Perceptually Based Approach for Planar Shape Morphing

Ligang Liu, Guopu Wang*,
Bo Zhang, Baining Guo, Harry Shum
Microsoft Research Asia
*Tsinghua University

Morphing Problem

- Gradual transformation of one shape into another
  - Wide applications
- Two steps
  - Vertex correspondence problem (Important)
  - Vertex path problem

Previous Work

- Vertex correspondence problem
  - Sederberg and Greenwood 92 [22]
  - Zhang 96 [26]
  - Cohen et al. 97 [5]
  - Hui and Li 98 [10]
  - Sebastian et al. 03 [20]
- Vertex path problem
  - Preserve geometric properties [3, 21]
  - Interpolating compatible triangulations [1,8]

Motivation

- Correspondence between features
  - Instead of correspondence between sampling points
  - Perceptually meaningful
  - Preserving feature correspondence

Overview

- Feature extraction
- Geometric shape properties for features
  - Based on a wide range of local region
- Similarity measurements between features
  - Penalty measurement of discarding feature
- Dynamic programming with “skips”

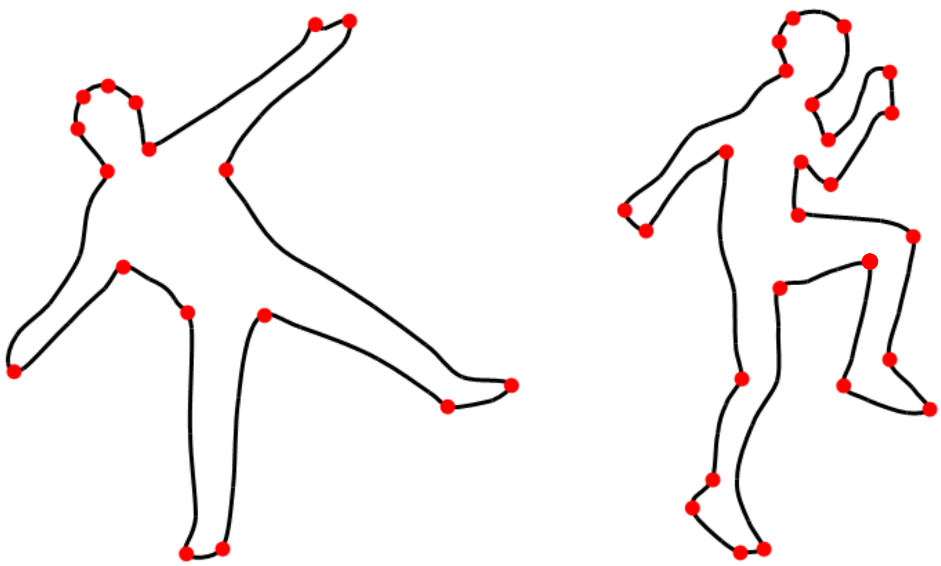
Feature Detection

- A feature is anything that is
  - Salient part
  - Localized
  - Meaningful
  - Detectable
- A well-studied research area
  - Computer vision, medical imaging, computational fluid dynamics

Our Approach

- Potential feature points
  - Curvature extrema
  - Cusp
  - Inflection points
  - Discontinuities of curvature
- An improved approach based on Chetverikov and Szabo's [4]
  - Simple and efficient

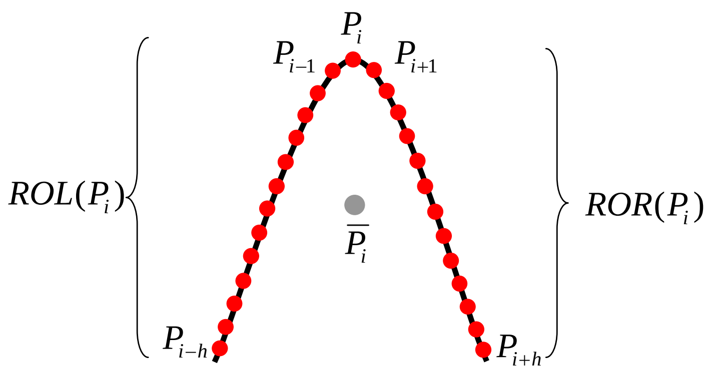
Results of Feature Detection



Local Shape Properties Near a Feature Point

- Local Neighborhood
  - Sampling points near feature point
  - Defined as region of support (ROS)
- Covariance matrix of ROS
  - Principle component analysis of ROS
  - Eigen structure of the covariance matrix

