Karlsruher Institut für Technologie

Deformation Transfer for Triangle Meshes

Final project presentation - Praktikum Diskrete Freiformflächen

Xinyi Zhang, Oliver Suchan | 27. June 2022

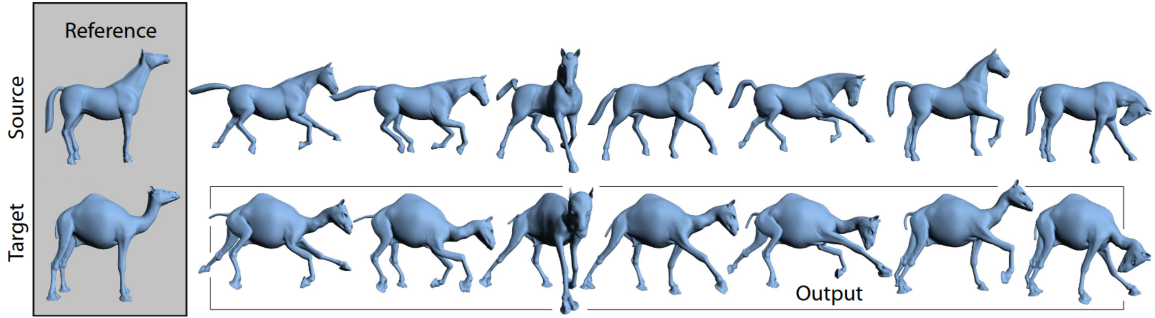
Table of Contents

Motivation

Deformation Transfer for Triangle Meshes

Robert W. Sumner Jovan Popović

Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology



Formulation

- represent source deformation as collection of affine transformations $\mathbf{Q}_j + \mathbf{d}_j$ per triangle $\Delta = (\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$

Formulation

- represent source deformation as collection of affine transformations $\mathbf{Q}_j + \mathbf{d}_j$ per triangle $\Delta = (\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$

$$\mathbf{Q}_j \mathbf{v}_i + \mathbf{d} = \tilde{\mathbf{v}}_i, \quad i = 1, \ldots, 4,$$

Formulation

- represent source deformation as collection of affine transformations $\mathbf{Q}_j + \mathbf{d}_j$ per triangle $\Delta = (\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$

$$\mathbf{Q}_j \mathbf{v}_i + \mathbf{d} = \tilde{\mathbf{v}}_i, \quad i = 1, \ldots, 4,$$

with

$$\mathbf{v}_4 = (\mathbf{v}_2 - \mathbf{v}_1) \times (\mathbf{v}_3 - \mathbf{v}_1) / \sqrt{|(\mathbf{v}_2 - \mathbf{v}_1) \times (\mathbf{v}_3 - \mathbf{v}_1)|}$$

Formulation

- represent source deformation as collection of affine transformations $\mathbf{Q}_j + \mathbf{d}_j$ per triangle $\Delta = (\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$

$$\mathbf{Q}_j \mathbf{v}_i + \mathbf{d} = \tilde{\mathbf{v}}_i, \quad i = 1, \ldots, 4,$$

with

$$\mathbf{v}_4 = (\mathbf{v}_2 - \mathbf{v}_1) \times (\mathbf{v}_3 - \mathbf{v}_1) / \sqrt{|(\mathbf{v}_2 - \mathbf{v}_1) \times (\mathbf{v}_3 - \mathbf{v}_1)|}$$

- closed form expression for $\mathbf{Q}_j$:

$$\mathbf{Q}_j = \tilde{\mathbf{V}} \mathbf{V}^{-1}$$
$$\mathbf{V} = [\mathbf{v}_2 - \mathbf{v}_1 \quad \mathbf{v}_3 - \mathbf{v}_1 \quad \mathbf{v}_4]$$
$$\tilde{\mathbf{V}} = [\tilde{\mathbf{v}}_2 - \tilde{\mathbf{v}}_1 \quad \tilde{\mathbf{v}}_3 - \tilde{\mathbf{v}}_1 \quad \tilde{\mathbf{v}}_4]$$

Formulation

- set of *correspondences*

$$M = \{(s_1, t_1), (s_2, t_2), \ldots, (s_{|M|}, t_{|M|})\}$$

Formulation

- set of *correspondences*

$$M = \{(s_1, t_1), (s_2, t_2), \ldots, (s_{|M|}, t_{|M|})\}$$

target triangle t_i should deform like source triangle s_i.

