An image processing approach to surface matching

Felix Schmid

Recent advances in 3D shape matching

Structure

- Basic Idea and Motivation
- Image Matching
- Surface Matching
- Matching Algorithm
- Results

Basic Idea

- Goal: Match two surfaces
- Reduce the 3D-problem to a 2D-problem
- Solve the problem in 2D

Motivation

- Image matching is already heavily researched
- Computation in 2D is less complicated and faster
- Especially for highly resolved meshes

Surface Matching

Correlate two Surface Patches M_A and M_B through Deformation:

$$\phi_M : M_A \rightarrow R^3$$

In a way that corresponding regions in M_A are mapped to regions in M_B

avoid finding these maps in 3D

-> match parameter domains containing the information

Surface Matching

Measure deformation from one parameter space to the other as energy

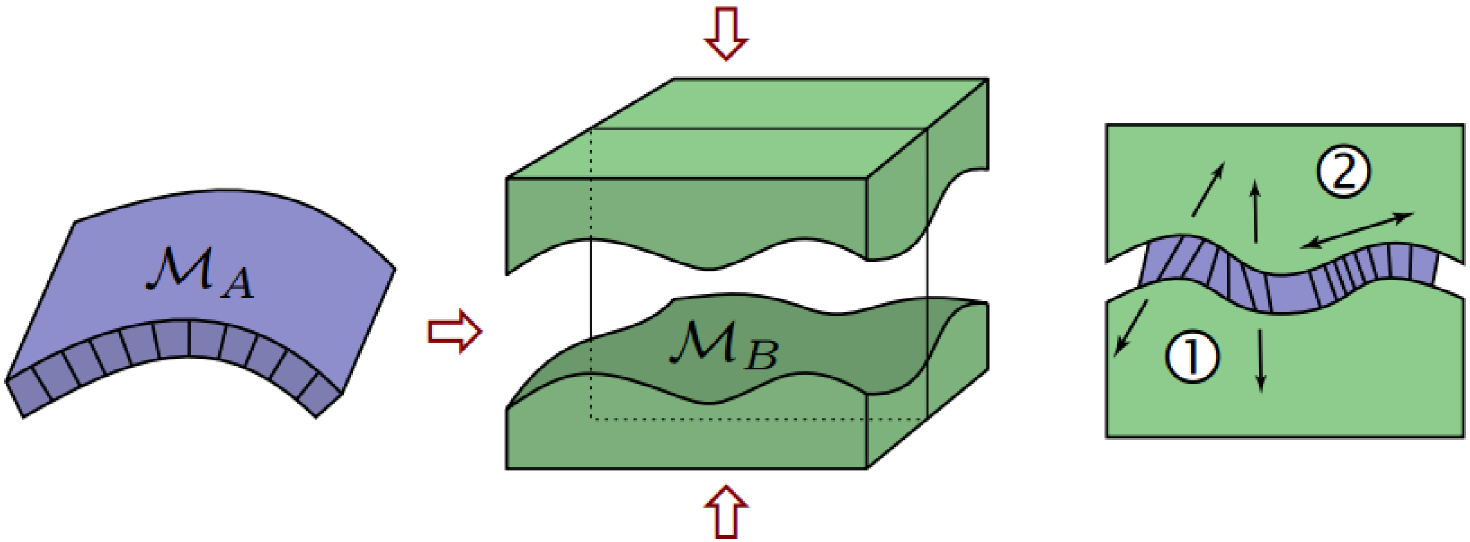
-> minimize the energy to get the optimal match

Three different types of energy

- Regularization energy
- Bending energy
- Feature energy

Surface Matching

Visualization: Pressing a thin shell M_A into a mould of the surface M_B



Bending and stretching represented as energies

Parametrization

Mapping from parameter domain ω to surface $M \in R^3$:

$$x : \omega \to M$$

X has to be a bijection therefore we can define a metric g:

$$g = Dx^T Dx$$

With $Dx \in R^{3,2}$ as the jacobian of x

Describes distortion of length, area and angles in parametrization

Parametrization

Distortion from Surface M to domain ω under x^{-1}

Average change of length of tangent vectors

$$\sqrt{tr(g^{-1})}$$

Average change of area

$$\sqrt{det(g^{-1})}$$

Needed later ensure correct energy measurement

Overwiev

From now on we assume we have the following:

