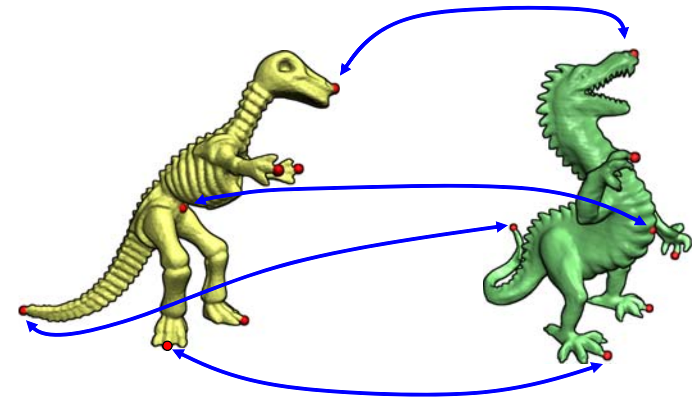
# Deformation-Drive Shape Correspondence

Hao (Richard) Zhang[1], Alla Sheffer[2], Daniel Cohen-Or[3],
Qingnan Zhou[2], Oliver van Kaick[1], and Andrea Tagliasacchi[1]

July 3, 2008

1 SFU gruvi 2 UBC Imager 3 TEL AVIV UNIVERSITY THE BLAVATNIK SCHOOL OF COMPUTER SCIENCE

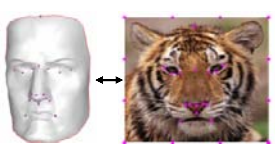
## The correspondence problem



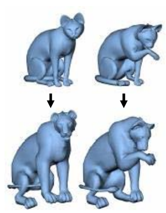


## A classic problem

- Fundamental to geometry processing
- Many applications
  - Attribute transfer, e.g., texture, animation, geometry



[Kraevoy et al. 04]



[Sumner & Popovic 04]



## A classic problem

- Fundamental to geometry processing
- Many applications
  - Attribute transfer, e.g., **texture, animation**
  - Statistical shape modeling, e.g., **SCAPE**

[Anguelov et al. 05]

## A classic problem

- Fundamental to geometry processing
- Many applications
  - Attribute transfer, e.g., **texture, animation**
  - Statistical shape modeling, e.g., **SCAPE**
  - Object recognition



## An intensely studied problem

- Different fields: computer vision, medical image analysis, computer graphics, etc.
- Different shape classes
- Rigid vs. non-rigid
- Discrete vs. continuous
- Global vs. partial

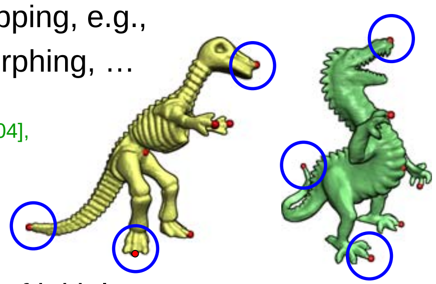
**Need to be more specific …**



## *Coarse feature* correspondence

- Anchors for continuous mapping, e.g., cross-parameterization, morphing, …

  [Schreiner et al. 04], [Kraevoy & Sheffer 04], [Cohen-Or et al. 98], [Gregory et al. 98], [Alexa et al. 00]



- Automatic correspondence of initial, sparse features is more difficult

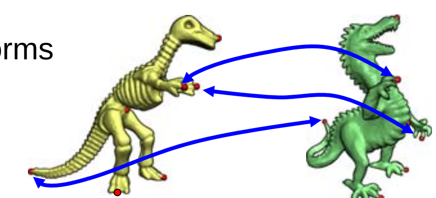


## *Non-rigid* correspondence

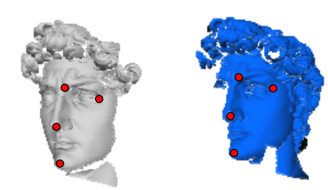
- Tolerate non-rigid transforms
- Most existing works are on rigid registration

  [Gelfand et al. 05], [Li & Guskov 05], [Huber & Hebert 03], [Huang et al. 06], [Gal & Cohen-Or 06]

