Image Deformation Using Moving Least Squares

Scott Schaefer, Travis McPhail,
Joe Warren

SIGGRAPH 2006

Presented by Teddy Zhang

Credit to Nirup Reddy

Image Deformation

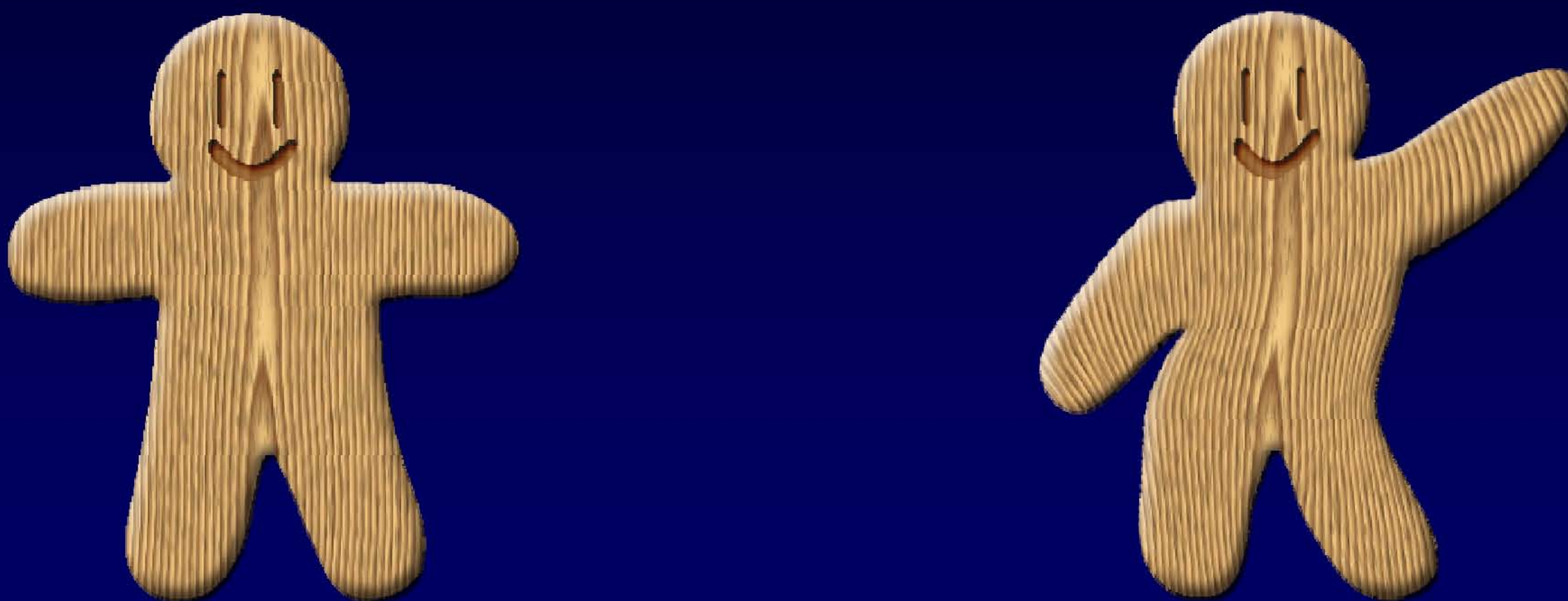
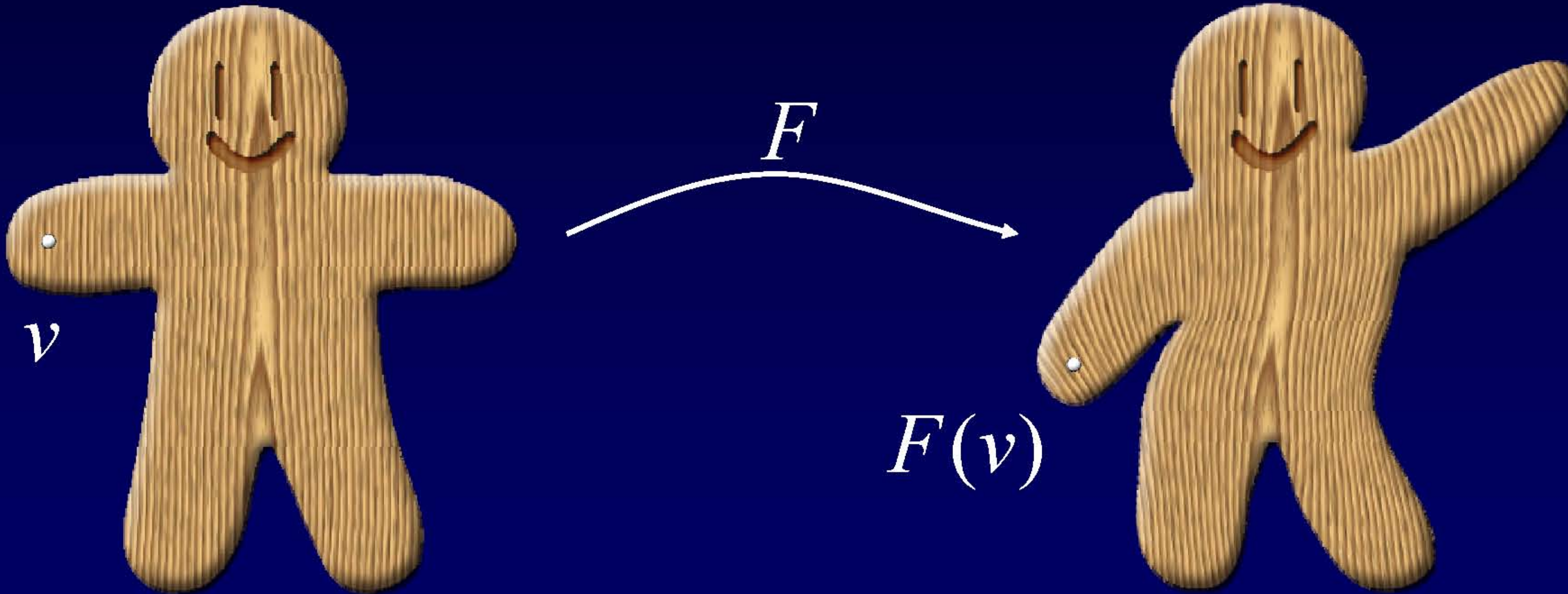
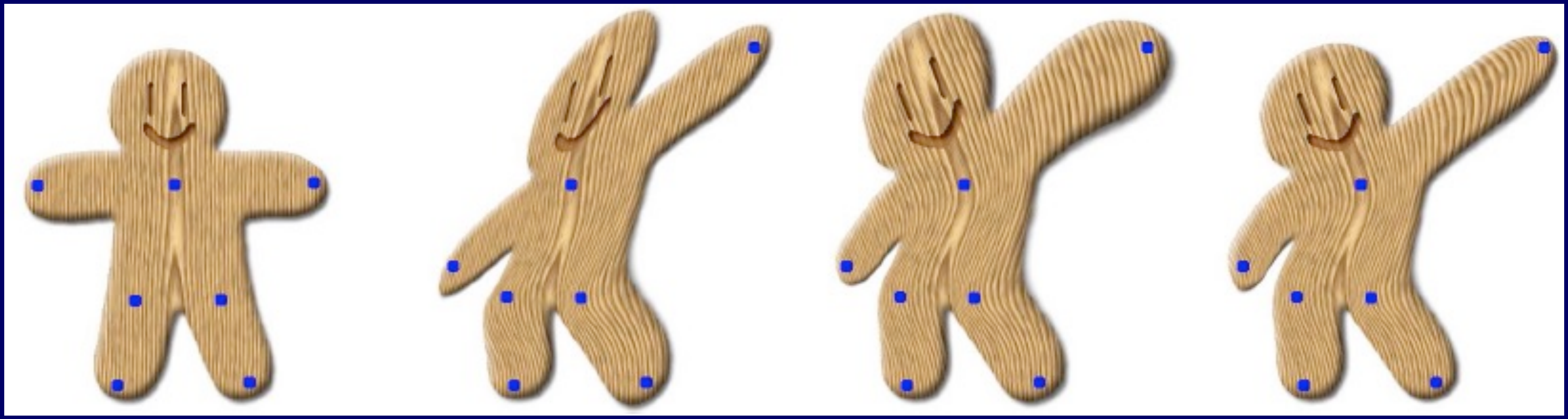


