in the *inner region* are fully deformed, the points in the *outer region* are no deform, and the points in the *blended region* are blended deformed. The scalar field p,q are defined as:

$$p(x,t) = \begin{cases} e(x,t) & r(x) < r_i \\ (1-b)\cdot e(x,t) + b \cdot 0 & r_i \leq r(x) < r_o \\ 0 & r_o \leq r(x) \end{cases} \tag{2}$$

$$q(x,t) = \begin{cases} f(x,t) & r(x) < r_i \\ (1-b)\cdot f(x,t) + b \cdot 0 & r_i \leq r(x) < r_o \\ 0 & r_o \leq r(x) \end{cases} \tag{3}$$

by a scalar field $r(x)$ we separate the three parts, and v is blended by the blending function $b = b(r(x))$ in the *blended region*. b is defined by Bezier representation[1]:

$$b(r) = \sum_{i=0}^{4} w_i B_i^4 \left(\frac{r - r_i}{r_o - r_i}\right) \tag{4}$$

[1]Bezier representation $B_n(f) = \sum\limits_{k=0}^{n} \begin{pmatrix} n \\ k \end{pmatrix} x^k (1-x)^{n-k} f(\frac{k}{n})$. $B_n(f)(x) \to f(x)$ as $n \to \infty$.

here let $w_0 = w_1 = w_2 = 0$ and $w_3 = w_4 = 1$[2]. Then the vector field should be:

$$\nabla p(x,t) = \begin{cases} \nabla e(x,t) & r(x) < r_i \\ [-\nabla b \cdot e(x,t) + (1-b)\nabla e(x,t)] \times [-\nabla b \cdot f(x,t) + (1-b)\nabla f(x,t)] & r_i \le r(x) < r_o \\ 0 & r_o \le r(x) \end{cases} \tag{5}$$

$$\nabla p(x,t) = \begin{cases} \nabla e(x,t) & r(x) < r_i \\ [-\nabla b \cdot e(x,t) + (1-b)\nabla e(x,t)] \times [-\nabla b \cdot f(x,t) + (1-b)\nabla f(x,t)] & r_i \le r(x) < r_o \\ 0 & r_o \le r(x) \end{cases} \tag{6}$$

3 Integration

$$x_{k+1} = \int_{t_k}^{t_{k+1}} v(x_t, t)dt + x_k \tag{7}$$

Algorithm 1 Integrator algorithm

```
procedure INIT PROCEDURE
    h ← T/time_slice
    i ← 0
    cur_t ← init_time
for i <time_slice do
    construct vector field v
    x = funcAdv(v(x, cur_t)) + x
    cur_t+ = h
```

4 Implicit tools

4.1 Sphere Deform Tool

Construst a constant vector field in the inner region $\mathbf{v} = (u, v, w)$, where $\mathbf{v}$ is the translation of the sphere's center, for that target choose two orthogonal vectors: $\mathbf{u}$ and $\mathbf{v}$, which $\mathbf{u} \times \mathbf{v} = \mathbf{v}$. Then e(x) and f(x) is:

$$e(x) = \mathbf{u}(\mathbf{x} - \mathbf{c}), f(x) = \mathbf{w}(\mathbf{x} - \mathbf{c}) \tag{8}$$

where $\mathbf{c}$ is the current center. The region function is defined as

$$r(\mathbf{x}) = (\mathbf{x} - \mathbf{c})^2 \tag{9}$$

[2] see Bezier Representation.

4.2 Bending Deform Tool

A linear vector field v is used to describe a rotation inside the inner region. Given a rotational axis by a center point **c** and the normalized axis direction **a**, Then the e(x) and f(x) is:

$$e(x) = \mathbf{a}(\mathbf{x} - \mathbf{c}), f(x) = (\mathbf{a} \times (\mathbf{x} - \mathbf{c}))^2 \tag{10}$$

from quation up, we can derive that the angular velocity in the inner region is 2, take $Line(\mathbf{a}, \mathbf{c})$ as axis.[3] The region function is defined as

$$r(\mathbf{x}) = \mathbf{b} \cdot (\mathbf{x} - \mathbf{c}) \tag{11}$$

Innfer from the paper, **b** rotate the half of the inner regions rotation. So in each time_step update **b** and recaculate r(x) and update vector field.

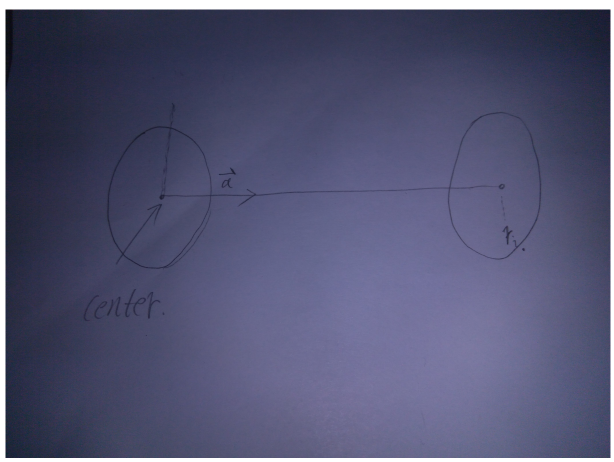
$$\mathbf{b} = rotate\_mat * \mathbf{b} \tag{12}$$