- Parametrizations x_A and x_B of the surfaces M_A and M_B with parameter spaces ω_A and ω_B
- Corresponding metrics g_A and g_B

Now we investigate the deformation $\phi : \omega_A \to \omega_B$ and can define the deformation between the surface patches:

$$\phi_M \coloneqq x_B \circ \phi \circ {x_A}^{-1}$$

Measuring Distortion

Distortion from surface M_A onto M_B

In Elasticy the distortion under deformation ϕ is measured by the Cauchy-Green strain tensor:

$$D\phi^T D\phi$$

We want to adapt this to measure distortion between tangents on the surfaces

-> apply the metrics to get:

$$G[\phi] = {g_A}^{-1} D\phi^T (g_B \circ \phi) D\phi$$

The distortion tensor on tangent vectors in the parameter domain ω

Measuring Distortion

Similar to earlier we can define:

Average change of length of tangent vectors $\sqrt{tr(G[\phi])}$

Average change of area $\sqrt{det(G[\phi])}$

Therefore we can use $tr(G[\phi])$ and $\det(G[\phi])$ as variables for an energy density

$$E_{reg}[\phi] = \int_{\omega_A} W(trG[\phi], detG[\phi]\,)\sqrt{det\,g_A}\,d\xi$$

Energy Density

For the energy density W we use:

$$W(a,d) = a_l a + a_a\left(d + \left(1 + \frac{a_l}{a_a}\right)d^{-1}\right)$$

With weights $a_l, a_a > 0$ chosen depending on the importance of length and area distortion

Measuring Bending

Bending is measured in the change of normals

Variation of normals is represented by the shape operators S_A and S_B

For the bending of normals we can now derive:

$$tr(S_B \circ \phi) - tr(S_A)$$

Trace of the shape operator is the mean curvature so we define $h_A = tr(S_A)$ and $h_B = tr(S_B)$

Measuring Bending

Since mean curvature is easy to compute we base the bending energy on the squared difference of mean curvature:

$$E_{bend}[\phi] = \int_{\omega_A} (h_B \circ \phi - h_A)^2 \sqrt{\det g_A}\, d\xi$$

Feature Matching

Surfaces contain similar geometric or texture features that should be matched

User chosen feature lines create feature sets $F_{M_A} \subset M_A$ and $F_{M_B} \subset M_B$, as well as $F_A \subset \omega_A$ and $F_B \subset \omega_B$ the corresponding points on the domain

We want these points to match, this means:

$$\phi_M\left(F_{M_A}\right) = F_{M_B}$$

Feature Matching

This can be written as a difference:

$$F_{M_A} \,/\, \phi_M{}^{-1}\left(F_{M_B}\right) = \emptyset$$

Which leads to a third energy function:

$$E_F[\phi] = H^1(F_{M_A} \,/\, \phi_M{}^{-1}\left(F_{M_B}\right)) + H^1(F_{M_B} \,/\, \phi_M{}^{-1}\left(F_{M_A}\right))$$

Since this is difficoult to minimize numerically we go with another approximation

Feature Matching

The approximation involves the distance on the surface to the feature sets and leads to the following equation for the third energy:

$$\tilde{E}_F{}^{\epsilon}[\phi]$$

$$= \int_{\omega_A} (\eta^{\epsilon} \circ d_A\,(\xi))\big(\theta^{\epsilon} \circ d_A\big(\phi(\xi)\big)\big)\sqrt{det g_A}\, d\xi$$

$$+ \int_{\omega_B} (\eta^{\epsilon} \circ d_B\,(\xi))(\theta^{\epsilon} \circ d_B(\phi^{-1}\,(\xi)))\sqrt{det g_B}\, d\xi$$

With $\eta^{\epsilon}(s) = \frac{1}{\epsilon}\max(1 - \frac{s}{\epsilon}, 0)$ and $\theta^{\epsilon}(s) = \min(\frac{s^2}{\epsilon}, 1)$