Covariance Matrix



Region of support (ROS):

$$ROS_h(P_i) = \{P_j\}_{j=i-h}^{j=i+h}$$

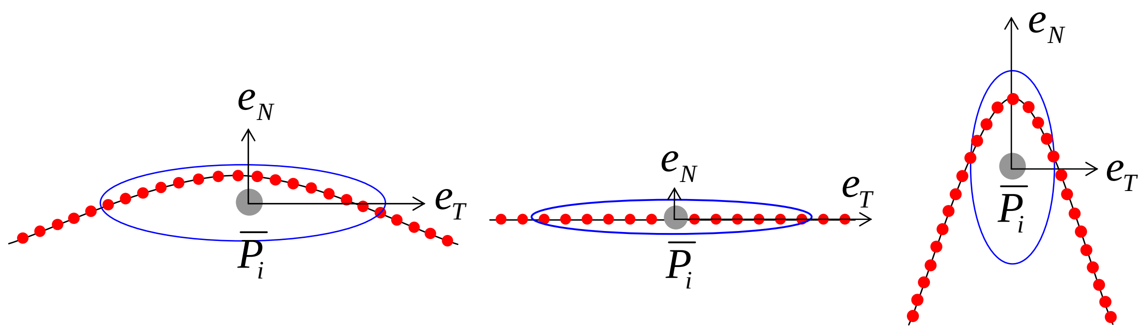
$\bar{P}_i$: center of $ROS_h(P_i)$

Covariance matrix:

$$C(P_i) = \frac{1}{2h+1} \sum_{j=i-h}^{i+h} (P_j - \bar{P}_i)(P_j - \bar{P}_i)^T$$

Eigen Structures of Covariance Matrix

λ_T, λ_N : eigenvalues of $C(P_i)$



$(a)\ \lambda_T > \lambda_N > 0$ $\quad (b)\ \lambda_T > 0, \lambda_N = 0$ $\quad (c)\ \lambda_N > \lambda_T > 0$

Geometric Quantity Properties of a Feature Point

(1) feature variation

$$\sigma(P_i) = \frac{\lambda_N}{\lambda_N + \lambda_T}$$

(2) feature side variation

$$\tau(P_i) = \frac{\sigma(ROL(P_i)) + \sigma(ROR(P_i))}{2}$$

(3) feature size

$$\rho(P_i) = \frac{\rho(ROL(P_i)) + \rho(ROR(P_i))}{2}$$

Similarity Measurement of Two Feature Points

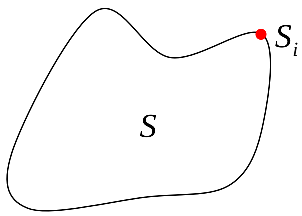
$$SimilarityCost(S_i, T_j) = \sum_{q=\sigma,\tau,\rho} \varpi_q \Delta_q(S_i, T_j)$$

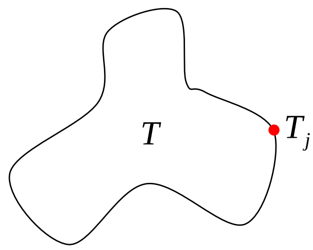
The differences of feature quantities:

$$\Delta_\sigma(S_i, T_j) = \left|\sigma(S_i) - \sigma(T_j)\right|$$

$$\Delta_\tau(S_i, T_j) = \frac{1}{2}(|\sigma(ROL(S_i)) - \sigma(ROL(T_j))| + |\sigma(ROR(S_i)) - \sigma(ROR(T_j))|)$$

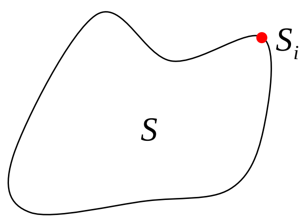
$$\Delta_\rho(S_i, T_j) = \frac{1}{2}(|\rho(ROL(S_i)) - \rho(ROL(T_j))| + |\rho(ROR(S_i)) - \rho(ROR(T_j))|)$$