Image Deformation





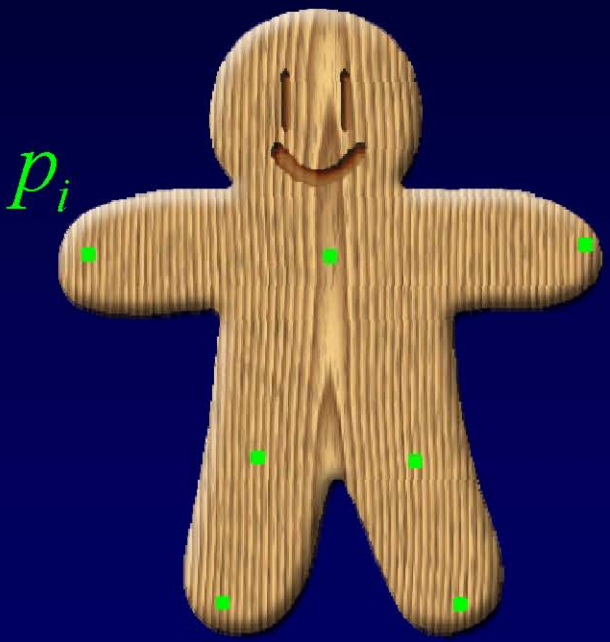
Input Affine Similarity Rigid

Introduction

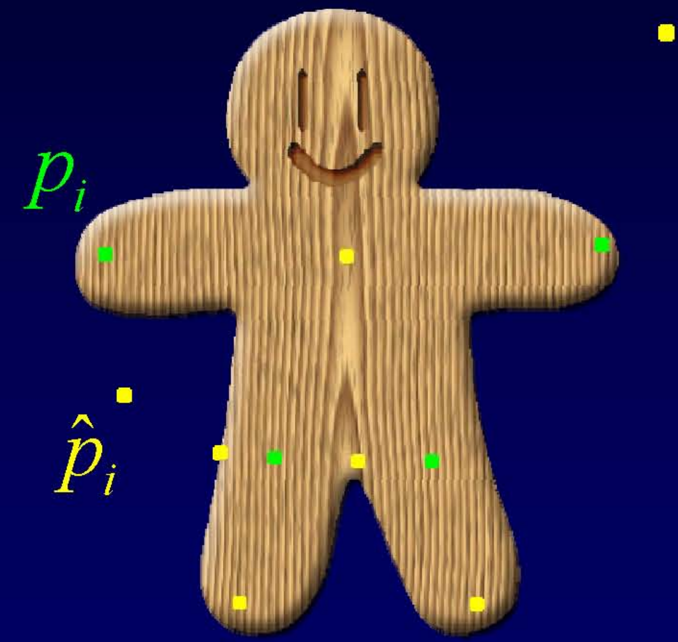
- Image deformation
  - Animation, morphing, medical imaging…
  - Controlled by handles: points, lines…
  - Function f
    - Interpolation: $f(p_i)=q_i$ (handles)
    - Smoothness ~ smooth deformations
    - Identity: $q_i=p_i \rightarrow f(x)=x$
  - Similar to scattered data interpolation