Formulation

- set of *correspondences*

$$M = \{(s_1, t_1), (s_2, t_2), \ldots, (s_{|M|}, t_{|M|})\}$$

  target triangle t_i should deform like source triangle s_i.
- yields minimization problem over the vertices

$$\min_{\tilde{\mathbf{v}}_1, \ldots, \tilde{\mathbf{v}}_n} \sum_{j=1}^{|M|} \left\| \mathbf{S}_{s_j} - \mathbf{T}_{t_j} \right\|_F^2 \tag{1}$$

Formulation

- set of *correspondences*

$$M = \{(s_1, t_1), (s_2, t_2), \ldots, (s_{|M|}, t_{|M|})\}$$

  target triangle t_i should deform like source triangle s_i.
- yields minimization problem over the vertices

$$\min_{\tilde{\mathbf{v}}_1,\ldots,\tilde{\mathbf{v}}_n} \sum_{j=1}^{|M|} \left\|\mathbf{S}_{s_j} - \mathbf{T}_{t_j}\right\|_F^2 \tag{1}$$

  or equivalently

$$\min_{\tilde{\mathbf{v}}_1,\ldots,\tilde{\mathbf{v}}_n} \left\|\mathbf{c} - \mathbf{A}\tilde{\mathbf{x}}\right\|_2^2 \tag{2}$$

  with $\mathbf{c}$ representing source transformations, $\tilde{\mathbf{x}}$ vector of unknown deformed vertex locations.

How to construct A?

Karlsruher Institut für Technologie

$$\min_{\tilde{\mathbf{v}}_1,\ldots,\tilde{\mathbf{v}}_n} \|\mathbf{c} - \mathbf{A}\tilde{\mathbf{x}}\|_2^2 \tag{2}$$

- it holds $\mathbf{T} = \tilde{\mathbf{V}}\mathbf{V}^{-1}$ where $\tilde{\mathbf{V}} = [\tilde{\mathbf{v}}_2 - \tilde{\mathbf{v}}_1 \quad \tilde{\mathbf{v}}_3 - \tilde{\mathbf{v}}_1 \quad \tilde{\mathbf{v}}_4]$.

How to construct A?

$$\min_{\tilde{\mathbf{v}}_1,\ldots,\tilde{\mathbf{v}}_n} \|\mathbf{c} - \mathbf{A}\tilde{\mathbf{x}}\|_2^2 \tag{2}$$

- it holds $\mathbf{T} = \tilde{\mathbf{V}}\mathbf{V}^{-1}$ where $\tilde{\mathbf{V}} = [\tilde{\mathbf{v}}_2 - \tilde{\mathbf{v}}_1 \quad \tilde{\mathbf{v}}_3 - \tilde{\mathbf{v}}_1 \quad \tilde{\mathbf{v}}_4]$.
- Want to write $\mathbf{T} = \mathbf{M}\tilde{\mathbf{v}}$ as product of matrix $\mathbf{M}$ and unknown positions $\tilde{\mathbf{v}}$.

How to construct A?

Karlsruher Institut für Technologie

$$\min_{\tilde{\mathbf{v}}_1,\dots,\tilde{\mathbf{v}}_n} \|\mathbf{c} - \mathbf{A}\tilde{\mathbf{x}}\|_2^2 \tag{2}$$

