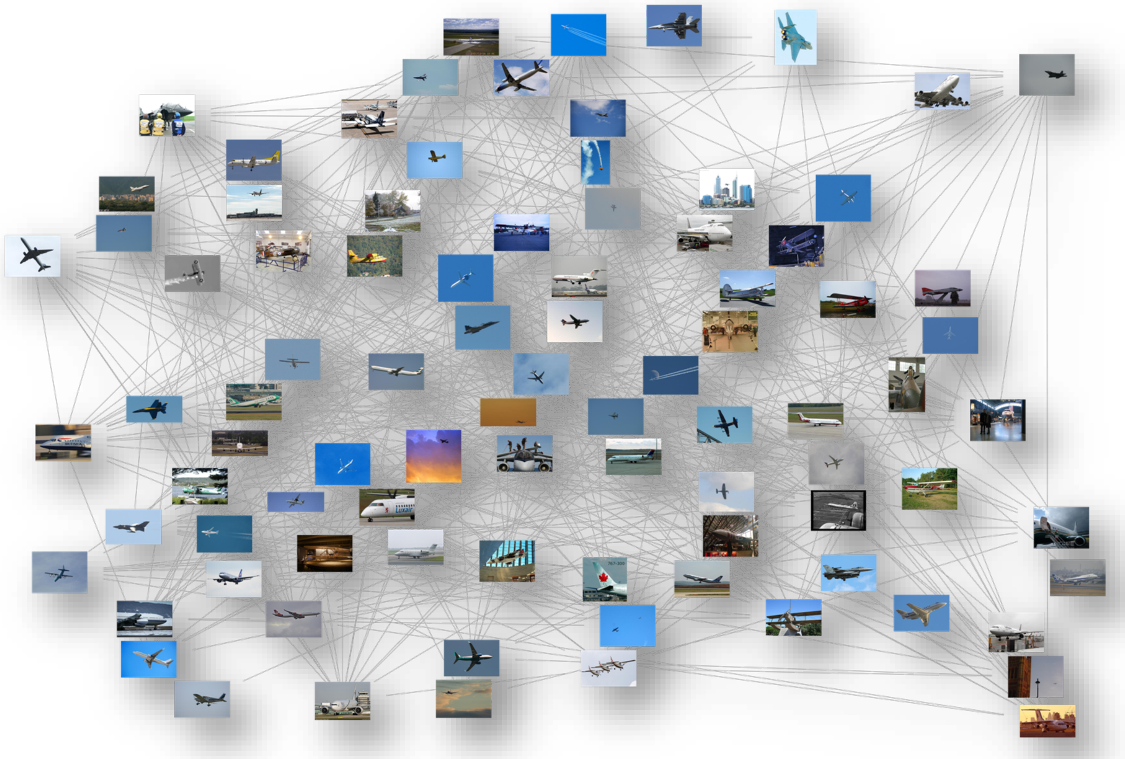
Functional Map Synchronization and Applications

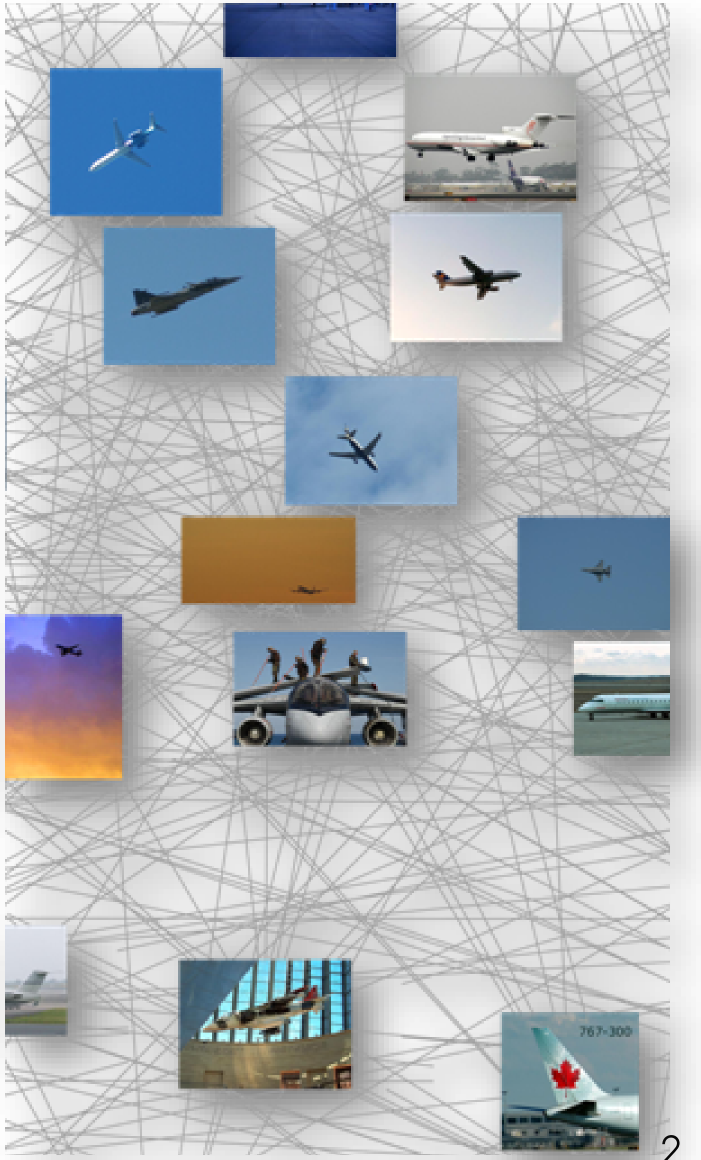


Leonidas Guibas
Stanford University

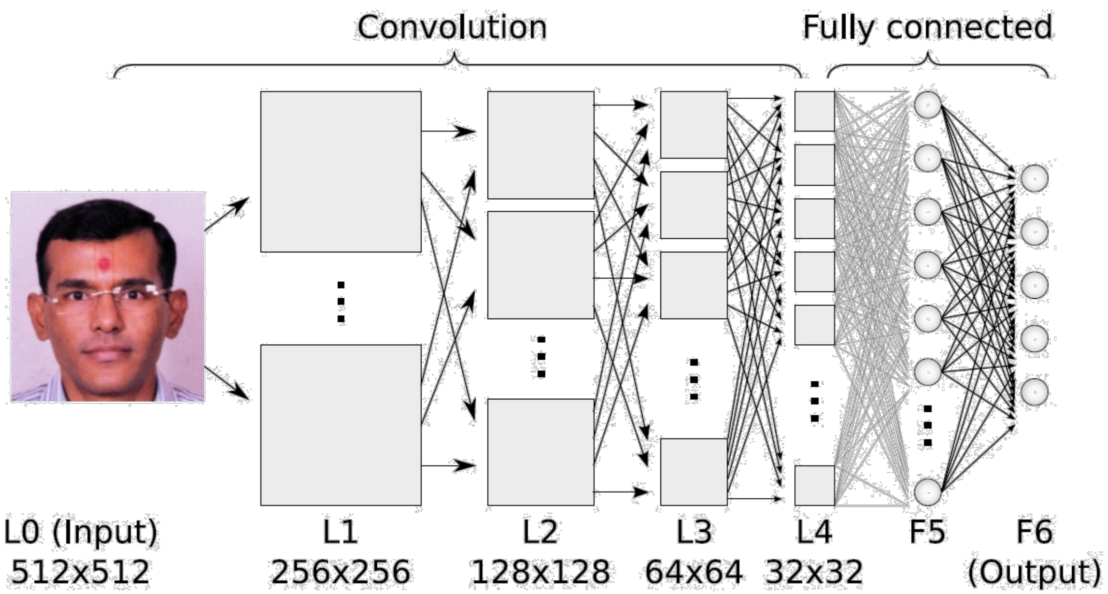
Leonidas Guibas Laboratory
Geometric Computing

Talk Outline

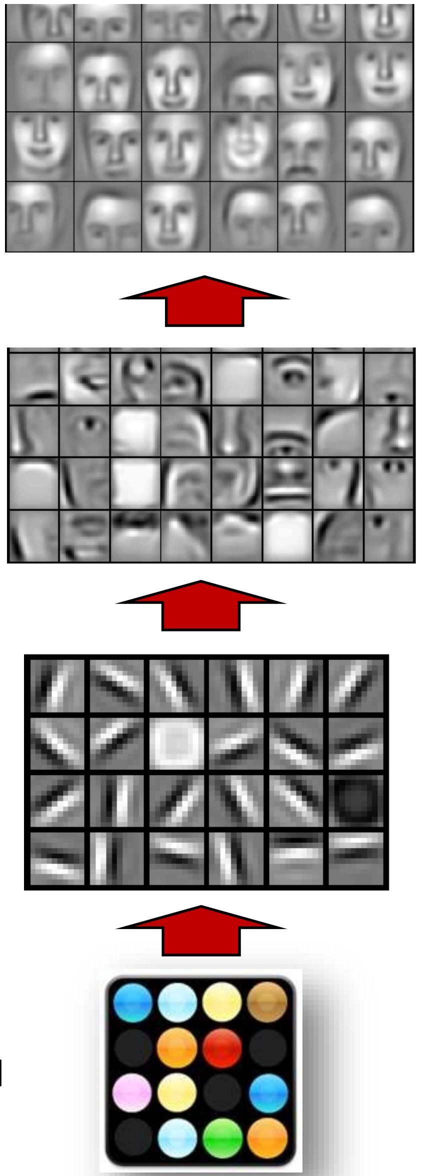
- Focus on maps and correspondences for visual data such as images or 3D models
- Develop a functional representation for such maps
- Build networks of functional maps connecting different data and discuss synchronization problems on such networks
- Demonstrate how such synchronizations helps to clean up maps and leads to emergence of shared semantic abstractions between the data



Vertical (Deep) Learning Networks



[Makwana, 2016]



[Lee et al., 2009]

Data-driven feature learning at ascending abstraction layers

“Deep” nets

Horizontal Networks

Similarity as a communications channel

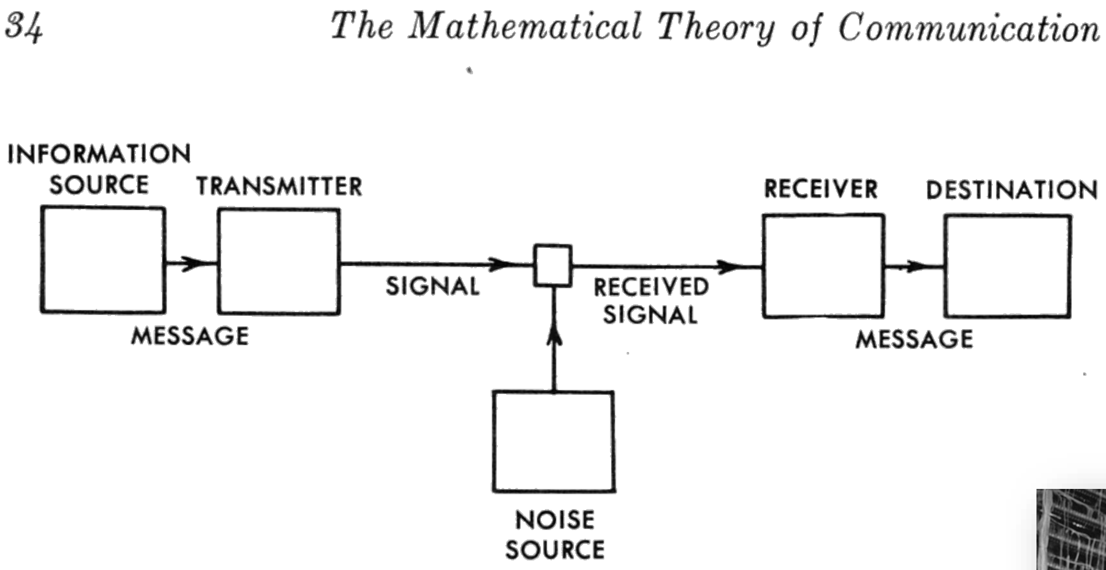
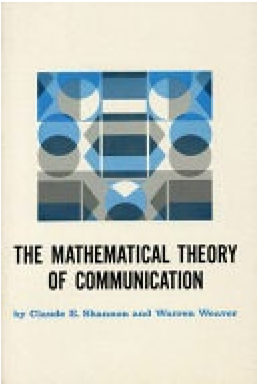
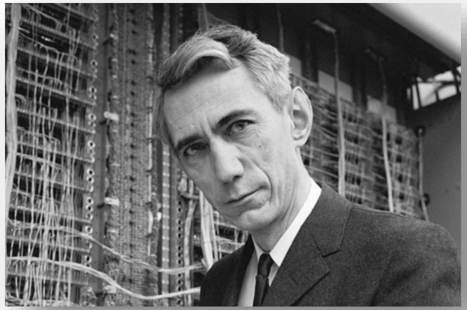


Fig. 1. — Schematic diagram of a general communication system.