Least Squares Deformation

Least Squares Deformation



Least Squares Deformation



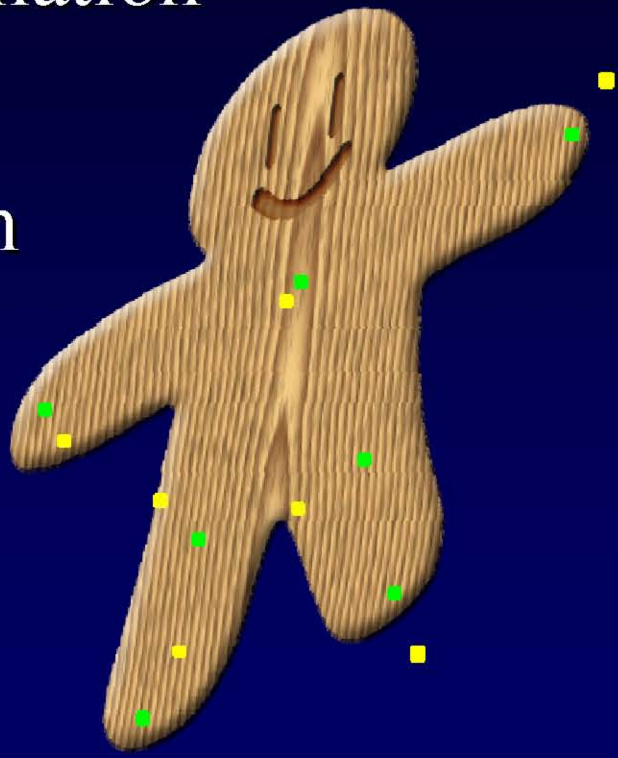
Least Squares Deformation

- Find the best affine transformation that maps p to $\hat{p}$
- Unique, closed-form solution
- Apply transformation to each point in the image

$$\sum_i | F(p_i) - \hat{p}_i |^2$$

Least Squares Deformation

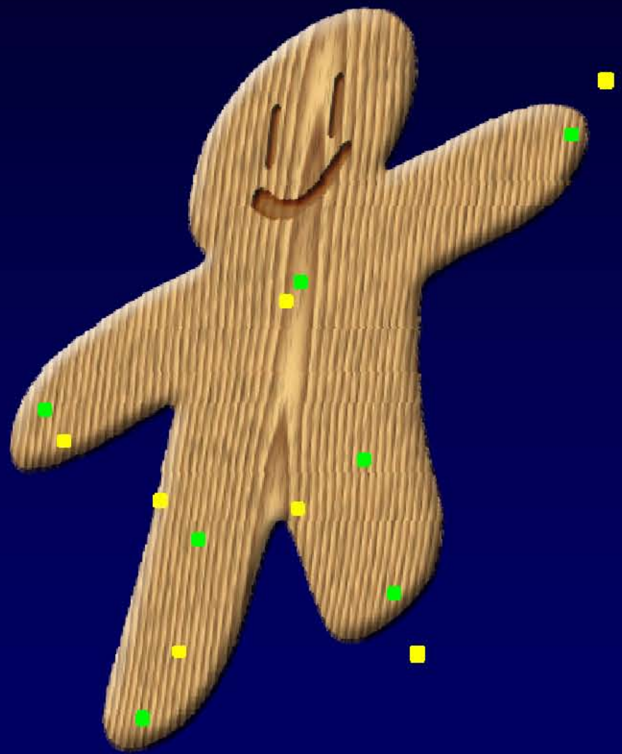
- Find the best affine transformation that maps p to $\hat{p}$
- Unique, closed-form solution
- Apply transformation to each point in the image



Moving Least Squares Deformation

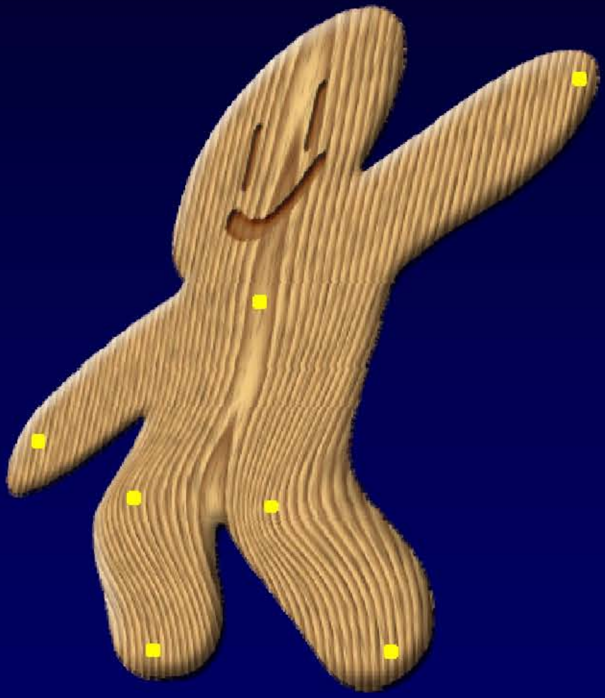
- Find the best transformation that maps p to $\hat{p}$
- Weight each term in least squares problem by $\frac{1}{|p_i - v|}$
- Different transformation for each point in image

$$\sum_i w_i \,|\, F(p_i) - \hat{p}_i \,|^2$$



Moving Least Squares Deformation

- Find the best transformation that maps p to $\hat{p}$
- Weight each term in least squares problem by $\frac{1}{|p_i - v|}$
- Different transformation for each point in image



