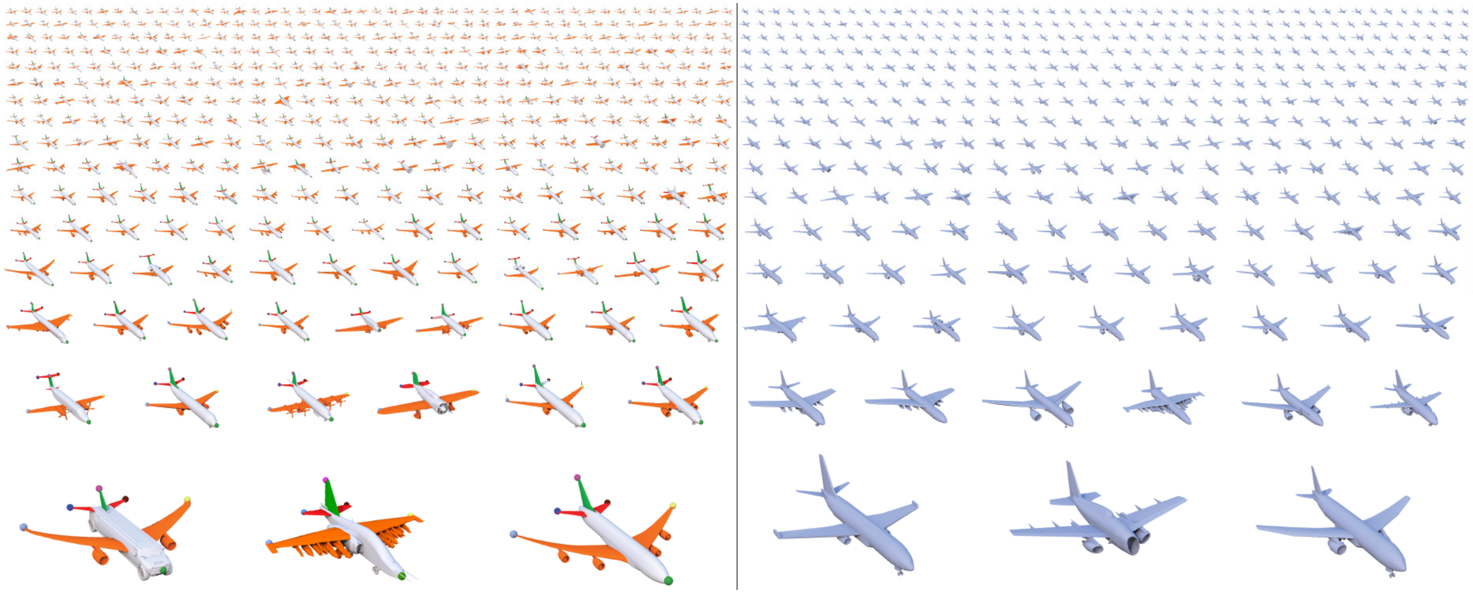
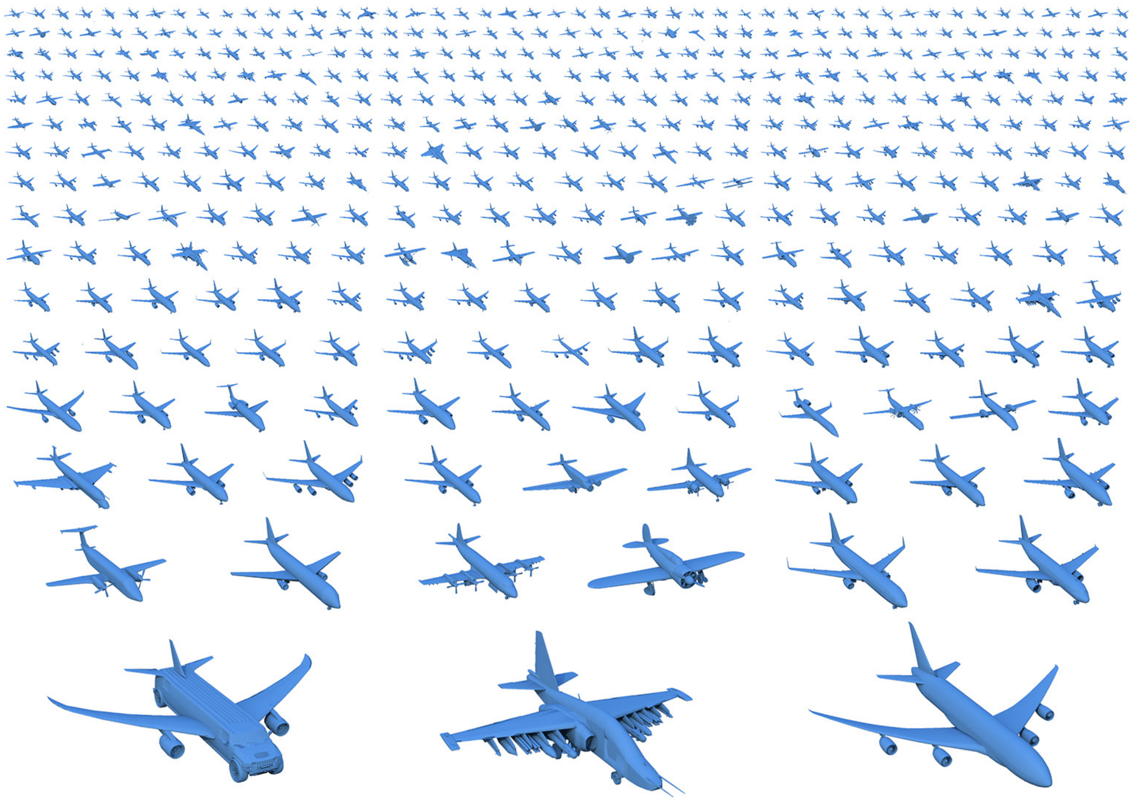


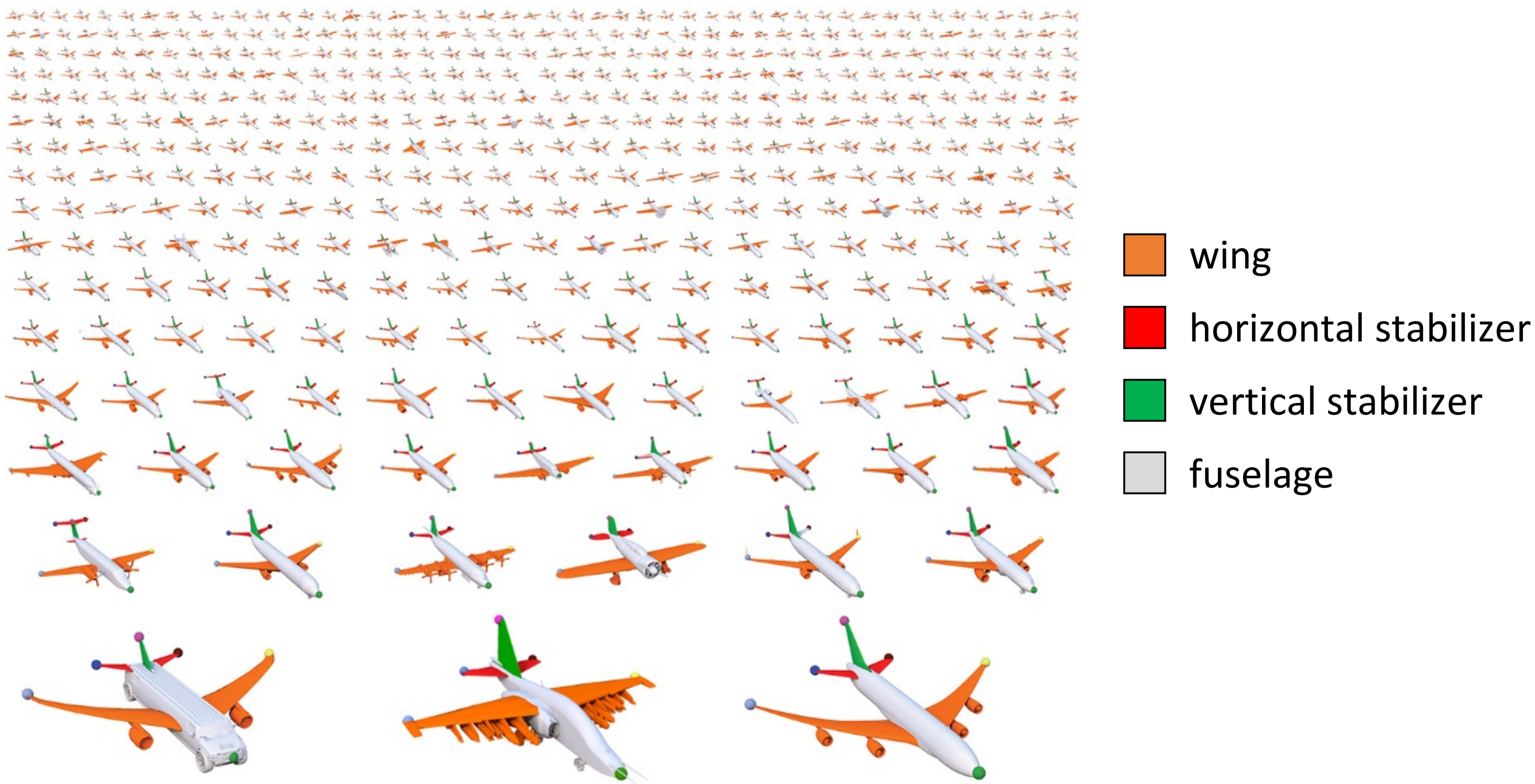
Analysis and Synthesis of 3D Shape Families via Deep-Learned Generative Models of Surfaces

Haibin Huang, Evangelos Kalogerakis, Benjamin Marlin
University of Massachusetts Amherst

