DISCUSSION LOG

Sikang Yan

University of Kaiserslautern

yan@rhrk.uni-kl.de

February 5, 2019

KW13

In our cloth simulation, we follow the precedure written by Rohmer et al.. We consider to begin with the *Defomation gradient* $\mathbf{F}$, which is defined by

$$\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}}, \tag{1}$$

where the $\mathbf{x}$ denotes the deformed vector and the $\mathbf{X}$ denotes the reference vector.

Since we have no further information about the mapping from $\mathbf{X}$ to $\mathbf{x}$, we use the vector $(\mathbf{u_1}, \mathbf{u_2})$ and $(\overline{\mathbf{u_1}}, \overline{\mathbf{u_2}})$ to approximate the Defomation gradient, which is defined as

$$\mathbf{F} = \begin{bmatrix} \mathbf{u_1}, \mathbf{u_2} \end{bmatrix} \begin{bmatrix} \overline{\mathbf{u_1}}, \overline{\mathbf{u_2}} \end{bmatrix}^{-1}, \tag{2}$$

Attention should be paid especially:

- eq.(2) characrize only the 2D deformation of each triangle.
- in Rohmer et al. $\mathbf{F}$ is symbolised as $\mathbf{T}$.

We provide here our code preceed with concrete data:

- $faces(i, j)$ is the jth vertex of the ith triangle, here we choose the $face(0, 0), face(0, 1), face(0, 2)$ as examples and calculate the $VecT$ and $VecR$ for $face(0, 0)$.

cloth_vec

$$VecT = el_1 VertT_1 - el_1 VertT_2; el_1 VertT_1 - el_1 VertT_3 \quad (3)$$

$$VecR = el_1 VertR_1 - el_1 VertR_2; el_1 VertR_1 - el_1 VertR_3 \quad (4)$$

cloth_vec

$$VecT = [0.771842 - 0.0144887 \ 6.39045] - [0.780121 - 0.0186188 \ 6.37318]$$
$$(5)$$

$$[0.771842 - 0.0144887 \ 6.39045] - [0.737177 - 0.00912791 \ 6.39292$$
$$(6)$$

$$VecR = [0.759919 - 0.015194 \ 6.38401] - [0.767822 - 0.0212492 \ 6.3669];$$
$$(7)$$

$$[0.7599190.015194 \ 6.38401] - [0.726977 - 0.00749985 \ 6.38692]$$
$$(8)$$

$$(9)$$

such that

cloth_vec

$$VecT = [-0.0079 \ 0.0061 \ 0.0171 \ 0.0329 \ -0.0077 \ -0.0029]; \qquad (10)$$
$$VecR = [-0.0083 \ 0.0041 \ 0.0173 \ 0.0347 \ -0.0054 \ -0.0025]; \qquad (11)$$
$$\qquad (12)$$

where the first 3 entries of $VecR$ is the vector $\mathbf{u_1}$ and the last 3 entries of $VecR$ is the vector $\mathbf{u_2}$. $VecT$ analogiously.

cloth_eig_2D

we use here *Eigen* :: *Map* to transform the vector *VecT* and *VecR* to $2 * 2$ 2D deformation gradient **F**.

$$\mathbf{F} = \begin{bmatrix} -0.0083 & 0.0347 \\ 0.0041 & -0.0054 \end{bmatrix} \begin{bmatrix} -0.0079 & 0.0329 \\ 0.0061 & -0.0077 \end{bmatrix}^{-1} \tag{13}$$

hence the **F** has a rotation information **R** and a stretch information **U**,

$$\mathbf{F} = \mathbf{RU}. \tag{14}$$

we use $\mathbf{F}^T\mathbf{F}$ to eliminate the ratotion information to obtain $\det \mathbf{R} = 1$

$$\mathbf{F}^T\mathbf{F} = \left(\mathbf{RU}\right)^T\mathbf{RU} \tag{15}$$

$$= \mathbf{U}^T\mathbf{R}^T\mathbf{RU} \tag{16}$$

$$= \mathbf{U}^2 \tag{17}$$

$$= \mathbf{C} \tag{18}$$

and using decomposition, if we have the form

$$\mathbf{C} = \lambda_1^2\mathbf{v}_1\mathbf{v}_1^T + \lambda_2^2\mathbf{v}_2\mathbf{v}_2^T \tag{19}$$

then we could obtain the **U** by applying

$$\mathbf{U} = \lambda_1\mathbf{v}_1\mathbf{v}_1^T + \lambda_2\mathbf{v}_2\mathbf{v}_2^T \tag{20}$$

the main backdraw of Rohmer et al. is that this is only applied for 2D problem, thus we need a new algorithm, which can also take the consideration for the vertical deformation. Therefore, the *Kabisch Algorithm* is introduced.

// to do

use $< boost >$ to collect all ply files name, which are stored in _filename
as well as _output in $std :: vector$. Later I could only pass the index i and
$i + 1$ for the inputs, and the eigenvector can be calculated automatically
and saved.
Problem might be the allocation memory (?) since I don't delete any
temporary datas while the calculation. Let's see.
need the binary files.