  - Low-dim transform space
  - Strict rigidity constraints



Non-rigid



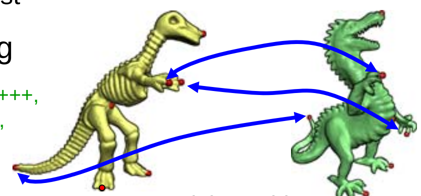
Rigid registration [Gelfand et al. 05]

## *Partial* matching

- Matching parts of the shapes
- Higher combinatorial complexity
  - Partial matching set not known
- Most approaches via optimization
  - Hard to define what is the "best"
- Applied in rigid/affine setting

Relaxation labeling: [Rosenfeld et al. 76]+++,
Assignment: [Gold & Rangarajan 96]+++,
[Funkhouser & Shilane 06],
[Gelfand et al. 05], etc.

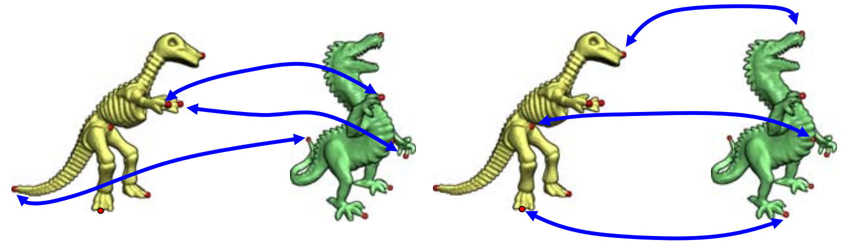
Self-similarity
[Gal & Cohen-Or 06]



Partial matching



## Allow *greater geometry variability*



Pose + non-uniform scaling

Local shape variability

**Not registration …**



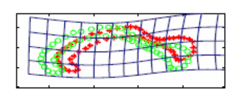
## Non-rigid registration

- Overlapping patches: geometry repeats
- Rigidity constraints still useful, e.g., with articulation only
- Precise registration, not coarse feature correspondence



## Other non-rigid works

- Works in vision, medical imaging
  - Limited shape variability

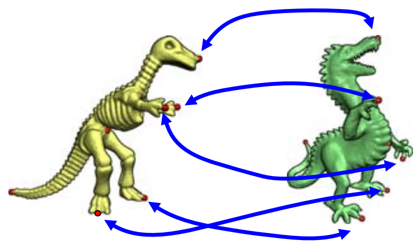


[Wang et al. 06]

[Vaillant & Glaunes 05]

## Deal with *symmetry* in shape

- Cannot be resolved with purely intrinsic approaches,
  - e.g., use of pair-wise geodesic distances between features in graph matching
- Symmetry breaking calls for user intervention,
  - e.g., SCAPE [Angeulov et al. 05]

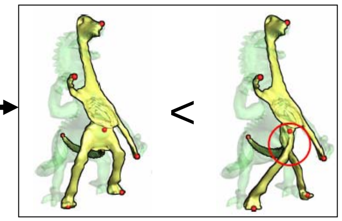




## Solution: a more global approach

- Local vs. global criteria
  - Local: feature region similarity
  - Global: global consistency of correspondence
  - **Local criterion less reliable with large shape variations**
- Emphasis on global via non-rigid mesh deformation

Correspondence cost = effort to deform one mesh into other







## A result



## The deformation idea

- An old idea, e.g., [Sederberg & Greenwood 92]
  - Works in 2D
  - Energy = bending (angle) + stretching (edge length)
  - Others rely on extrinsic criterion or parameterized models [Blanz & Vetter 99], [Sheldon 00], etc
- First time **surface (mesh) deformation** is used to solve general non-rigid (partial) shape correspondence

# Our contributions

- Deformation-driven, automatic feature correspondence
  - Handles variations in pose, local scale, part composition, geometric details
- Self distortion cost
  - Deformation energy **measured on surface of deformed mesh**
  - Feature similarity and geodesic distances do not enter cost
  - Symmetry breaking (**surface distortion**) + partial matching
- Combinatorial search (priority-driven search)
  - Exploration of large solution space
  - Avoid initial alignment or local minima



