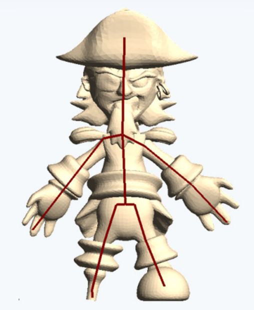
Predicting Animation Skeletons for 3D Articulated Models via Volumetric Nets

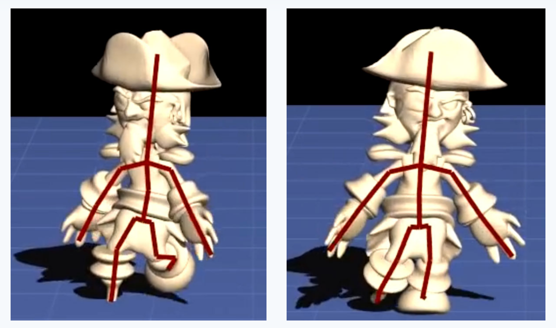
Zhan Xu — *UMass Amherst*
Yang Zhou — *UMass Amherst*
Evangelos Kalogerakis — *UMass Amherst*
Karan Singh — *University of Toronto*

Overview

Motivation: Jointed skeletons are compact shape representations that are useful for shape analysis, recognition, modeling, and animation.
Goal: Given a single input 3D model, our method produces an animation skeleton tailored for its structure, geometry and underlying part mobility.



Example of a 3D model and its animation skeleton

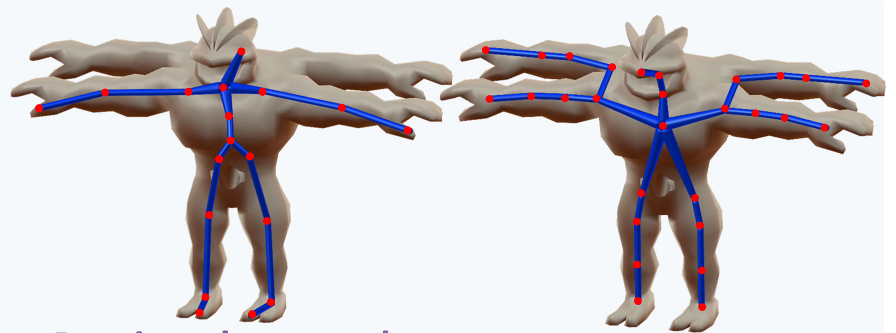


Animated model

[Baran et al. 2007]

Challenges: Predicting joints from a single shape snapshot without any extra information is under-constrained; A good skeleton should capture the mobility of underlying articulating parts; User control is often desirable to determine the level-of-detail of the output skeleton.

Earlier work: Fits pre-defined hand-crafted skeletal templates with fixed sets of joints to a 3D model. Limited to specific shape classes. Does not handle geometric and structural variability of input shapes.



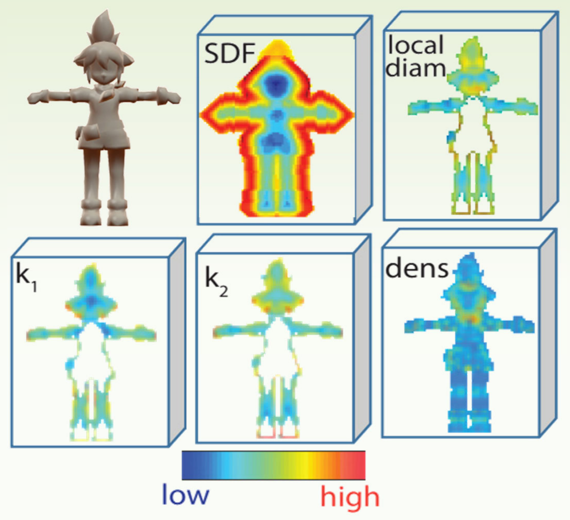
Previous best work (Pinocchio) — Our method

Our approach: a deep learning approach for predicting animation skeletons trained on a large repository of rigged 3D models.