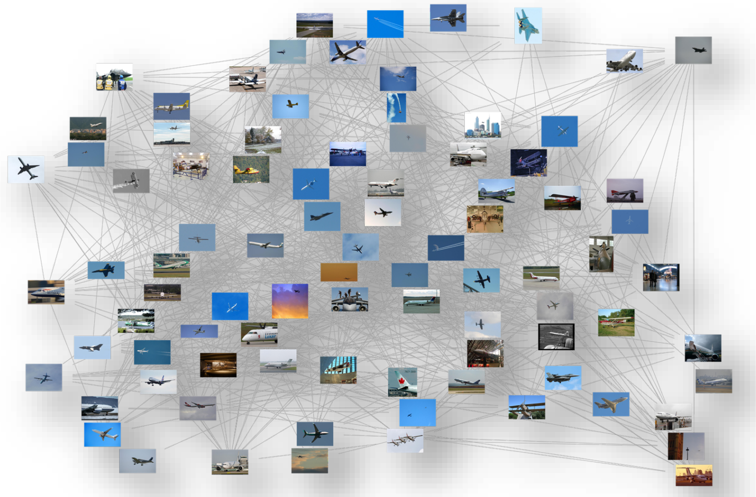
Claude Shannon

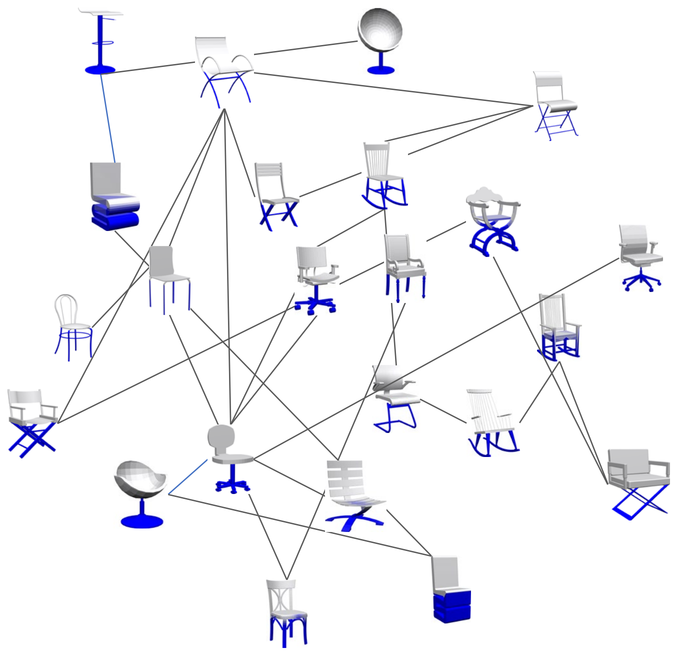


The Network View: Information Transport Between Visual Data

Networks of Images

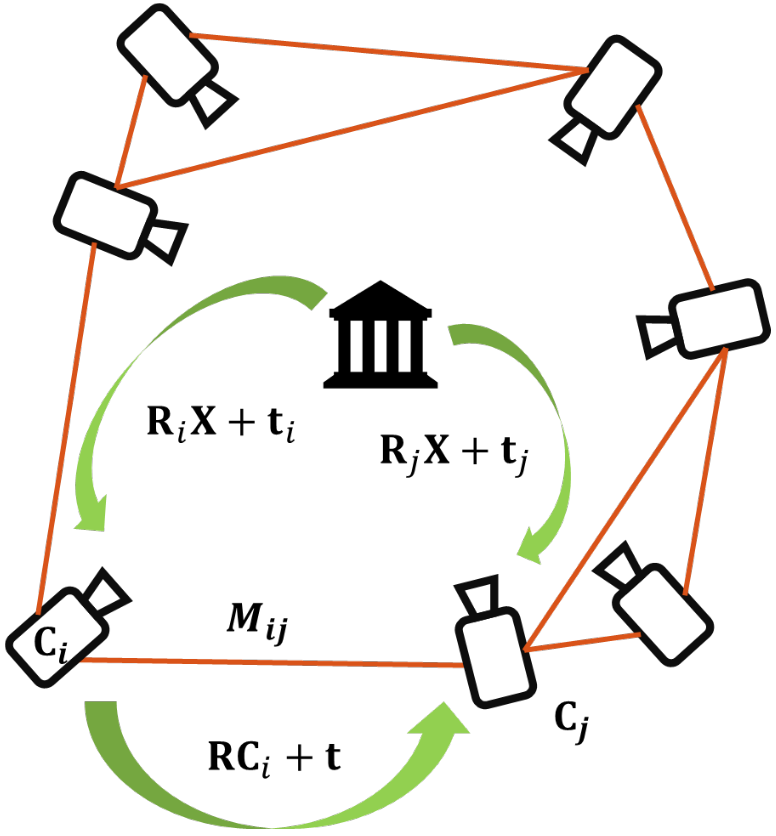
Or of Shapes, Or of Both

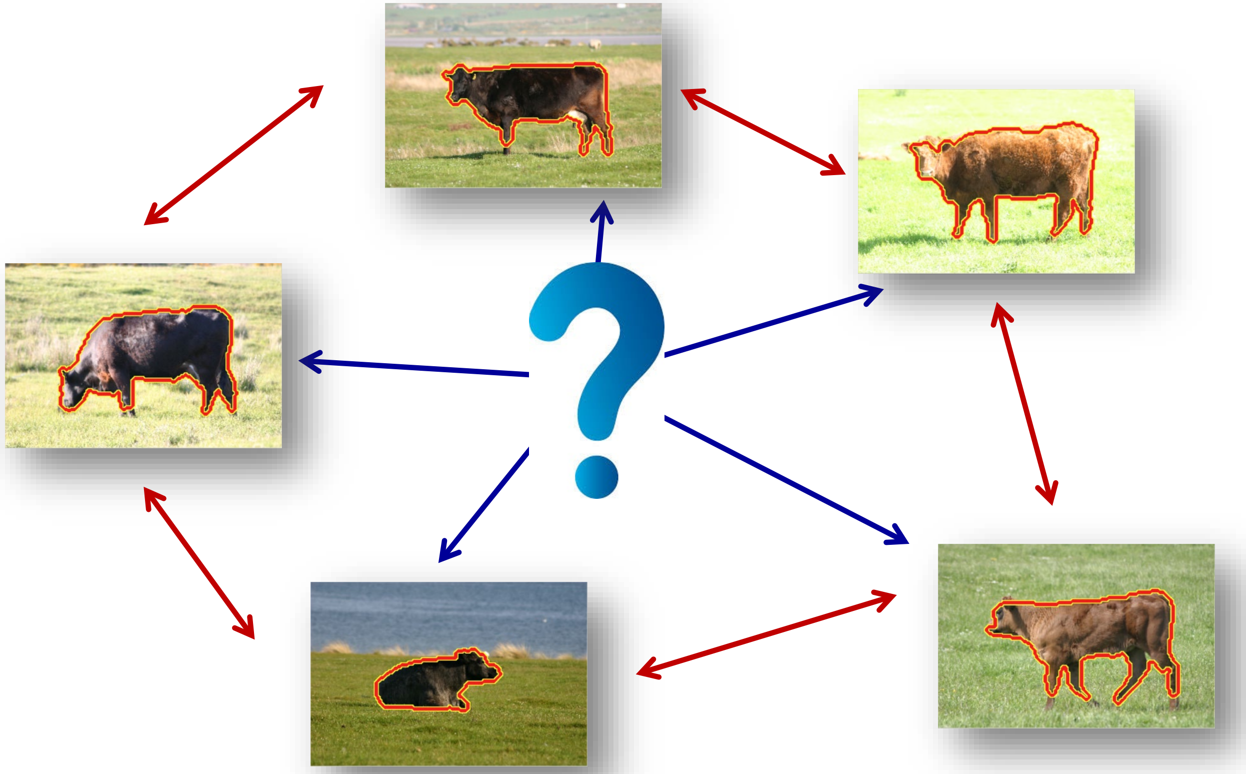




Relations Between Visual Data

Abstraction Emergence from the Network

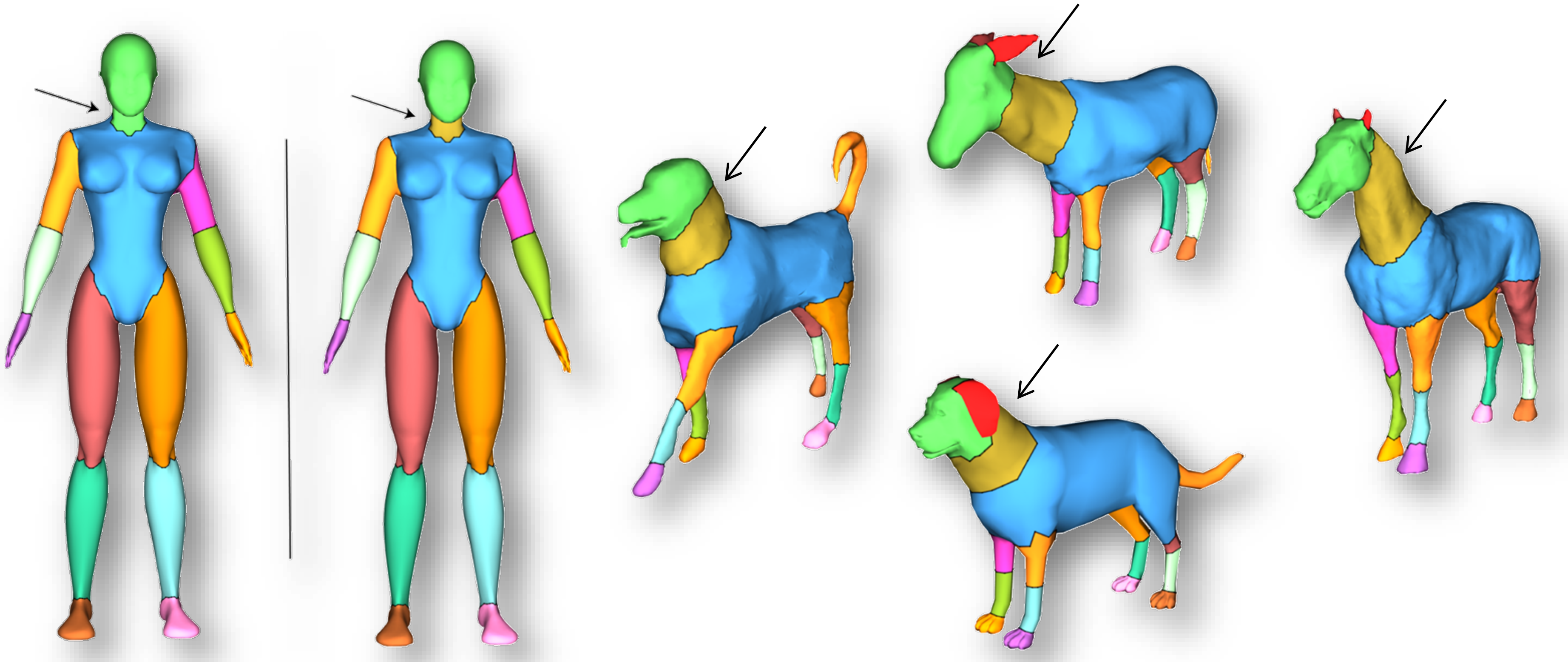




Each Data Set Is Not Alone

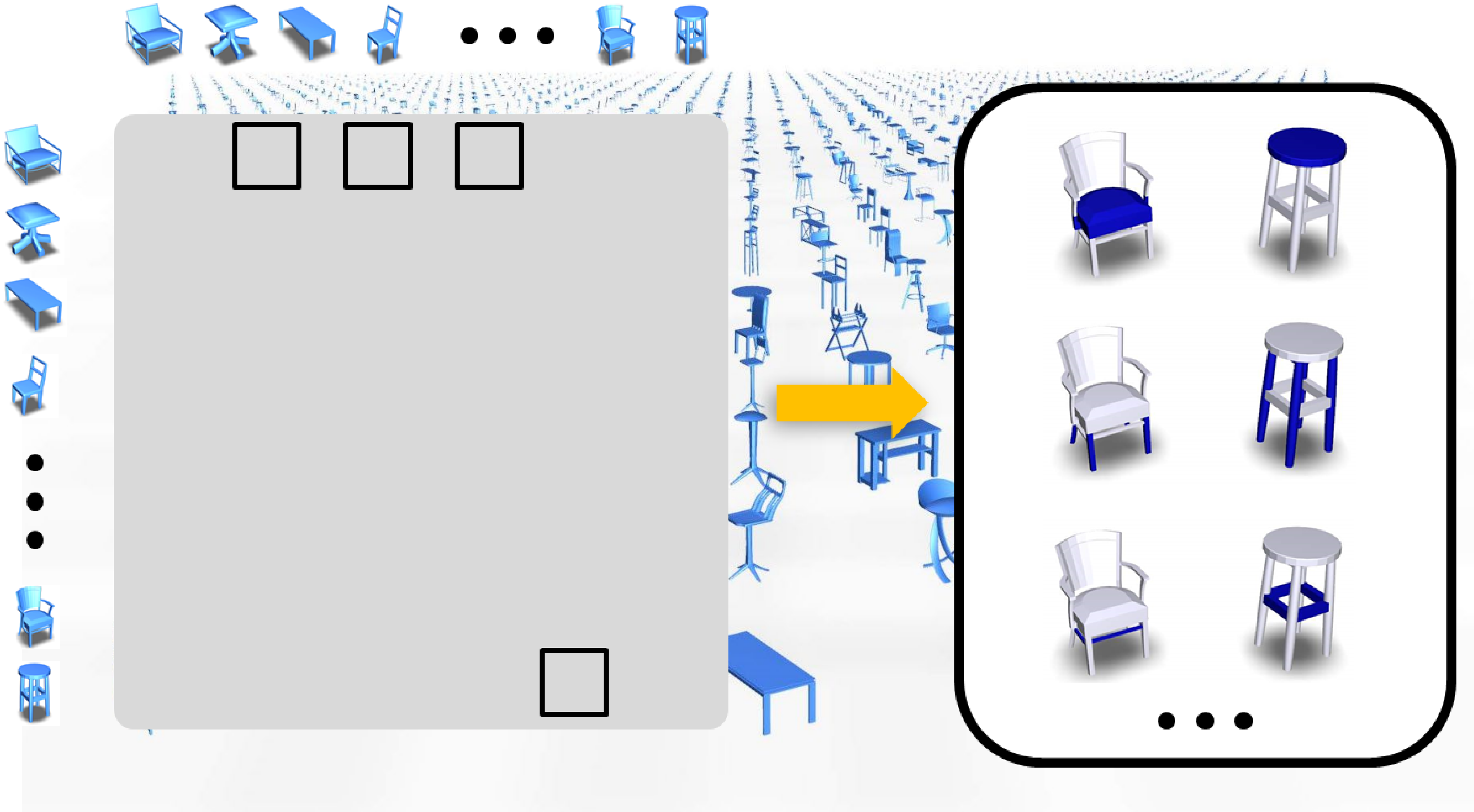
[Q. Huang, V. Koltun, L. Guibas, 2011]

- The interpretation of a particular piece of geometric data is deeply influenced by our interpretation of other related data



3D Segmentation

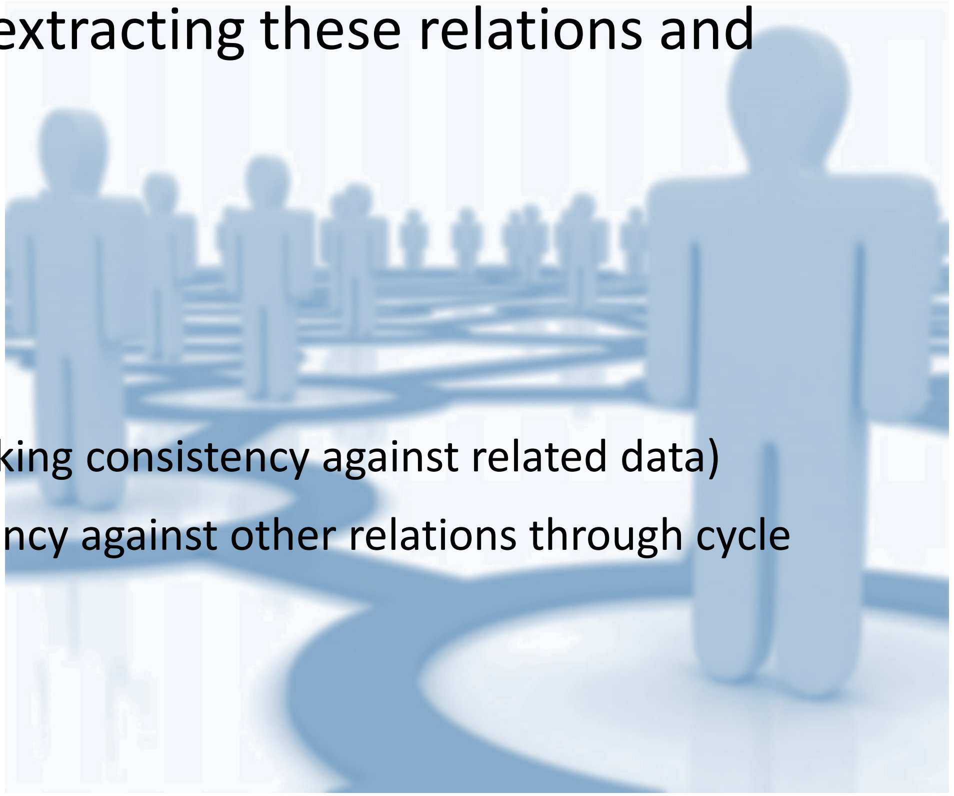
Semantic Structure Emerges from the Network



[Q. Huang, F. Wang, L. Guibas, ’14]

Societies, or Social Networks of Data Sets

Our understanding of data can greatly benefit from extracting these relations and building relational networks.



We can exploit the relational network to

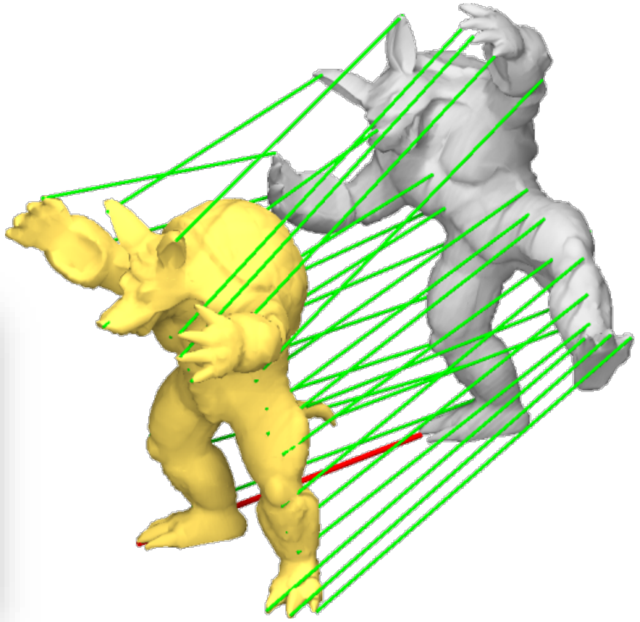
- transport information around the network
- assess the validity of operations or interpretations of data (by checking consistency against related data)
- assess the quality of the relations themselves (by checking consistency against other relations through cycle closure, etc.)
- extract shared structure among the data

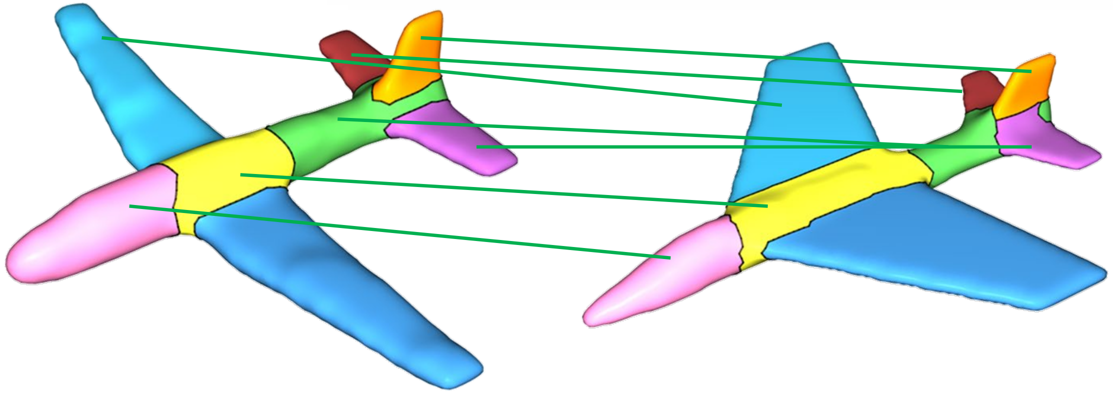
Thus the network becomes the great regularizer in joint data analysis.

Maps as
First-Class Citizens

Relationships as Correspondences or Maps

- Multiscale mappings
  - Point/pixel level
  - part level

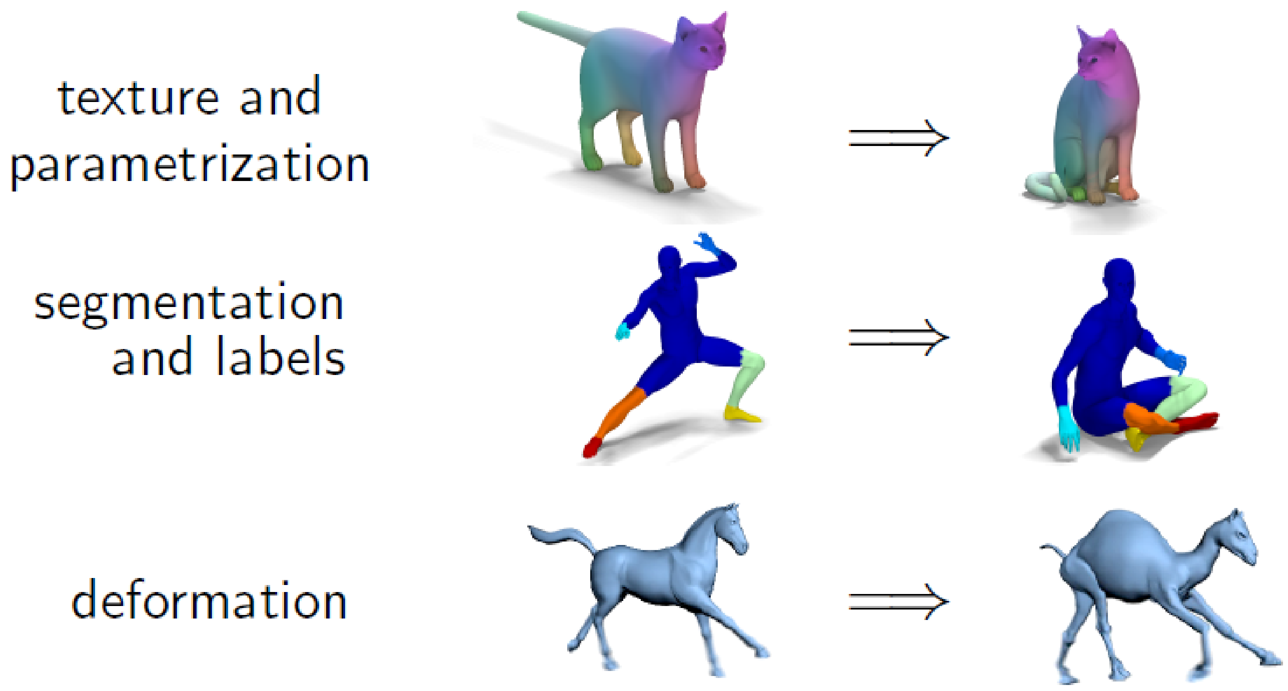




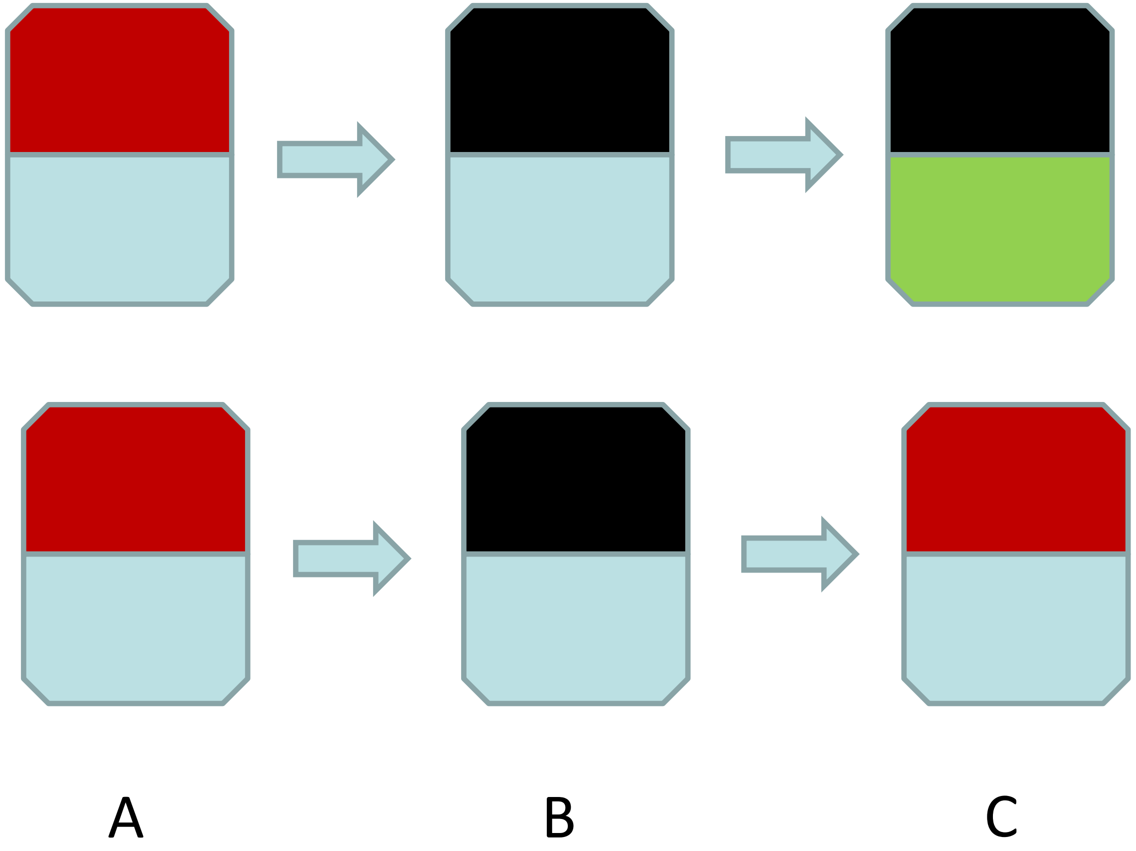
Maps capture what is the same or similar across two data sets

Matching Enables Information Transport

Correspondences or Maps are Information Transporters



Maps vs. Distances/Similarities Networks vs. Graphs

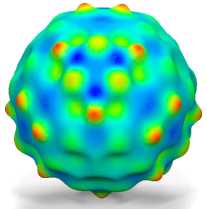


Persistence of correspondences

Technical Approach: Function Spaces and Functional Maps

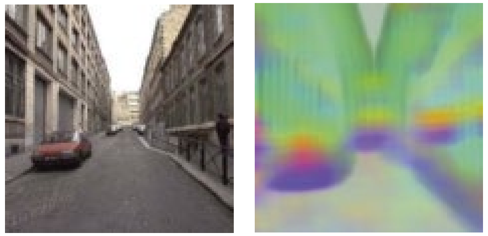
A Dual View: Functions and Operators

- Functions on data
  - Properties, attributes, descriptors, part indicators, etc.
  - But also beliefs, opinions, etc
- Operators on functions
  - Maps of functions to functions
    - Laplace-Beltrami operator on a manifold



Curvature

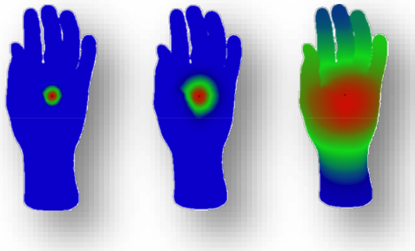
Parts



SIFT flow, C. Liu 2011

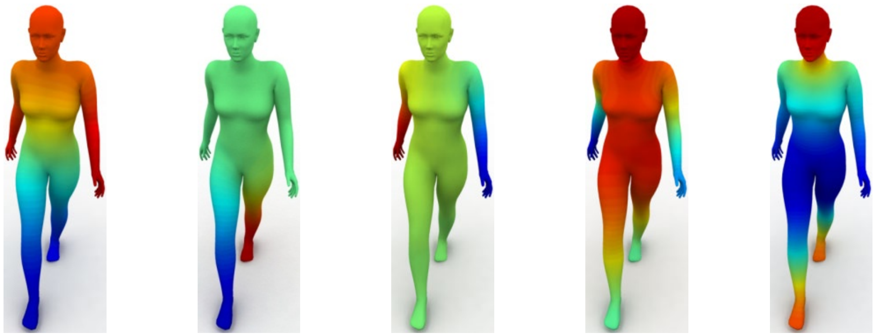
$$M$$

$$\Delta : C^\infty(M) \to C^\infty(M), \Delta f = \text{div } \nabla f$$



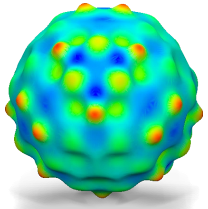
$$\frac{\partial u}{\partial t} = -\Delta u$$

heat diffusion



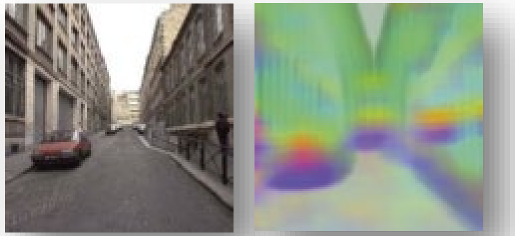
Laplace Beltrami eigenfunctions

Knowledge as Functions



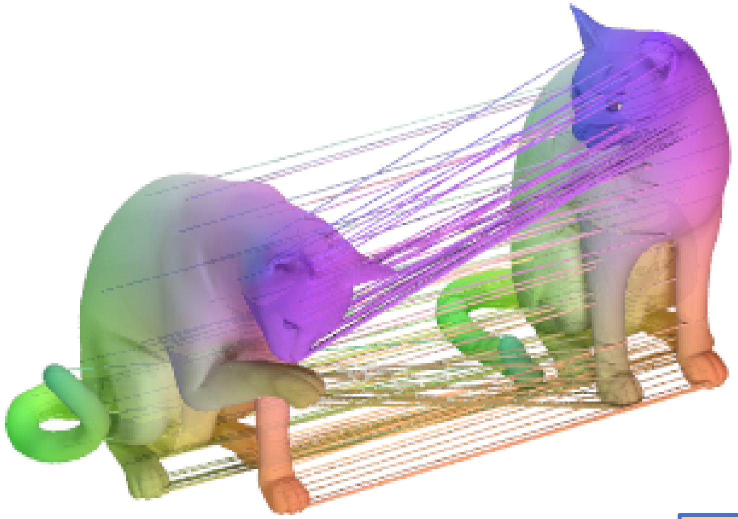
Curvature

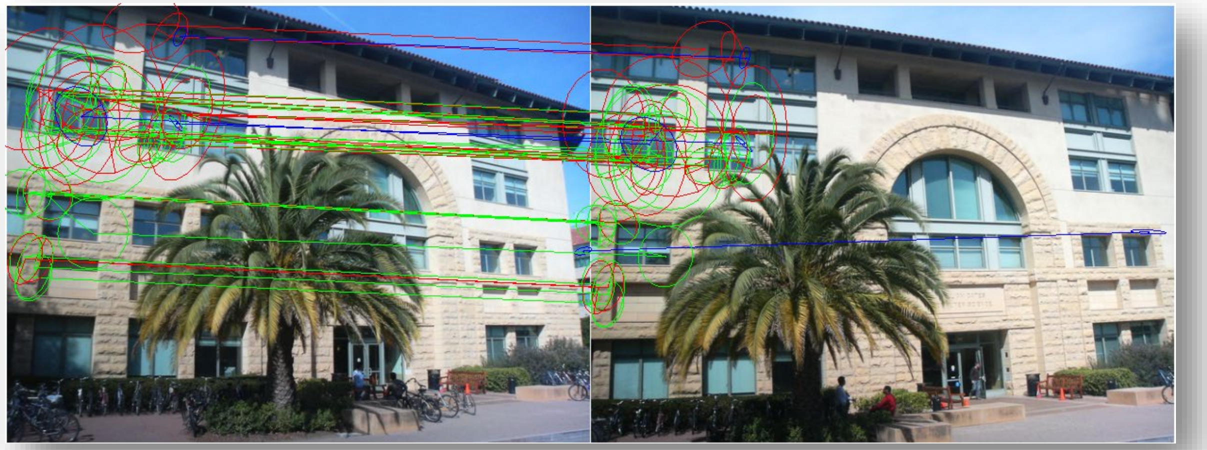
Parts



SIFT flow, C. Liu 2011

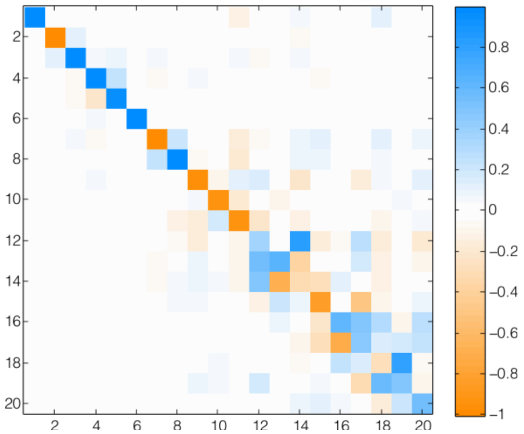
Knowledge towers over visual data:
function spaces