Given an input 3D shape family…

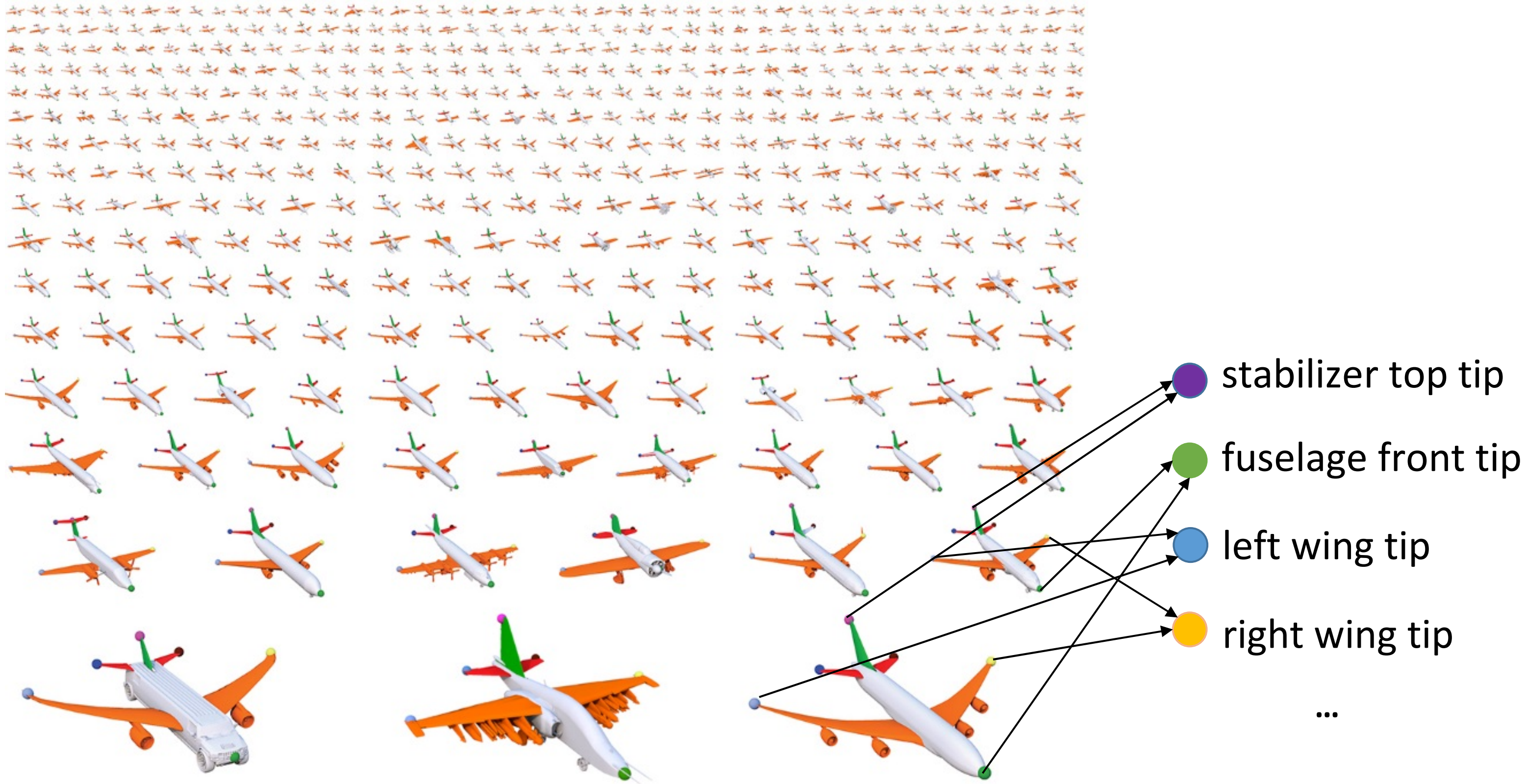


Goal 1/4: part segmentation & labeling



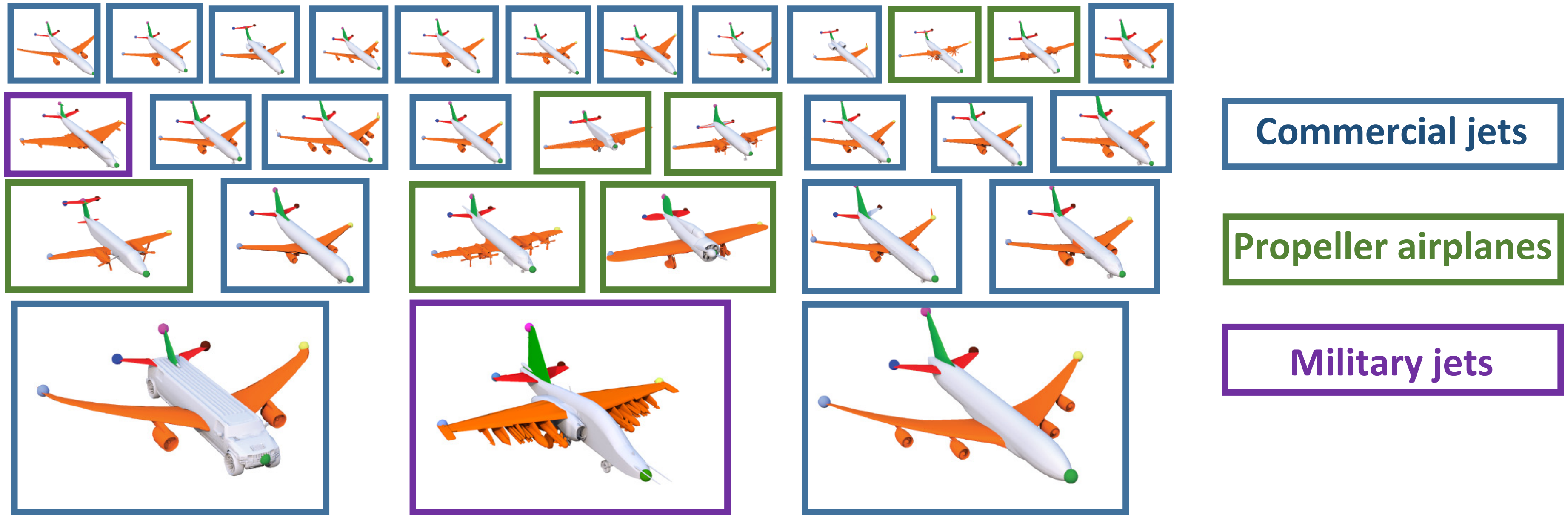
Applications: assembly-based modeling, manufacturing…

Goal 2/4: correspondences



Applications: texture transfer, exploration, morphing…

Goal 3/4: learn shape descriptors & fine-grained classification



Applications: text-based shape retrieval, database organization…

Goal 4/4: shape synthesis

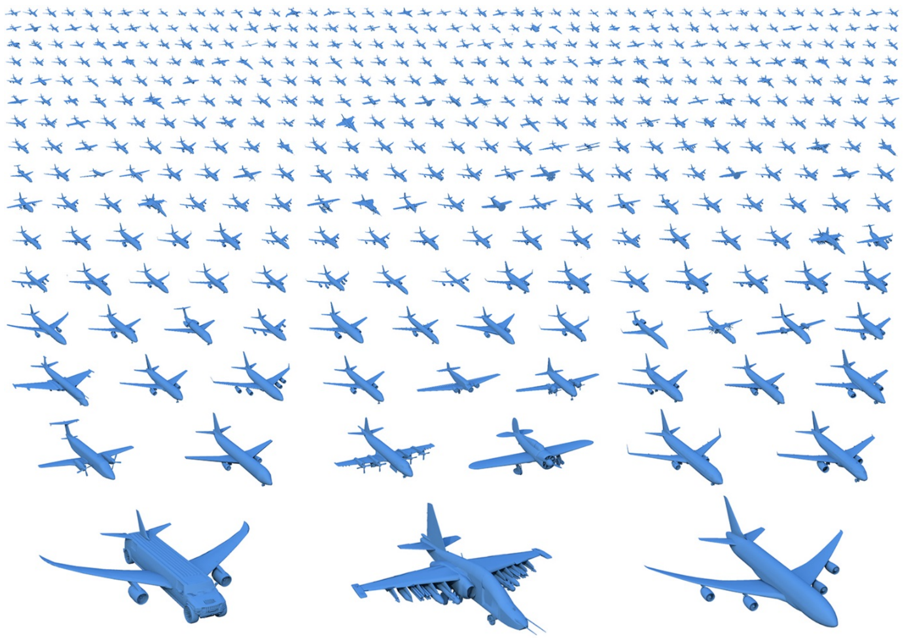
Applications: virtual worlds with lots of (infinite?) content variations,
training data for vision algorithms…

Challenges

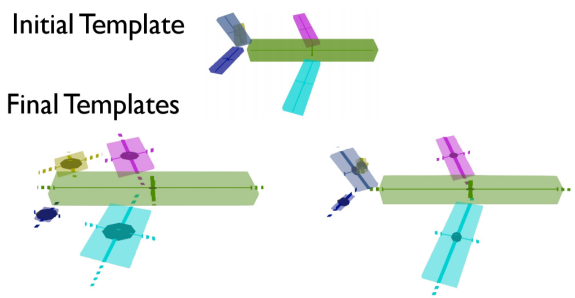
Shape families can have large **structural & geometric variability**.

All tasks **depend on each other**.

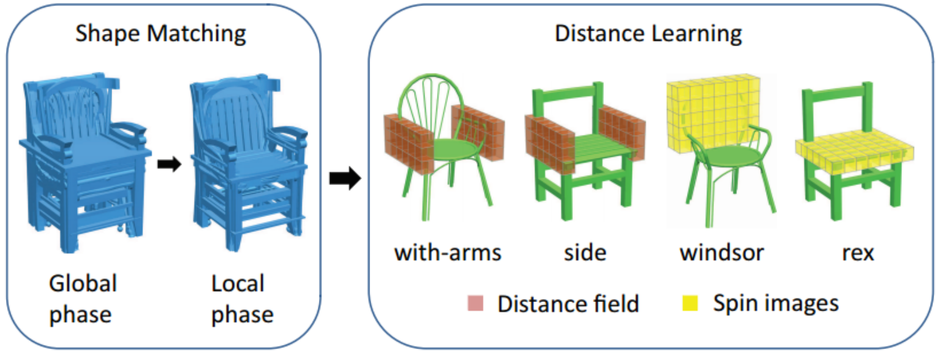
Generate **surface geometry**.