Key ideas of our method:

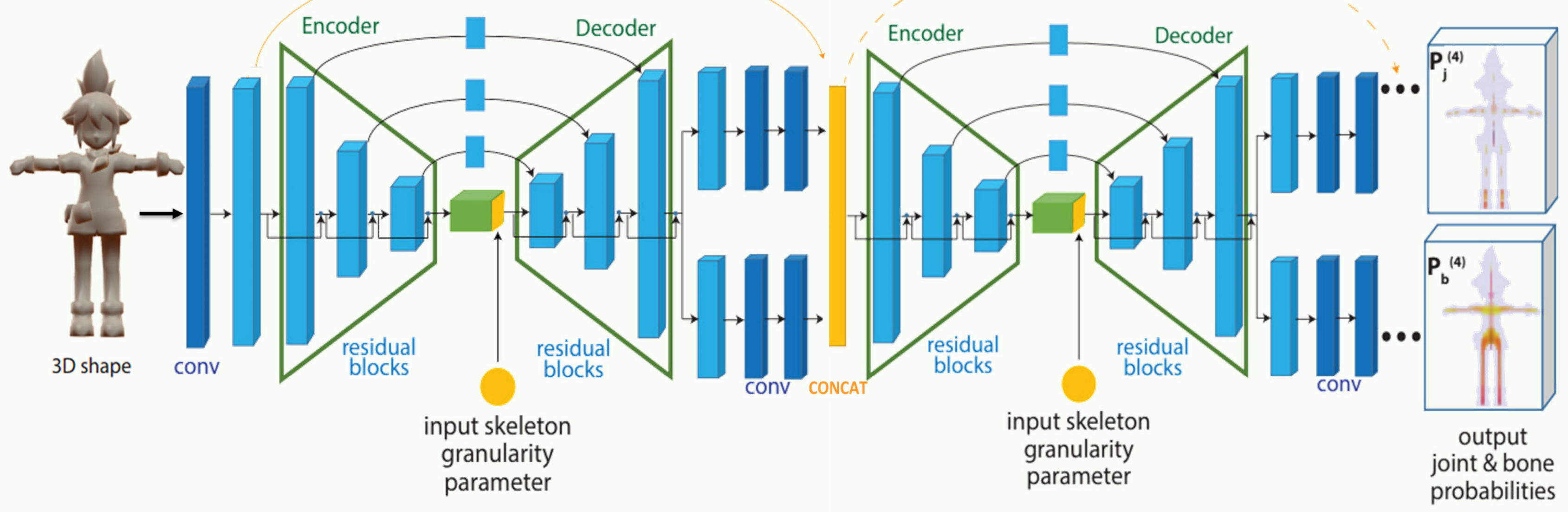
- **Simultaneously predicts joints & bones** through a 3D hourglass architecture.
- **Combines multiple geometric cues** including mesh, surface, volumetric features from the input 3D shape to predict an animation skeleton.
- Allows **user control** to synthesize skeletons with varying level-of-detail via a single, optional input parameter.
- Learns a **generic model**: our method extracts plausible skeletons for a large variety of input shapes, such as humanoids, quadrupeds, birds, fish, and so on.

Method



Input features: the shape is discretized into a 88^3 volumetric grid. Voxels store the Signed Distance Function (SDF) representation of the shape augmented with principal surface curvature information, local shape diameter, and mesh vertex density.

Architecture: stacked 3D hourglass module predicting joint and bone probabilities.



Training: we generate target joint & bone probabilities by placing a Gaussian distribution for each joint & bone voxel.