4.3 Twist Deform Tool

By using a linearly increasing rotation defined by:

$$e(x) = (a \cdot (x - c))^2, f(x) = (a \times (x - c))^2 \tag{13}$$

we construct a twist vector field in the inner region.[4]



The region function is defind as:

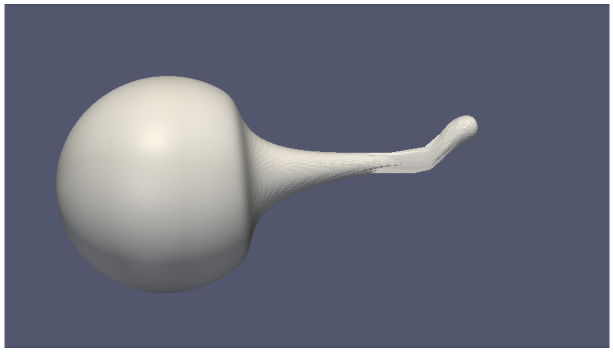
$$r(x) = a \cdot (x - c) \tag{14}$$

[3]see Appendix.

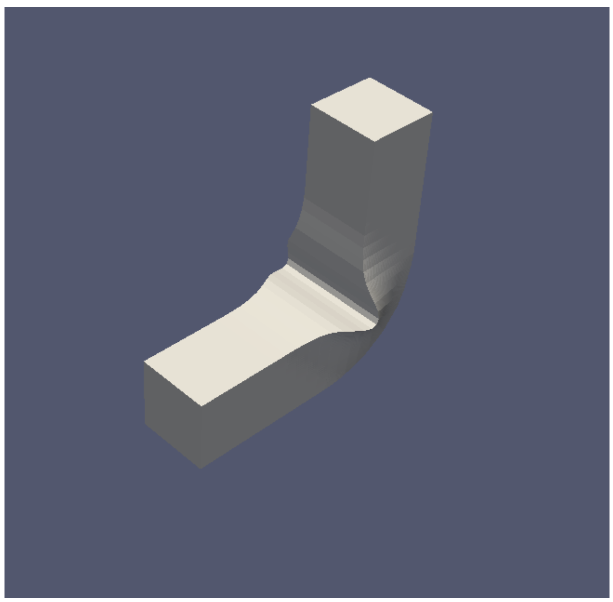
[4]we can construct a more understanble scalar field e(x), see Appendix

5 Result

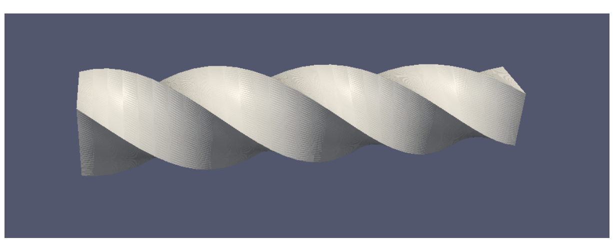
Sphere Deform Tool:



Bend Deform Tool:

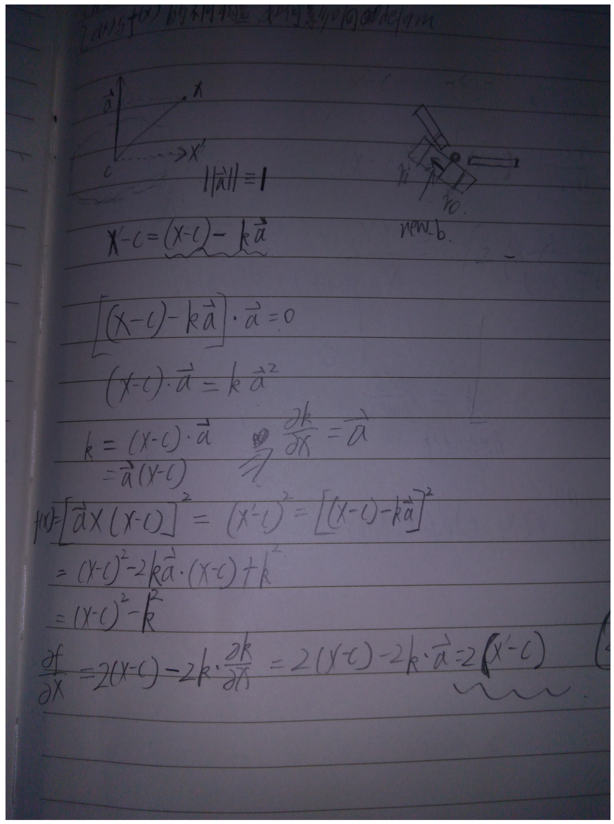


Twist Deform Tool:



6 Appendix

Caculate the Bend Deform Tool's angular velocity:



Construct a scalar field let

$$\nabla e = (\mathbf{a} \cdot (x - c))\mathbf{a} \tag{15}$$

then e(x) define as:

$$\begin{aligned} e(x) = & \frac{a_0}{2}(x_0 - c_0)^2 + \frac{a_1}{2}(x_1 - c_1)^2 + \frac{a_2}{2}(x_2 - c_2)^2 \\ & + a_0 a_1 (x_1 - c_1) x_0 + a_0 a_2 (x_2 - c_2) x_0 + a_1 a_2 (x_2 - c_2) x_1 \\ & - a_1 a_0 c_0 x_1 - a_2 a_0 c_0 x_2 - a_2 a_1 c_1 x_2 \end{aligned} \tag{16}$$