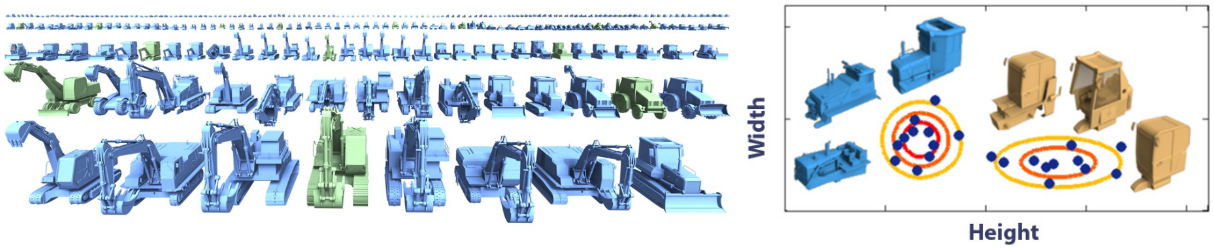
Related work: modeling structural & geometric variability



Box part templates
[Kim et al. 13]

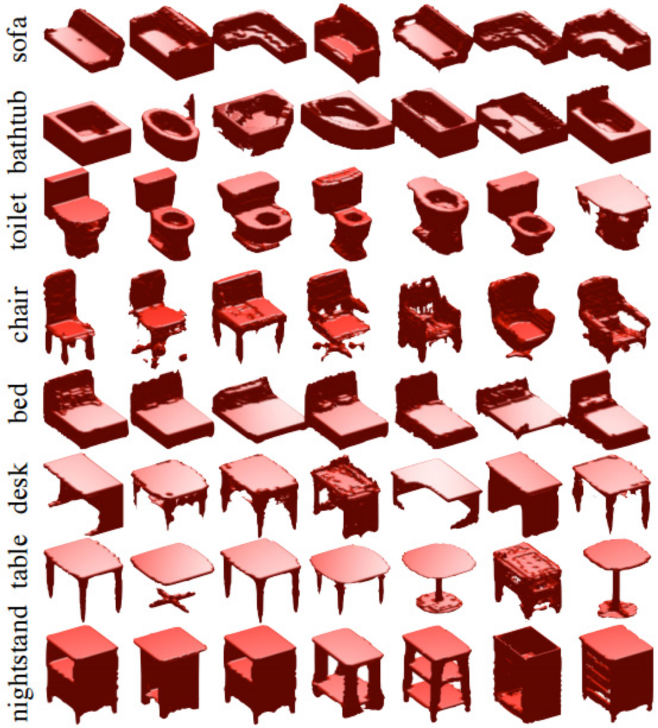


Non-rigid alignment / distance learning
[Huang et al. 13]



Probabilistic model for shape synthesis
[Kalogerakis et al. 12]

Related Work: deep learning



Deep representations for volumetric shapes
[Wu et al. 15]

Joint Analysis and Synthesis

Probabilistic Model for analysis:

- Learns template geometry and deformations
- Estimates fuzzy point correspondences
- Segments and labels parts

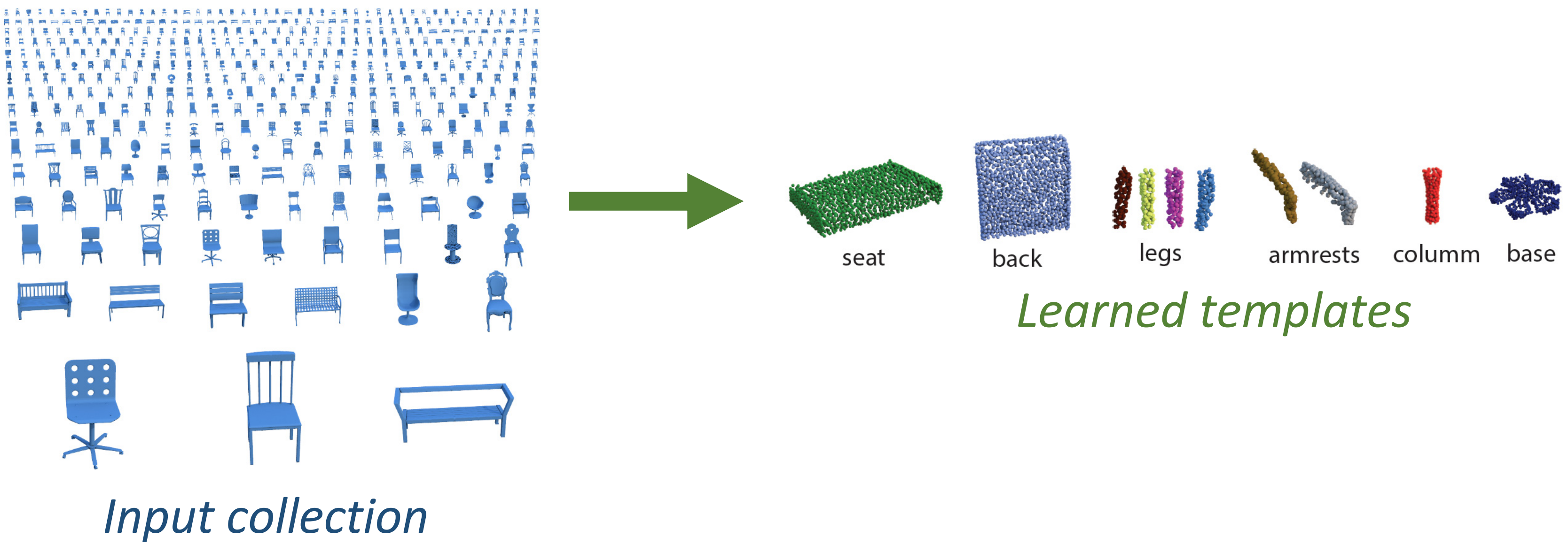
Deep Probabilistic Model for synthesis:

- Generates point-sampled surfaces
- Learns class-specific descriptors

Combine both models and jointly learn them from input collection

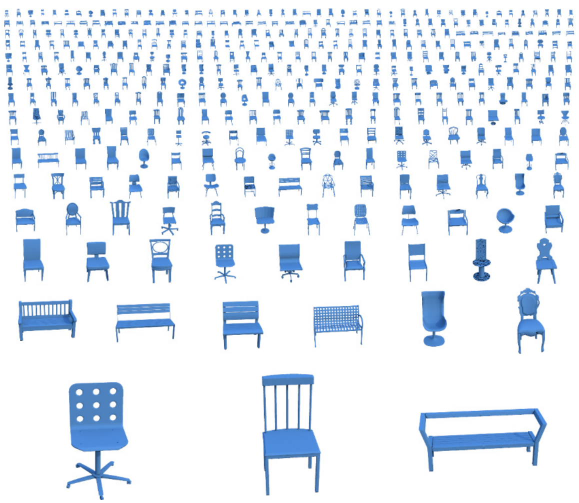
Probabilistic model

Given the input shapes, estimate assignments to random variables representing template geometry

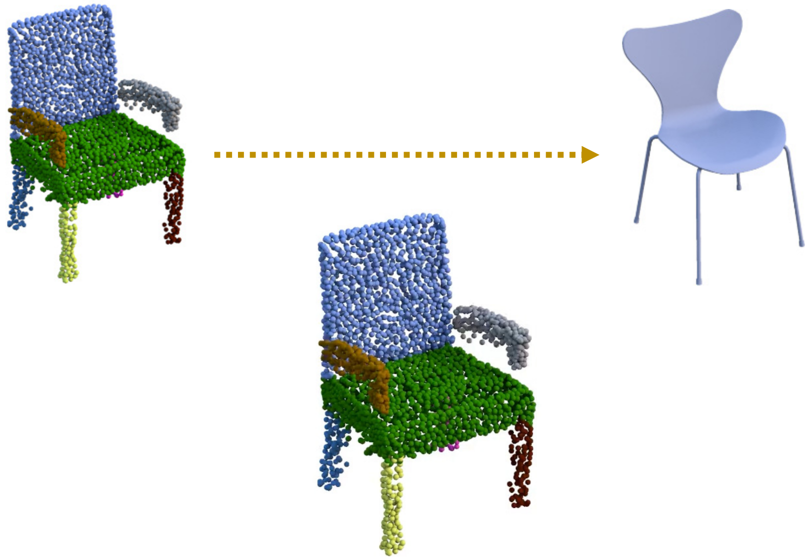


Probabilistic model

Given the input shapes, estimate assignments to random variables representing template geometry, template deformations for each shape



Input collection

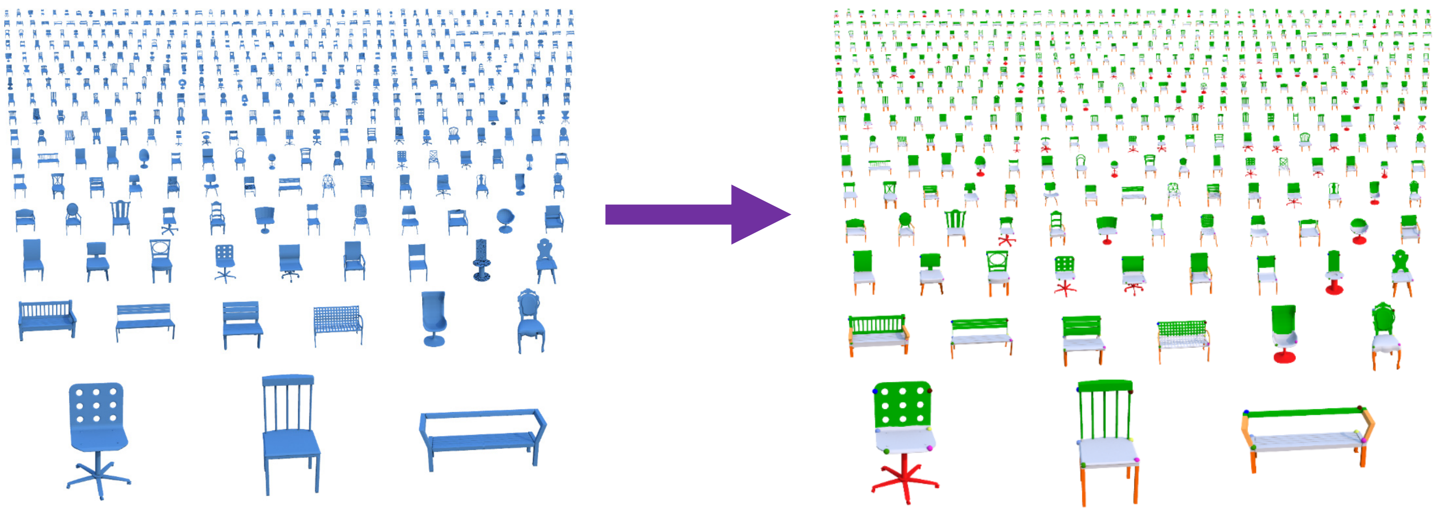


Non-rigid deformation

[video]