# Algorithm overview

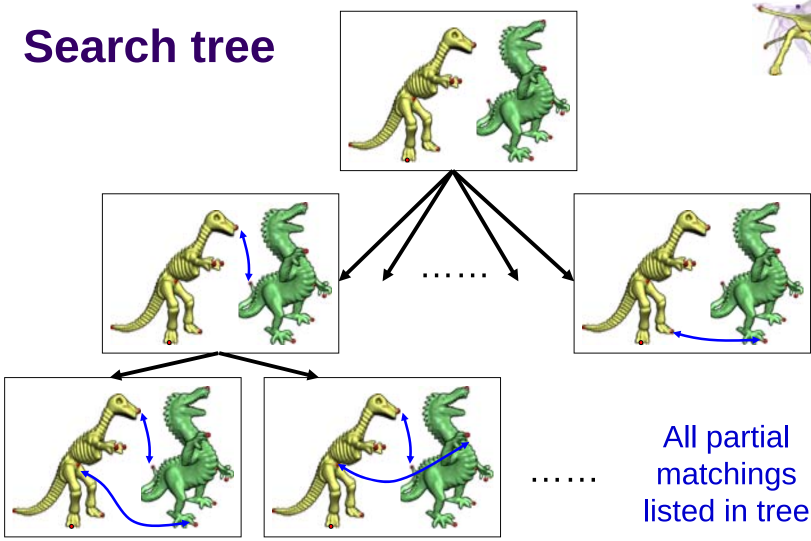
Step 1: feature extraction

Step 2: combinatorial search

- Priority = distortion cost
- Pruning by feature similarity and geodesic distance



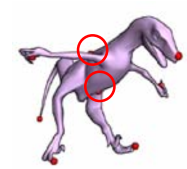
# Search tree



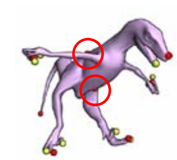



# Step 1: Feature extraction

- Edge features unstable under articulation
- Our choice: **part extremities**
  - Most prominent and stable features of parts
  - Critical points of **average geodesic distance (AGD) fields** [Hilaga et al. 01]
  - **Poison disk sampling** prioritized by prominence values (AGD)
  - Local maxima: part extremities
  - Local minima: central part of body



Poisson disk radius $\gamma = 0.2$

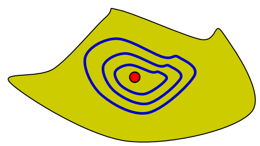


$\gamma = 0.1$

## Step 2: tree search

- Each node is a potential candidate solution
- Candidates are prioritized by correspondence costs — **best-first search** strategy
- Thresholds on
  - Pair-wise feature similarity via **curvature maps** [Gatzke et al. 05]
    Collect average curvature in geodesic bins → 1D signature
  - Total geodesic distortions

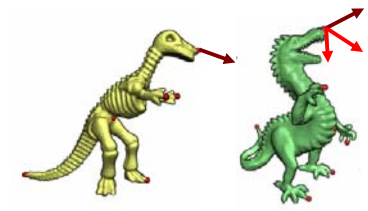
  for **pruning** candidate solutions





## Mesh deformation

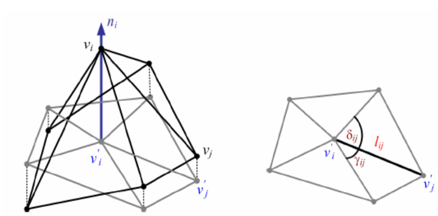
- Need efficient and robust mesh deformation
  - Applied to evaluate each candidate solution
- Use the linear differential (rotation-invariant) scheme of [Lipman et al. 05]
- Target local frames estimated via rigid alignment of matched vertices and normals





## Distortion energy/cost

- Measured on deformed mesh — self-distortion
- Symmetrize to remove order dependence
- Actual distortion computed via **mean-value encoding** [Kraevoy & Sheffer 06]
  - Does not depend on rotated normals from rigid alignment
  - More accurate distortion error estimate

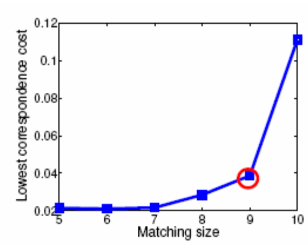



Mean-value encoding [Kraevoy & Sheffer 06]