Penalty Measurement of Discarding a Feature Point

$$DiscardingCost(S_i) = \sum_{q=\sigma,\tau,\rho} \varpi_q \left|q(S_i)\right|$$



Minimization Problem

A correspondence is a mapping between feature points:

$$J : \{S_i\} \to \{T_j\}$$

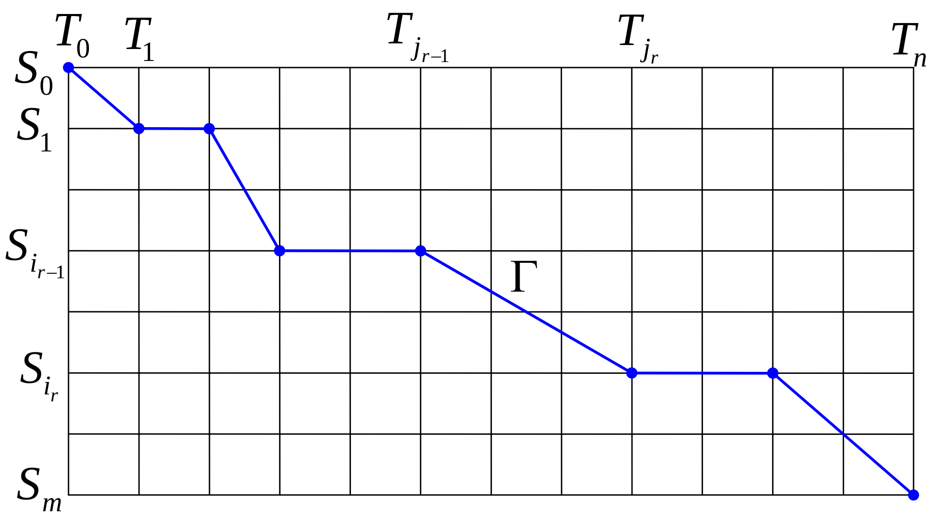
Similarity function of S and T:

$$SimilarityCost(S,T;J) = \sum_{i=0}^{m-1} SimilarityCost(S_i, T_{J(i)})$$

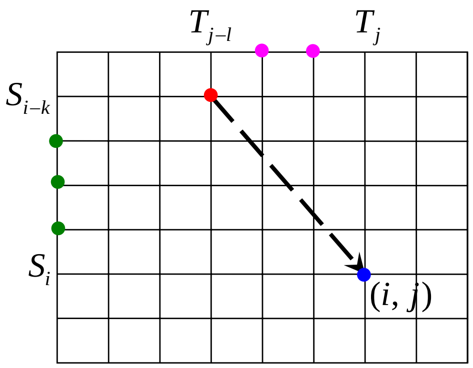
Minimization problem:

$$\min_J SimilarityCost(S,T;J)$$

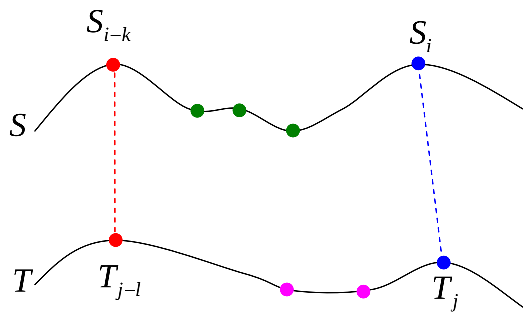
Dynamic Programming (DP)



Skips in DP Graph



Update cost of node (*i*,*j*) from node (*i-k*, *j-l*) with skips



Correspondence between feature points

(The other feature points are discarded)

Discarding Cost for Skips

Penalty cost of discarding the feature points between S_{i-k} and S_i :

$$DiscardingCost\big(S(i-k \mid i)\big) = \sum_{p=i-k+1}^{i-1} DiscardingCost(S_p)$$