- it holds $\mathbf{T} = \tilde{\mathbf{V}}\mathbf{V}^{-1}$ where $\tilde{\mathbf{V}} = [\tilde{\mathbf{v}}_2 - \tilde{\mathbf{v}}_1 \quad \tilde{\mathbf{v}}_3 - \tilde{\mathbf{v}}_1 \quad \tilde{\mathbf{v}}_4]$.
- Want to write $\mathbf{T} = \mathbf{M}\tilde{\mathbf{v}}$ as product of matrix $\mathbf{M}$ and unknown positions $\tilde{\mathbf{v}}$.
- Set

$$\mathbf{B} := \begin{pmatrix} -(v_{11} + v_{21} + v_{31}) & v_{11} & v_{21} & v_{31} \\ -(v_{12} + v_{22} + v_{32}) & v_{12} & v_{22} & v_{32} \\ -(v_{13} + v_{23} + v_{33}) & v_{13} & v_{23} & v_{33} \end{pmatrix}$$

and thus

$$\mathbf{T} \simeq \begin{pmatrix} \mathbf{B} & 0 & 0 \\ 0 & \mathbf{B} & 0 \\ 0 & 0 & \mathbf{B} \end{pmatrix} \tilde{\mathbf{v}}. \tag{3}$$

How to construct A?

Karlsruher Institut für Technologie

- it holds $\mathbf{A} \in \mathbb{R}^{9|M| \times (3n+3m)}$, $\mathbf{c} \in \mathbb{R}^{9|M|}$,
  $|M|$ amount of correspondences, m amount of triangles, n amount of vertices.

How to construct A?

- it holds $\mathbf{A} \in \mathbb{R}^{9|M| \times (3n+3m)}$, $\mathbf{c} \in \mathbb{R}^{9|M|}$,
  $|M|$ amount of correspondences, m amount of triangles, n amount of vertices.
- therefore, 9 rows of **A** and **c** correspond to 1 triangle correspondence
- and the columns of **A** correspond to the coordinates of the vertices and sqrt normalized triangle normals.

How to calculate M?

Goal

Calculate the correspondences

$$M = \{(s_1, t_1), (s_2, t_2), \ldots, (s_{|M|}, t_{|M|})\}. \tag{4}$$

Deform the source mesh *into* the target mesh.

How to calculate M?

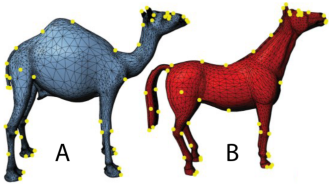
Goal

Calculate the correspondences

$$M = \{(s_1, t_1), (s_2, t_2), \ldots, (s_{|M|}, t_{|M|})\}. \quad (4)$$

Deform the source mesh *into* the target mesh.

- Additional input: small set of user selected marker points



How to calculate *M*?

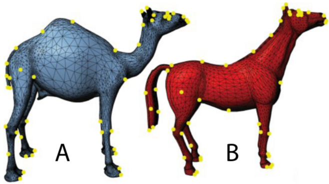
Goal

Calculate the correspondences

$$M = \{(s_1, t_1), (s_2, t_2), \ldots, (s_{|M|}, t_{|M|})\}. \quad (4)$$

Deform the source mesh *into* the target mesh.

- Additional input: small set of user selected marker points



- marker force source vertices to match exactly target vertices.

Correspondence problem

Karlsruher Institut für Technologie

Correspondence problem

$$\min_{\tilde{\mathbf{v}}_1,\ldots\tilde{\mathbf{v}}_n} E(\mathbf{v}_1,\ldots,\mathbf{v}_n,\mathbf{c}_1,\ldots,\mathbf{c}_n) = w_S E_S + w_I E_I + w_C E_C \tag{5}$$

subject to marker constraints

Correspondence problem

Correspondence problem

$$\min_{\tilde{\mathbf{v}}_1,\ldots\tilde{\mathbf{v}}_n} E(\mathbf{v}_1,\ldots,\mathbf{v}_n,\mathbf{c}_1,\ldots,\mathbf{c}_n) = w_S E_S + w_I E_I + w_C E_C \quad \text{subject to marker constraints} \tag{5}$$

Deformation Smoothness E_S

transformations for adjacent triangles should be equal

$$E_S(\mathbf{v}_1,\ldots,\mathbf{v}_n) = \sum_{i=1}^{m} \sum_{j\in \mathrm{adj}(i)} \|\mathbf{T}_i - \mathbf{T}_j\|_F^2.$$