Probabilistic model

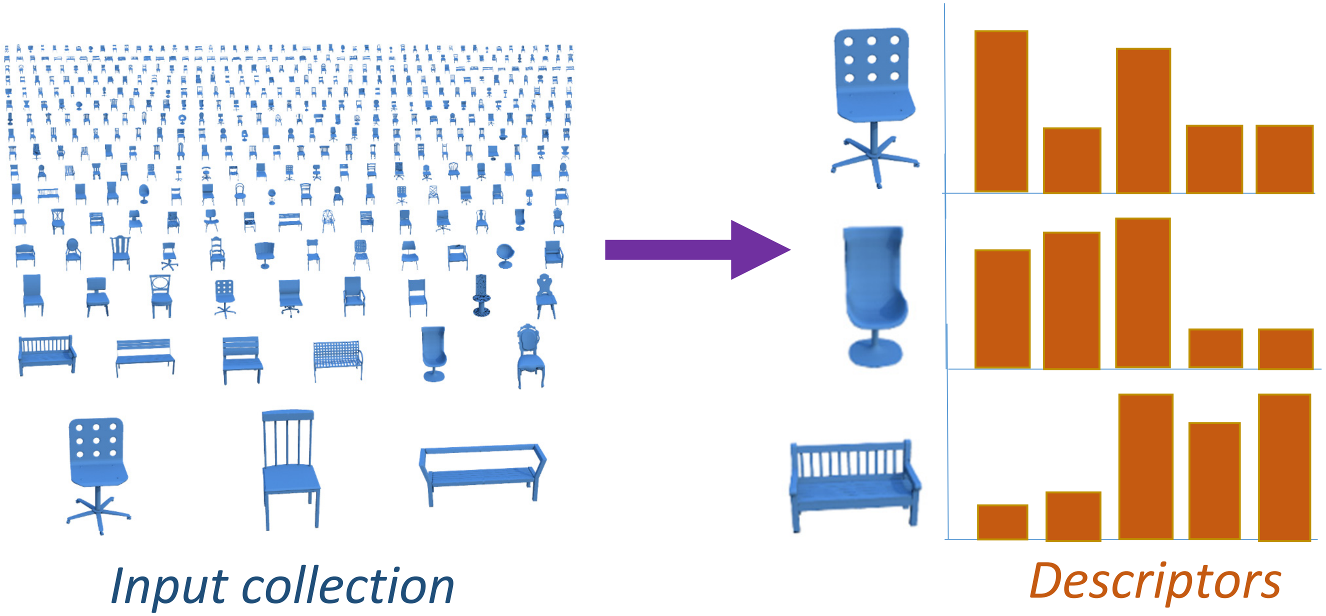
Given the input shapes, estimate assignments to random variables representing template geometry, template deformations for each shape, surface part labels and point labels for each shape



Input collection *Segmented & labeled shapes*

Probabilistic model

Given the input shapes, estimate assignments to random variables representing template geometry, template deformations for each shape, surface part labels and point labels for each shape, and descriptors for each shape.



Probabilistic model

Given the input shapes, estimate assignments to random variables representing template geometry, template deformations for each shape, surface part labels and point labels for each shape, and descriptors for each shape.

Maximize:

$$P_{crf}(\boldsymbol{Y}, \boldsymbol{D}, \boldsymbol{S}, \boldsymbol{U} \mid X) \cdot P_{bsm}(\boldsymbol{D}, \boldsymbol{H})$$

Probabilistic model

Given the input shapes, estimate assignments to random variables representing template geometry, template deformations for each shape, surface part labels and point labels for each shape, and descriptors for each shape.

Maximize:

$$\boxed{P_{crf}(\boldsymbol{Y}, \boldsymbol{D}, \boldsymbol{S}, \boldsymbol{U} \mid X)} \cdot P_{bsm}(\boldsymbol{D}, \boldsymbol{H})$$

Probabilistic model for analysis:

how likely are the shape & template attributes given each input surface geometry?

Probabilistic model

Given the input shapes, estimate assignments to random variables representing template geometry, template deformations for each shape, surface part labels and point labels for each shape, and descriptors for each shape.

Maximize:

$$P_{crf}(\boldsymbol{Y}, \boldsymbol{D}, \boldsymbol{S}, \boldsymbol{U} \mid X) \cdot \boxed{P_{bsm}(\boldsymbol{D}, \boldsymbol{H})}$$

"Generative" probabilistic model:

How likely is the output deformed geometry?

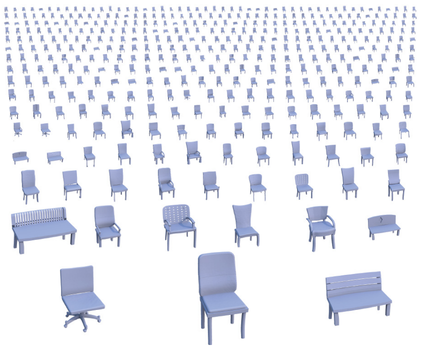
Ensures consistency within the input collection

Probabilistic model

Given the input shapes, estimate assignments to random variables representing template geometry, template deformations for each shape, surface part labels and point labels for each shape, and descriptors for each shape.

Maximize:

$$P_{crf}(\boldsymbol{Y}, \boldsymbol{D}, \boldsymbol{S}, \boldsymbol{U} \mid X) \cdot \boxed{P_{bsm}(\boldsymbol{D}, \boldsymbol{H})}$$



New Collection

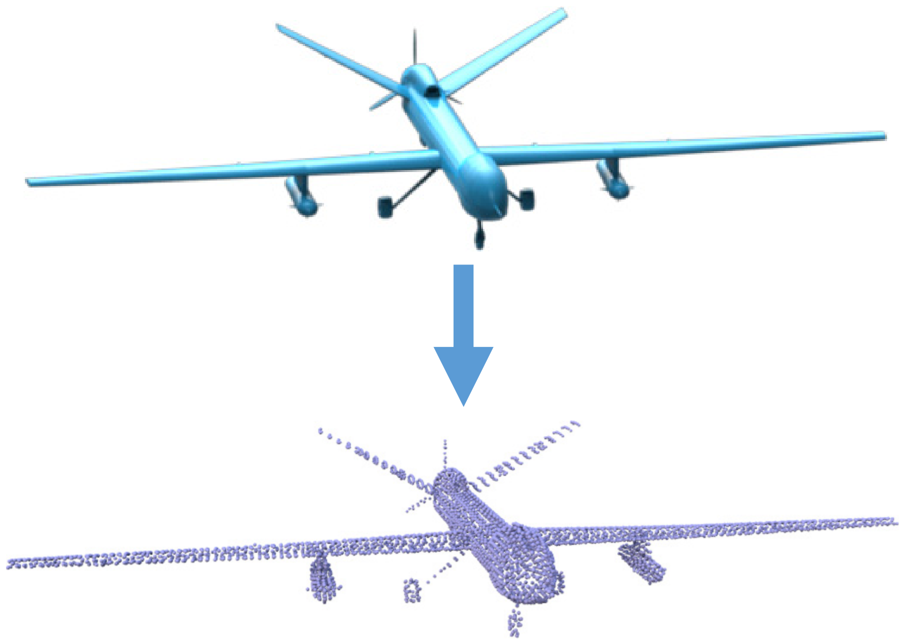
"Generative" probabilistic model:

How likely is the output deformed geometry?

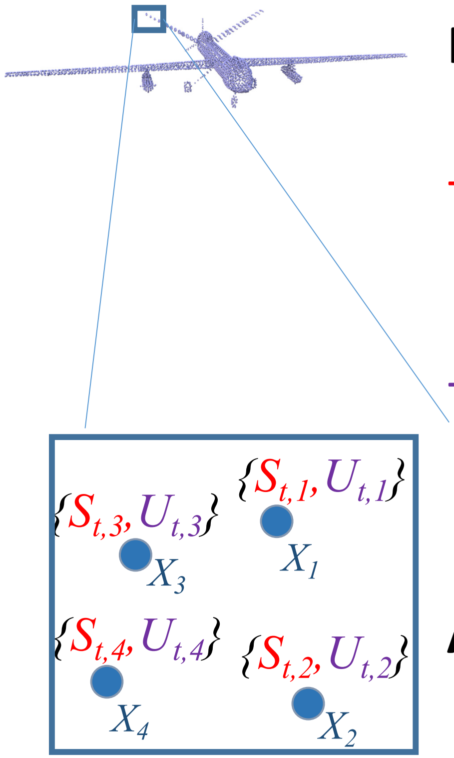
Ensures consistency within the input collection

Input shape representation

Each input surface is **uniformly sampled:**



Surface random variables



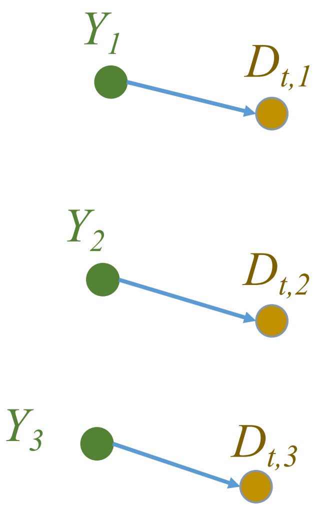
For each surface point p of shape t with position $X_{t,p}$:

- Part label: $S_{t,p}$
  {fuselage, left wing, right wing…}

- Point label: $U_{t,p}$
  {left wing tip, front fuselage tip …}

Assignments are probabilistic ("fuzzy" correspondences).

Template Random variables



For each template point k:

- Template point position: Y_k
  {point position in 3D space}

- Deformed template point position for each input shape t: $D_{t,k}$
  {point position in 3D space}

Template geometry will be estimated.