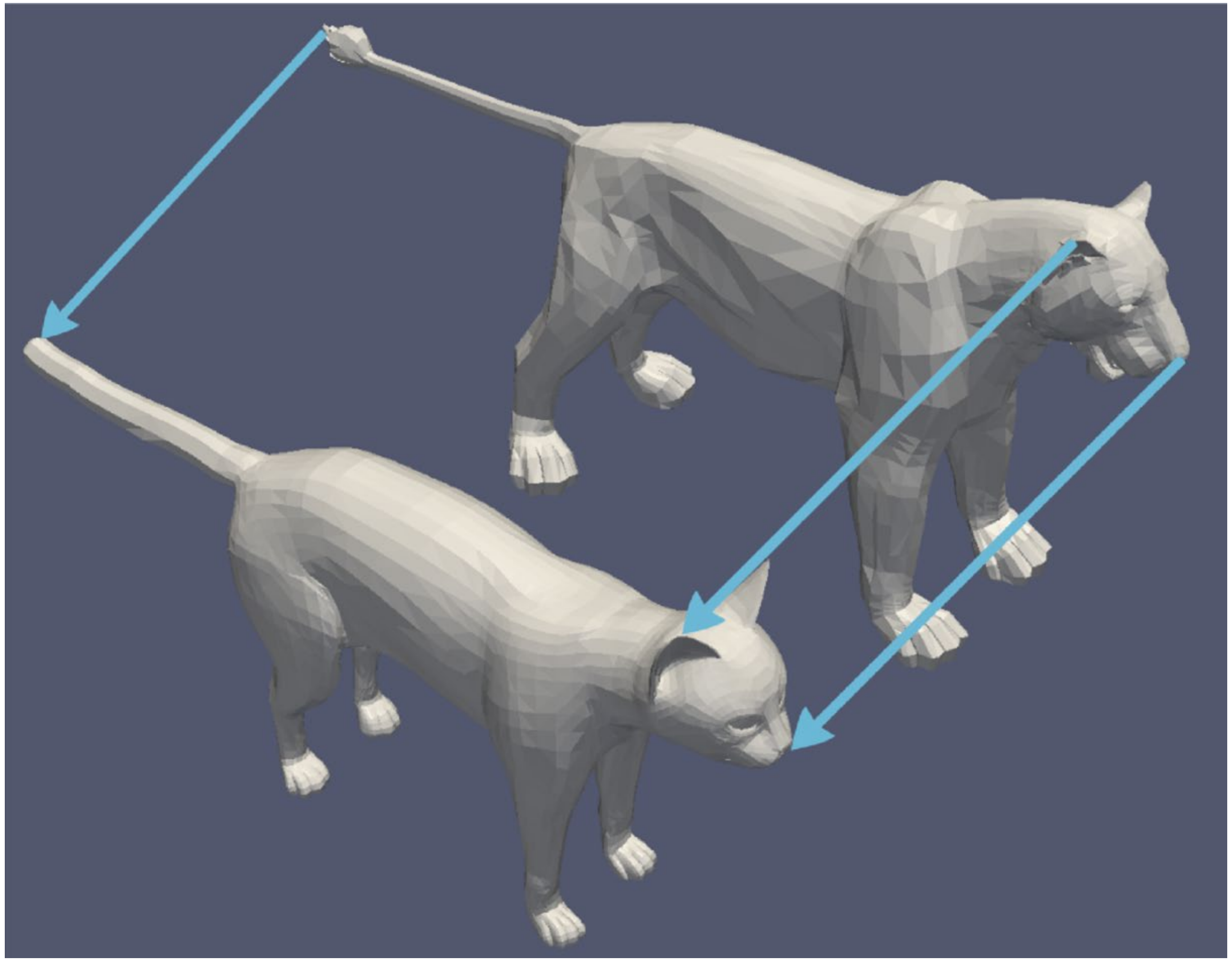


Functional Maps (a.k.a. Operators)

[M. Ovsjanikov, M. Ben-Chen, J. Solomon, A. Butscher, L. G., Siggraph ’12]

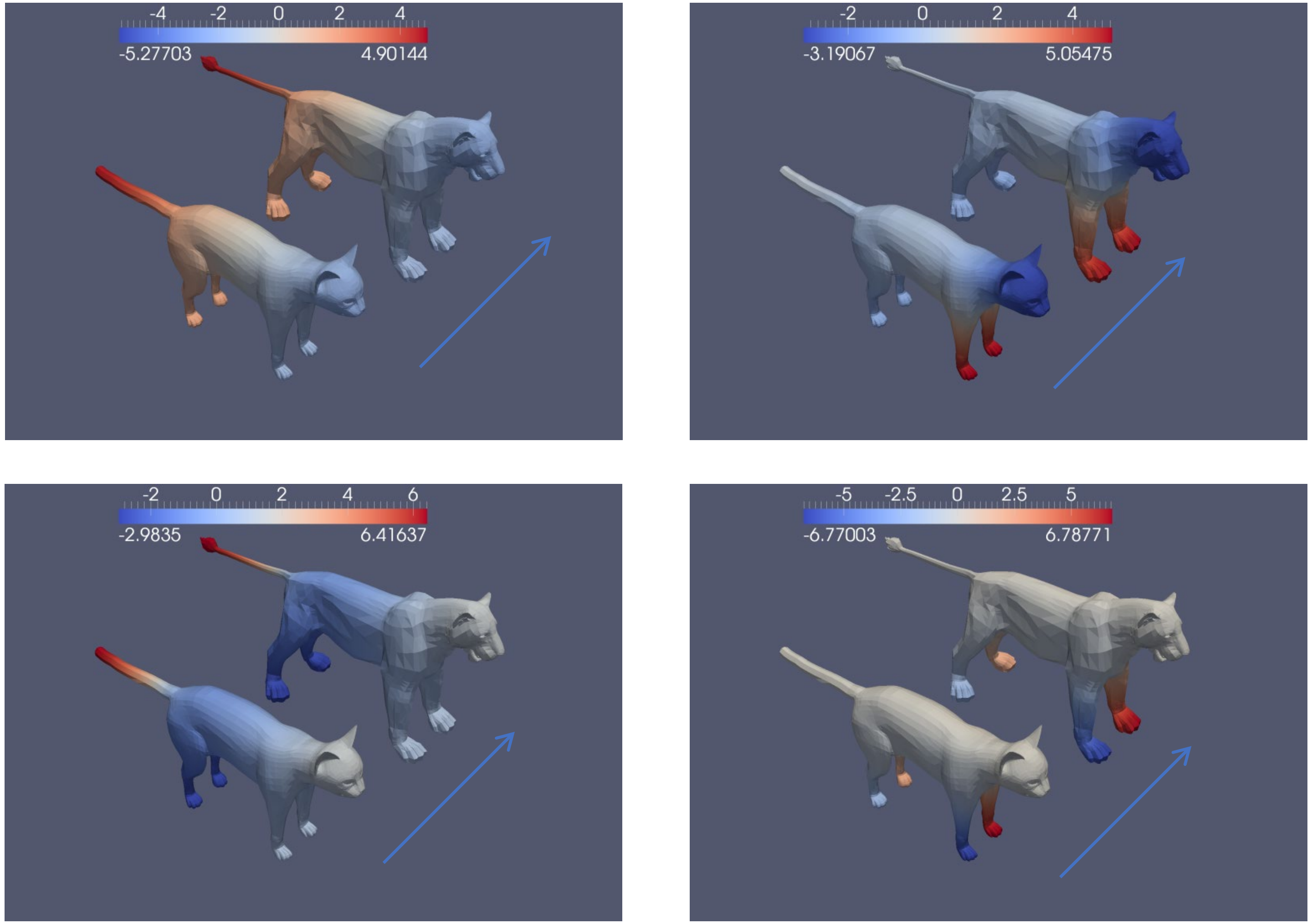


Starting from a Regular P2P Map ϕ



ϕ: lion → cat

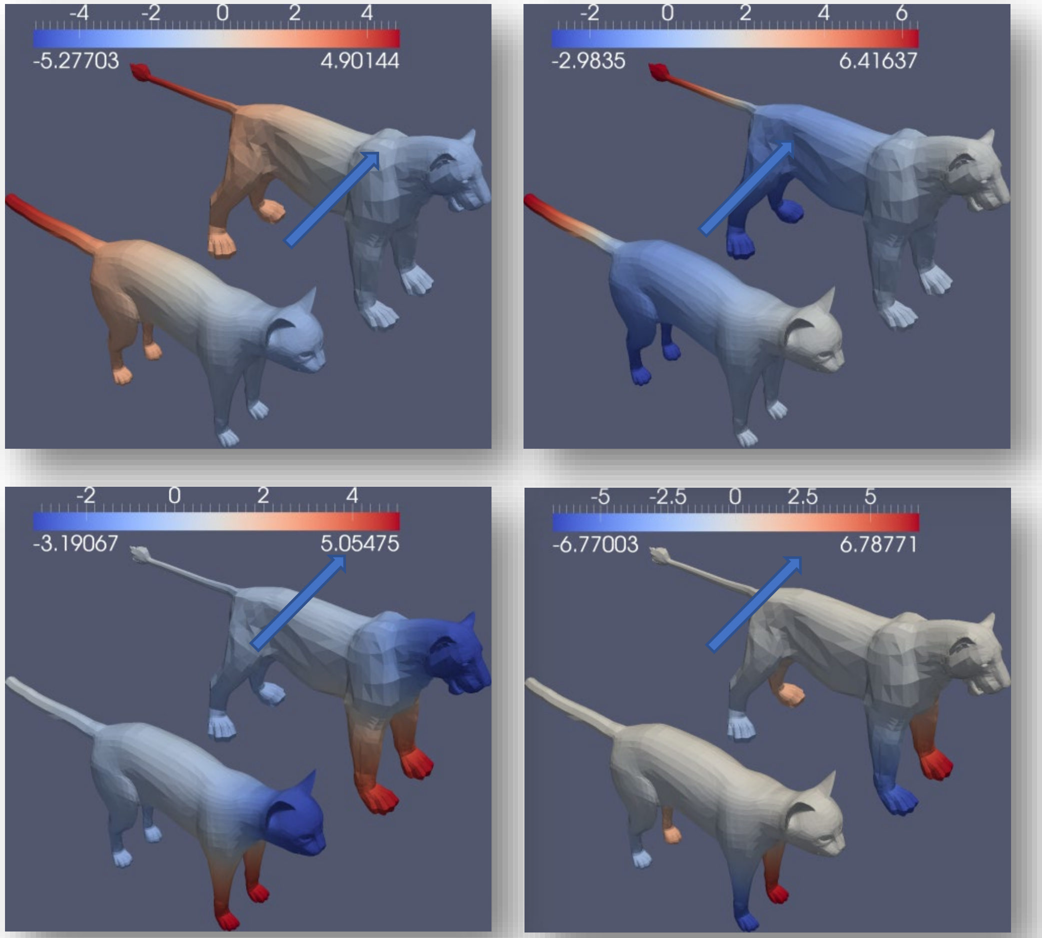
Attribute Transfer via Pull-Back



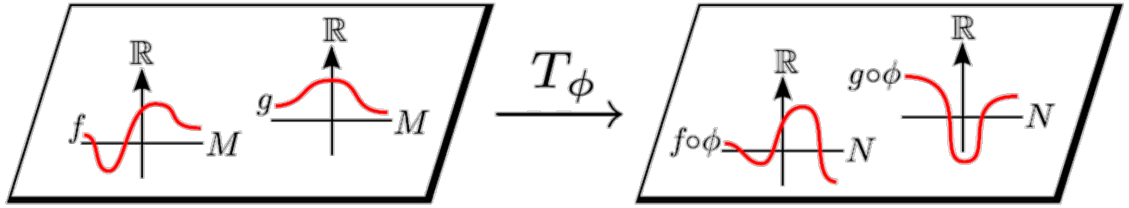
T_{ϕ}: cat → lion

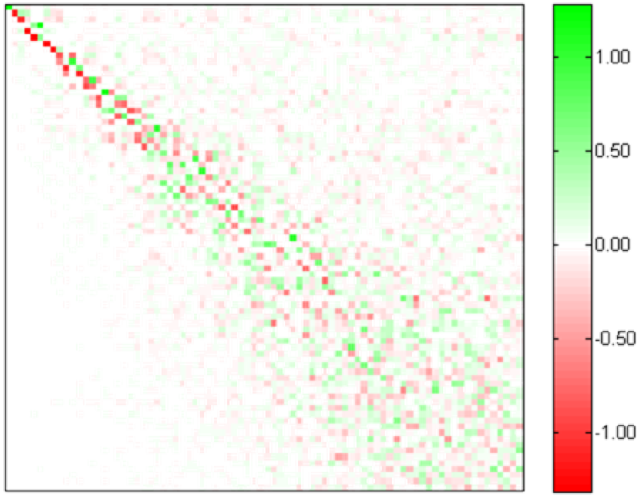
A Contravariant Functor

from cat to lion



Functions on cat are transferred to lion using T_ϕ





T_ϕ is a linear operator (matrix)

$$T_\phi : L^2(cat) \rightarrow L^2(lion)$$

Functional Map: From P2P to F2F

$$\boxed{\phi : M \to N}$$

$$\boxed{T_\phi : L^2(N) \to L^2(M)}$$

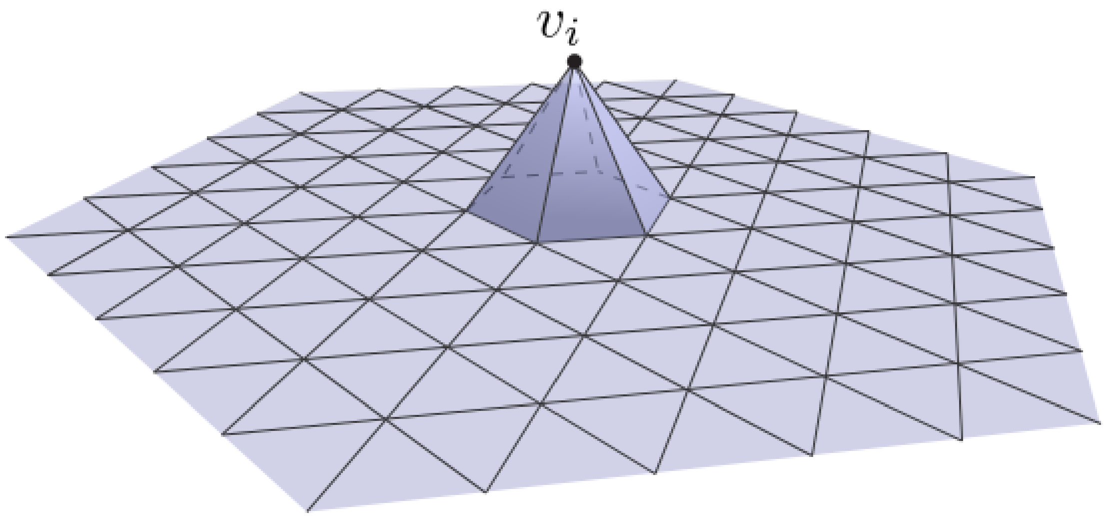
Dual of a point-to-point map

Bases for a Function Space

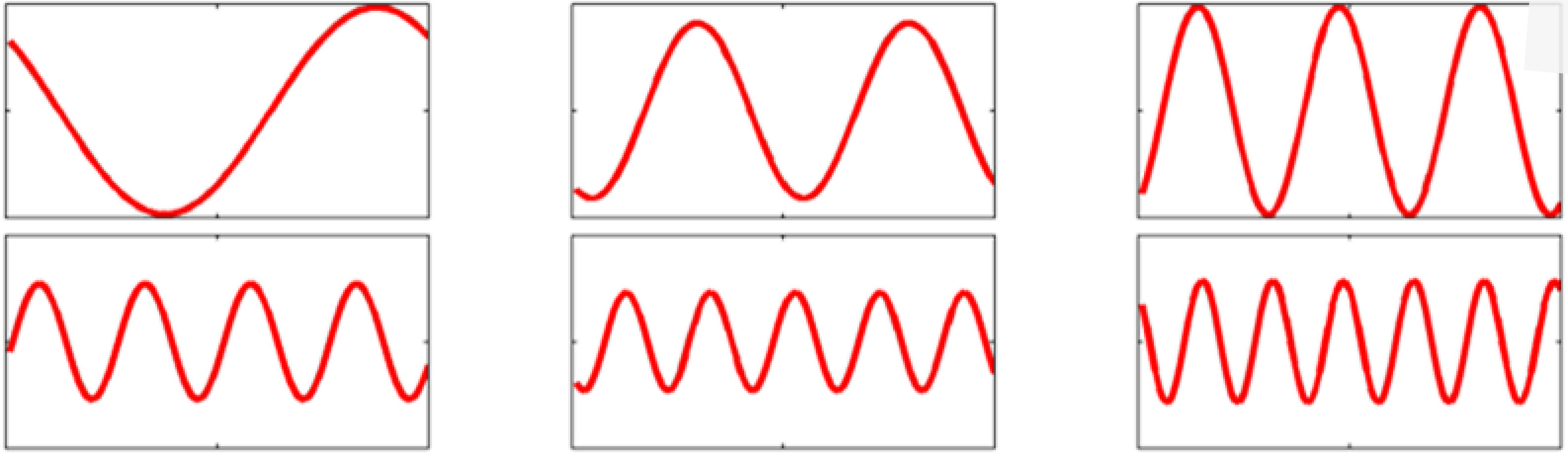
Point basis

Finite-element basis

Local bases



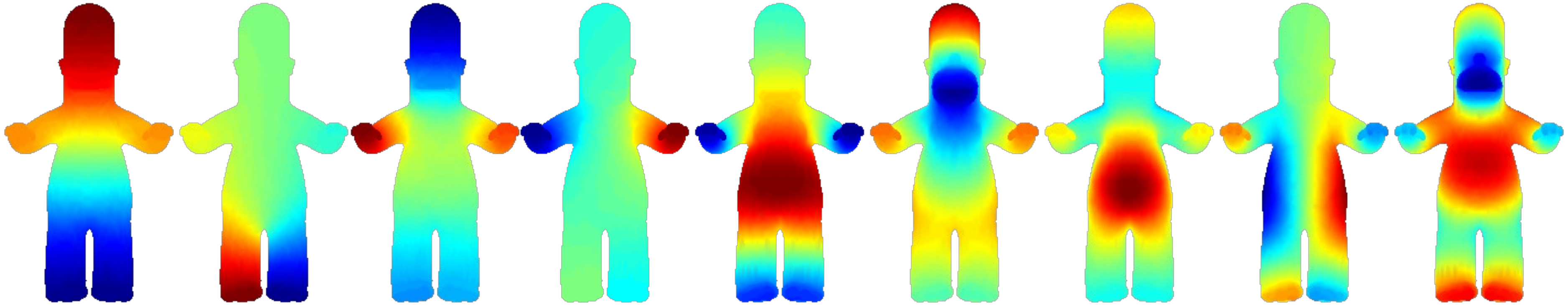
Hierarchical Bases for a Function Space



Fourier

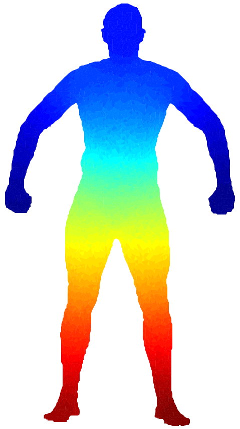
Laplace-Beltrami

global support



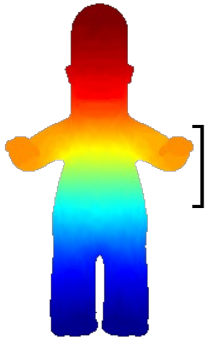
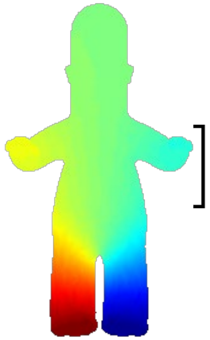
Application of Basis

$f(x) = a_1 \cdot$  $+ a_2 \cdot$  $+ a_3 \cdot$  $+ \cdots$

$\phi :$  $\longrightarrow$ 

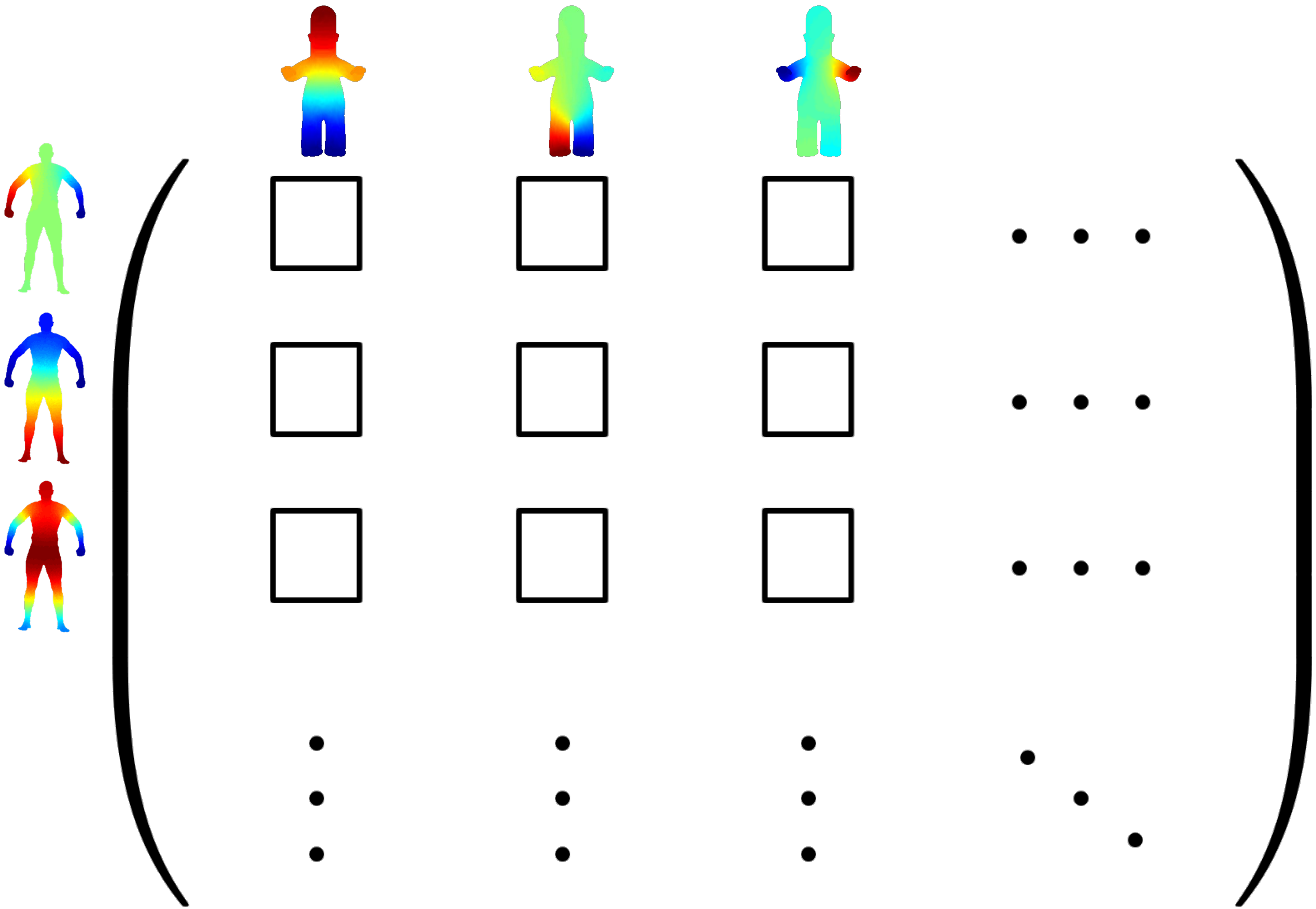
Application of Basis

$T_\phi[f](x) = T_\phi[a_1 \cdot$  $+ a_2 \cdot$  $+ a_3 \cdot$  $+ \cdots]$

$= a_1 T_\phi[$  $] + a_2 T_\phi[$  $] + a_3 T_\phi[$  $] + \cdots$

Enough to know these

Functional Map Matrix



Functional Map Representation

Definition

For a fixed choice of basis functions $\{\phi^M\}$ and $\{\phi^N\}$, and a bijection $T : M \rightarrow N$, define its **functional representation** as a matrix C, s.t. for all $f = \sum_i a_i \phi_i^M$, if $T_F(f) = \sum_i b_i \phi_i^N$ then:

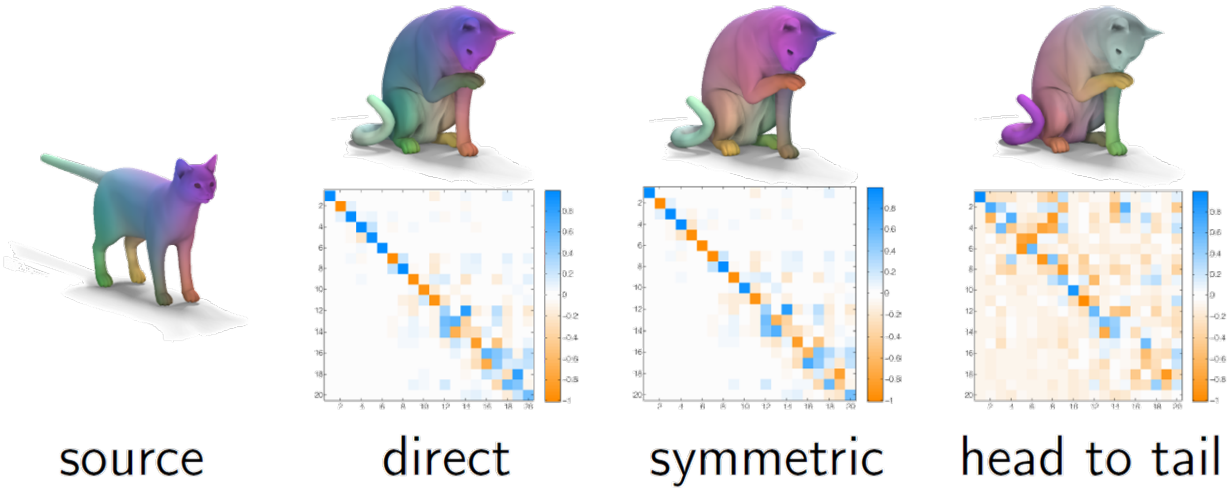
$$\mathbf{b} = C\mathbf{a}$$

If $\{\phi^M\}$ and $\{\phi^N\}$ are both orthonormal w.r.t. some inner product, then

$$C_{ij} = \left\langle T_F(\phi_i^M), \phi_j^N \right\rangle .$$

Maps as Linear Operators

- An ordinary shape map lifts to a linear operator mapping the function spaces
- With a truncated hierarchical basis, compact representations of functional maps are possible as ordinary matrices
- Map composition becomes ordinary matrix multiplication
- Functional maps can express many-to-many associations, generalizing classical 1-1 maps



Using truncated Laplace-Beltrami basis

Estimating the Mapping Matrix

Suppose we don't know *C*. However, we expect a pair of functions $f : M \to \mathbb{R}$ and $g : N \to \mathbb{R}$ to correspond. Then, *C* must satisfy:

$$C\mathbf{a} \approx \mathbf{b}$$

where $f = \sum_i \mathbf{a_i}\phi_i^M, \quad g = \sum_i \mathbf{b}_i\phi_i^N$

Given enough $\{\mathbf{a_i}, \mathbf{b}_i\}$ pairs in correspondence, we can recover C through a linear least squares system.