Moving Least Squares Deformation

- Build image deformations based on handles (p_i, q_i)
  - Interpolation
  - Smoothness
  - Identity
- Given a point v, solve for the best $l_v(x)$:

$$\sum_i w_i \,| l_v(p_i) - q_i |^2 \qquad w_i = \frac{1}{| p_i - v |^{2\alpha}}$$

- Deformation function $f(v)=l_v(v)$
  - Locally defined --- MOVING Least Squares
  - Satisfy the three properties

Affine Transform

- Affine transform: $l_v(x)=xM+T$
- Translation can be removed: $T=q_*-p_*M$

$$p_* = \frac{\sum_i w_i p_i}{\sum_i w_i} \qquad q_* = \frac{\sum_i w_i q_i}{\sum_i w_i}$$

- So, $l_v(x)=(x-p_*)M+q_*$
- The new cost function:

$$\sum_i w_i \,|\, \hat{p}_i M - \hat{q}_i \,|^2$$

$$\hat{p}_i = p_i - p_* \qquad \hat{q}_i = q_i - q_*$$

- M could be different class of transformations

Affine Deformations

- Solution

$$M = \left(\sum_i \hat{p}_i^T w_i \hat{p}_i\right)^{-1} \sum_j w_j \hat{p}_j^T \hat{q}_j$$

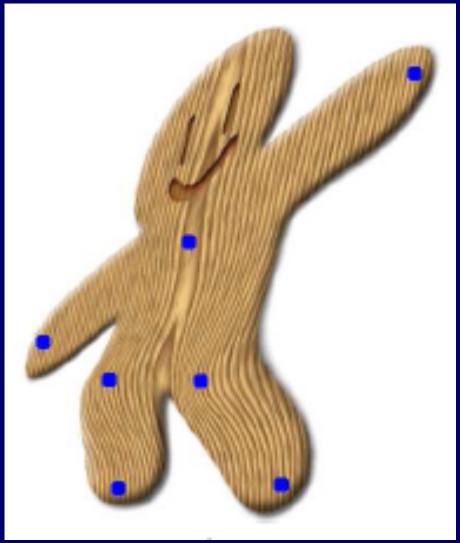
- The deformation function

$$f_a(v) = \sum_j A_j \hat{q}_j + q_*$$

$$A_j = (v - p_*)\left(\sum_i \hat{p}_i^T w_i \hat{p}_i\right)^{-1} \hat{p}_i^T$$

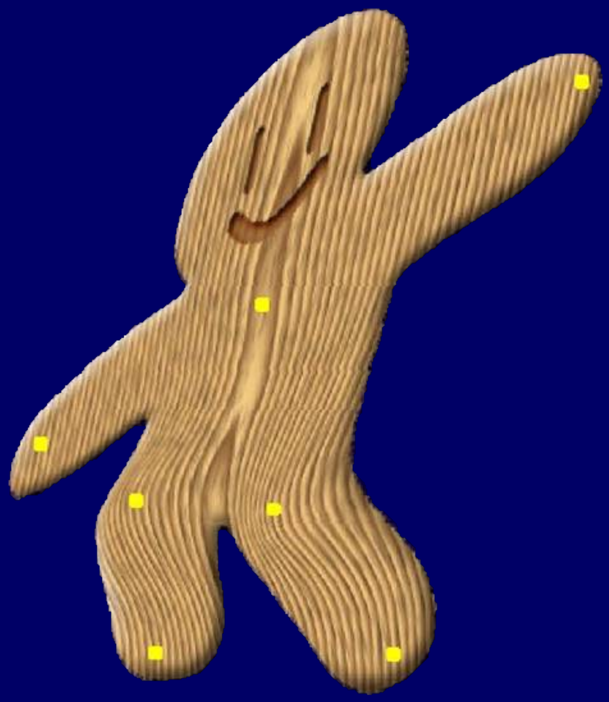
  - A_j can be precomputed

- Contains non-uniform scaling and shear



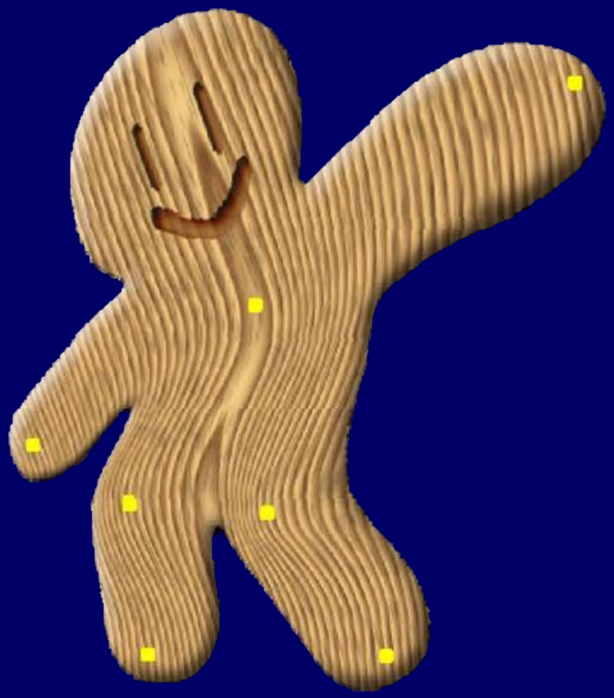
Similarity Deformations

- Only include translation, rotation, and uniform scaling
- Remove shear from deformation



Similarity Deformations

- Only include translation, rotation, and uniform scaling
- Remove shear from deformation



Similarity Deformations (Cont.)