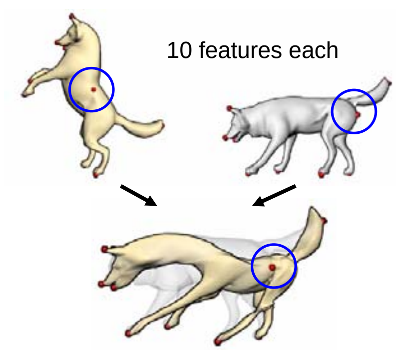


## Optimal (partial) matching size

- Finding the **largest jump in correspondence cost**

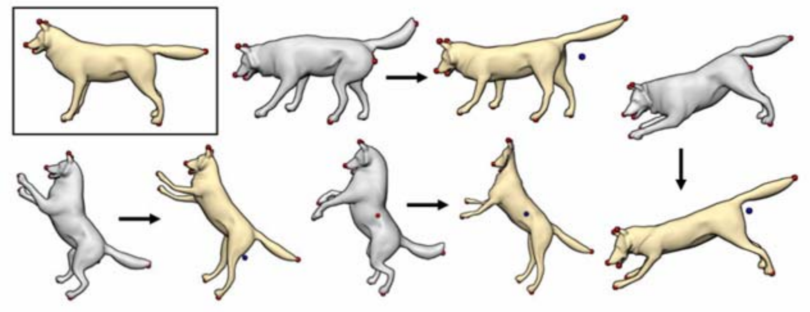



Plot of cost curve




**Wrong matching of size 10**

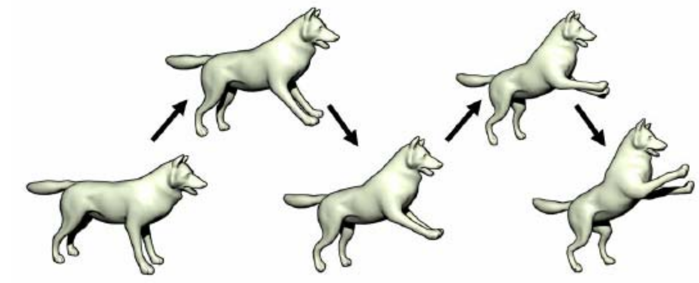
# Results: articulation only



- **Fully automatic**: 10 features selected + tree search
- **All parameters and thresholds fixed** throughout



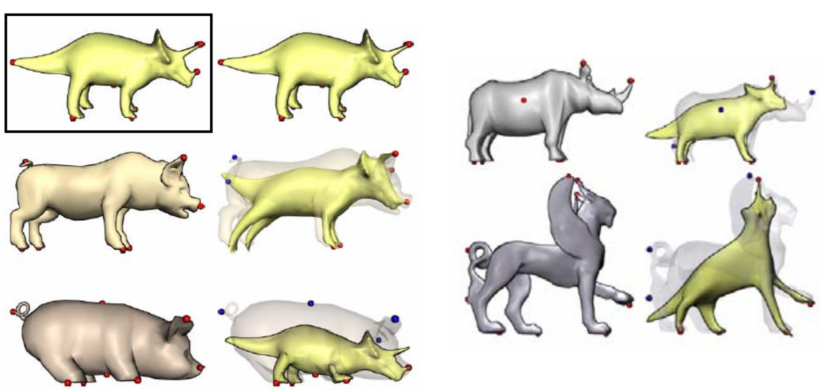
# Results: shape morphing



- Based on cross-parameterization [Kraevoy & Sheffer 04]