Function Preservation Constraints

Suppose we don't know *C*. However, we expect a pair of functions $f : M \to \mathbb{R}$ and $g : N \to \mathbb{R}$ to correspond. Then, *C* must be s.t.

$$C\mathbf{a} \approx \mathbf{b}$$

Function preservation constraint is quite general and includes:

- Descriptor preservation (e.g. Gaussian curvature, spin images, HKS, WKS).
- Landmark correspondences (e.g. distance to the point).
- Part correspondences (e.g. indicator function).
- Texture preservation

"Probe functions"

injection of low-level knowledge or supervision

Commutativity Regularization

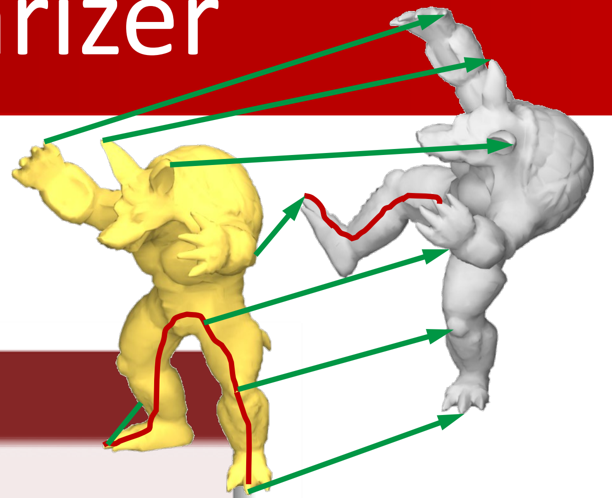
In addition, we can phrase an operator commutativity constraint: given two operators $S_1 : \mathcal{F}(M, \mathbb{R}) \rightarrow \mathcal{F}(M, \mathbb{R})$ and $S_2 : \mathcal{F}(N, \mathbb{R}) \rightarrow \mathcal{F}(N, \mathbb{R})$

$$
\begin{array}{ccc}
\mathcal{F}(M, \mathbb{R}) & \xrightarrow{C} & \mathcal{F}(N, \mathbb{R}) \\
{\scriptstyle S_1}\downarrow & & \downarrow{\scriptstyle S_2} \\
\mathcal{F}(M, \mathbb{R}) & \xrightarrow{C} & \mathcal{F}(N, \mathbb{R})
\end{array}
$$

Thus: $CS_1 = S_2 C$ or $\|CS_1 - S_2 C\|$ should be minimized

Note: this is a linear constraint on C. S_1 and S_2 could be symmetry operators or e.g. Laplace-Beltrami or heat operators.

Isometry (Length Preservation) Regularizer



Lemma 1:

The mapping is *isometric*, if and only if the functional map matrix commutes with the Laplacian:

$$C\Delta_1 = \Delta_2 C$$

Differentiate and then transport

Transport and then differentiate

Δ_1 Laplacian on Shape 1
Δ_2 Laplacian on Shape 2

Conformal (Angle Preservation) Regularization

Lemma 3:

If the mapping is *conformal* if and only if:

$$C^T \Delta_1 C = \Delta_2$$

Using these regularizations, we get a very efficient shape matching method.

Volume Preservation Regularizer

Lemma 2:

The mapping is *locally volume preserving*, if and only if the functional map matrix is *orthonormal*:

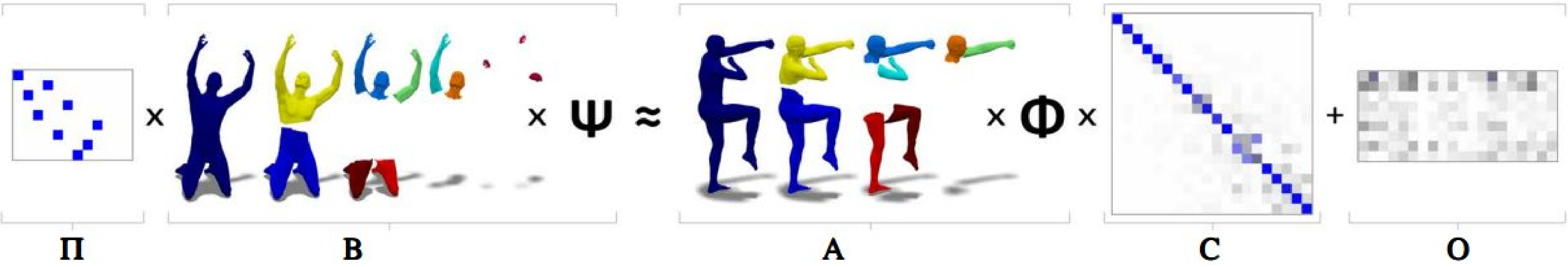
$$C^T C = \mathrm{I}$$

Rotations/reflections in functions space

Sparcity in a Localized Basis

$$\boxed{\min \|C\|_{2,1}}$$

Sum of Euclidean norms of cols



Sparse Modeling of Intrinsic Correspondences (Pokrass, Bronstein[2], Sprechmann, Sapiro)

Basic FMaps Pipeline

Given a pair of shapes: $\mathcal{M}, \mathcal{N}$

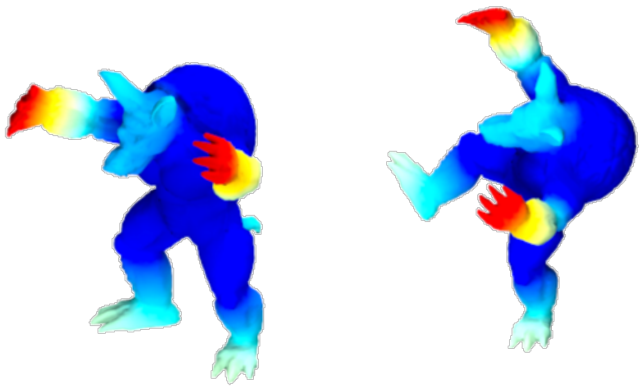
1. Compute the first *k* (~80-100) eigenfunctions of the Laplace-Beltrami operator. Store them in matrices: $\Phi_{\mathcal{M}}, \Phi_{\mathcal{N}}$

2. Compute descriptor functions (e.g., Heat Kernel Signatures) on $\mathcal{M}, \mathcal{N}$. Express them in $\mathbf{A}, \mathbf{B}$, as columns of $\Phi_{\mathcal{M}}, \Phi_{\mathcal{N}}$

3. Solve $C_{\text{opt}} = \arg\min_{C} \|C\mathbf{A} - \mathbf{B}\|^2 + \|C\Delta_{\mathcal{M}} - \Delta_{\mathcal{N}}C\|^2 + \ \ldots$

   $\Delta_{\mathcal{M}}, \Delta_{\mathcal{N}}$: diagonal matrices of eigenvalues of LB operator

4. Convert the functional map C_{opt} to a point to point map *T*.



From Functional to Point-to-Point Maps

- Can try transporting delta functions individually -- expensive

Primal

$$p \mapsto q$$

Dual

$$p \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \mapsto \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} q$$

$$\delta_x = (\phi_1^M(x), \phi_2^M(x), \phi_3^M(x), \ldots)$$

From Functional to Point-to-Point Maps

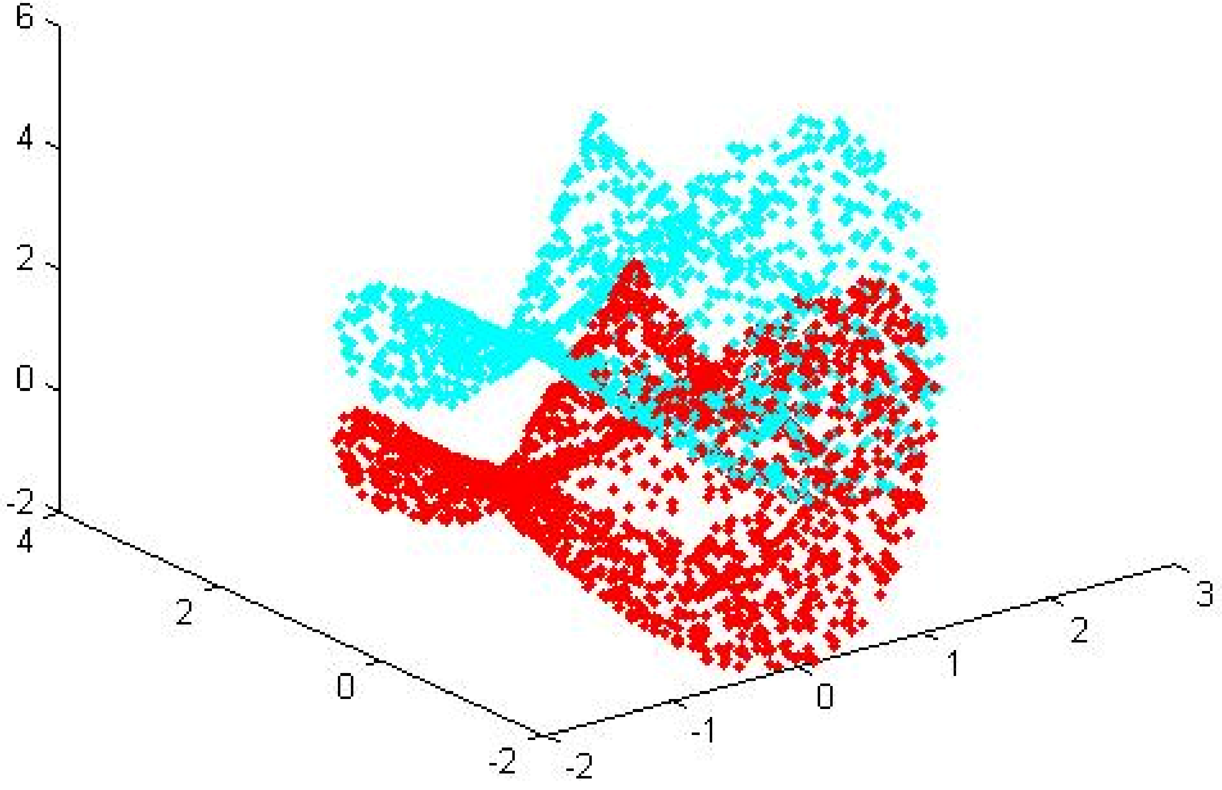
$$C\Phi_M^\top \leftrightarrow \Phi_N$$

Image of each point on surface M

Each point on surface N in LB basis

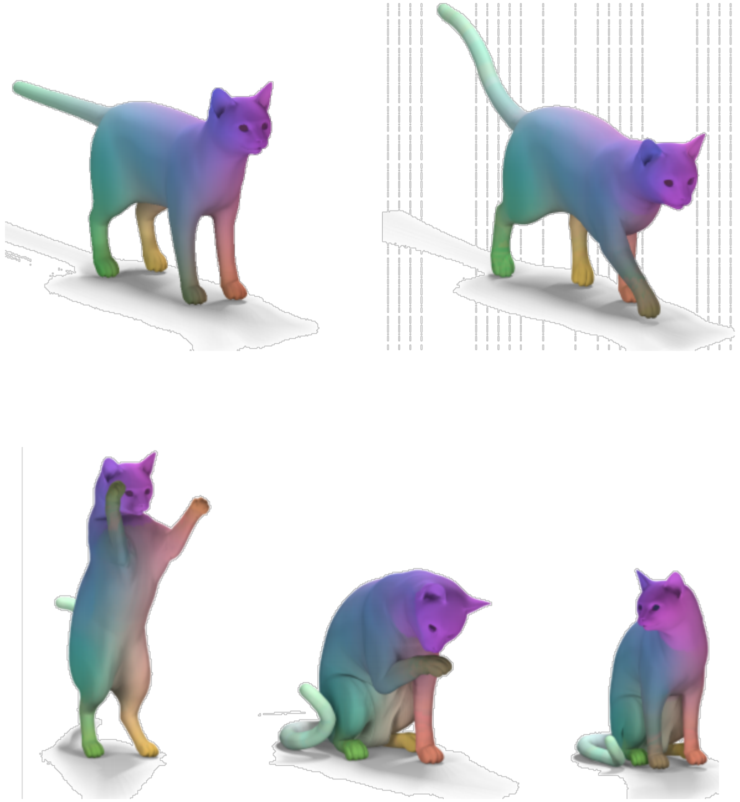
So transport, and then use nearest neighbor search

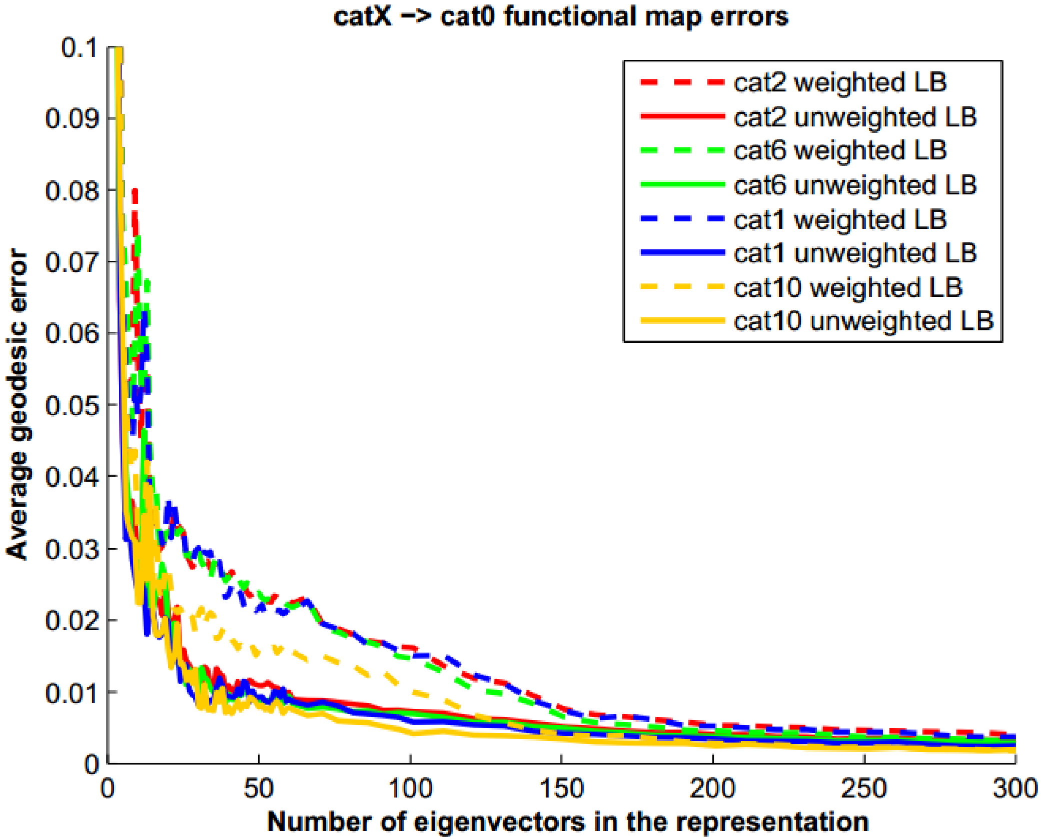
From F2F Back to P2P



ICP in Function Space!

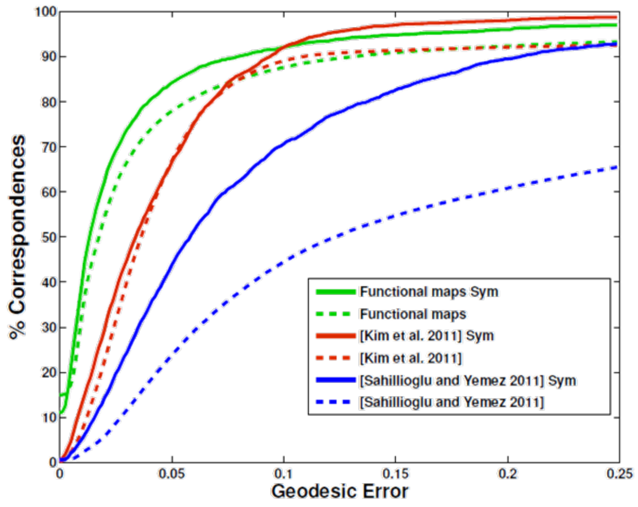
Ground Truth Comparison



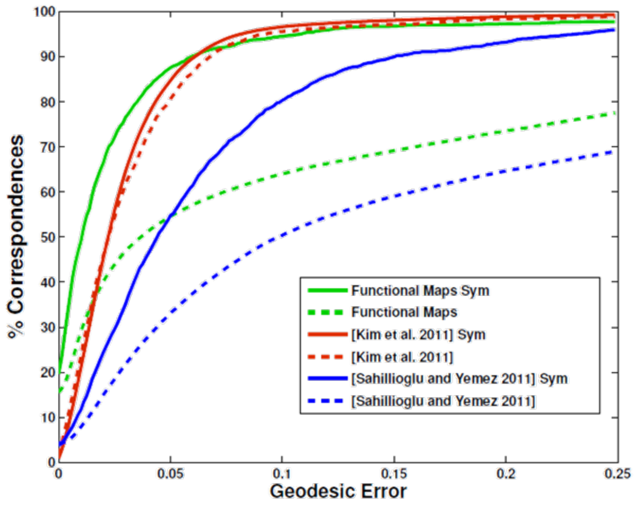


Map Estimation Quality

A very simple method that puts together a modest set of constraints and uses 100 basis functions outperforms state-of-the-art:



SCAPE



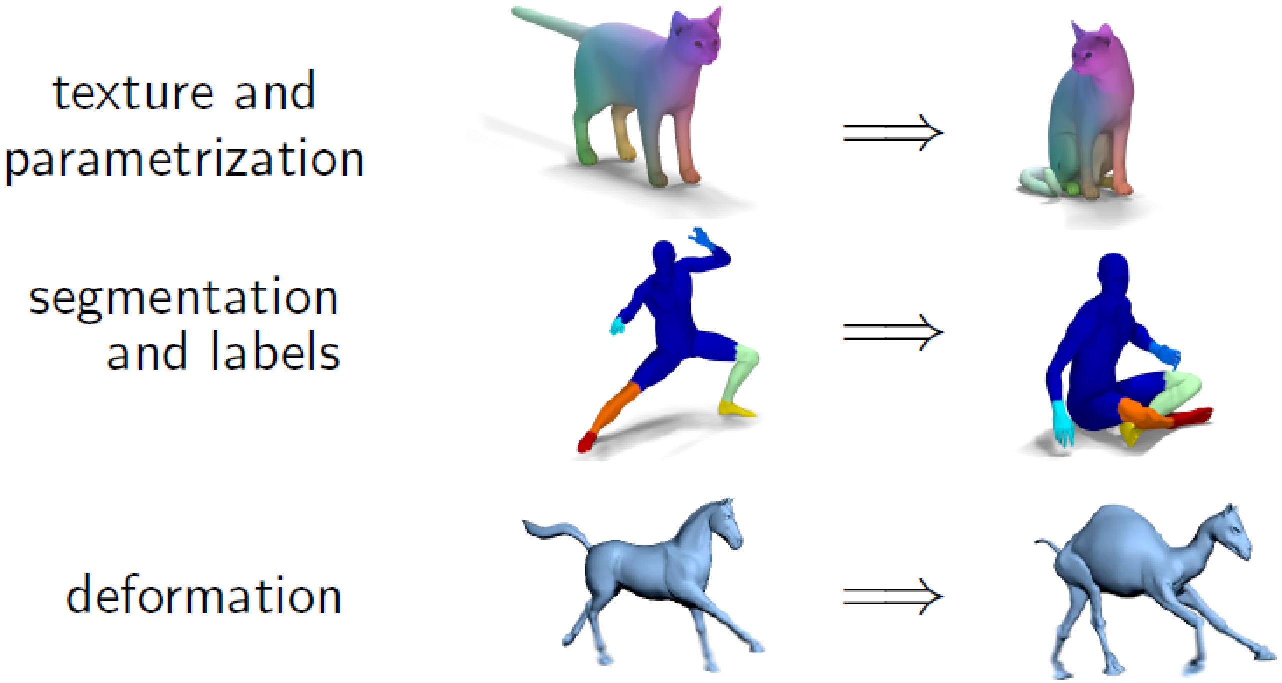
TOSCA

Roughly 10 probe functions + 1 part correspondence

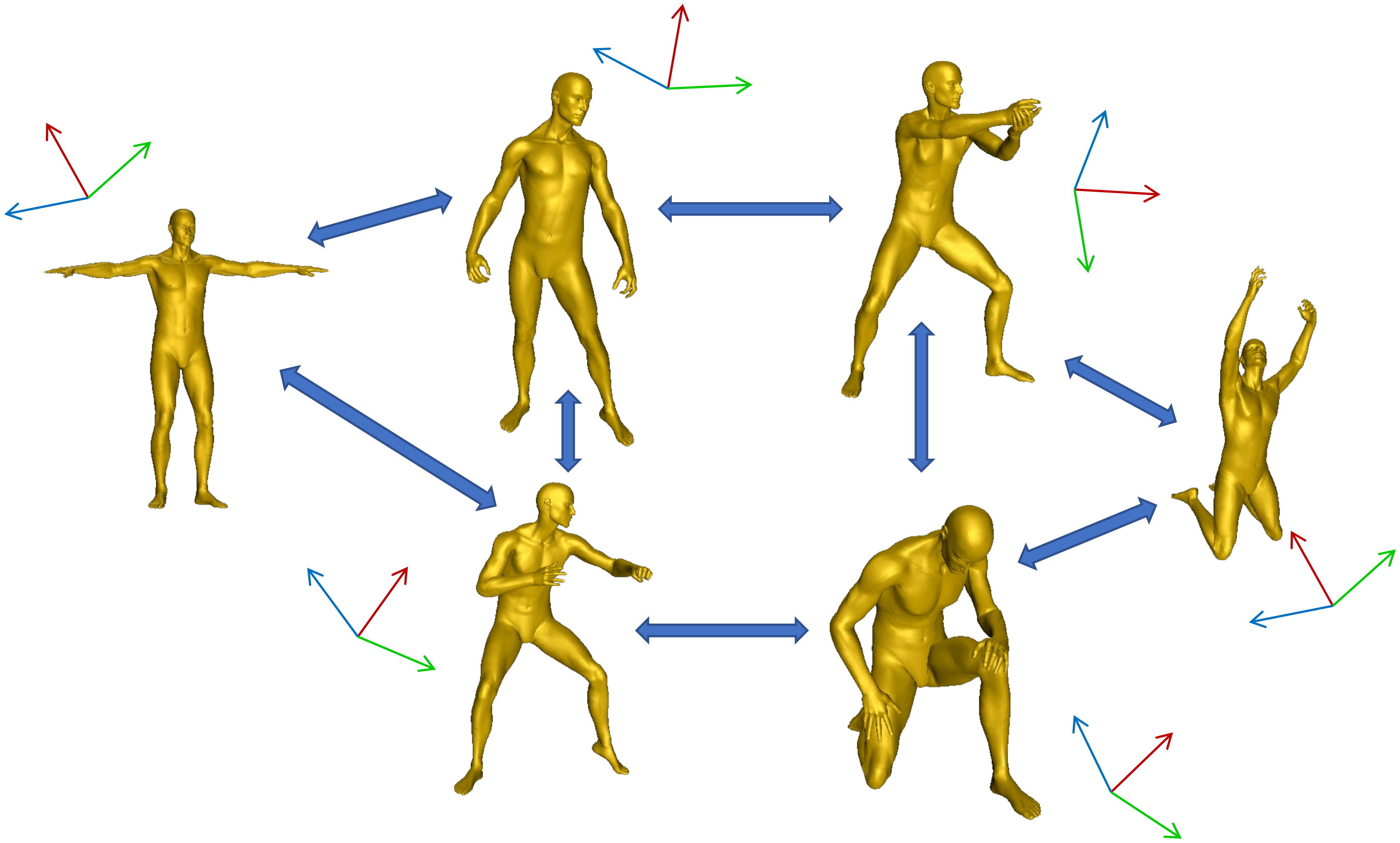
P2P Conversion Not Required : Segmentation Transfer