- A special subset of affine transforms
  - Translation
  - Rotation
  - Constraints: Uniform-scaling
- Requirement: $M^T M=\lambda^2 I$
- Define $M = \begin{pmatrix} M_1 & M_2 \end{pmatrix}$, where $M_2=M_1^{\perp}$
  - Cost function (Least squares problem) still quadratic in M_1

$$\sum_i w_i \left| \begin{pmatrix} \hat{p}_i \\ -\hat{p}_i^{\perp} \end{pmatrix} M_1 - \hat{q}_i^T \right|^2$$

where $(x,y)^{\perp}=(-y,x)$

Similarity Deformations (Cont.)

- Solution for matrix *M*

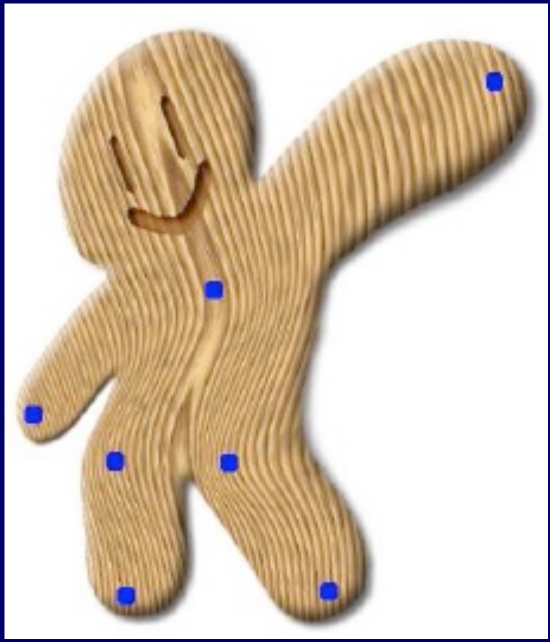
$$M = \frac{1}{\mu_s} \sum_i w_i \begin{pmatrix} \hat{p}_i \\ -\hat{p}_i^{\perp} \end{pmatrix} \begin{pmatrix} \hat{q}_i^T & -\hat{q}_i^{\perp T} \end{pmatrix}$$

where $\mu_s = \sum_i w_i \hat{p}_i \hat{p}_i^T$

- Solution for deformation function

$$f_s(v) = \sum_i \hat{q}_i \left(\frac{1}{\mu_s} A_i\right) + q_*$$

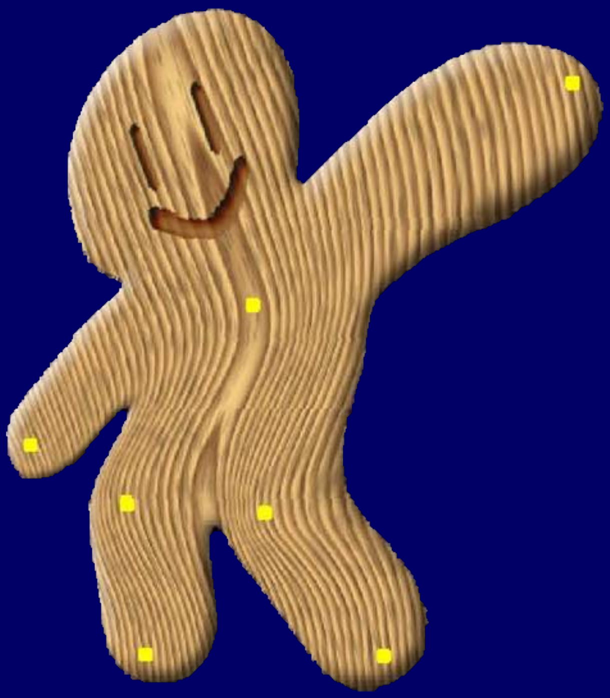
where $A_i = w_i \begin{pmatrix} \hat{p}_i \\ -\hat{p}_i^{\perp} \end{pmatrix} \begin{pmatrix} v - p_* \\ -(v - p_*)^{\perp} \end{pmatrix}^T$



- Property
  - Preserves angles better than affine deformations
  - Local scaling can hurt realism

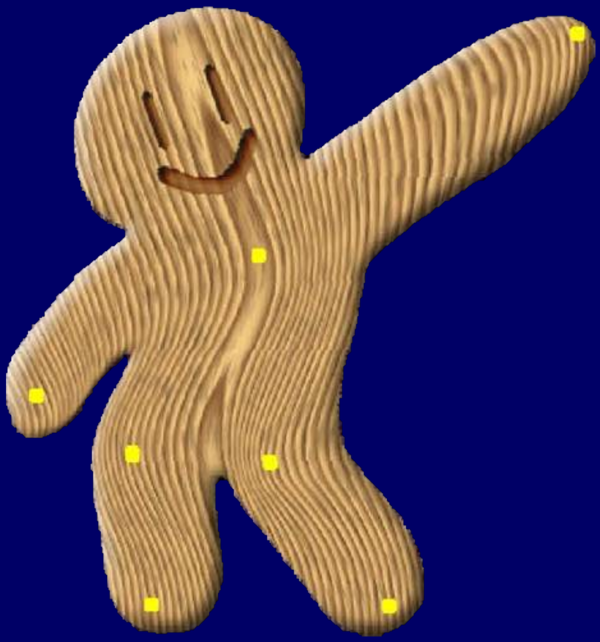
Rigid Deformations

- Deformations should not include scaling and shear
  - [Alexa 2000; Igarashi et al. 2005]
- Best rigid transformation can be found from best similarity transformation
  - Remove local uniform scaling $M^T M = I$