Factors

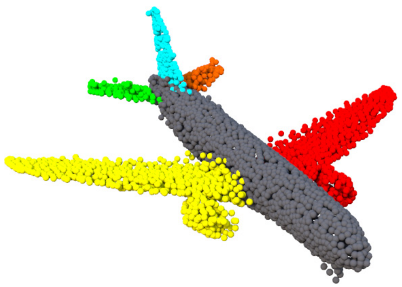
All these random variables **are not independent** of each other!

The probabilistic model is defined through **factors** over variable assignments.

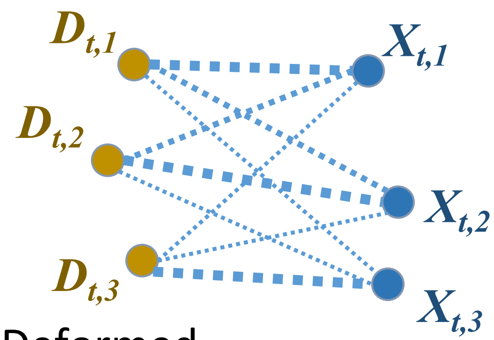
The higher the factor values, **the more compatible** the assignments are.

Unary deformation factor

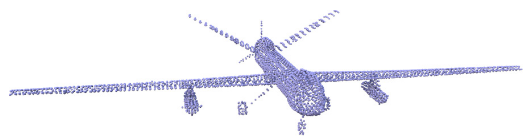
Encourage template points deform towards corresponding, closeby surface points.



Learned
template surface



Deformed
template surface



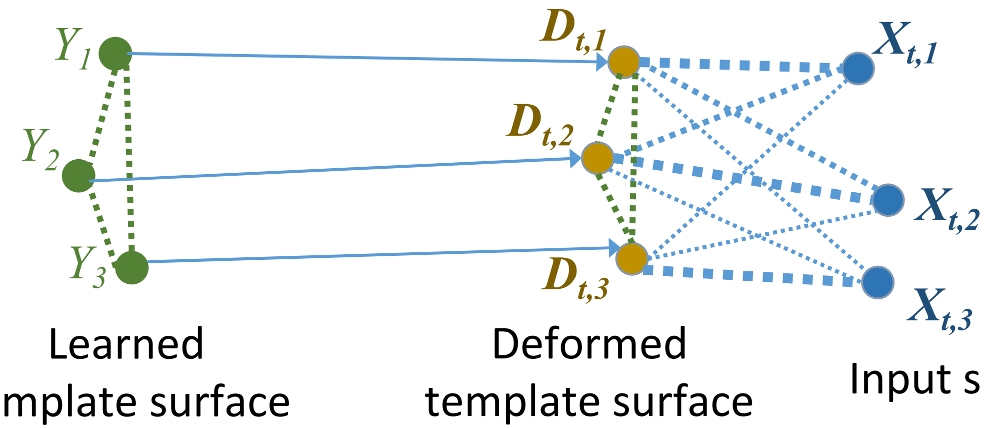
Input shape surface t

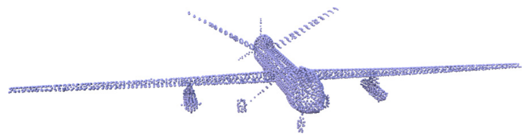
$$\varphi_1(\mathbf{D}_{t,k},\ U_{t,p}=k,\ \mathbf{X}_{t,p})=exp\{-.5[\mathbf{D}_{t,k}-\mathbf{X}_{t,p}]^T\,\boldsymbol{\Sigma}_1^{-1}\,[\mathbf{D}_{t,k}-\mathbf{X}_{t,p}]\}$$

Learned covariance matrix

Deformation smoothness factor

Encourages smoothness in the deformations of the template parts.



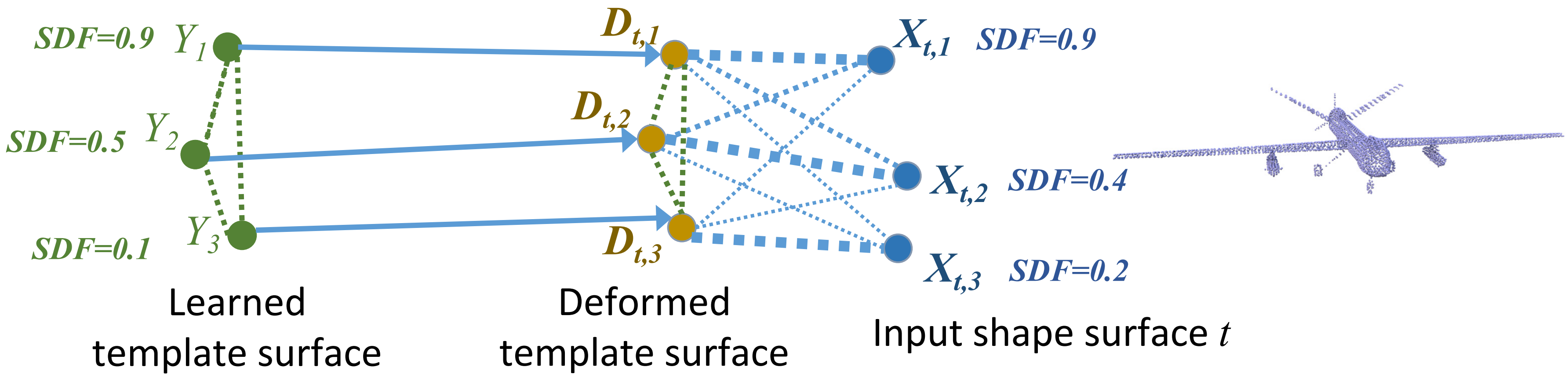


$$\varphi_2(D_{t,k}, D_{t,k'}, Y_k, Y_{k'}) = exp\{-.5[(D_{t,k}-D_{t,k'}) - (Y_k-Y_{k'})]^T \Sigma_2^{-1} [(D_{t,k}-D_{t,k'}) - (Y_k-Y_{k'})]\}$$

Learned covariance matrix

Correspondence factor

Encourage correspondences between template and surface points with similar geometry

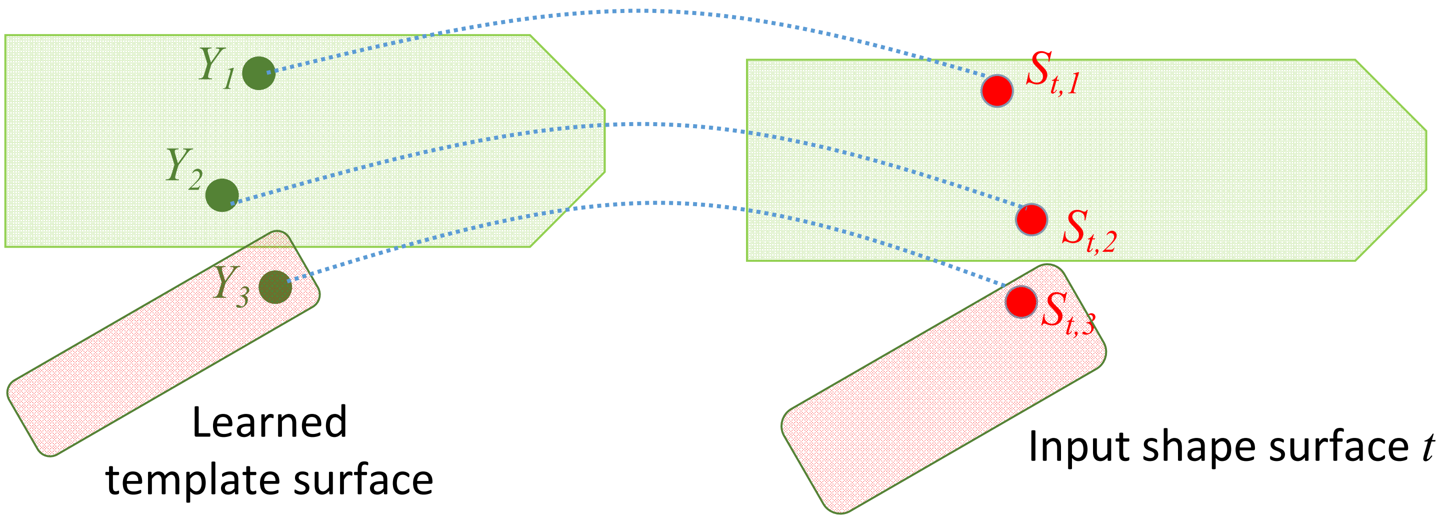


$$\varphi_3(U_{t,p}=k)=exp\{-.5[f_k - f_{t,p}]^T \Sigma_3^{-1} [f_k - f_{t,p}]\}$$

Learned covariance matrix

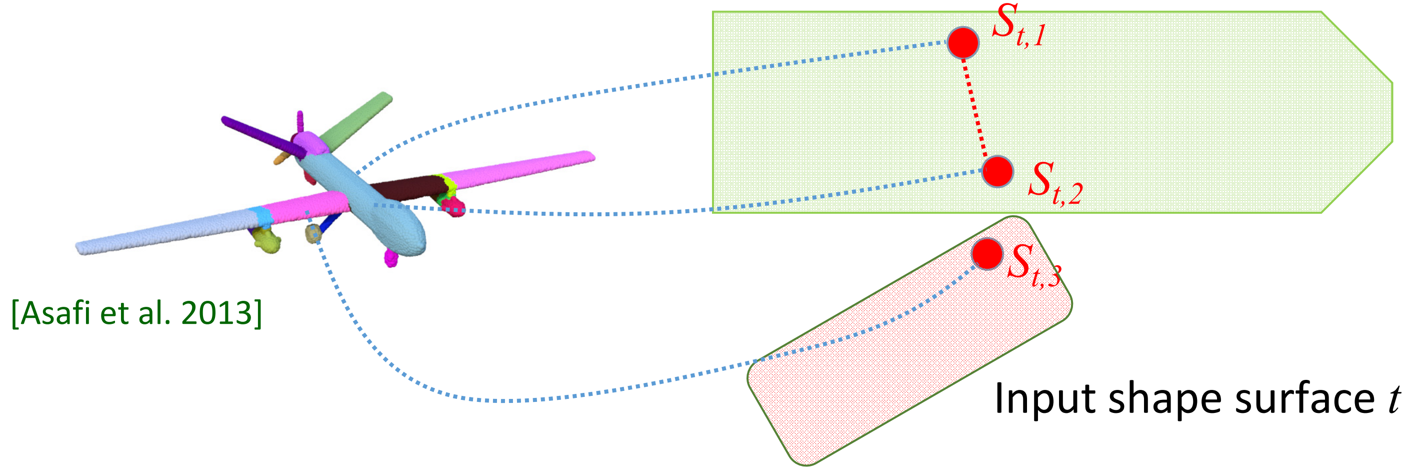
Segmentation factor

Encourage consistent part labels between corresponding template and surface points