Consistency of Network Transport

Correspondences or Maps are Information Transporters



Map Networks for Related Data



Networks of “samenesses”

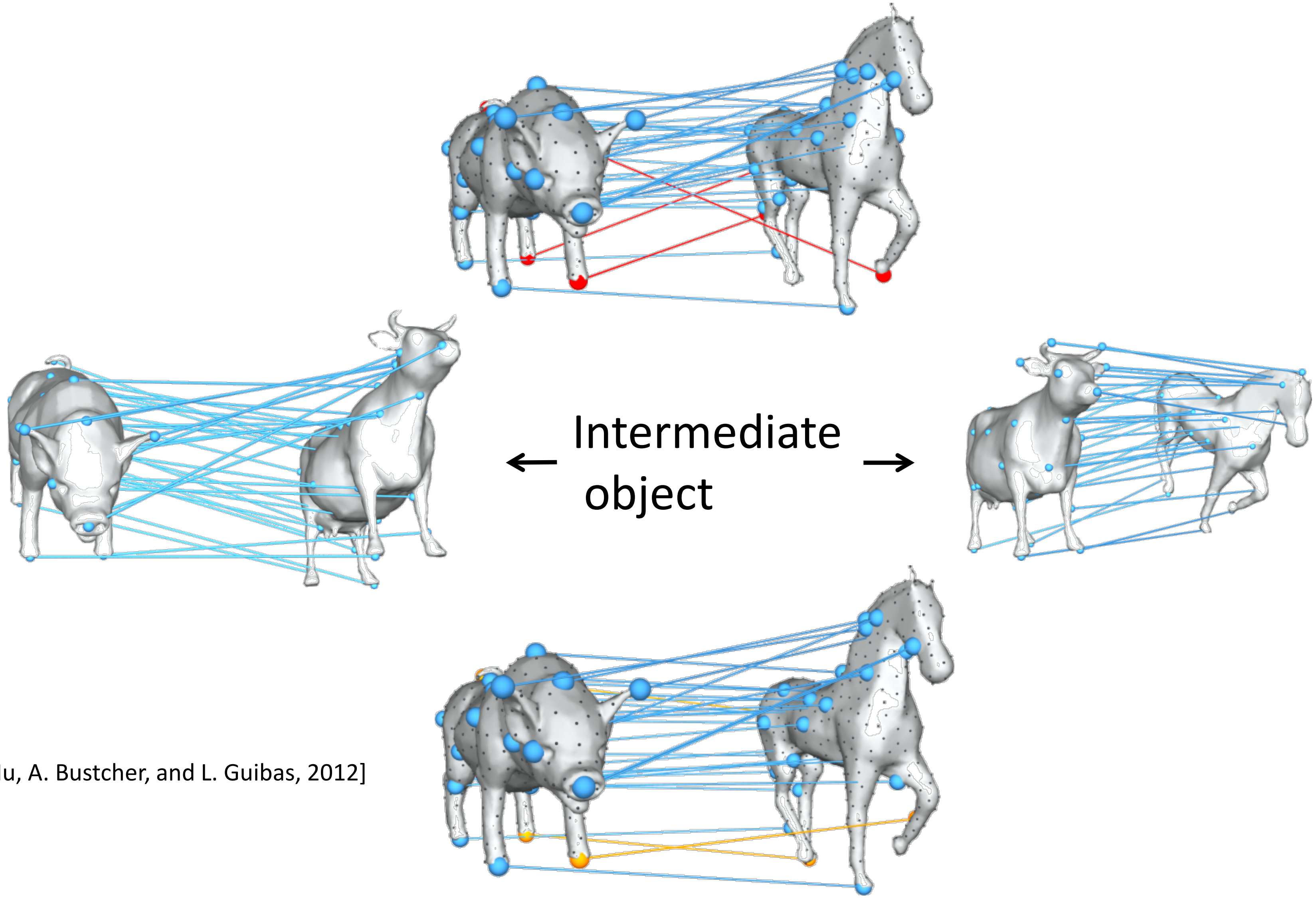
Path Invariance or Loop Closure

Maps are composable, algebraic objects

Maps processing

- We desire path invariance or cycle closure
- What if
  - Real map diagrams don’t commute?
  - Fix the maps, by enforcing cycle consistency!
  - Map recovery is possible even with 50% error!

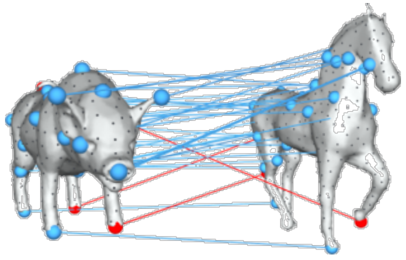
Fixing Maps



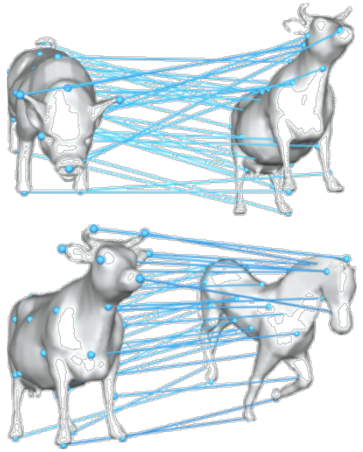
[Q. Huang, G. Zhang, L. Gao, S. Hu, A. Bustcher, and L. Guibas, 2012]

Cycle-Consistency ≡ Low-Rank

- In a functional map network, commutativity, path-invariance, or cycle-consistency are equivalent to a low rank or semidefiniteness condition on a big mapping matrix



$$X = \begin{pmatrix} I_m & X_{1,2} & \cdots & X_{1,n} \\ X_{1,2} & I_m & \cdots & \cdots \\ \vdots & \vdots & I_m & X_{(n-1),n} \\ X_{n,1} & \vdots & X_{n,(n-1)} & I_m \end{pmatrix}.$$



- Conversely, such a low-rank condition can be used to
  - regularize and clean up functional maps
  - extract shared structure

Map Synchronization by Matrix Factorization

$$X = \begin{bmatrix} I_m & X_{12} & \cdots & X_{1n} \\ X_{21} & I_m & \cdots & \vdots \\ \vdots & \vdots & \ddots & X_{n-1,n} \\ X_{n1} & \cdots & X_{n,n-1} & I_m \end{bmatrix}$$

$$X_{ij} = X_{j1} X_{i1}^T$$

$$= \begin{bmatrix} I_m \\ \vdots \\ X_{n1} \end{bmatrix} \begin{bmatrix} I_m & \cdots & X_{n1}^T \end{bmatrix}$$

Structure Emergence Through the Network

Entity Extraction in Images

[F. Wang, Q. Huang, L. G., ICCV ’13]

- Task: jointly segment a set of related images
  - same object, different viewpoints/scales:



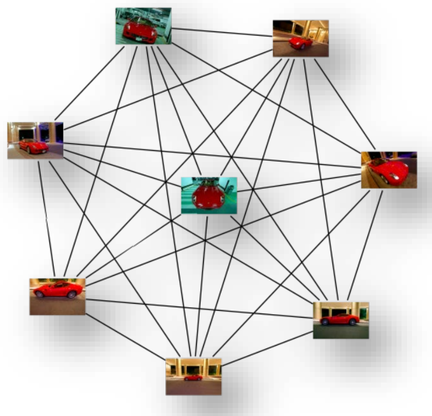

  - similar objects of the same class:



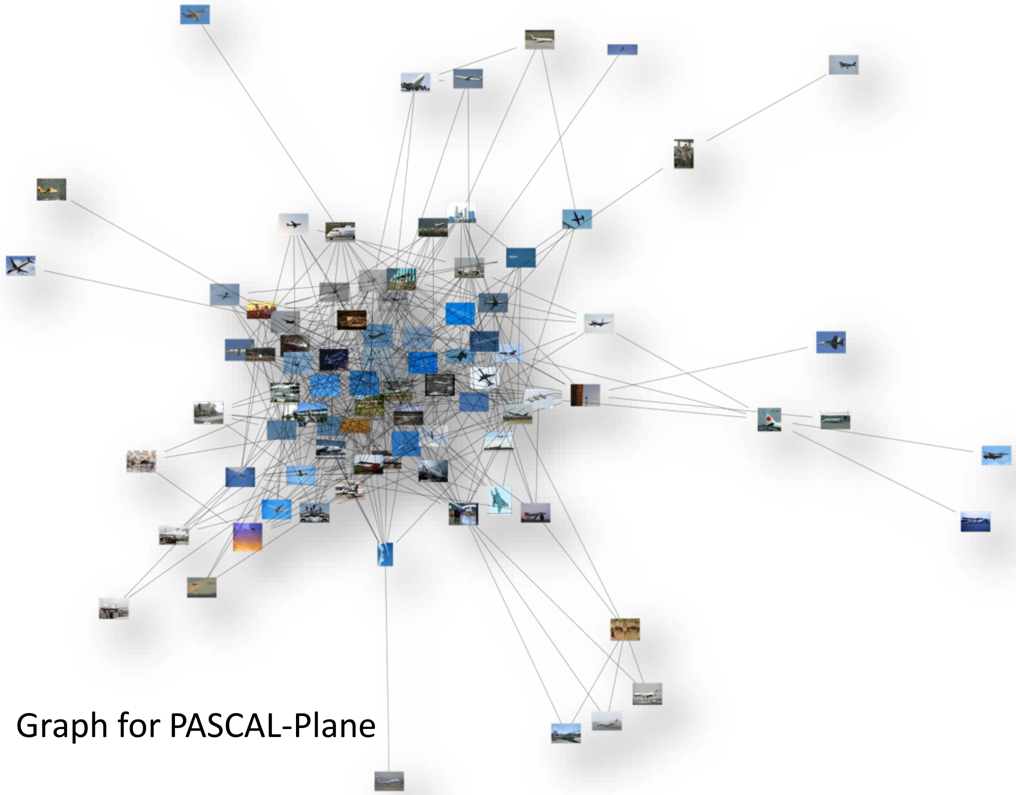

- Benefits and challenges:
  - Images can provide weak supervision for each other
  - But exactly how should they help each other? How to deal with clutter and irrelevant content?

Co-Segmentation via an Image Network

- Image similarity graph based on GIST
  - Each edge has global image similarity w_{ij} and functional maps in both directions;
  - Sparse if large.



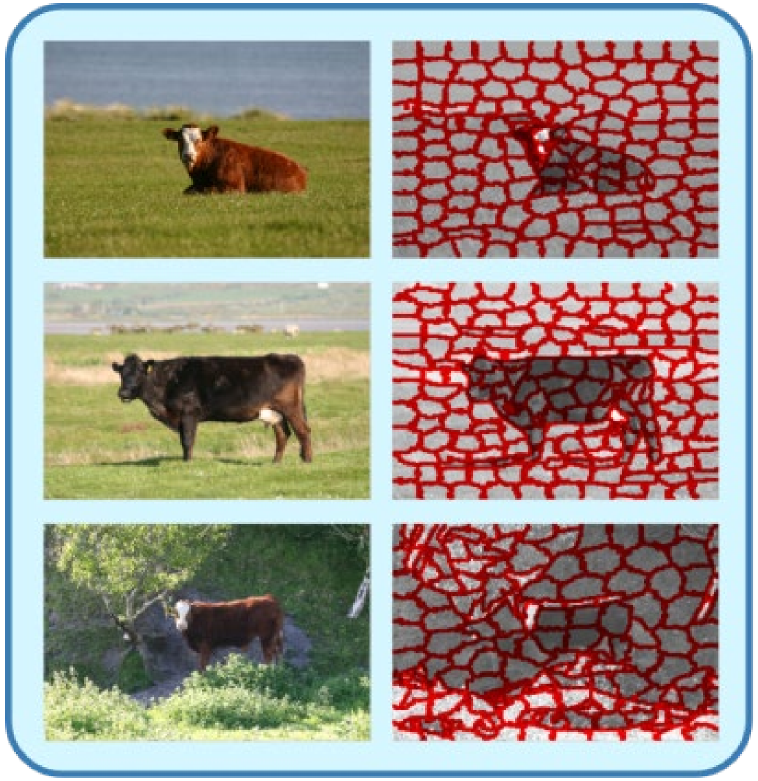
Graph for iCoseg-Ferrari



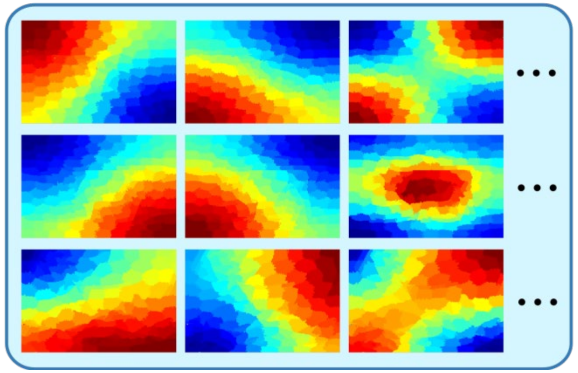
Graph for PASCAL-Plane

Superpixel Representation

- Over-segment images into super-pixels
- Build a graph on super-pixels
  - Nodes: super-pixels
  - Edges weighted by length of shared boundary



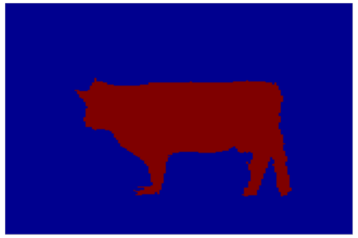
Encoding Functions over Graphs



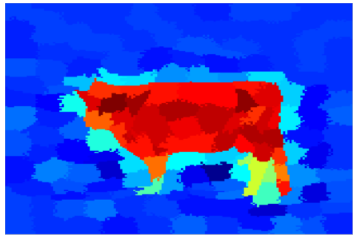
- Basis of functional space
  - : First M Laplacian eigenfunctions of the graph

$$f = \sum_{j=1}^{M} f_j b_i^j = B_i \mathbf{f}$$

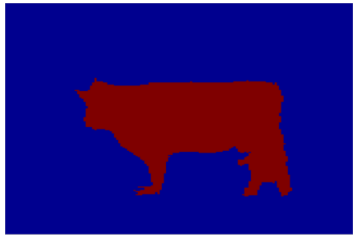
- Reconstruct any function with small error (M=30)



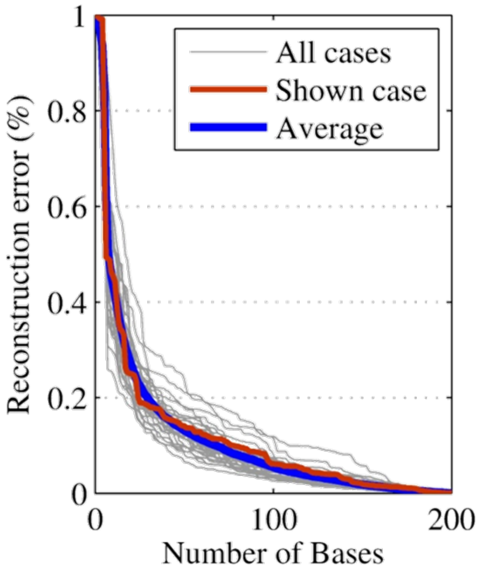
Binary indicator function



Reconstructed function



Thresholded reconstructed function

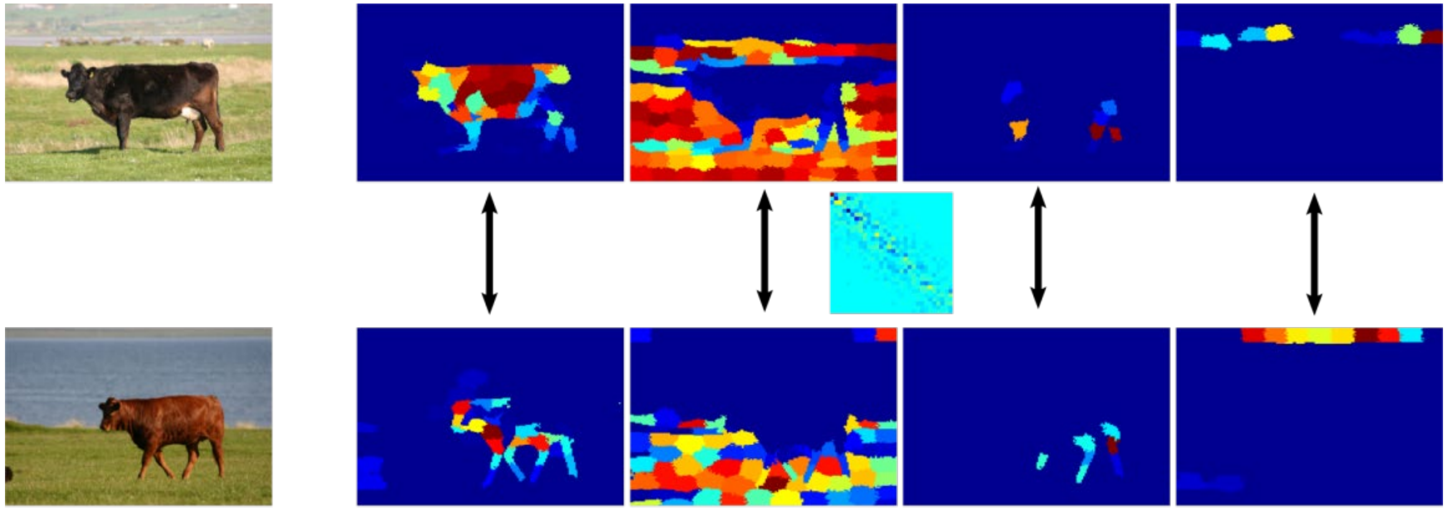


Reconstruction error

Joint Estimation of Functional Maps, I

- Functional map:
  - A linear map between functions in two functional spaces
  - Can be recovered by a set of probe functions

$$\mathbf{f}' = X_{ij}\mathbf{f} \qquad X_{ij} \in \mathcal{R}^{M \times M}$$

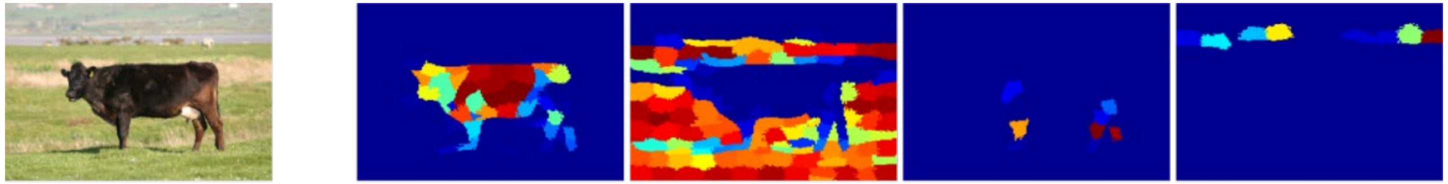


Joint Estimation of Functional Maps, I

- Recover functional maps by aligning image features:

$$f_{ij}^{\text{feature}} = \|X_{ij} D_i - D_j\|_1$$

- Features (probe functions) for each super-pixel:
  - average RGB color, 3-dimensional;
  - 64 dimensional RGB color histogram;
  - 300-dimensional bag-of-visual-words.