Partial Correspondence

We cannot assume that $\phi_M(M_A) = M_B$

Surfaces don‘t have to be exactly alike

We have to reduce our parameter domain to a subset of the initial domain:

$$\omega_A[\phi] = \phi^{-1}(\phi(\omega_A) \cap \omega_B)$$

And integrate over this domain to compute the energy

Partial Correspondence

These integrals are hard to treat numerically so we use another approximation

We create a superset ω and expand ω_A and ω_B in the areas where they are undefined with a harmonic extension

Additionally we introduce a characterisic function:

$$X^{\epsilon}{}_{A}(\xi) = \max(1 - \epsilon^{-1} dist(\xi, A), 0)$$

Which causes the energy contribution to be ignored after a certain distance away from $\omega_A[\phi]$

Partial Correspondence

We combine both characteristic functions to one:

$$X^{\epsilon}\ (\xi) = {X^{\epsilon}}_{\omega_A}(\xi)\, {X^{\epsilon}}_{\omega_B}(\phi(\xi))$$

And use it as an additional facter in the energy integrand for the bending and feature energy

Final energy function

$$E[\phi] = \beta_{bend} E_{bend}[\phi] + \beta_{reg} E_{reg}[\phi] + \beta_F E_F[\phi]$$

Where β_{bend}, β_{reg} and β_F are weights for the user to define preferences

To find a good matching ϕ we now have to minimize $E[\phi]$

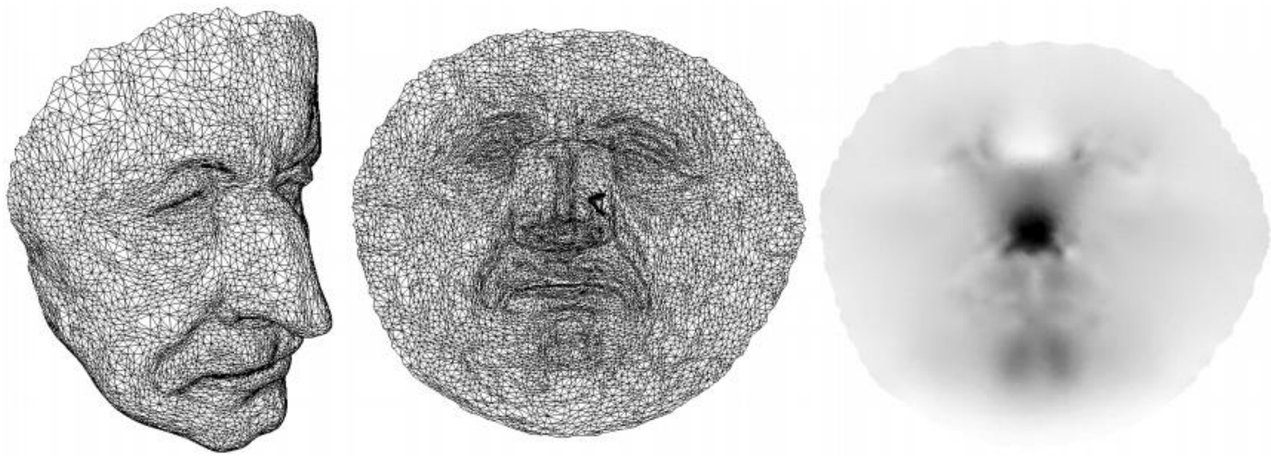
Algorithm Implementation

- Start with two triangle meshes we want to match and go through four steps:

1. Construct parametrization for the surfaces
2. Select features
3. Evaluate metric and mean curvature
4. Apply discretization and optimize the matching by minimizing the energy E