Penalty cost of discarding the feature points between T_{j-l} and T_j :

$$DiscardingCost\big(T(j-l \mid j)\big) = \sum_{p=j-l+1}^{j-1} DiscardingCost(T_p)$$

Cost of a Complete Path

$$Cost(S,T,\Gamma)=\sum_{r=1}^{R}\delta\left(S(i_{r-1}\,|\,i_r),T(j_{r-1}\,|\,j_r)\right)$$

where the similarity cost between $S(i_{r-1}\,|\,i_r)$ and $T(j_{r-1}\,|\,j_r)$ is defined by

$$\begin{aligned}\delta\left(S(i_{r-1}\,|\,i_r),T(j_{r-1}\,|\,j_r)\right)=&\ DiscardingCost\left(S(i_{r-1}\,|\,i_r)\right)\\ &+DiscardingCost\left(S(i_{r-1}\,|\,i_r)\right)\\ &+\lambda\cdot SimilarityCost(S_{i_r},T_{j_r})\end{aligned}$$

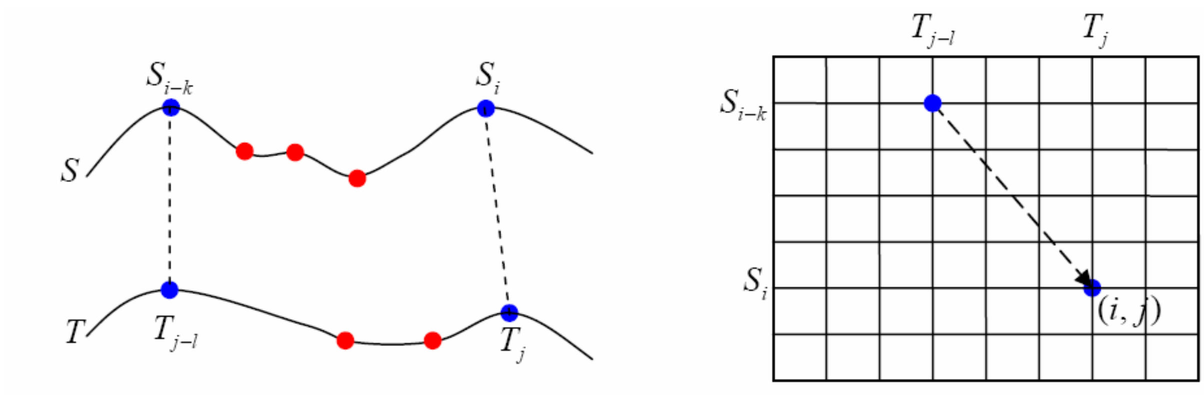
high values of λ encourage discarding feature points

low values of λ inhibit discarding feature points

Implementations

Optimum cost of incomplete path ending at node :

$$node(i,j)=\min_{k,l\geq 0}\left[node(i-k,j-l)+\delta\left(S(i-k\,|\,i),T(j-l\,|\,j)\right)\right]$$



The case of k=3, l=2

Run-Time Complexity

$O(mn)$ if there is no skip in the DP graph

$O(C^2mn)$ if restricting discarding to C feature points ($C \ll \min(m,n)$)