Rigid Deformations

- Deformations should not include scaling and shear
  - [Alexa 2000; Igarashi et al. 2005]
- Best rigid transformation can be found from best similarity transformation
  - Remove local uniform scaling $M^T M = I$



Rigid Deformations (Cont.)

- Solution for matrix *M*

$$M = \frac{\sum_i w_i \begin{pmatrix} \hat{p}_i \\ \hat{p}_i^{\perp} \end{pmatrix} \begin{pmatrix} \hat{q}_i^T & \hat{q}_i^{\perp T} \end{pmatrix}}{\sqrt{\left(\sum_i w_i \hat{q}_i \hat{p}_i^T\right)^2 + \left(\sum_i w_i \hat{q}_i \hat{p}_i^{\perp T}\right)^2}}$$

- Solution for deformation function

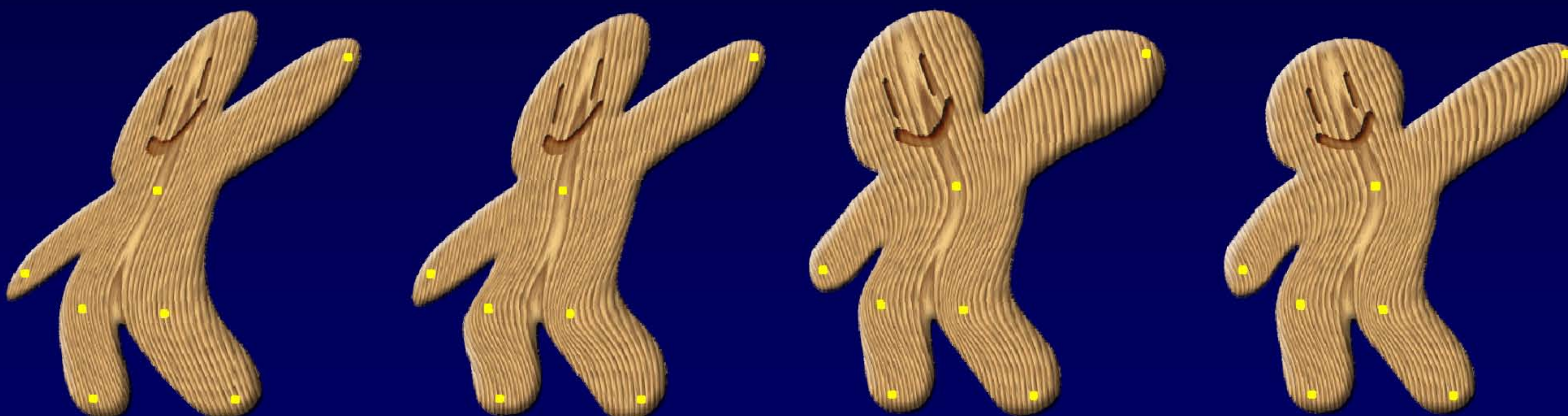
$$f_r(v) = |v - p_*| \frac{\vec{f}_r(v)}{|\vec{f}_r(v)|} + q_*.$$

where $\vec{f}_r(v) = \sum_i \hat{q}_i A_i$ and A_i is as in similarity deformations.

- Limited precomputation

Comparison

- Rigid transformations create the most realistic looking deformations



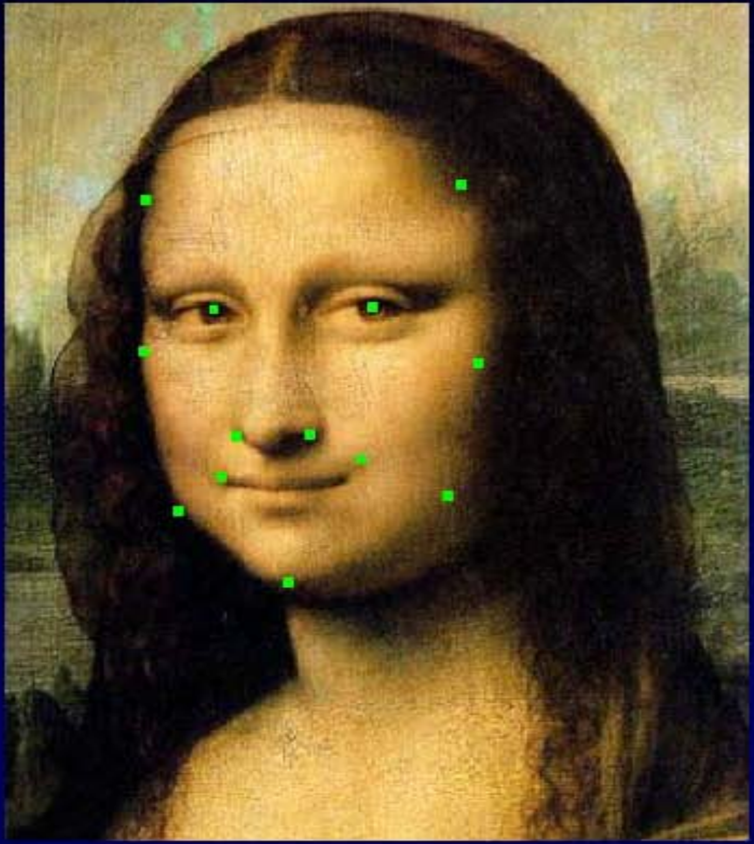
Thin-Plate [Bookstein 1989] | Affine MLS | Similarity MLS | Rigid MLS