Correspondence problem

Correspondence problem

$$\min_{\tilde{\mathbf{v}}_1,\ldots\tilde{\mathbf{v}}_n} E(\mathbf{v}_1,\ldots,\mathbf{v}_n,\mathbf{c}_1,\ldots,\mathbf{c}_n) = w_S E_S + w_I E_I + w_C E_C \quad \text{subject to marker constraints} \tag{5}$$

Deformation Identity E_I

transformations should be equal to identity

$$E_I(\mathbf{v}_1,\ldots,\mathbf{v}_n) = \sum_{i=1}^{m} \|\mathbf{T}_i - \mathbf{I}\|_F^2.$$

Correspondence problem

Correspondence problem

$$\min_{\tilde{\mathbf{v}}_1,\ldots\tilde{\mathbf{v}}_n} E(\mathbf{v}_1,\ldots,\mathbf{v}_n,\mathbf{c}_1,\ldots,\mathbf{c}_n) = w_S E_S + w_I E_I + w_C E_C \tag{5}$$

subject to marker constraints

Closest valid point term E_C

position of each vertex of source mesh equal to closest valid point on target mesh

$$E_C(\mathbf{v}_1,\ldots,\mathbf{v}_n,\mathbf{c}_1,\ldots,\mathbf{c}_n) = \sum_{i=1}^{n} \left\|\mathbf{v}_i - \mathbf{c}_i\right\|^2$$

Correspondence problem

Correspondence problem

$$\min_{\tilde{\mathbf{v}}_1,\ldots\tilde{\mathbf{v}}_n} E(\mathbf{v}_1,\ldots,\mathbf{v}_n,\mathbf{c}_1,\ldots,\mathbf{c}_n) = w_S E_S + w_I E_I + w_C E_C \quad (5)$$

subject to marker constraints

- solve this problem in two phases
  1. Ignore closest point term: $w_S = 1.0$, $w_I = 0.001$, $w_C = 0$.

Correspondence problem

Correspondence problem

$$\min_{\tilde{\mathbf{v}}_1,\ldots\tilde{\mathbf{v}}_n} E(\mathbf{v}_1,\ldots,\mathbf{v}_n,\mathbf{c}_1,\ldots,\mathbf{c}_n) = w_S E_S + w_I E_I + w_C E_C \quad (5)$$

subject to marker constraints

- solve this problem in two phases
  1. Ignore closest point term: $w_S = 1.0$, $w_I = 0.001$, $w_C = 0$.
  2. Increase w_C from 1.0 to 5000.0 in 4 steps and update closest points after each iteration.

Correspondence problem

Correspondence problem

$$\min_{\tilde{\mathbf{v}}_1,\ldots\tilde{\mathbf{v}}_n} E(\mathbf{v}_1,\ldots,\mathbf{v}_n,\mathbf{c}_1,\ldots,\mathbf{c}_n) = w_S E_S + w_I E_I + w_C E_C \quad (5)$$

subject to marker constraints

- solve this problem in two phases
  1. Ignore closest point term: $w_S = 1.0$, $w_I = 0.001$, $w_C = 0$.
  2. Increase w_C from 1.0 to 5000.0 in 4 steps and update closest points after each iteration.
- after source mesh is transformed into target mesh

Correspondence problem

Correspondence problem

$$\min_{\tilde{\mathbf{v}}_1,\ldots\tilde{\mathbf{v}}_n} E(\mathbf{v}_1,\ldots,\mathbf{v}_n,\mathbf{c}_1,\ldots,\mathbf{c}_n) = w_S E_S + w_I E_I + w_C E_C \quad \text{subject to marker constraints} \tag{5}$$

- solve this problem in two phases
  1. Ignore closest point term: $w_S = 1.0$, $w_I = 0.001$, $w_C = 0$.
  2. Increase w_C from 1.0 to 5000.0 in 4 steps and update closest points after each iteration.
- after source mesh is transformed into target mesh
  - compute *compatible* triangles:

Correspondence problem

Correspondence problem

$$\min_{\tilde{\mathbf{v}}_1,\ldots\tilde{\mathbf{v}}_n} E(\mathbf{v}_1,\ldots,\mathbf{v}_n,\mathbf{c}_1,\ldots,\mathbf{c}_n) = w_S E_S + w_I E_I + w_C E_C \quad \text{subject to marker constraints} \tag{5}$$