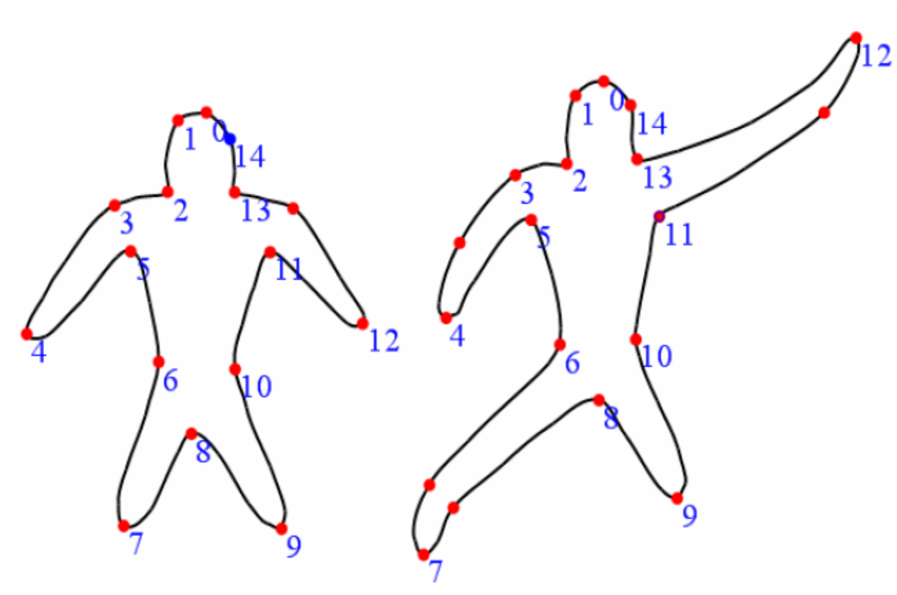
where

m : the number of feature points of shape S

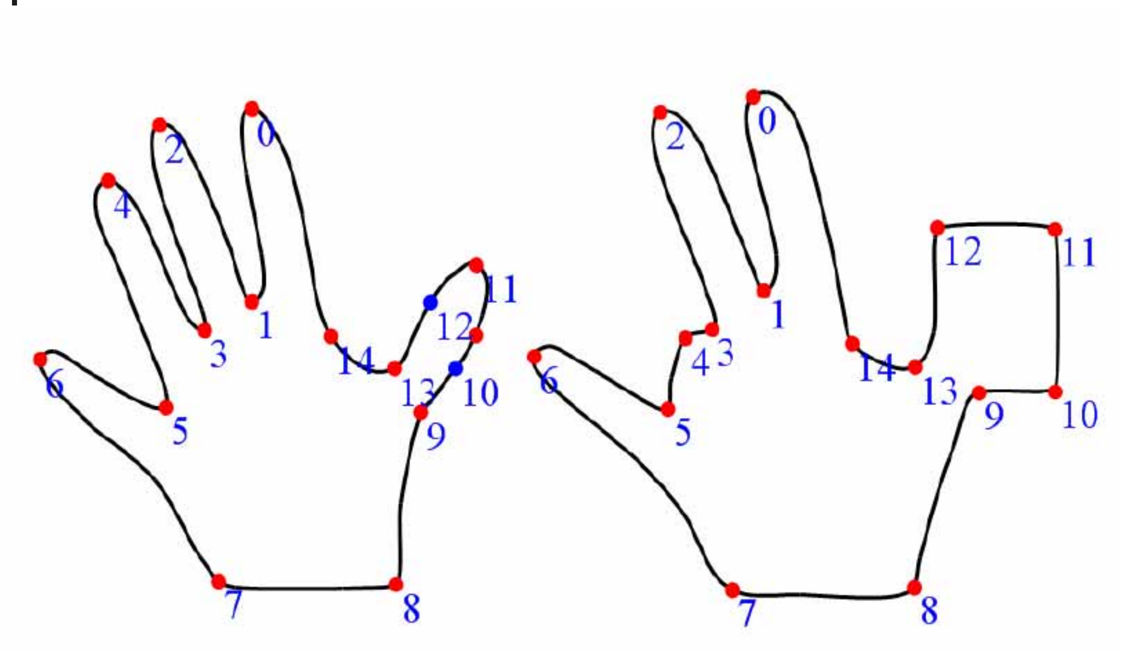
n : the number of feature points of shape T

Faster than the previous methods because the number of feature points is much less than the number of sampling points in the previous methods!