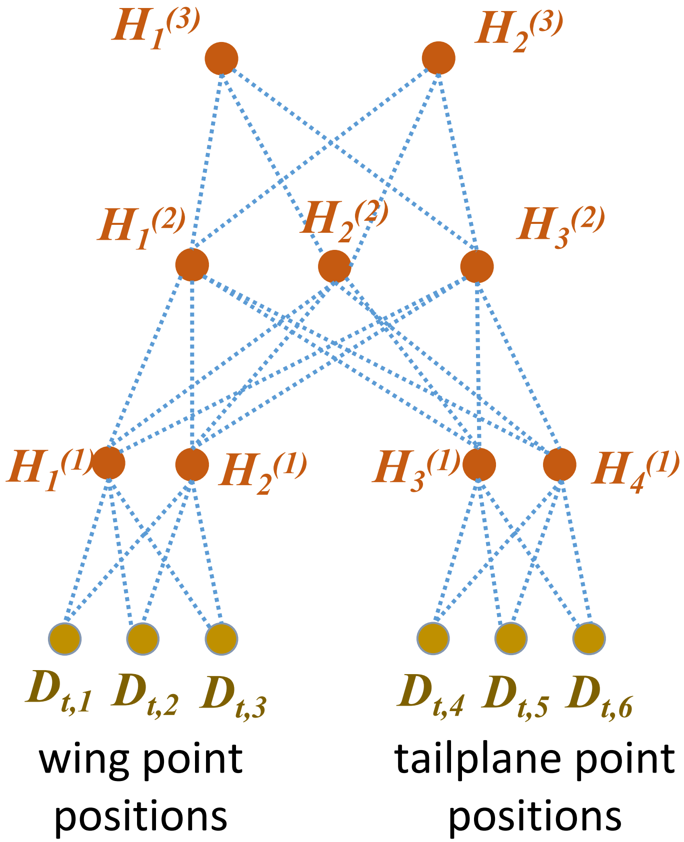


Segmentation smoothness factor

Encourage surface points within convex patches and similar geometry to have similar part label.