Comparison:

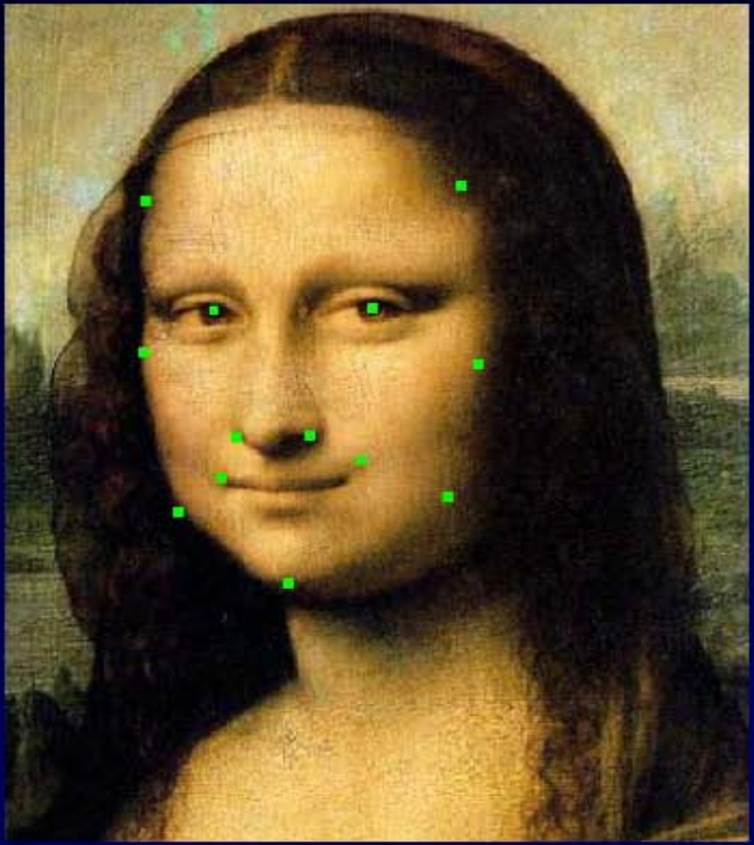
Affine	Similarity	Rigid
Non-uniform scaling and shear	Local scaling can often lead to undesirable Deformations But **Preserves angles**	Deformation is quite realistic
Computation time very low abt 1.5ms	Highest was computed as 3.4ms	Highest was computed 3.8ms