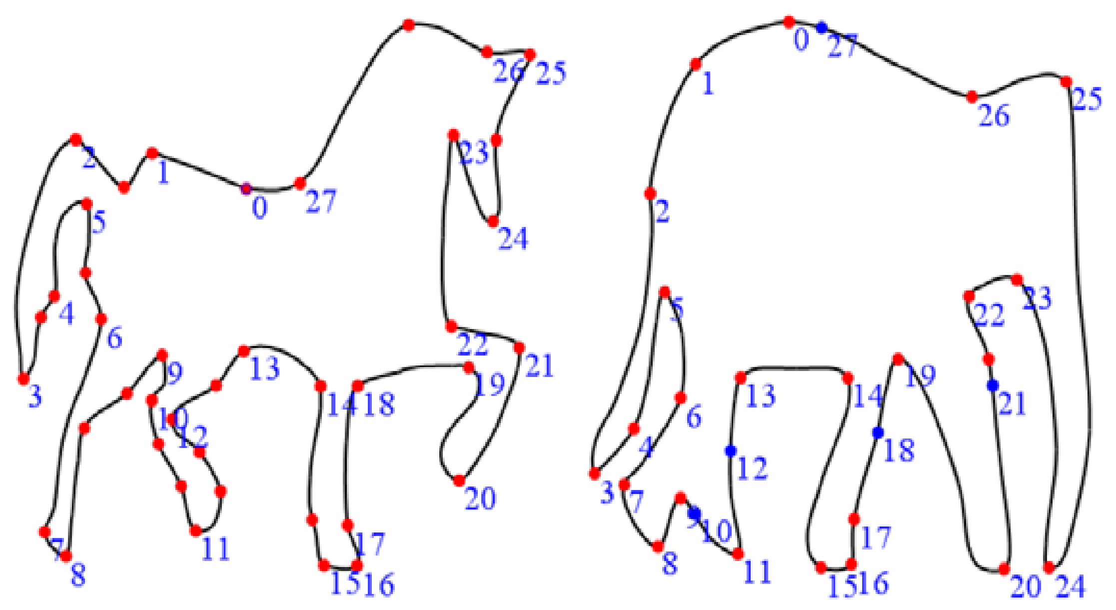
Correspondence Results (1)



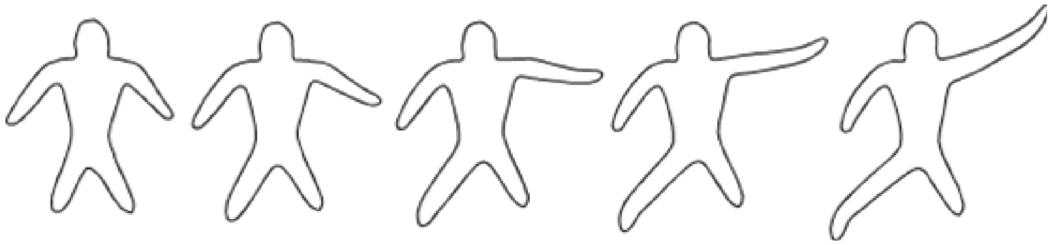
Correspondence Results (2)



Correspondence Results (3)



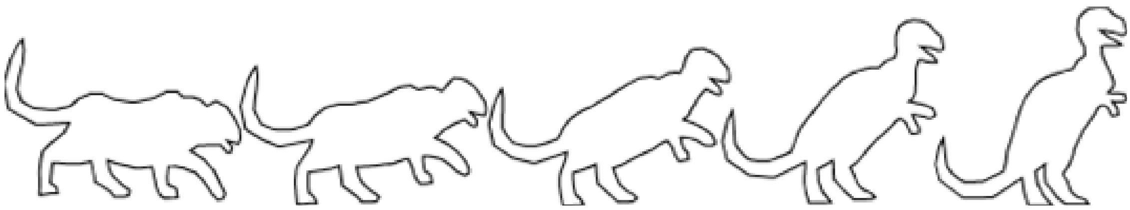
Morphing Results (1)



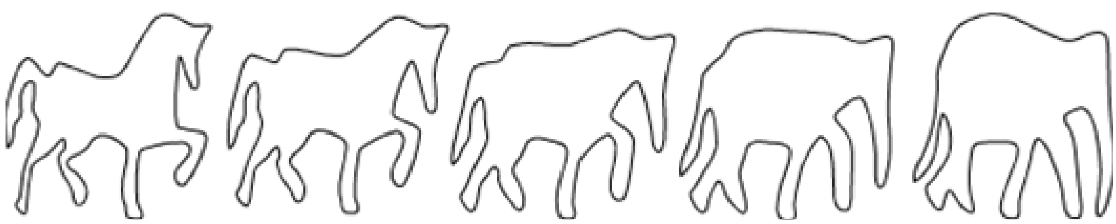
(a)