Generative model

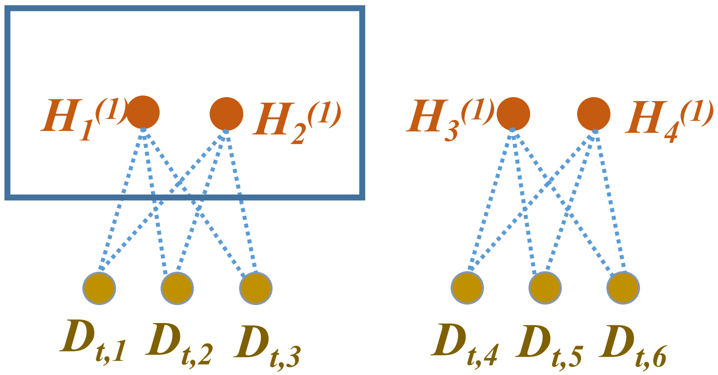


Generative model

wing point positions

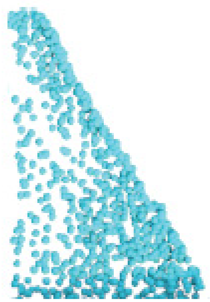
tailplane point positions

Generative model

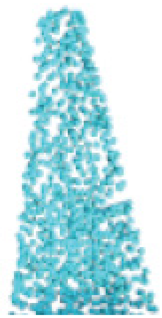


wing point positions

tailplane point positions

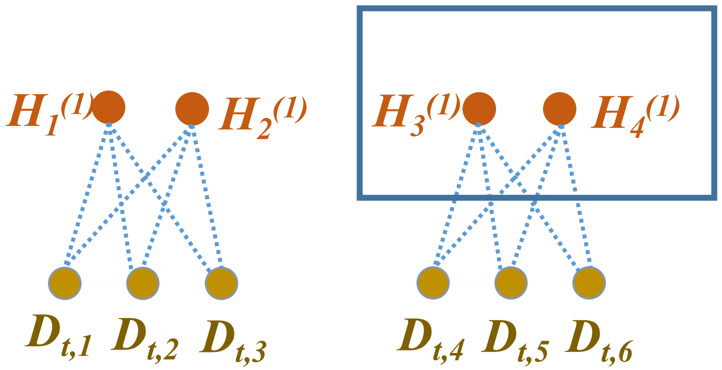


delta-shaped wing



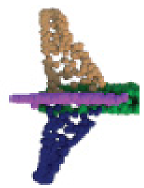
straight wing

Generative model



wing point positions

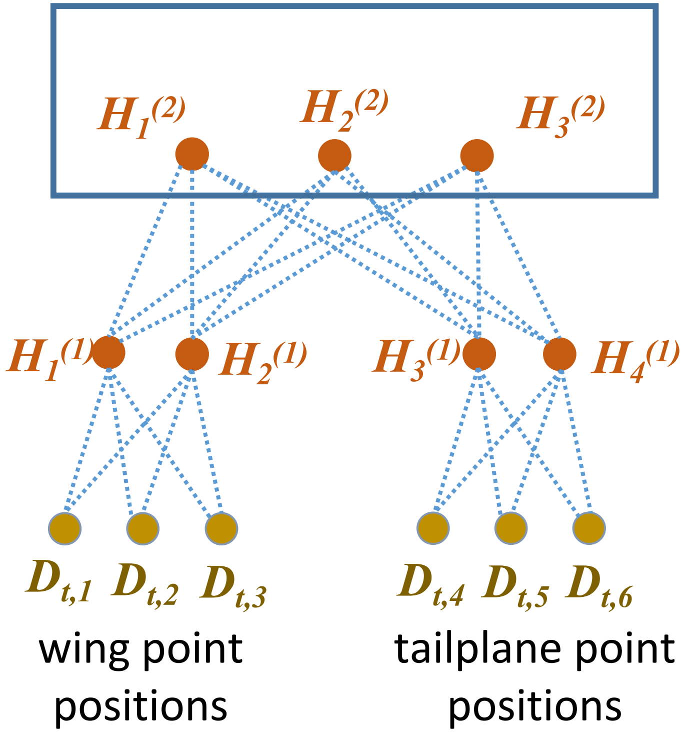
tailplane point positions

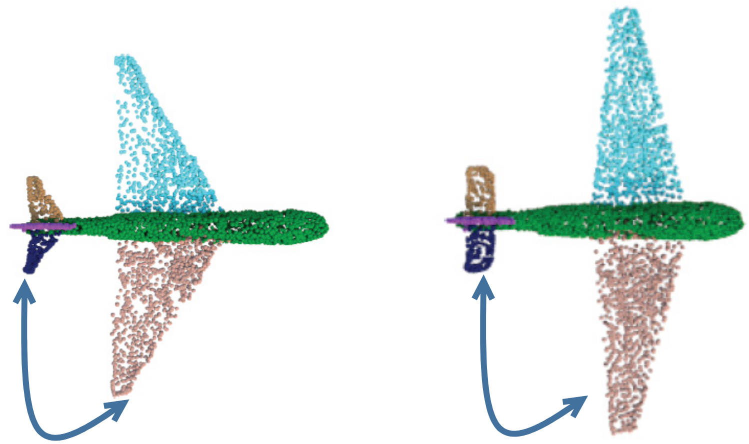


swept tailplanes

straight tailplanes

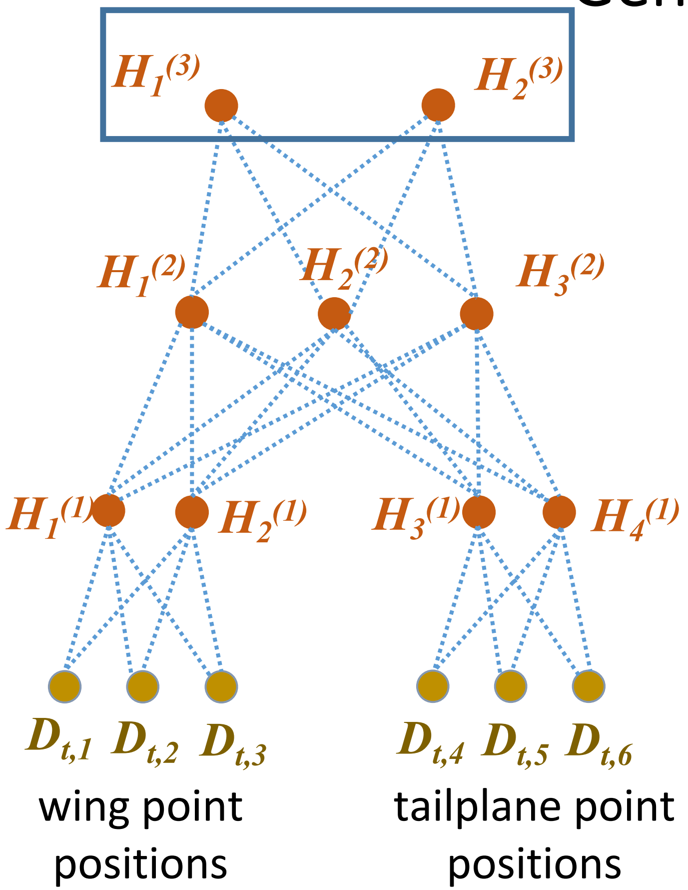
Generative model

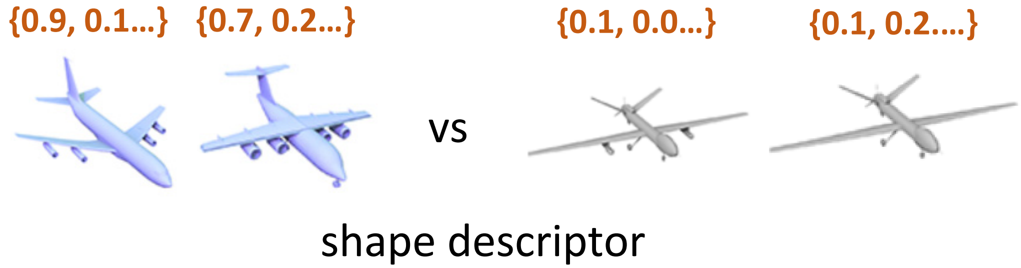




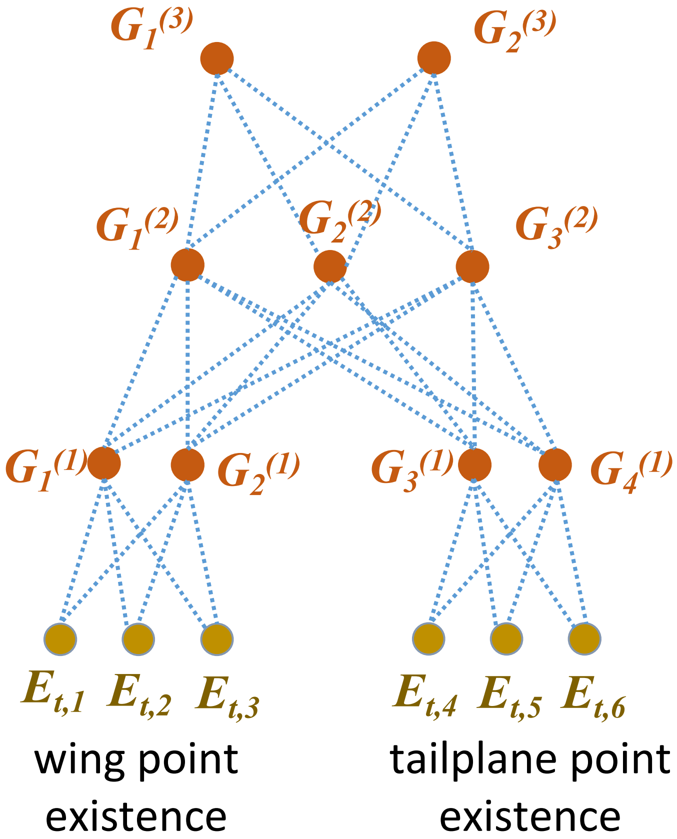
correlations between part configurations

Generative model





Generative model



Inference / parameter learning

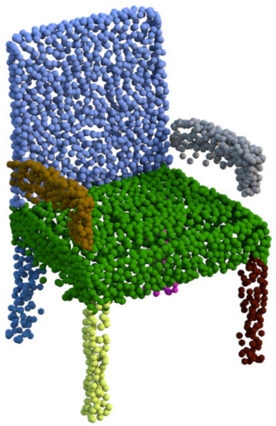
Parameter learning:

- **Piecewise training** for covariance matrices of the analysis part (12 parameters)
- **Contrastive divergence** for the generative model parameters (millions)

Mean-field inference:

- **Efficient** compared to other forms of inference
- **Convergence** properties
- **Approximate**
- **Requires initialization** (template parts from co-segmented shapes – we use [Kim et al., 2013] to initialize the segmentations)

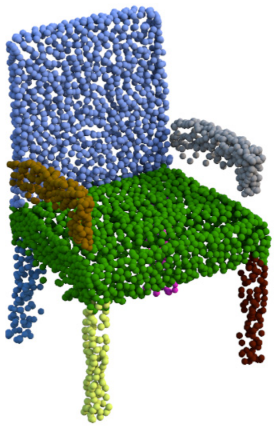
Deformation examples



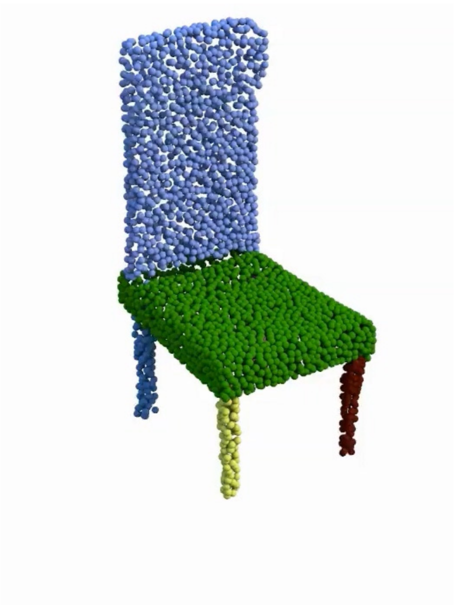
Template surface

Input surface

Deformation examples



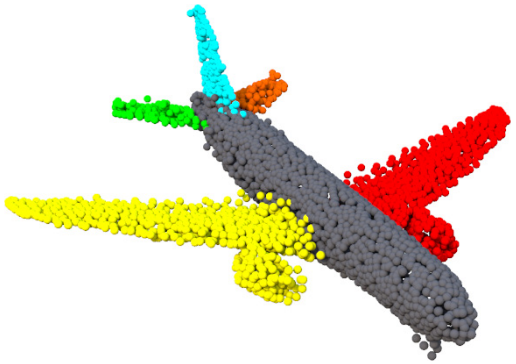
Template surface



Input surface

[video]

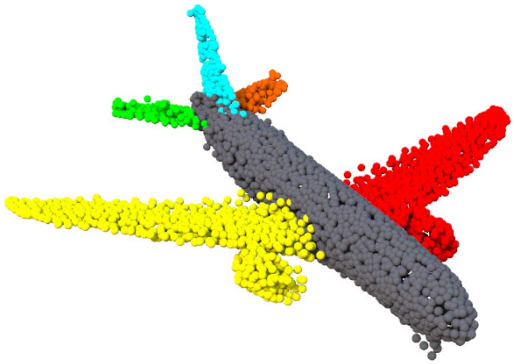
Deformation examples



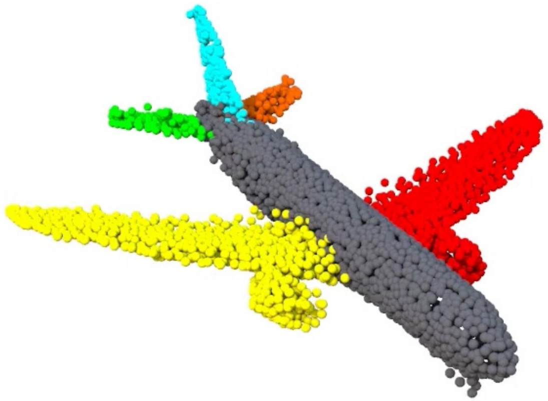
Template surface

Input surface

Deformation examples



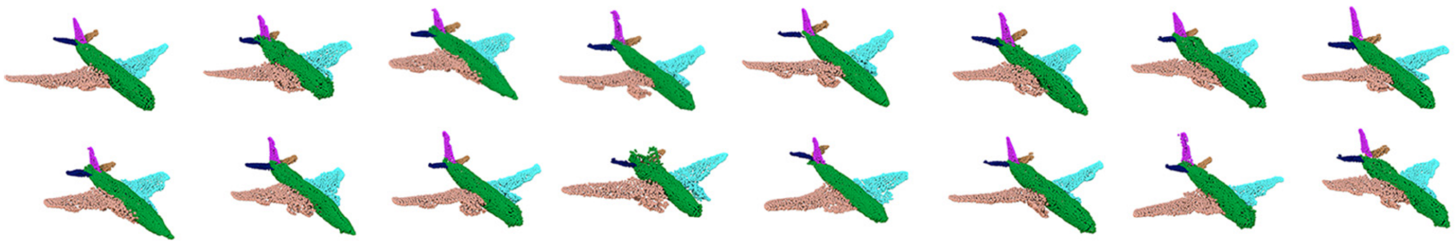
Template surface



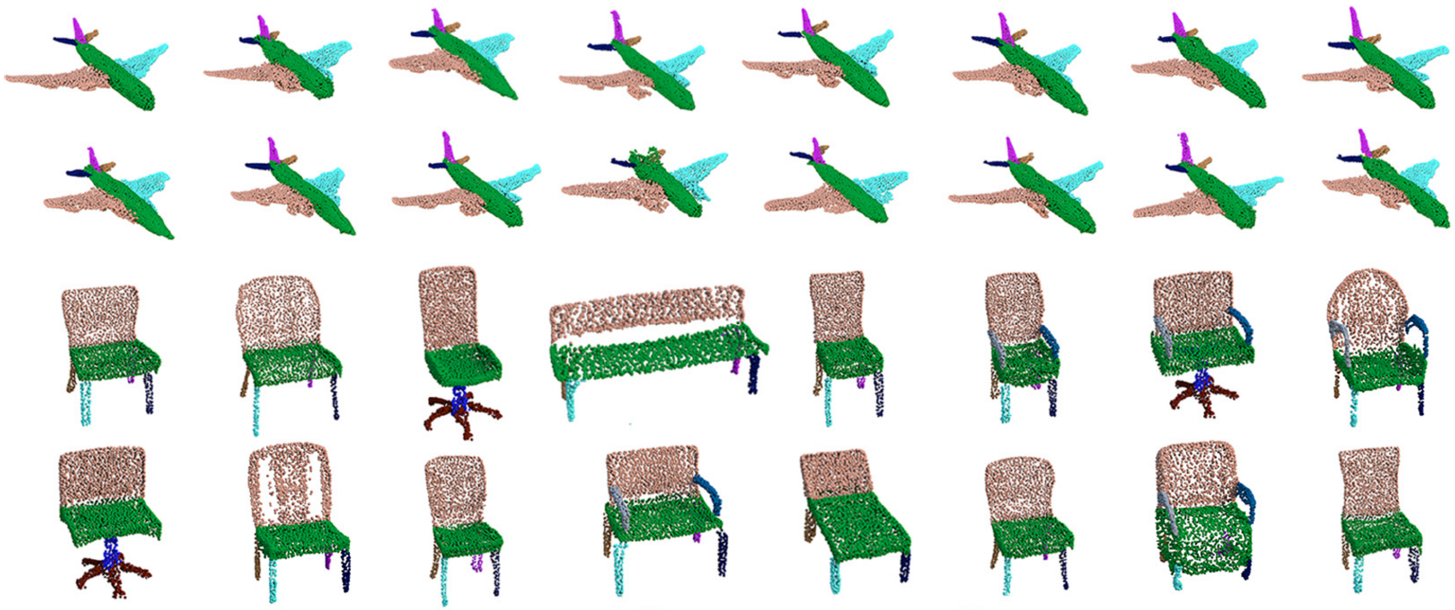
Input surface

[video]