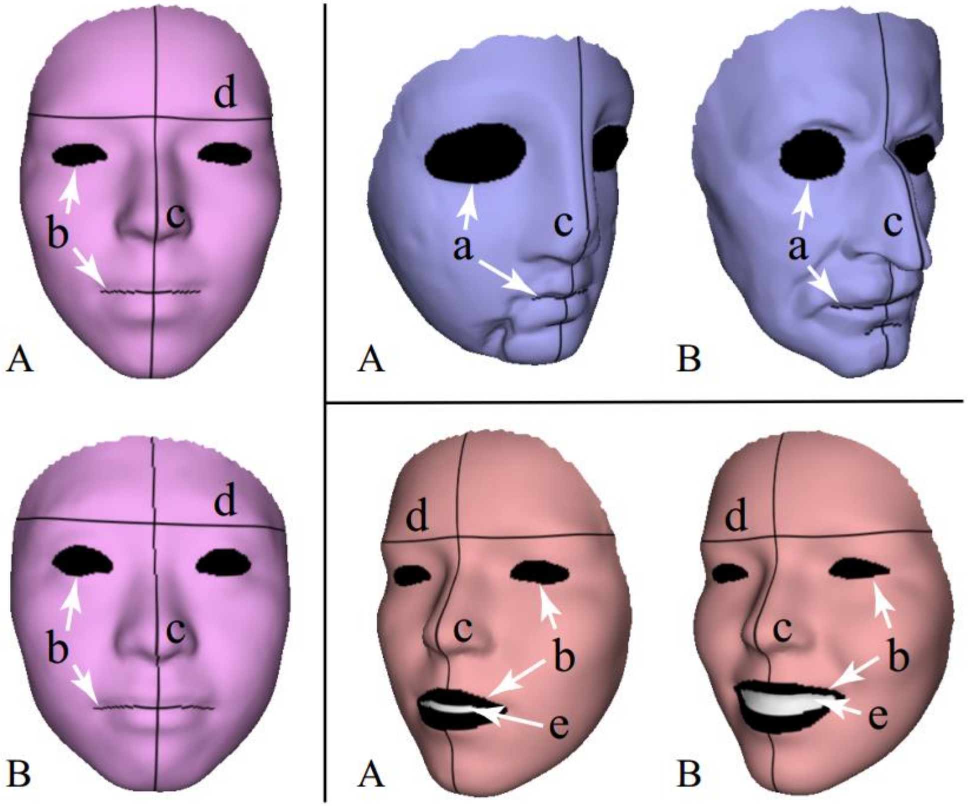
Algorithm Implementation

- Find parametrization optimized for low angle, area and length distortion
- Normalize parameter domains so that they are subsets of $\omega \coloneqq [0,1]^2$



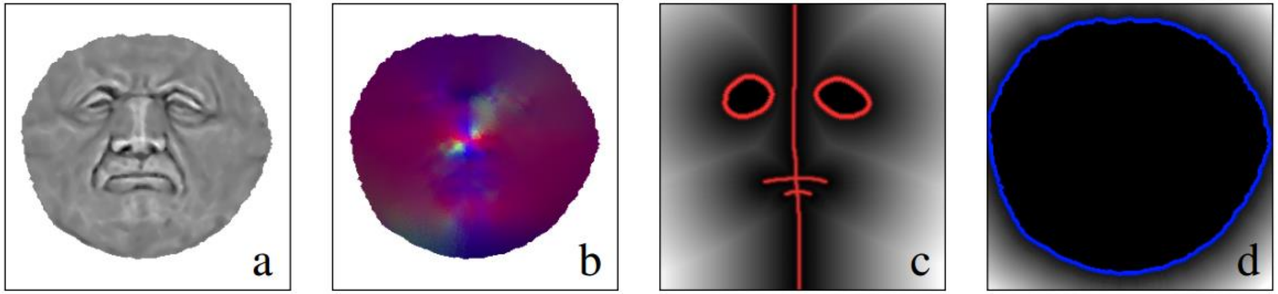
Algorithm Implementation

- Select Feature Lines to control the match



Algorithm Implementation

- Compute surface quantities:
  - Mean curvature
  - Metric tensors
  - Distance to the feature sets
- Since they are all constant they can be discretized into a texture in the beginning



Algorithm Implementation

- Due to the fact that the Energy E is higly non-linear we use a multiscale approach
- Therefore we create a pyramid of images based on a gaussian filter with respect to the surface metric
- Minimize for every level
- Stop when the difference is sufficiently small
- Calculate rest via bi-linear interpolation

Algorithm Implementation

- Descrete deformation $\phi^{k}: \omega \to R^2$ represented as array of vector
- One array at every point in the image grid
- Evaluate energy integrals over every pixel
- Compute gradient for conjugate gradient method