(b)

Morphing Results (2)

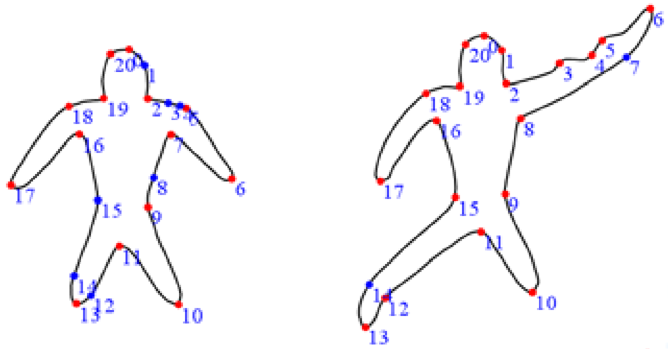


(c)

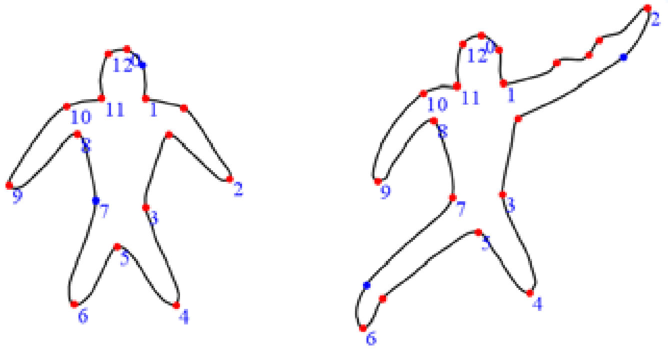


(d)

Noise Resistance

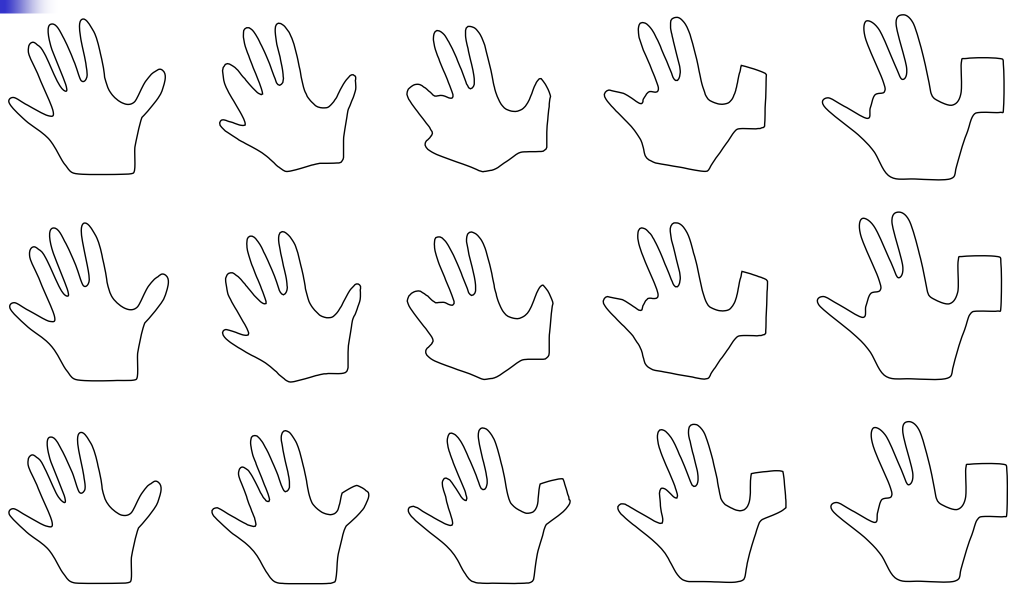


(a) C=0



(b) C=2

Comparative Results (1)