Sampled shapes



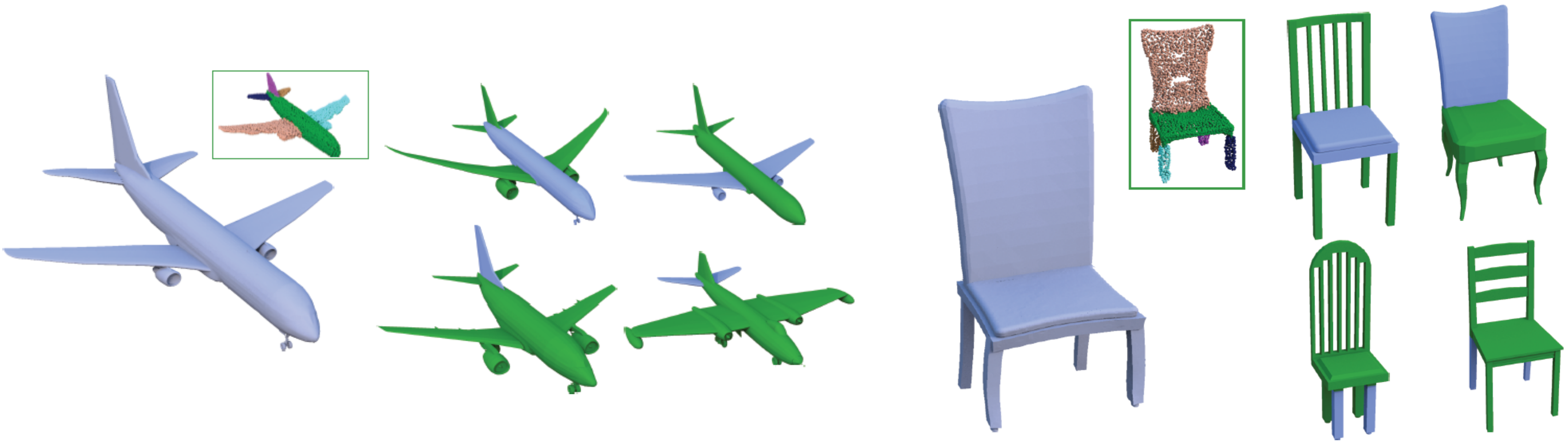
Sampled shapes



Sampled shapes

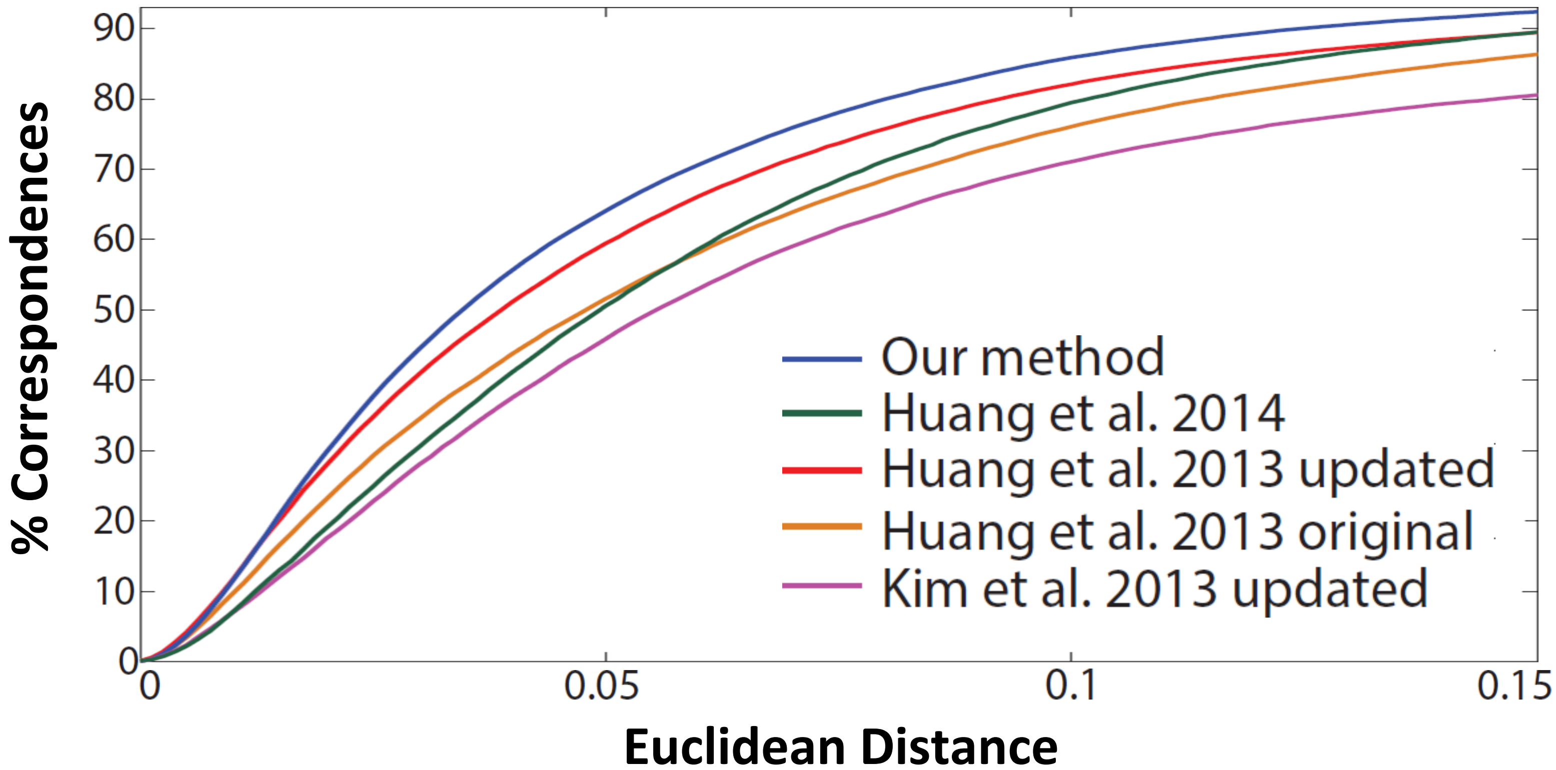
Assembly-based modeling + deformation

(Embedded deformation for shape manipulation, Sumner et al., 2007)



Correspondence quality – BHCP dataset

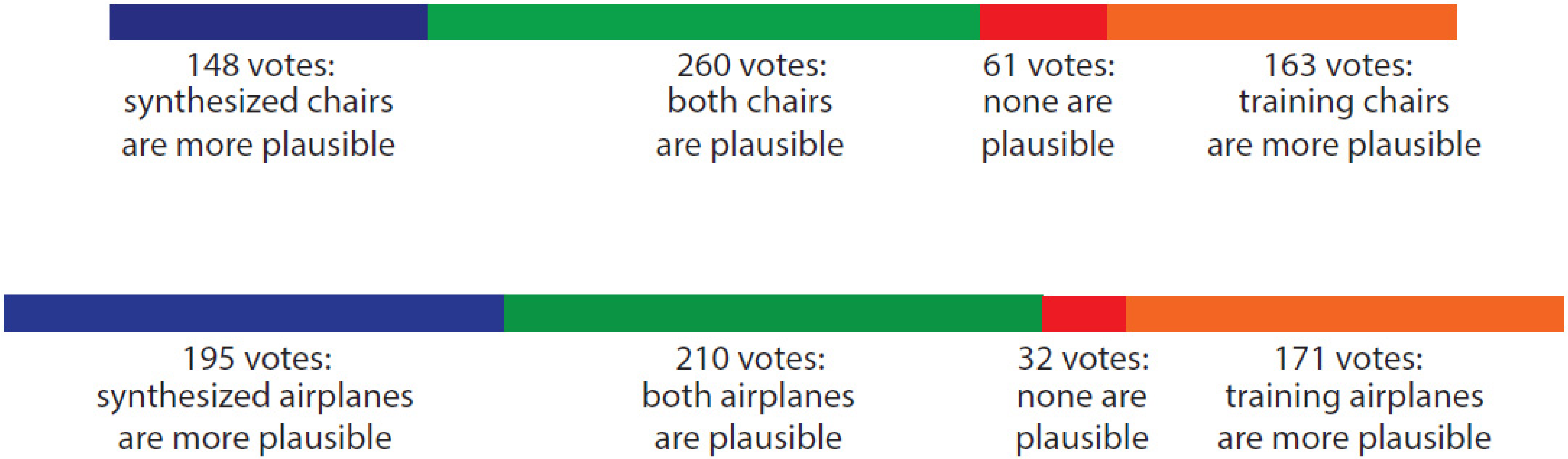
18.2% more correct predictions within .05 distance than Kim et al.'s 2013 method.



Segmentation quality

Category (Dataset)	Num. shapes	Kim et al. (our init.)	Our method
Bikes (BHCP)	100	76.8	82.3
Chairs (BHCP)	100	81.2	86.8
Helicopters (BHCP)	100	80.1	87.4
Planes (BHCP)	104	85.8	89.6

MTurk User study: plausibility of synthesized shapes



Summary

- **Joint** segmentation, correspondence, non-rigid deformation, descriptor learning, shape synthesis
- **Analysis-by-synthesis**
- **Deep learning** for modeling shape surfaces

Future Work

- **Full shape synthesis** from scratch without assembly
- **Reconstruction**, link descriptor to other forms of user input
- More **robust and automatic initialization schemes**
- **What is the best input shape representation?**

Thank you!

Acknowledgements: *Qi-xing Huang, Vladimir Kim, Siddhartha Chaudhuri*
Szymon Rusinkiewicz, anonymous reviewers

Our project web page (code, data, supp. material):

http://people.cs.umass.edu/~hbhuang/publications/bsm/index.html