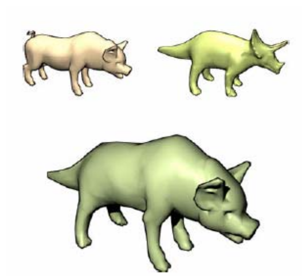
# Results: larger shape variations



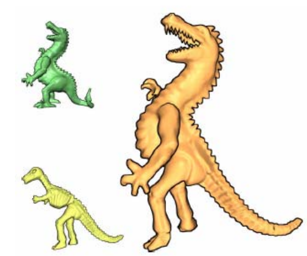
Observe partial matching



# Results: shape blending



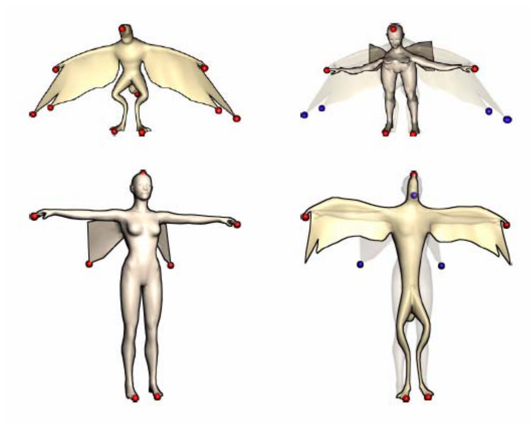
A “prehistoric” pig



Raptor under modern medical practice

- Again, based on dense cross-parameterization

## More correspondence results



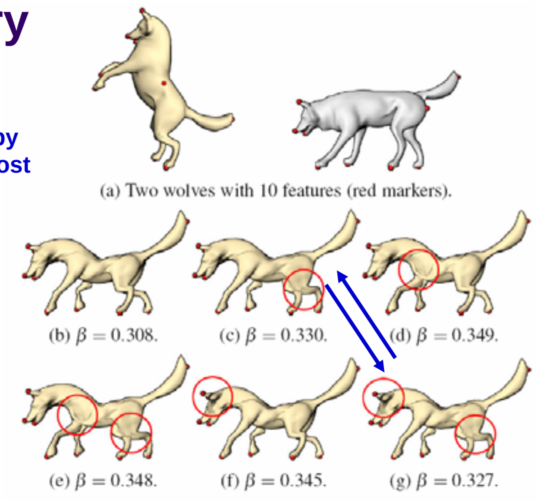
July 3, 2008, Copenhagen, Denmark



## Symmetry

(b) → (g): sorted by our deformation cost

$\beta$: total geodesic distortion by the correspondence

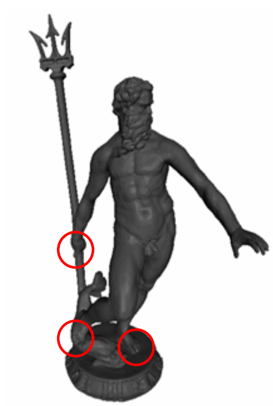



July 3, 2008, Copenhagen, Denmark



## Limitations

- **High search cost:** 20 min to > 1 hr
  - Vertex counts: 600 to 3,500
  - Price to pay for full autonomy (conservative parameters and thresholds)
- Reliance on extremity features
- Coarse correspondence
  - Can be refined (even correct local errors)

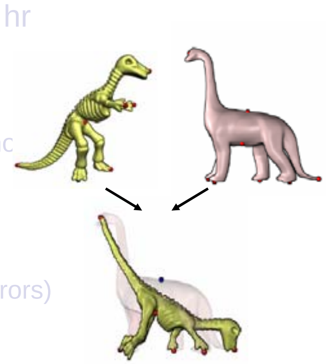


July 3, 2008, Copenhagen, Denmark



## Limitations

- High search cost: 20 min to > 1 hr
  - Vertex counts: 600 to 3,500
  - Price to pay for full autonomy (conservative parameters and thresholds)
- Reliance on extremity features
- Coarse correspondence
  - Can be refined (even correct local errors)
- Conflict between local vs. global
  - Deformation criterion not always intuitive



July 3, 2008, Copenhagen, Denmark

## A challenging case



## Lessons learned

- Non-rigid correspondence very difficult
  - Feature extraction
  - High-quality feature similarity helps!
    — **stretching/scaling** is the problem
  - Combinatorial complexity
  - Price to pay for large shape variations + partial matching



## Lessons learned

- Non-rigid correspondence very difficult
  - Feature extraction
  - High-quality feature similarity helps!
    — **stretching/scaling** is the problem
  - Combinatorial complexity
  - Price to pay for large shape variations + partial matching
- Is un-trained, fully automatic correspondence too much?
  - Incorporation of **prior knowledge**? How?



## Future works

- More robust local shape descriptors
  - **Feature-sensitive and part-aware** neighborhood traversal
- Incorporation of **prior knowledge**
- Any fresh idea for shape correspondence
  - Move away from existing optimization-based framework

## Acknowledgement

- Co-authors from three institutions:

  SFU, UBC, Tel-Aviv

- Funding: NSERC and MITACS
- Mesh models from ISDB and AIM@SHAPE
- Anonymous reviewers



Thank you!