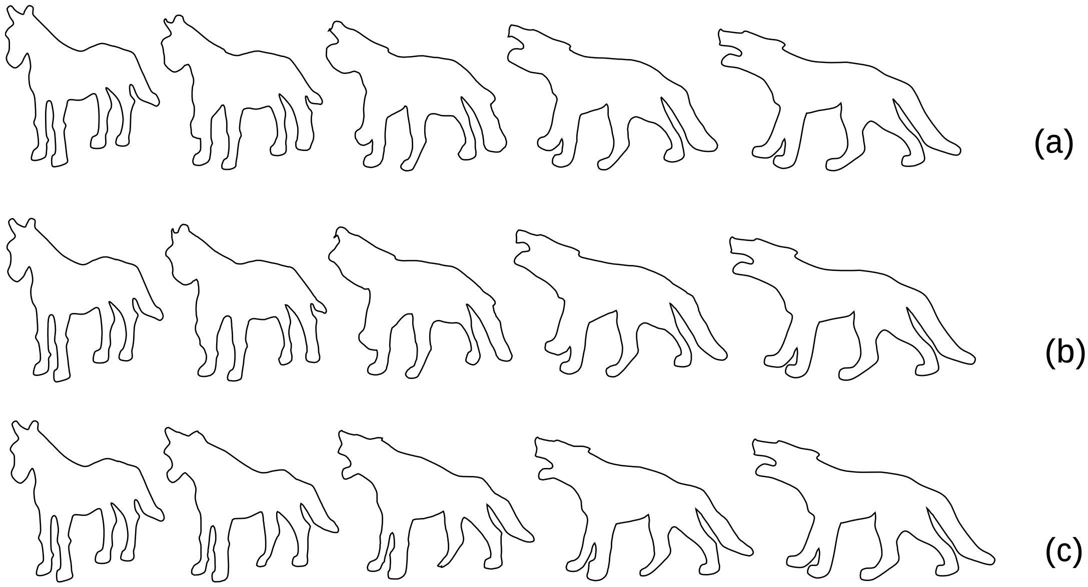
(a)

(b)

(c)

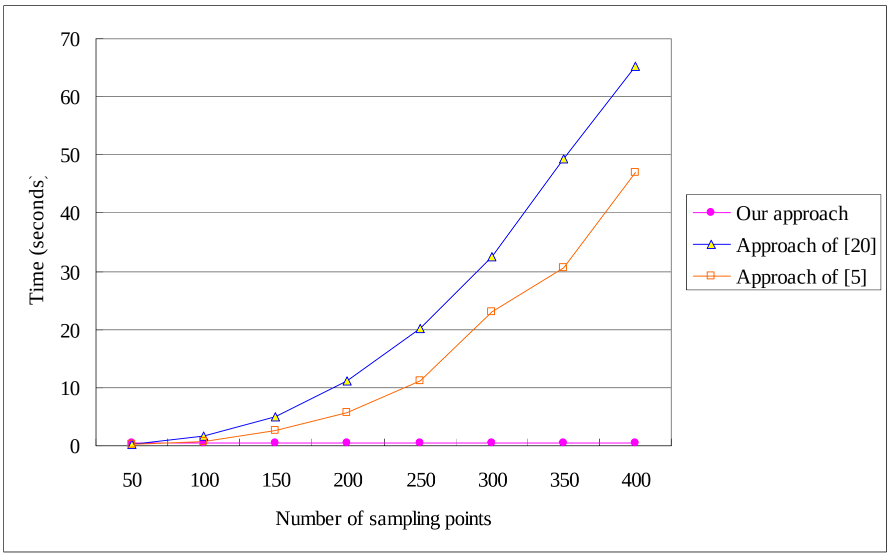
(a) Approach of [5]; (b) Approach of [20]; (c) Our approach

Comparative Results (2)



(a) Approach of [5]; (b) Approach of [20]; (c) Our approach

Comparisons of Running Time



Conclusions

- A novel approach based on matching feature points
  - Feature preserved
  - Meaningful for feature matching
- Dynamic programming process with skips
  - Penalty cost for discarding a feature point
  - Noise resistant to small or unimportant feature
- Fast and robust
  - Not sampling points in DP graph
  - Can be used in real-time system

Problems and Future Work

- Can not avoid self-intersection
- Might fail for simple shapes like circle or helical curve
- Local but not global shape feature
- Will be affected by large noise

Thank you!