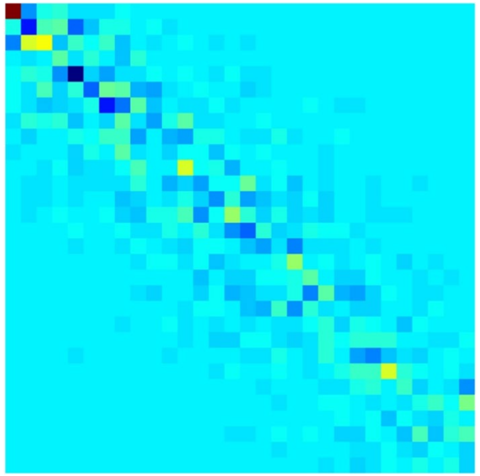
Joint Estimation of Functional Maps, II

- Regularization term:

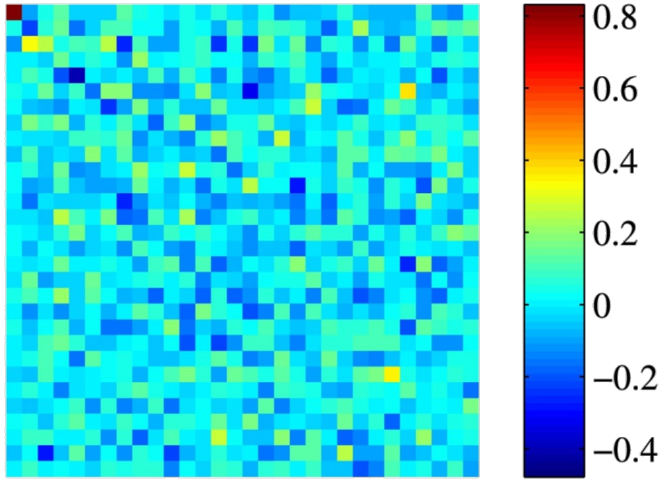
Λ_i, Λ_j diagonal matrices of Laplacian eigenvalues

$$f_{ij}^{\text{reg}} = \|X_{ij}\Lambda_i - \Lambda_j X_{ij}\|^2$$

  - Correspond bases of similar spectra
  - Enforce sparsity of map



Map with regularization



Map without regularization

Joint Estimation of Functional Maps, III

- Incorporating map cycle consistency:
  - A transported function along any loop should be identical to the original function:

$$X_{i_k i_0} \cdots X_{i_1 i_2} X_{i_0 i_1} \mathbf{f} = \mathbf{f} \quad \Longleftrightarrow \quad X_{ij} Y_i = Y_j, \quad \forall (i,j) \in \mathcal{G}$$

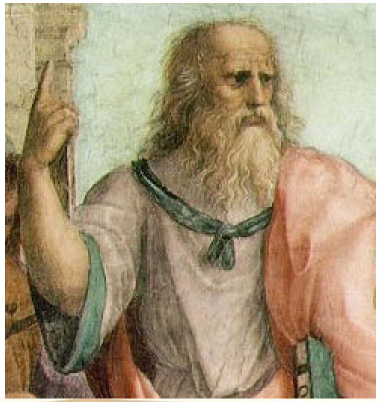
  - Consistency term:

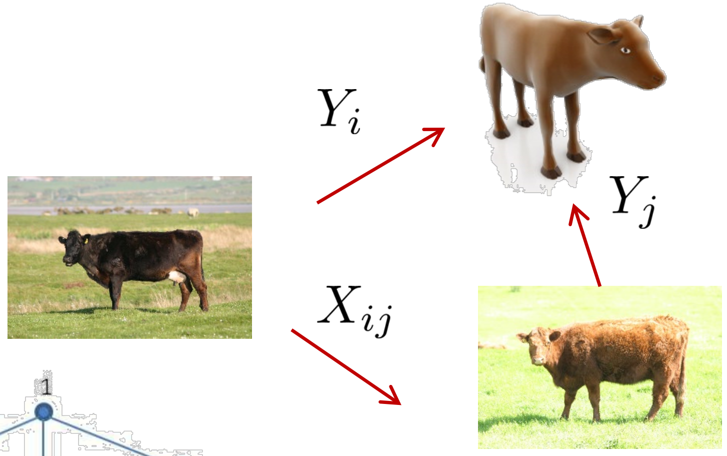
$$f^{\text{cons}} = \sum_{(i,j)\in\mathcal{G}} w_{ij} f_{ij}^{\text{cons}} = \sum_{(i,j)\in\mathcal{G}} w_{ij} \|X_{ij} Y_i - Y_j\|_{\mathcal{F}}^2$$

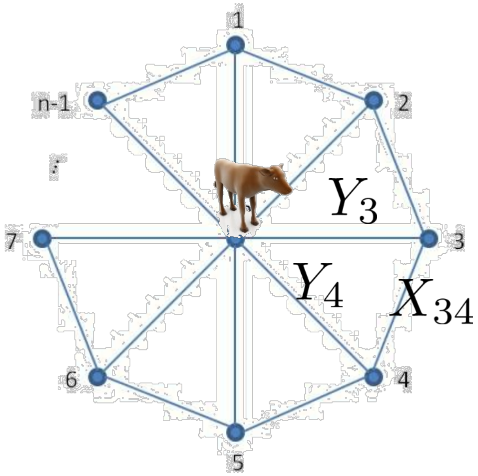
Image global similarity weight via GIST

Joint Estimation of Functional Maps, III

- Plato's allegory of the cave: a latent space

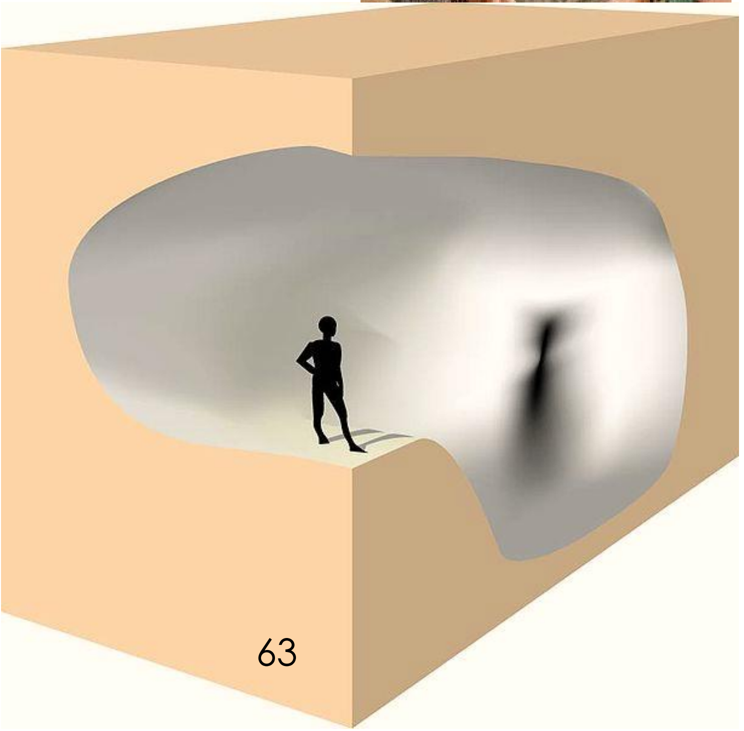






$$X_{ij} \approx Y_j^{-1} Y_i$$

X 30x30, *Y* 30x20



Joint Estimation of Functional Maps, IV

- Overall optimization

$$\min \quad \sum_{(i,j)\in\mathcal{G}} w_{ij}\left(f_{ij}^{\text{feature}} + \mu f_{ij}^{\text{reg}} + \lambda f_{ij}^{\text{cons}}\right)$$
$$s.t. \quad Y^T Y = I_m$$

- Alternating optimization:
  - Fix Y, solve X $\Rightarrow$ Independent QP problems

$$X_{ij}^{\star} = \arg\min_X \left(f_{ij}^{\text{feature}} + \mu f_{ij}^{\text{reg}} + \lambda f_{ij}^{\text{cons}}\right)$$

  - Fix X, solve Y $\Rightarrow$ Eigenvalue problem

$$\min \quad \mathrm{trace}(Y^T W Y)$$
$$s.t. \quad Y^T Y = I_m$$

$$W_{ij} = \begin{cases} \sum_{(i,j')\in\mathcal{G}} w_{ij'}(I_m + X_{ij'}^T X_{ij'}) & i = j \\ -w_{ij}(X_{ji} + X_{ij}^T) & (i,j)\in\mathcal{G} \\ 0 & \text{otherwise.} \end{cases}$$

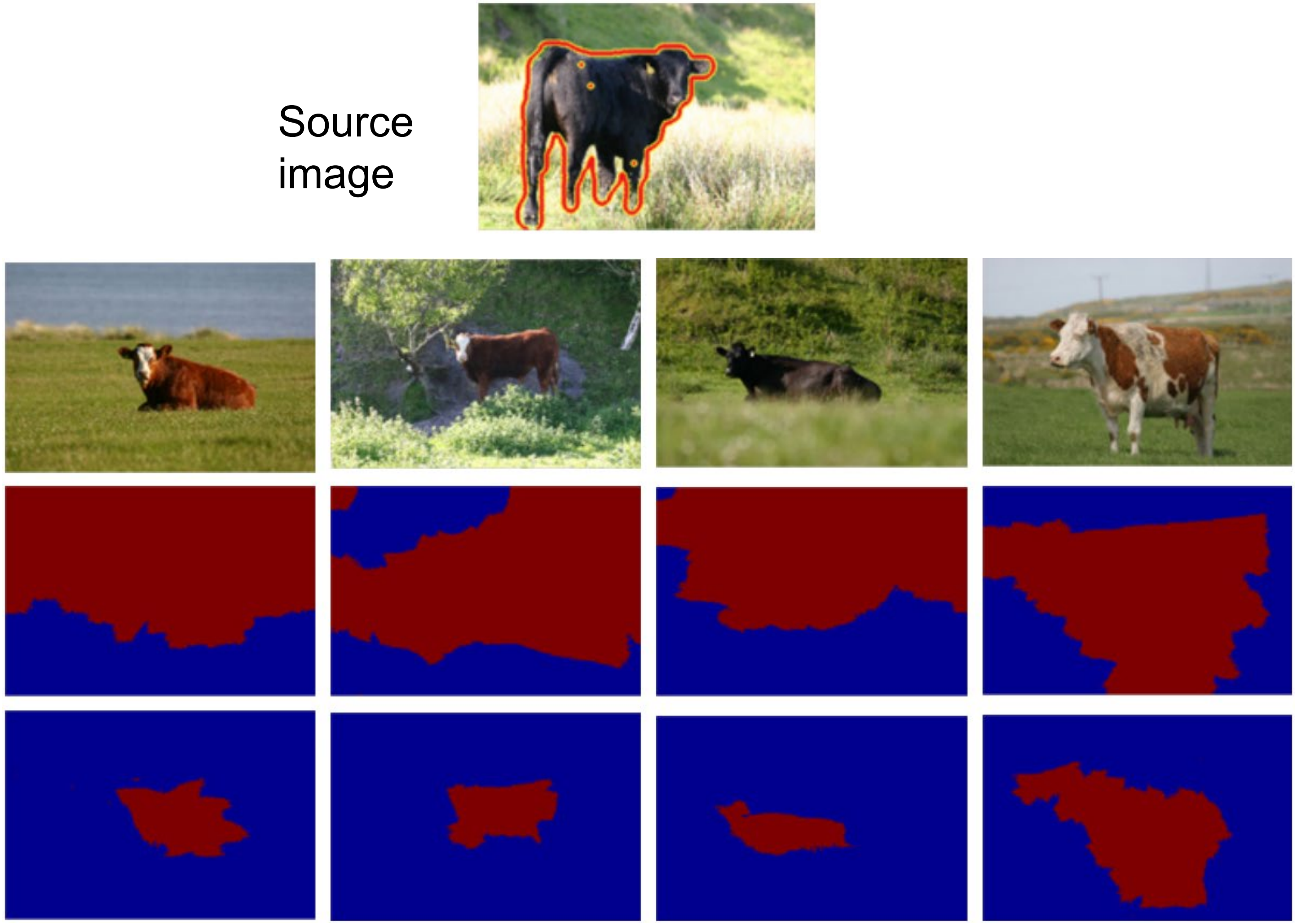
Consistency Matters

Source image

Target image

Without cycle consistency

With cycle consistency



Generating Consistent Segmentations

- Two objectives for segmentation functions
  - consistent under functional map transportation

$$f^{\text{map}} = \sum_{(i,j)\in\mathcal{G}} w_{ij}\|X_{ij}\mathbf{f}_i - \mathbf{f}_j\|_{\mathcal{F}}^2$$

consistent

We look for network fixed points!

  - and agreement with normalized cut scores:

$$f^{\text{seg}} = \sum_{i=1}^{N} \mathbf{f}_i^T B_i^T L_i B_i \mathbf{f}_i$$

Easy to incorporate labeled images with ground truth segmentation

- Joint optimization:

$$\min\ f^{\text{seg}} + \gamma f^{\text{map}} \quad s.t.\ \sum_{i=1}^{N} \|\mathbf{f}_i\|^2 = 1$$

Eigen-decomposition problem

Experiments

- iCoseg dataset
  - Very similar or the same object in each class;
  - 5~10 images per class.
- MSRC dataset
  - Similar objects in each class;
  - ~30 images per class.
- PASCAL data set
  - Retrieved from PASCAL VOC 2012 challenge;
  - All images with the same object label;
  - Larger scale;
  - Larger variability.

- iCoseg data set
- New unsupervised method
  - Mostly outperforms other unsupervised methods
  - Sometimes even outperforms supervised methods
  - Supervised input is easily added and further improves the results

Kuettel'12 (Supervised)		Unsupervised Fmaps
Image+transfer	Full model	
87.6	91.4	90.5

Supervised method

Class	Joulin '10	Rubio '12	Vicente '11	Fmaps -uns
Alaska Bear	74.8	86.4	90.0	**90.4**
Red Sox Players	73.0	90.5	90.9	**94.2**
Stonehenge1	56.6	87.3	63.3	**92.5**
Stonehenge2	86.0	88.4	88.8	87.2
Liverpool FC	76.4	82.6	87.5	**89.4**
Ferrari	85.0	84.3	89.9	**95.6**
Taj Mahal	73.7	88.7	91.1	**92.6**
Elephants	70.1	75.0	43.1	**86.7**
Pandas	84.0	60.0	92.7	88.6
Kite	87.0	89.8	90.3	**93.9**
Kite panda	73.2	78.3	90.2	**93.1**
Gymnastics	90.9	87.1	91.7	90.4
Skating	82.1	76.8	77.5	78.7
Hot Balloons	85.2	89.0	90.1	**90.4**
Liberty Statue	90.6	91.6	93.8	**96.8**
Brown Bear	74.0	80.4	95.3	88.1
Average	78.9	83.5	85.4	**90.5**

- MSRC

Unsupervised performance comparison

Class	N	Joulin '10	Rubio '12	Fmaps -uns
Cow	30	81.6	80.1	**89.7**
Plane	30	73.8	77.0	**87.3**
Face	30	84.3	76.3	**89.3**
Cat	24	74.4	77.1	**88.3**
Car(front)	6	87.6	65.9	87.3
Car(back)	6	85.1	52.4	**92.7**
Bike	30	63.3	62.4	**74.8**

Supervised performance comparison

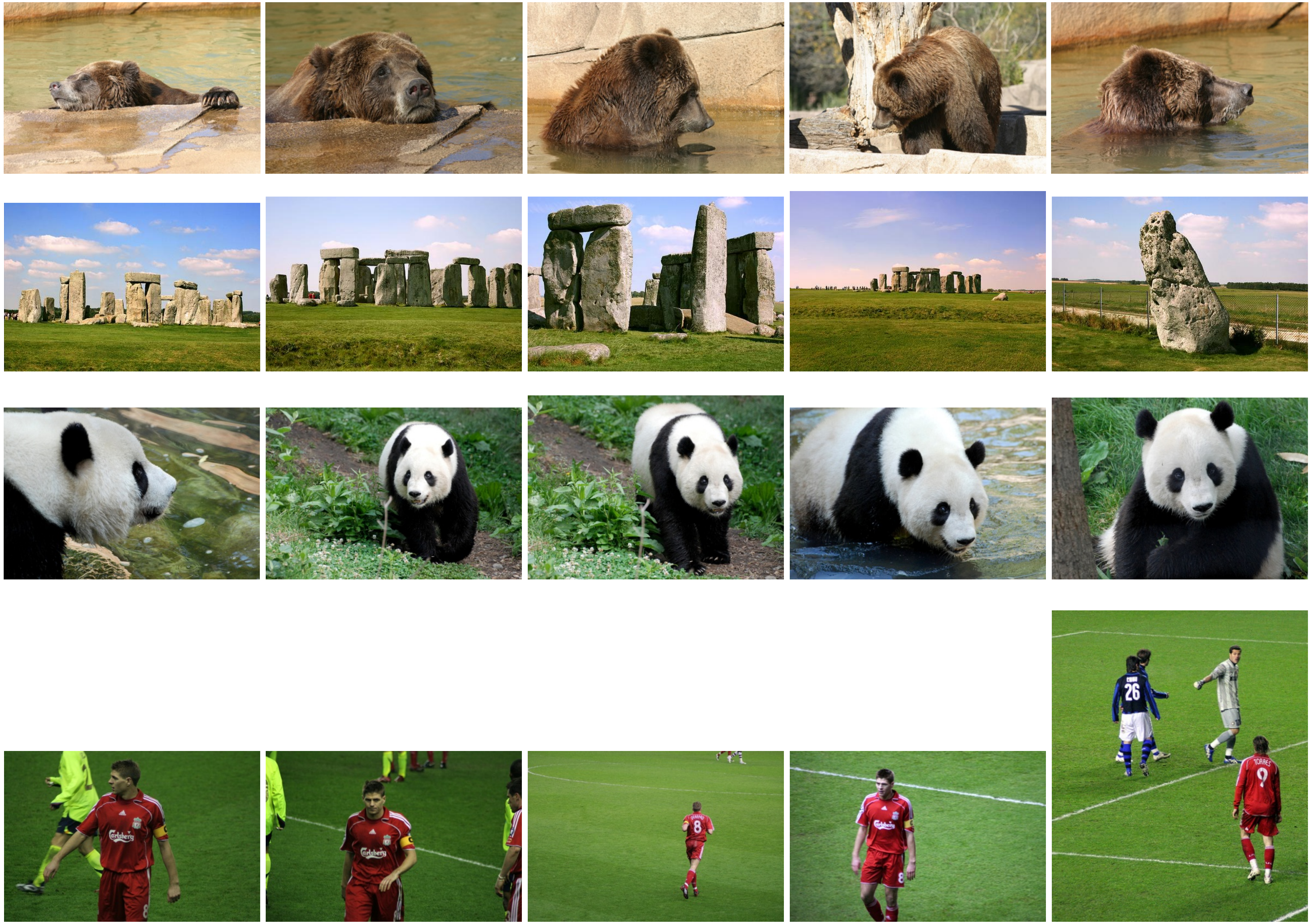
Class	Vicente '11	Kuettel '12	Fmaps -s
Cow	94.2	92.5	**94.3**
Plane	83.0	86.5	**91.0**
Car	79.6	88.8	83.1
Sheep	94.0	91.8	**95.6**
Bird	95.3	93.4	**95.8**
Cat	92.3	92.6	**94.5**
Dog	93.0	87.8	91.3

- PASCAL

Class	N	L	Kuettel '12	Fmaps -s	Fmaps -uns
Plane	178	88	90.7	**92.1**	89.4
Bus	152	78	81.6	**87.1**	80.7
Car	255	128	76.1	**90.9**	82.3
Cat	250	131	77.7	**85.5**	82.5
Cow	135	64	82.5	**87.7**	85.5
Dog	249	121	81.9	**88.5**	84.2
Horse	147	68	83.1	**88.9**	87.0
Sheep	120	63	83.9	**89.6**	86.5

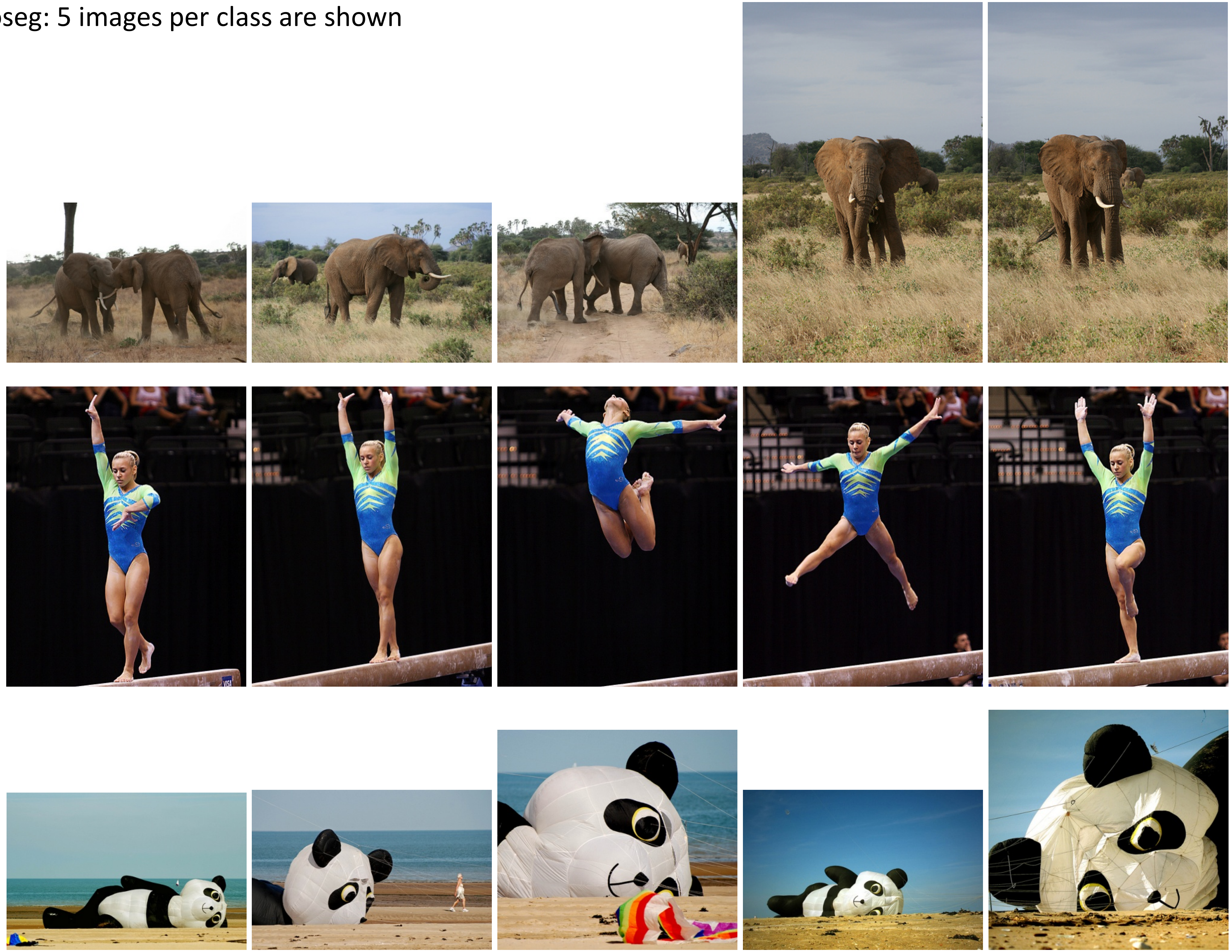
- New method mostly outperforms the state-of-the-art techniques in both supervised and unsupervised settings