- solve this problem in two phases
  1. Ignore closest point term: $w_S = 1.0$, $w_I = 0.001$, $w_C = 0$.
  2. Increase w_C from 1.0 to 5000.0 in 4 steps and update closest points after each iteration.
- after source mesh is transformed into target mesh
  - compute *compatible* triangles:
  - centroids are within certain threshold,

Correspondence problem

Correspondence problem

$$\min_{\tilde{\mathbf{v}}_1,\ldots\tilde{\mathbf{v}}_n} E(\mathbf{v}_1,\ldots,\mathbf{v}_n,\mathbf{c}_1,\ldots,\mathbf{c}_n) = w_S E_S + w_I E_I + w_C E_C \quad (5)$$

subject to marker constraints

- solve this problem in two phases
  1. Ignore closest point term: $w_S = 1.0$, $w_I = 0.001$, $w_C = 0$.
  2. Increase w_C from 1.0 to 5000.0 in 4 steps and update closest points after each iteration.
- after source mesh is transformed into target mesh
  - compute *compatible* triangles:
  - centroids are within certain threshold,
  - angle between normals less than 90°.

Test setup

Results were produced on a Lenovo Thinkpad X1

- Memory: 15 GiB
- Processor: Intel Core i7-8550U CPU @ 1.8 GHz x 8
- Graphics: Mesa Intel UHD Graphics 620

Test setup

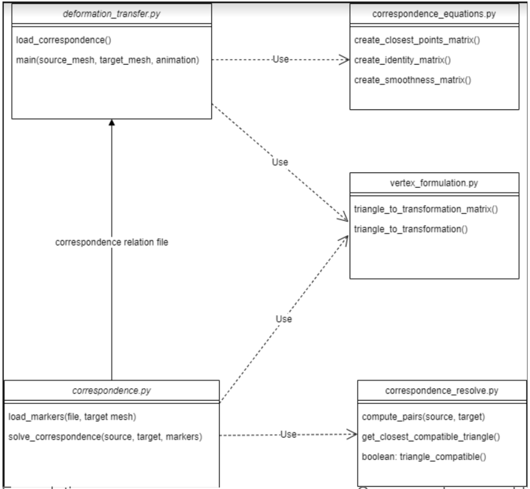
Results were produced on a Lenovo Thinkpad X1

- Memory: 15 GiB
- Processor: Intel Core i7-8550U CPU @ 1.8 GHz x 8
- Graphics: Mesa Intel UHD Graphics 620

Implemented in Python 3.8 using

- numpy (calculation of matrices)
- scipy (solving of the least squares problem)
- pymesh (work on meshes)

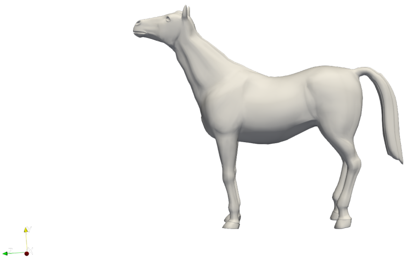
Implementation



Running times

- Input:

mesh	num vertices	num triangles
horse	8431	16843
camel	21887	43814

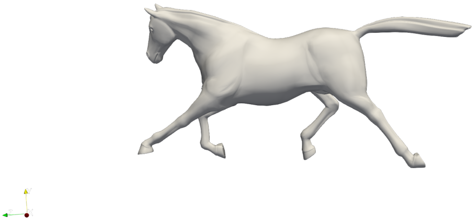


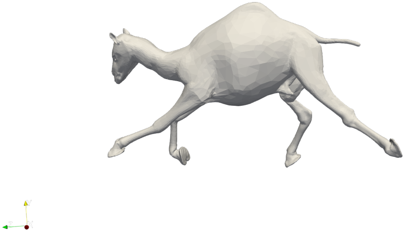
Running times

- Marker Points: 107.
- Solving Correspondence Problem: 741 s.

Running times

- Solving Deformation Problem: 20924 s.





Literatur

Backup-Teil

Folien, die nach `\beginbackup` eingefügt werden, zählen nicht in die Gesamtzahl der Folien.