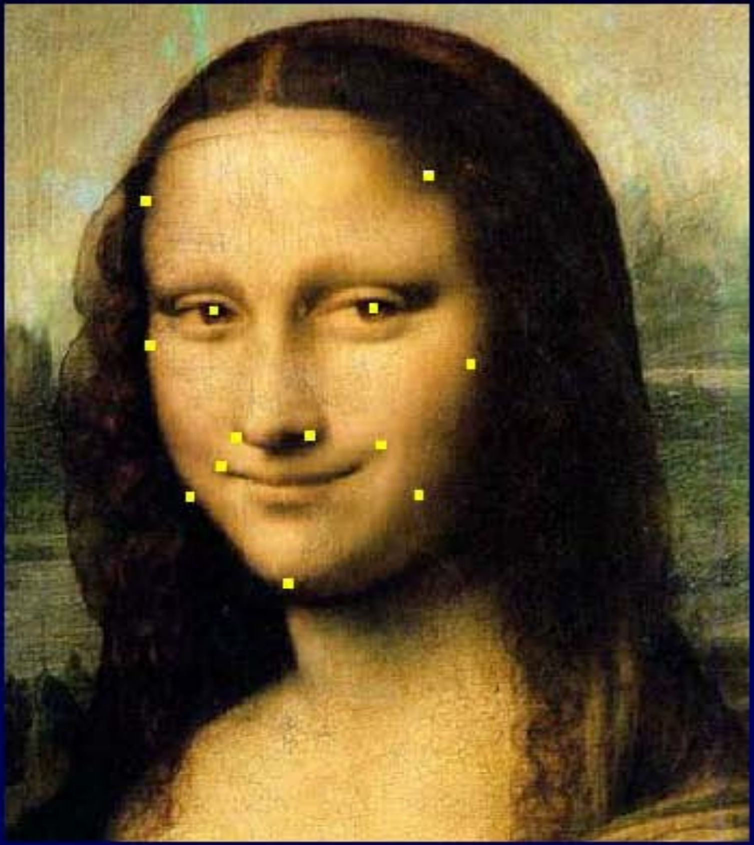
Examples

Examples



Examples



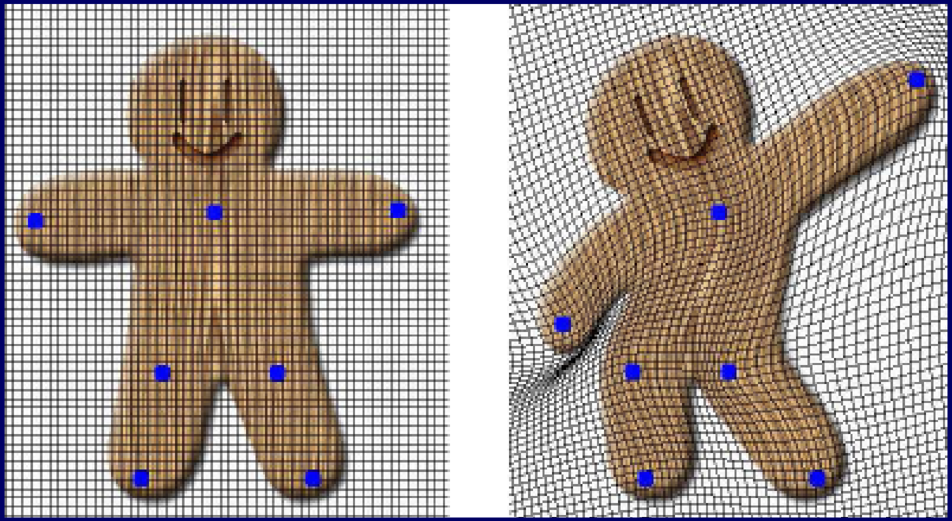


Examples

Some Rigid Deformation Results

Implementation

- Pixel by pixel --- expensive
- Deformation on a downsampled grid
- Fill quads using bilinear interpolation

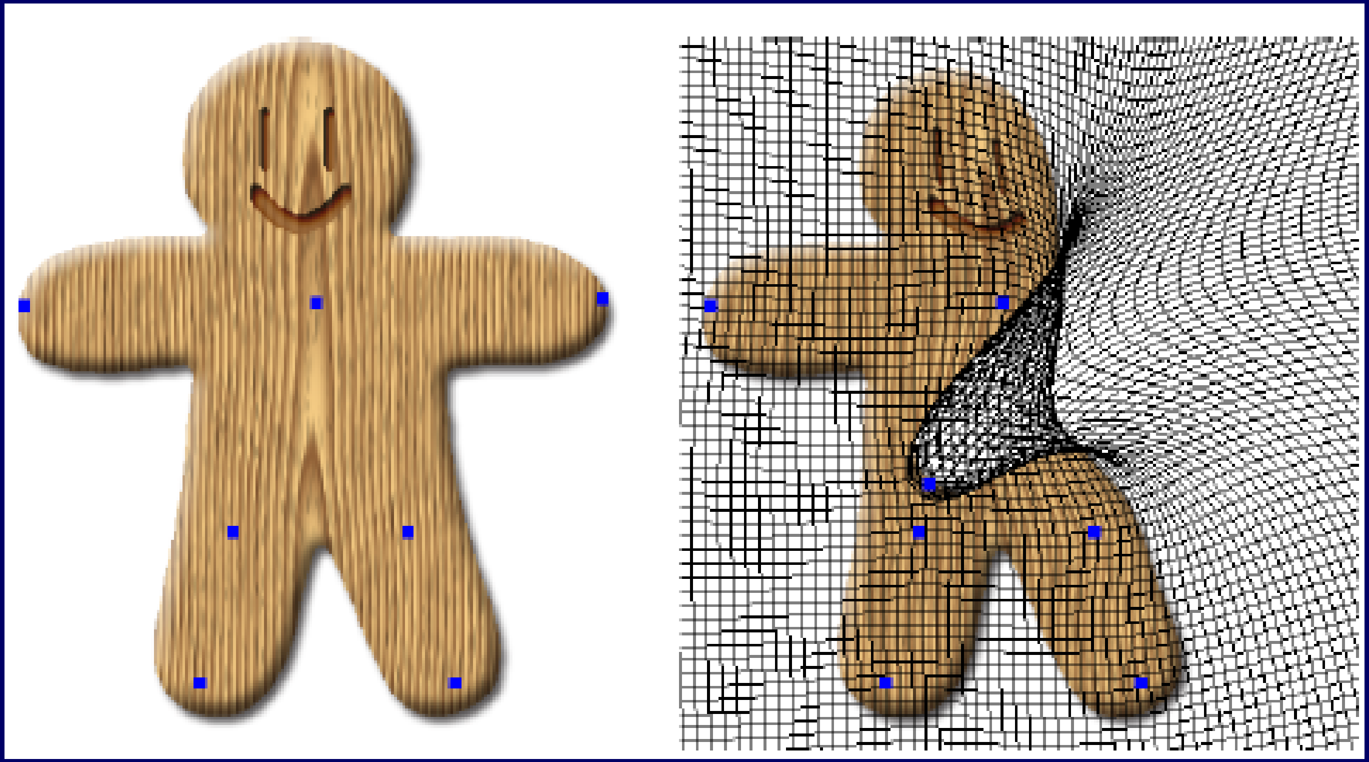


Timing

Method	Figure 1 (7 points)	Figure 4 (11 points)	Figure 5 (7 lines)
Affine MLS	1.5 ms	2.2 ms	1.5 ms
Similarity MLS	2.3 ms	3.4 ms	1.6 ms
Rigid MLS	2.6 ms	3.8 ms	3.3 ms

Grid size: 100x100

Limitation: Foldbacks



These situations occur when the sign of the Jacobian of f changes.

Conclusions

- Moving Least Squares
  - 2x2 linear system at each grid point
    - Real-time
- Similarity and rigid transformations
  - More realistic results
  - Extension: line segments
  - Closed-form expressions

Conclusions (Cont.)

- Future Work
  - Adding topological information
  - Generalizing to 3D to deform surfaces
  - Handles can be any curves