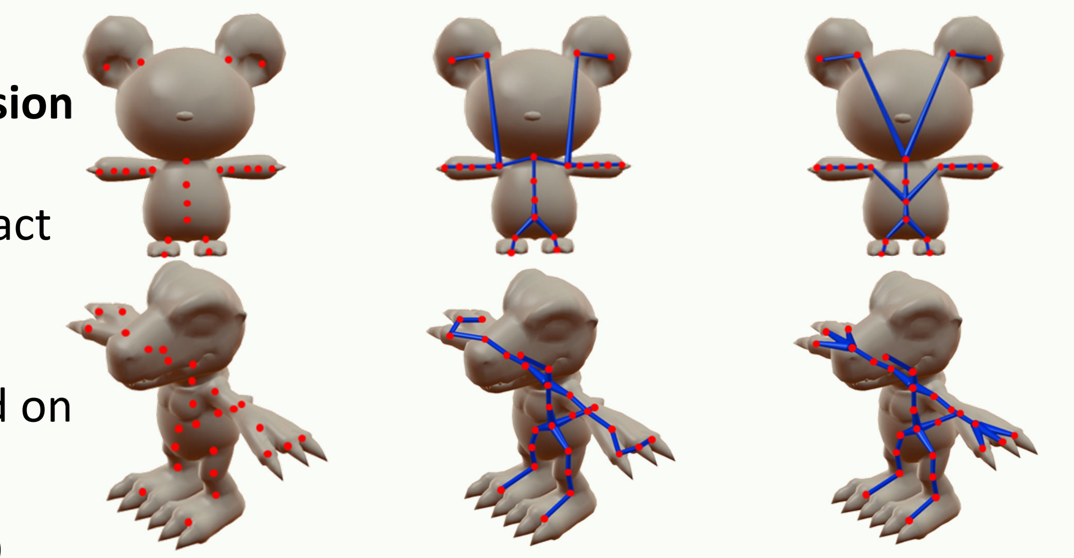
Loss:

$$L = \sum_{voxel\ v} M[v](L_j[v] + L_b[v])$$

L_j, L_b: cross entropies for joints & bones
M: mask (zero for voxels outside the shape)

Skeleton extraction:
(a) **non-maximum suppression** to obtain joints
(b) Prim's algorithm to extract a **minimum spanning tree** minimizing a cost function over candidate edges based on bone probabilities.

$$\text{cost}_{edge} = -\sum_{v \in edge} \log P_b(v)$$



predicted joints — MST with Euclidean distances as costs — MST using our bone probabilities

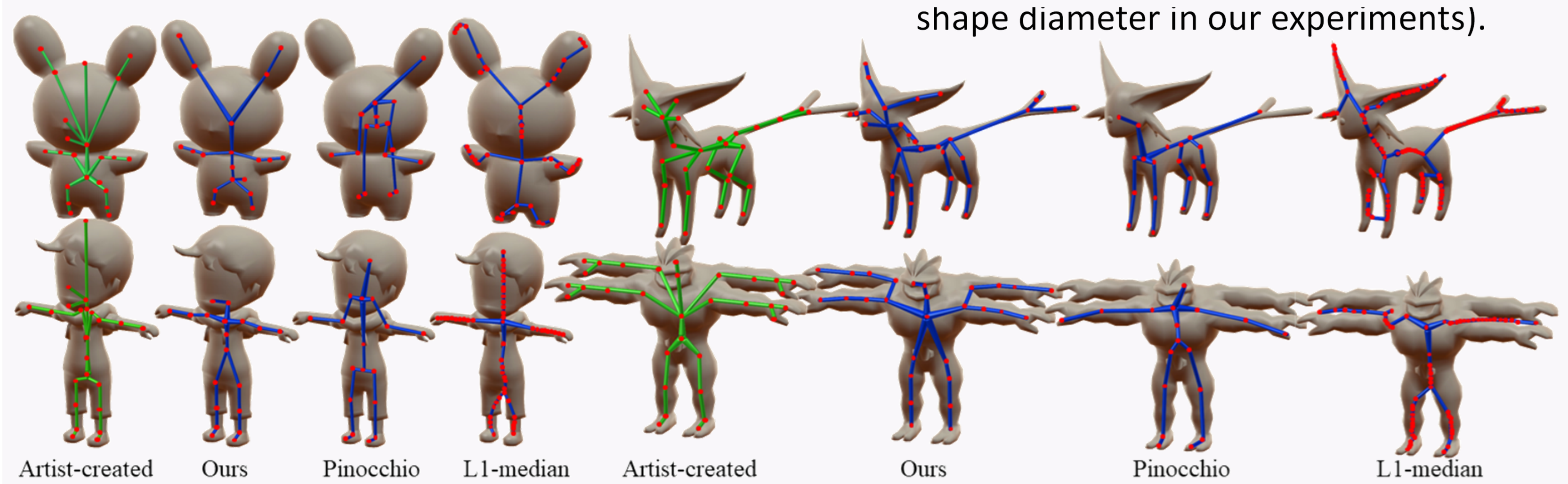
Results

Dataset: 3193 rigged models mined from the Models Resource repository
Average number of joints per character: 26.4
We split our dataset into 80% for training (2,554 models), 10% for hold-out validation (319 models), and 10% for testing (319 models).