Final formula

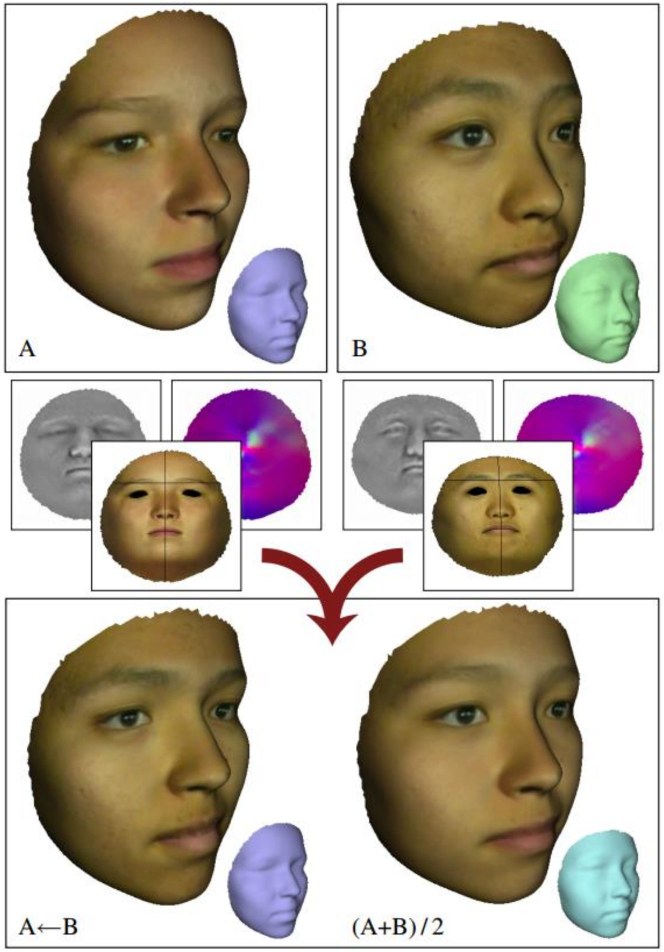
Once we have the discrete deformation ϕ we can now get the discrete surface matching deformation:

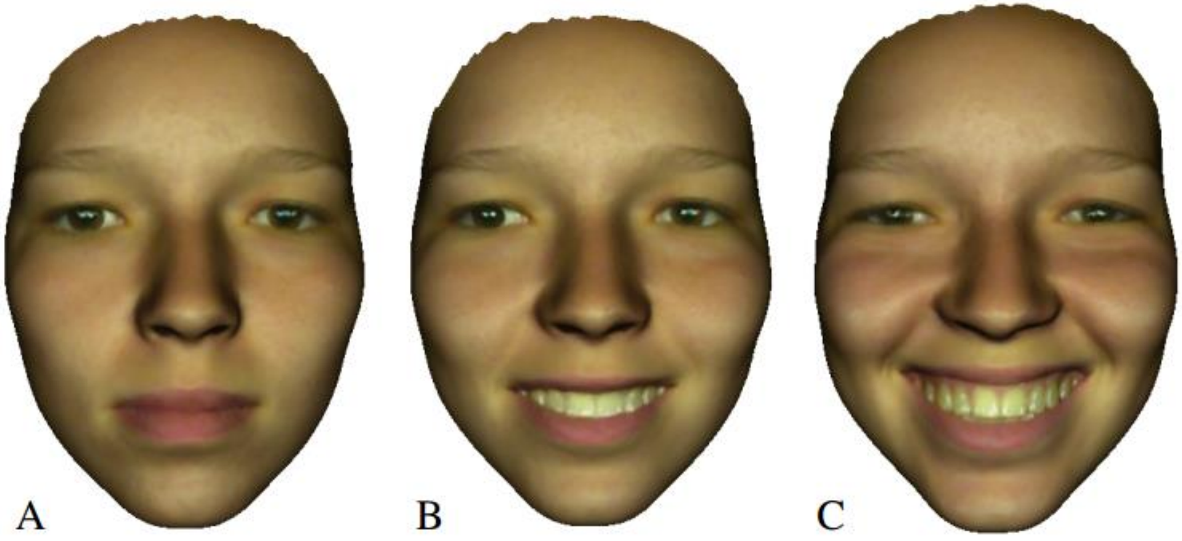
$$\phi_M(x) = (x_B \circ \phi \circ {x_A}^{-1})(x)$$

Which is defined on $x_A(\omega_A[\phi])$

This can now be applied directly to the surface

Examples





Examples