iCoseg: 5 images per class are shown

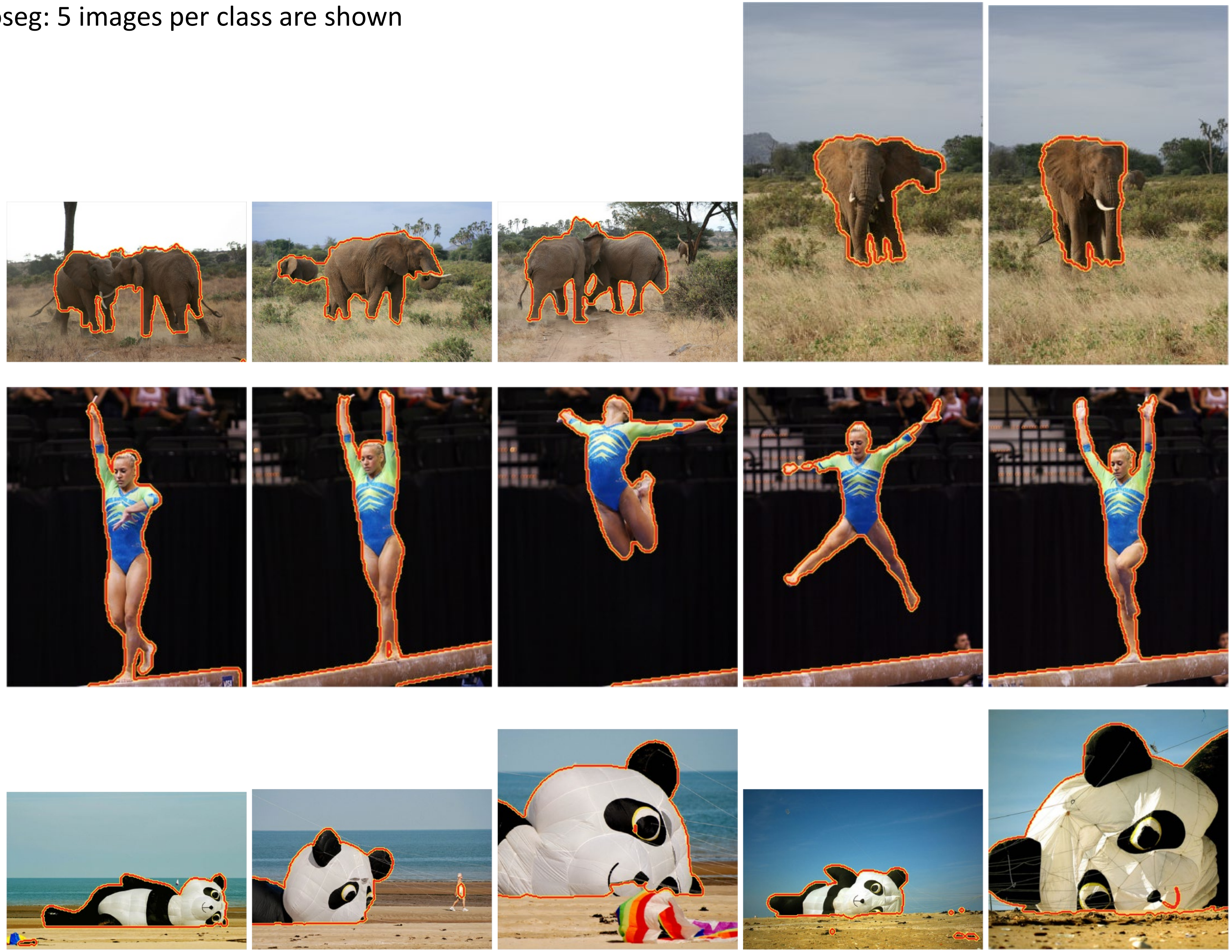


iCoseg: 5 images per class are shown

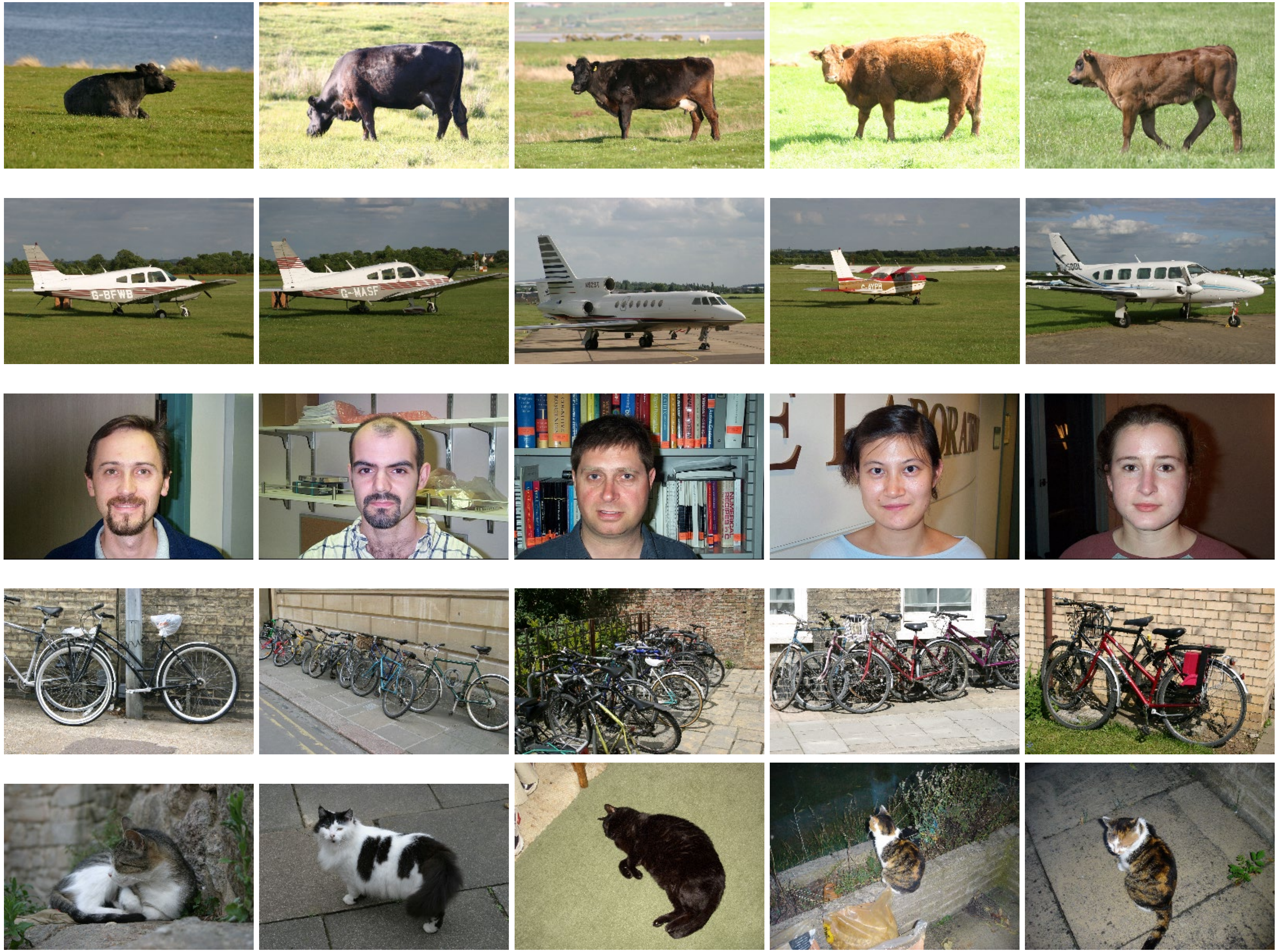
iCoseg: 5 images per class are shown



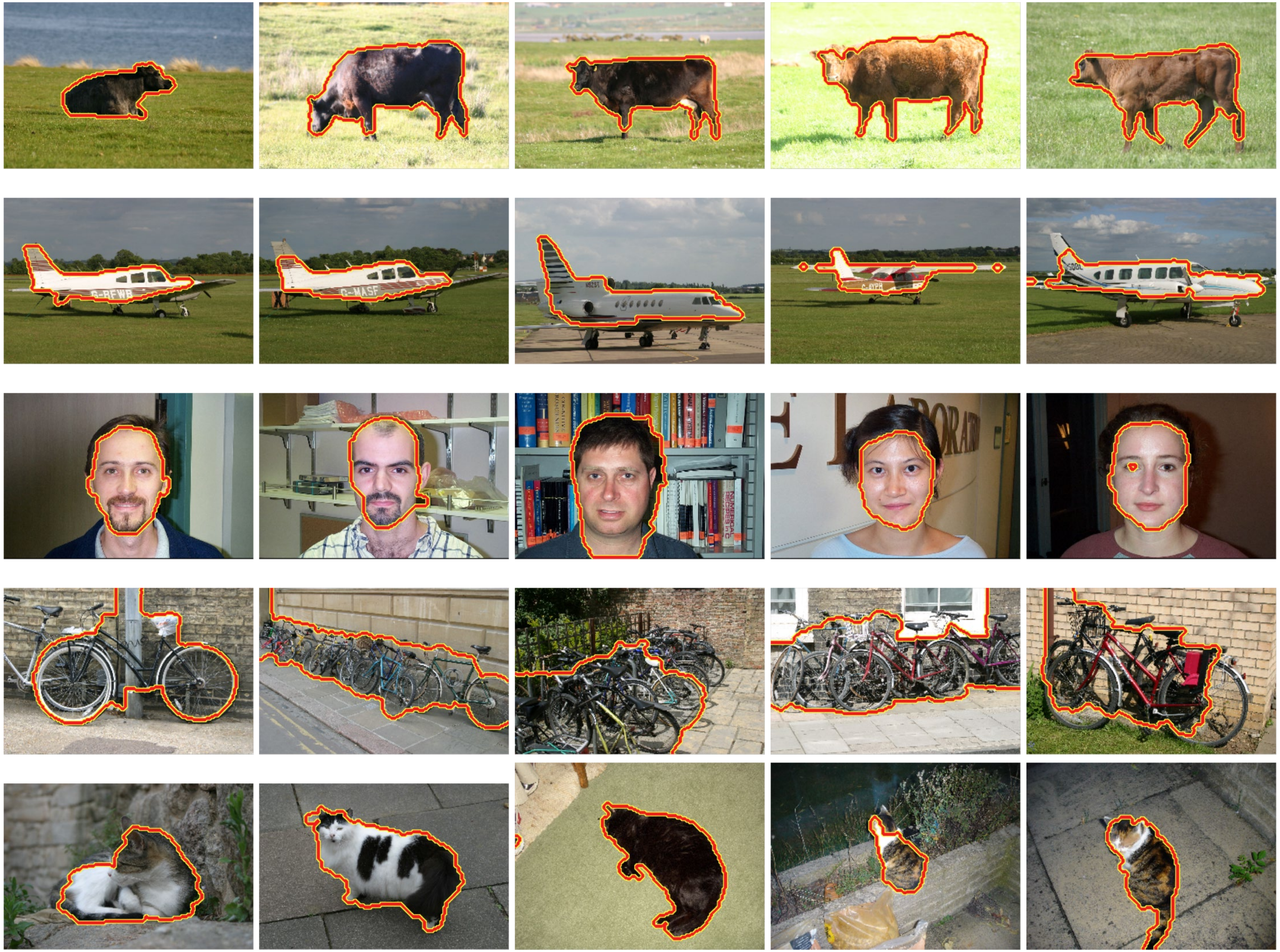
iCoseg: 5 images per class are shown



MSRC: 5 images per class are shown

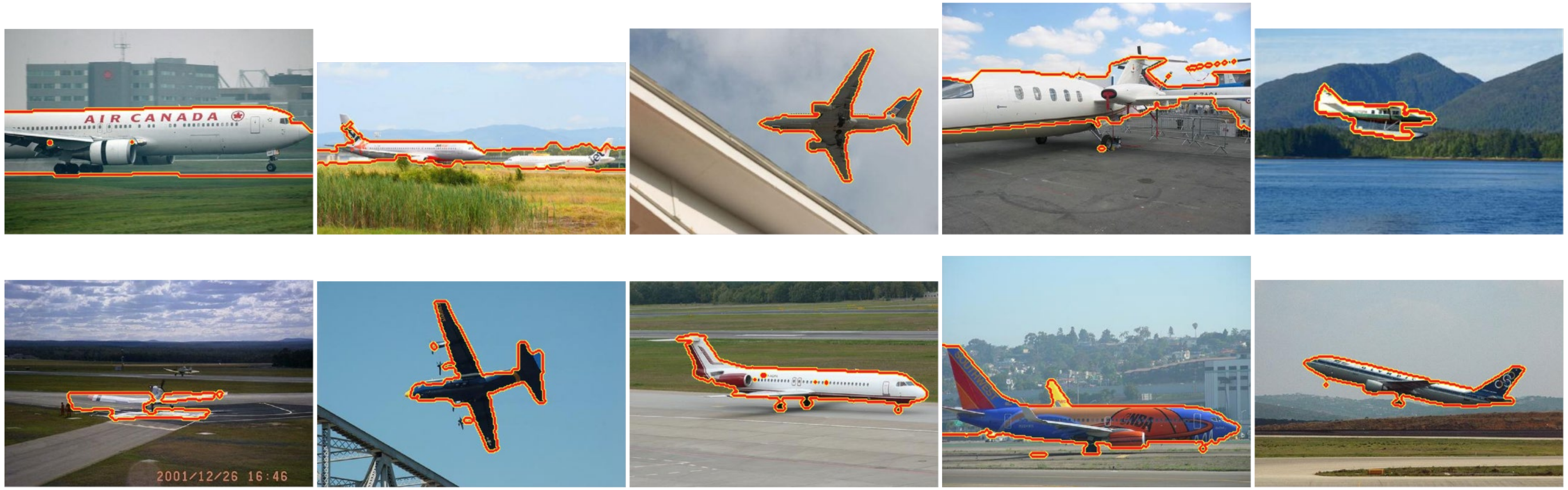


MSRC: 5 images per class are shown



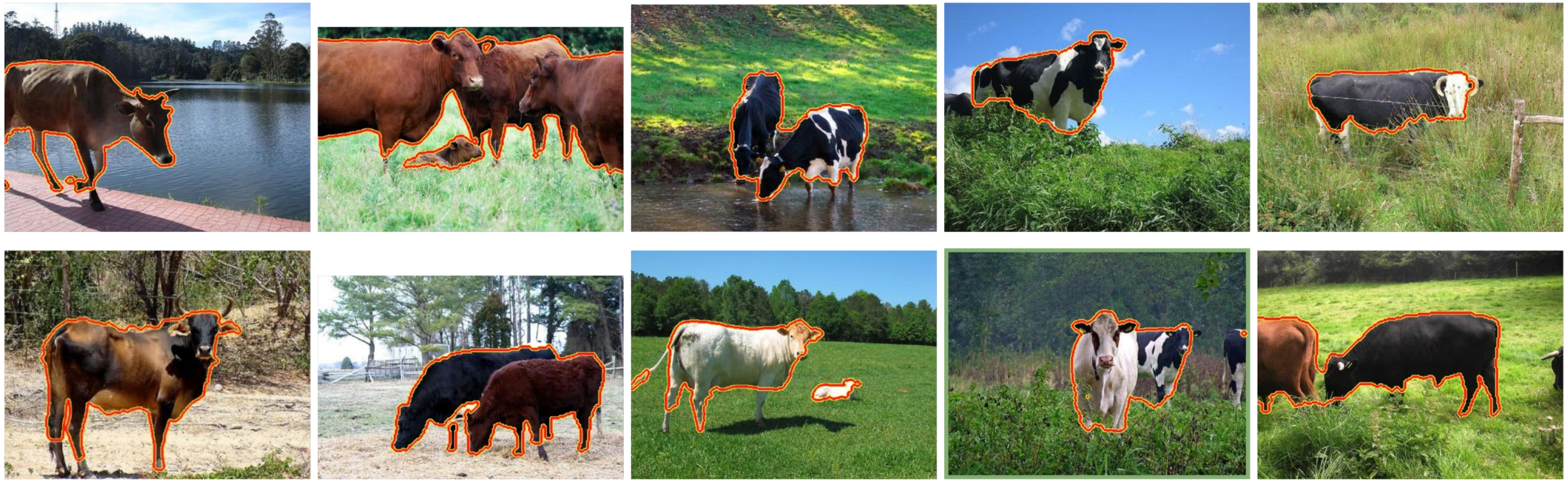
PASCAL: 10 images per class are shown

PASCAL: 10 images per class are shown



PASCAL: 10 images per class are shown

PASCAL: 10 images per class are shown



Multi-Class Co-Segmentation

[F. Wang, Q. Huang, M. Ovsjanikov, L. G., CVPR’14]

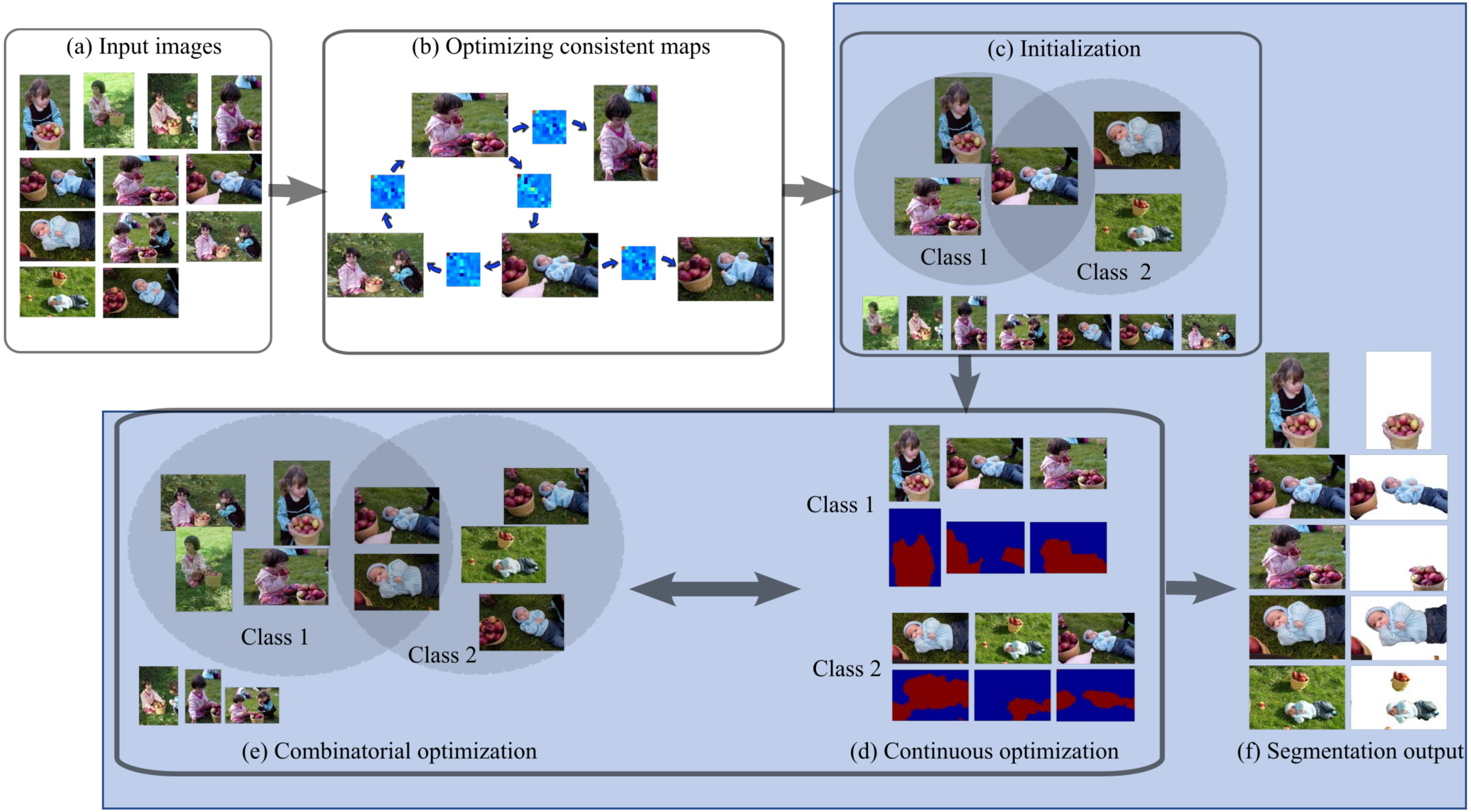
- Input:
  - A collection of N images sharing M objects
  - Each image contains a subset of the objects

- Output
  - Discovery of what objects appear in each image
  - Their pixel-level segmentation





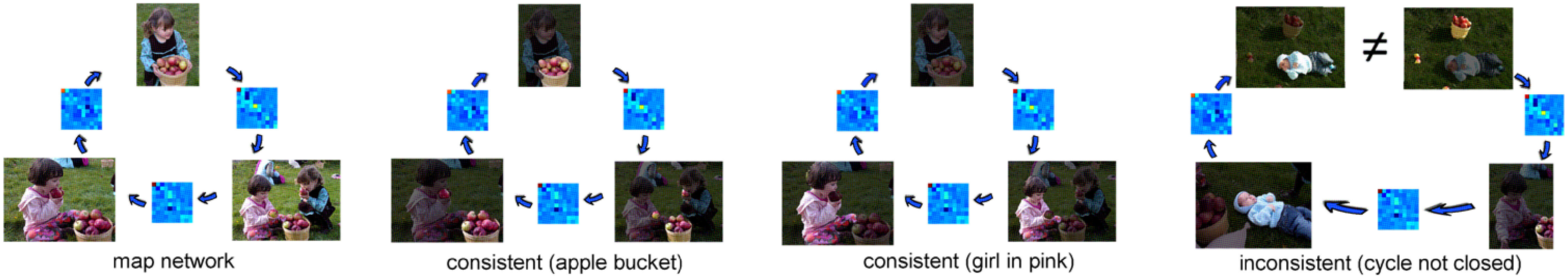
Framework



(a) Input images
(b) Optimizing consistent maps
(c) Initialization
Class 1
Class 2
Class 1
Class 2
Class 1
Class 2
(e) Combinatorial optimization
(d) Continuous optimization
(f) Segmentation output

Consistent Functional Maps

- Partial cycle consistency:



Must deal with non-total maps

Related to topological persistence / persistent homology

Consistent Functional Maps

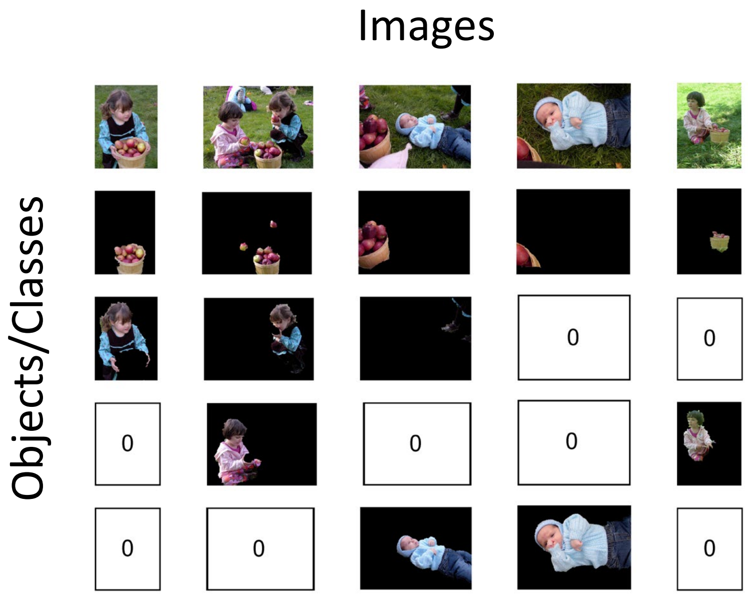
- Latent functions:
- Discrete variables:
- Relationship:
- Consistency:

$$\boldsymbol{Y}_i = (\boldsymbol{y}_{i1}, \cdots, \boldsymbol{y}_{iL})$$

$$\boldsymbol{z}_i = \{z_{il} \in \{0, 1\}, 1 \Box l \Box L\}$$

$$\mathbf{Y}_i \mathrm{Diag}(\mathbf{z}_i) = \mathbf{Y}_i$$

$$\boldsymbol{X}_{ij}\boldsymbol{Y}_i = \boldsymbol{Y}_j \mathrm{Diag}(\boldsymbol{z}_i), \quad (i, j) \in \mathcal{E}.$$



Optimizing Segmentation Functions

- Alternating between:
  - Continuous optimization:
    - Optimal segmentation functions in each class
  - Combinatorial optimization:
    - Class assignment by propagating segmentation functions

Continuous Optimization

- Optimize segmentations in each object class
  - Consistent with functional maps
  - Align with segmentation cues
  - Mutually exclusive

$$
\begin{aligned}
\min_{s_{ik}, i \in \mathcal{C}_k} \quad & \sum_{k=1}^{M} \sum_{(i,j) \in \mathcal{E} \cap (\mathcal{C}_k \times \mathcal{C}_k)} \|X_{ij} s_{ik} - s_{jk}\|^2 \\
& + \gamma \sum_{l \neq k} \sum_{i \in \mathcal{C}_k \cap \mathcal{C}_l} (s_{il}^T s_{ik})^2 + \mu \sum_{k=1}^{M} \sum_{i \in \mathcal{C}_k} s_{ik}^T L_i s_{ik} \\
\text{subject to} \quad & \sum_{i \in \mathcal{C}_k} \|s_{ik}\|^2 = |\mathcal{C}_k|, \quad 1 \leq k \leq K.
\end{aligned}
$$

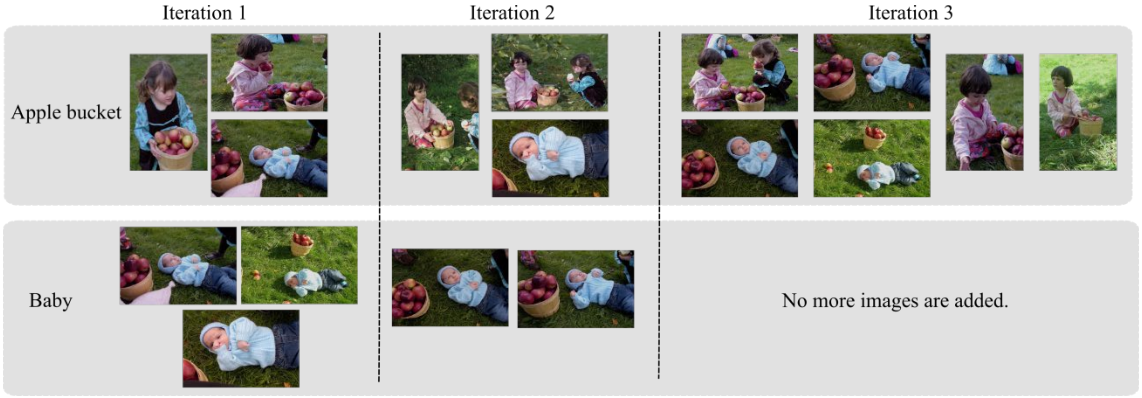
Combinatorial Optimization

- Expand each object class by propagating segmentations to other images

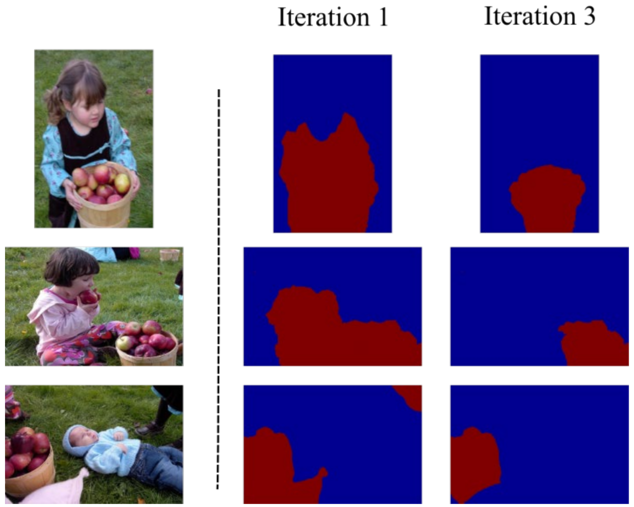
$$
\begin{aligned}
\max_{s_{ik}} \quad & \frac{1}{|\mathcal{N}(i) \cap \mathcal{C}_k|} \sum_{j \in \mathcal{N}(i) \cap \mathcal{C}_k} (s_{ik}^T X_{ji} s_{jk})^2 \\
& - \gamma \sum_{l \neq k, i \in \mathcal{C}_l} (s_{ik}^T s_{il})^2 - \mu s_{ik}^T L_i s_{ik} \\
\text{subject to} \quad & \|s_{ik}\|^2 = 1
\end{aligned}
$$