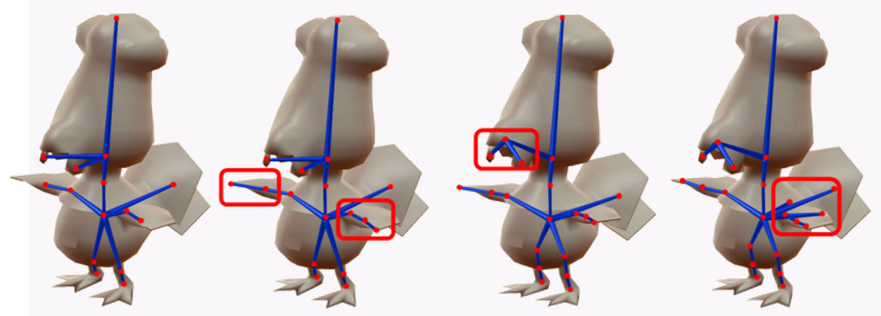
Method	CD-joint	CD-joint2bone	MR-pred	MR-ref
Pinocchio	7.4%	5.8%	55.8%	45.9%
L1-median	5.7%	4.4%	47.9%	63.2%
Ours	**4.6%**	**3.2%**	**62.1%**	**68.3%**

Quantitative evaluation for all methods

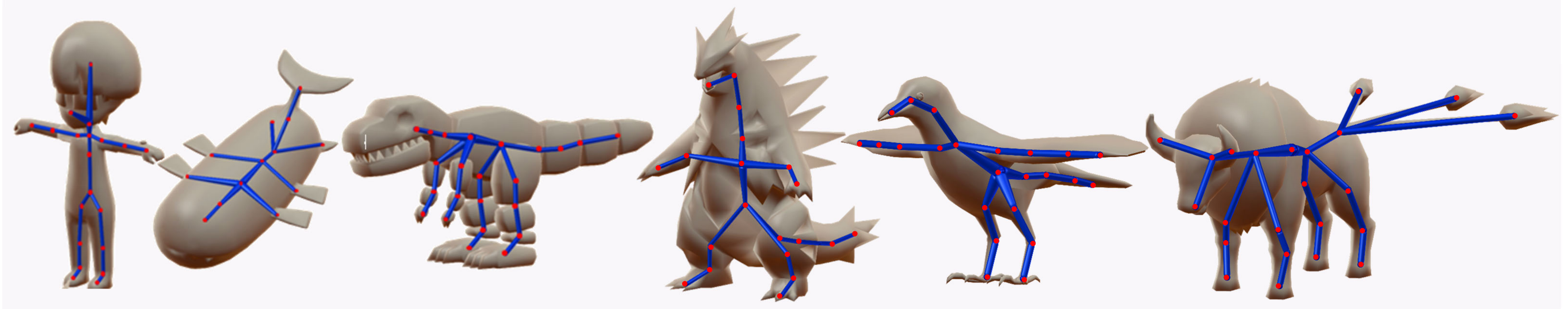
CD-joint/joint2bone: Chamfer distance between predicted joints and reference joints/bones.
MR-pred/ref: the percentage of predicted/reference joints whose distance to their nearest reference /predicted ones is lower than a prescribed tolerance (50% of local shape diameter in our experiments).



Comparisons of different methods for representative test characters. The green ones indicate the artist-created skeletons.



User control: Adjusting a single user parameter controls the granularity of the skeleton. Red boxes highlight changes in the output skeleton.



Results gallery

For paper, code and dataset, please visit our project page:
https://people.cs.umass.edu/~zhanxu/projects/AnimSkelVolNet/