Optimizing Segmentation Functions

- More images will be included in each object class



- Segmentation functions are improved during iterations



Experimental Results

- Accuracy
  - Intersection-over-union
  - Find the best one-to-one matching between each cluster and each ground-truth object.
- Benchmark datasets
  - MSRC: 30 images, 1 class (degenerated case);
  - FlickrMFC data set: 20 images, 3~6 classes
  - PASCAL VOC: 100~200 images, 2 classes

Experimental Results

class	N	M	Kim’12	Kim’11	Joulin ’10	Mukherjee ’11	Ours
Apple	20	6	40.9	32.6	24.8	25.6	**46.6**
Baseball	18	5	31.0	31.3	19.2	16.1	**50.3**
butterfly	18	8	29.8	32.4	29.5	10.7	**54.7**
Cheetah	20	5	32.1	40.1	50.9	41.9	**62.1**
Cow	20	5	35.6	43.8	25.0	27.2	**38.5**
Dog	20	4	34.5	35.0	32.0	30.6	**53.8**
Dolphin	18	3	34.0	47.4	37.2	30.1	**61.2**
Fishing	18	5	20.3	27.2	19.8	18.3	**46.8**
Gorilla	18	4	41.0	38.8	41.1	28.1	**47.8**
Liberty	18	4	31.5	41.2	44.6	32.1	**58.2**
Parrot	18	5	29.9	36.5	35.0	26.6	**54.1**
Stonehenge	20	5	35.3	49.3	47.0	32.6	**54.6**
Swan	20	3	17.1	18.4	14.3	16.3	**46.5**
Thinker	17	4	25.6	34.4	27.6	15.7	**68.6**
Average	-	-	31.3	36.3	32.0	25.1	**53.1**

Performance comparison on the MFCFlickr dataset

class	N	NCut	MNcut	Ours
Bike + person	248	27.3	30.5	**40.1**
Boat + person	260	29.3	32.6	**44.6**
Bottle + dining table	90	37.8	39.5	**47.6**
Bus + car	195	36.3	39.4	**49.2**
bus + person	243	38.9	41.3	**55.5**
Chair + dining table	134	32.3	30.8	**40.3**
Chair + potted plant	115	19.7	19.7	**22.3**
Cow + person	263	30.5	33.5	**45.0**
Dog + sofa	217	44.6	42.2	**49.6**
Horse + person	276	27.3	30.8	**42.1**
Potted plant + sofa	119	37.4	37.5	**40.7**

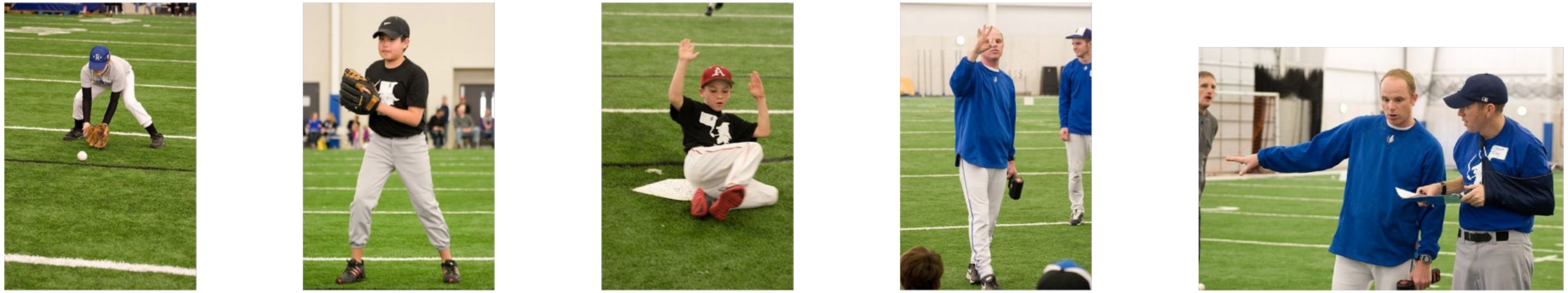
Performance comparison on the PASCAL-multi dataset

class	N	Joulin’10	Kim’11	Mukherjee’11	Ours
Bike	30	43.3	29.9	42.8	**51.2**
Bird	30	47.7	29.9	-	**55.7**
Car	30	59.7	37.1	52.5	**72.9**
Cat	24	31.9	24.4	5.6	**65.9**
Chair	30	39.6	28.7	39.4	**46.5**
Cow	30	52.7	33.5	26.1	**68.4**
Dog	30	41.8	33.0	-	**55.8**
Face	30	70.0	33.2	40.8	60.9
Flower	30	51.9	40.2	-	**67.2**
House	30	51.0	32.2	66.4	56.6
Plane	30	21.6	25.1	33.4	**52.2**
Sheep	30	66.3	60.8	45.7	**72.2**
Sign	30	58.9	43.2	-	**59.1**
Tree	30	67.0	61.2	55.9	**62.0**

Performance comparison on the MSRC dataset

Apple + picking

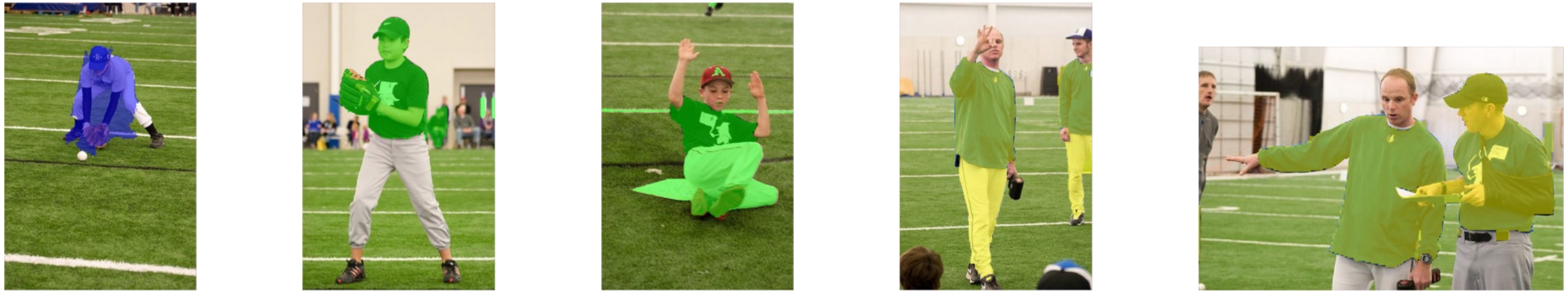
Baseball + kids



Butterfly + blossom

Apple + picking (red: apple bucket; magenta: girl in red; yellow: girl in blue; green: baby; cyan: pumpkin.)

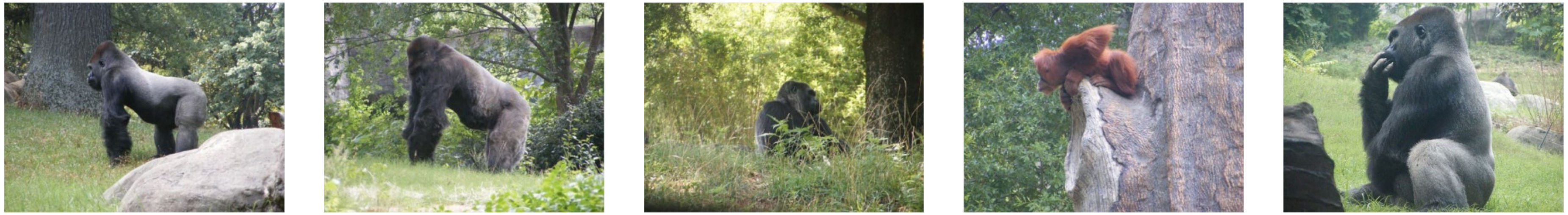
Baseball + kids (green: boy in black; blue: boy in grey; yellow: coach.)



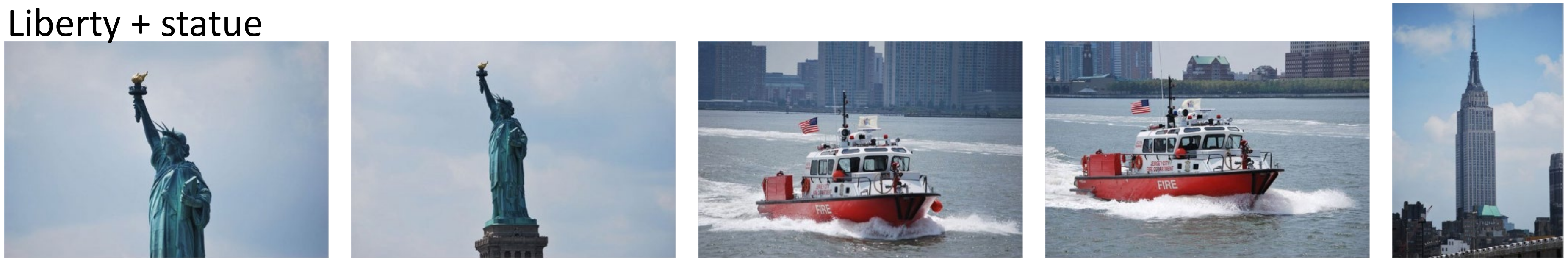
Butterfly + blossom (green: butterfly in orange; yellow: butterfly in yellow; cyan: red flower.)

Fishing + Alaska

Gorilla + zoo



Liberty + statue

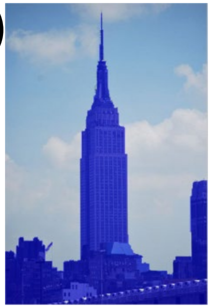


Parrot + zoo

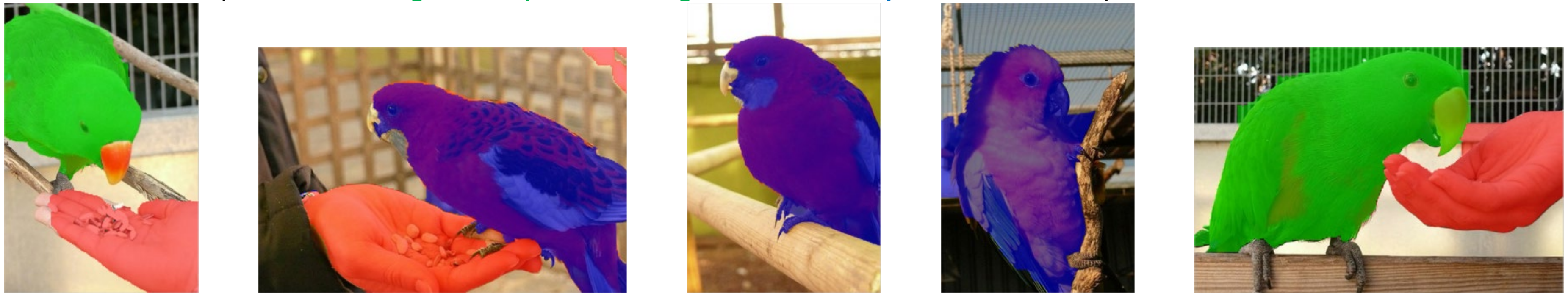
Fishing + Alaska (blue: man in white; green: man in gray; magenta: woman in gray; yellow: salmon.)

Gorilla + zoo (blue: gorilla; yellow: brown orangutan)

Liberty + statue (blue: empire state building; green: red boat; yellow: liberty statue.)



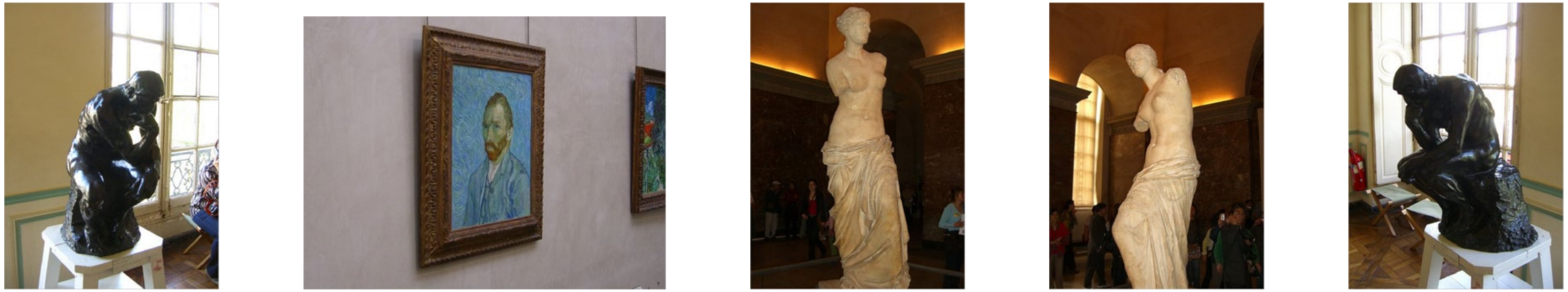
Parrot + zoo (red: hand; green: parrot in green; blue: parrot in red.)



Stonehenge

Swan + zoo

Thinker + Rodin



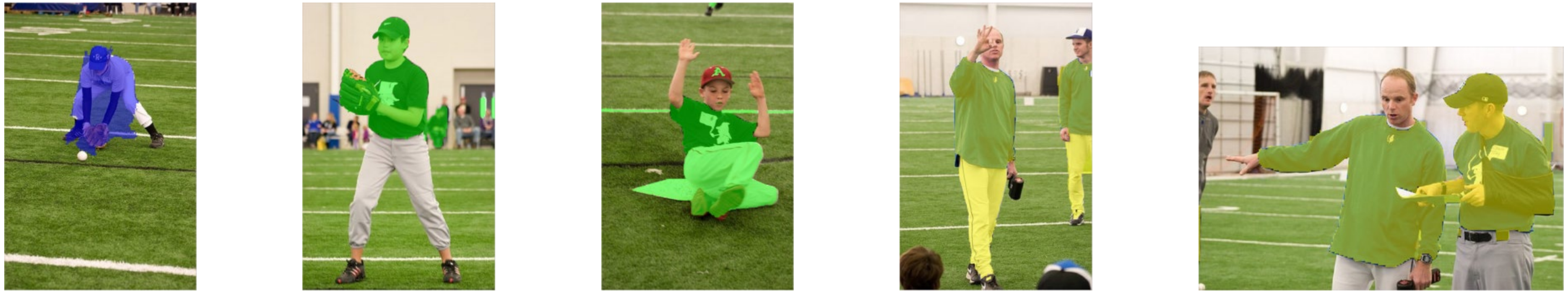
Stonehenge (blue: cow in white; yellow: person; magenta: stonehenge.)

Swan + zoo (blue: gray swan; green: black swan.)

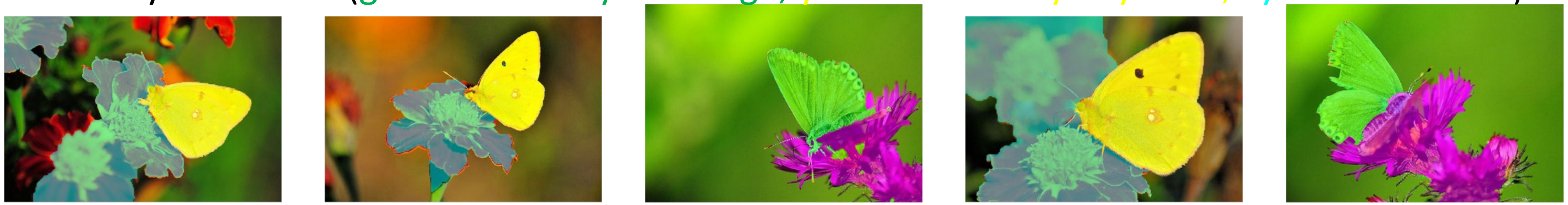
Thinker + Rodin (red: sculpture Thinker; green: sculpture Venus; blue: Van Gogh.)

Apple + picking (red: apple bucket; magenta: girl in red; yellow: girl in blue; green: baby; cyan: pumpkin.)

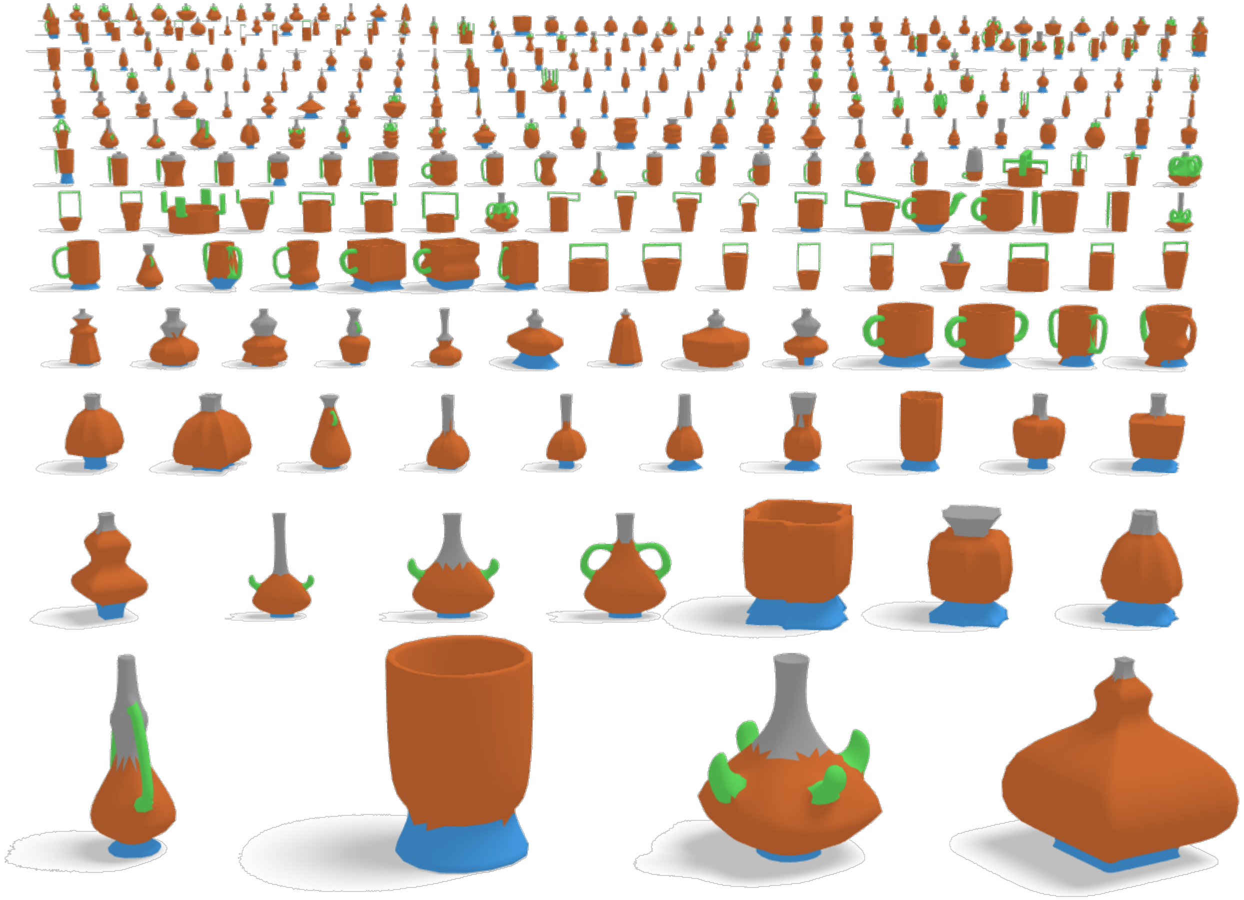
Baseball + kids (green: boy in black; blue: boy in grey; yellow: coach.)



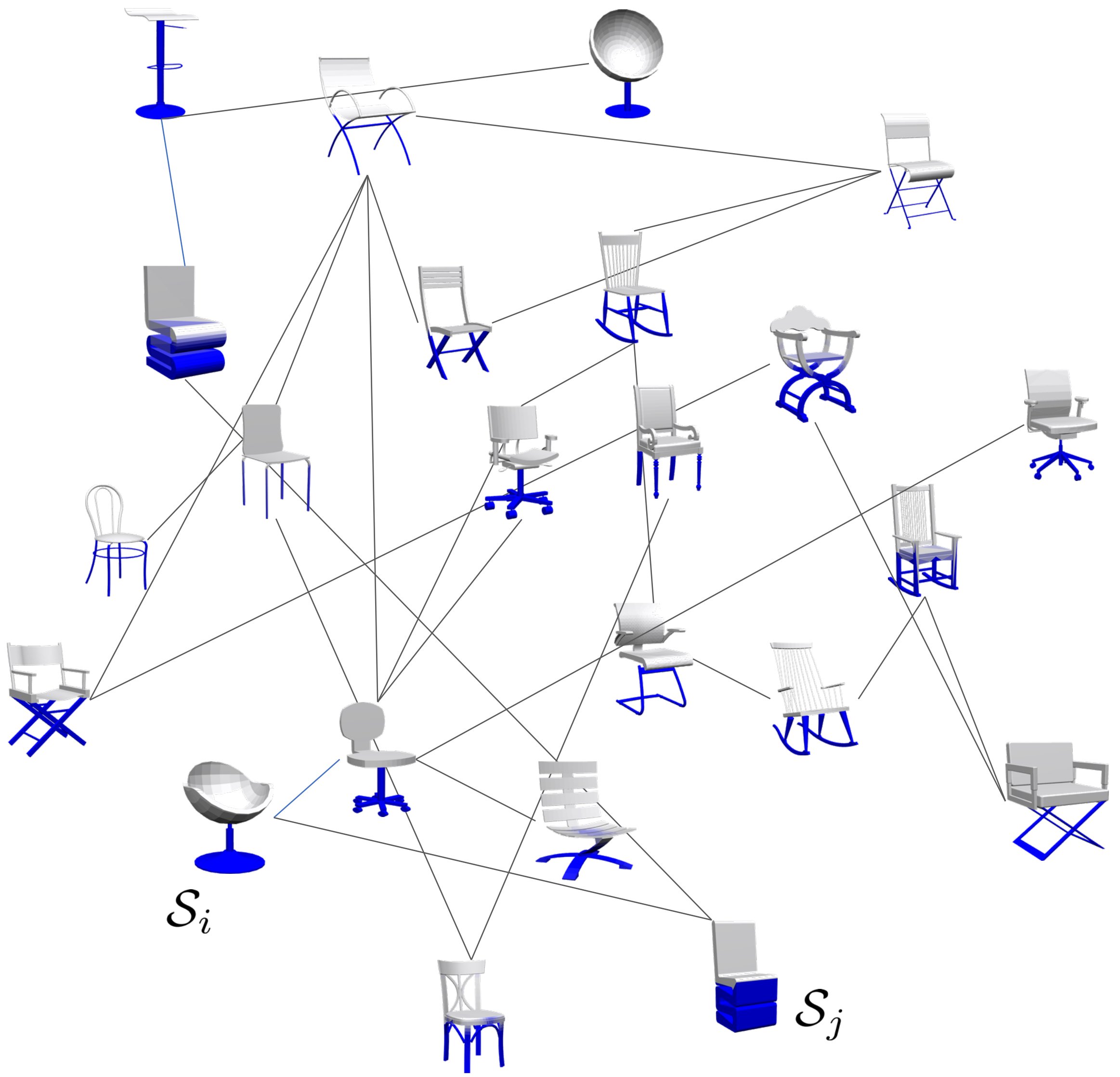
Butterfly + blossom (green: butterfly in orange; yellow: butterfly in yellow; cyan: red flower.)



Consistent Shape Segmentation



First Build a Network

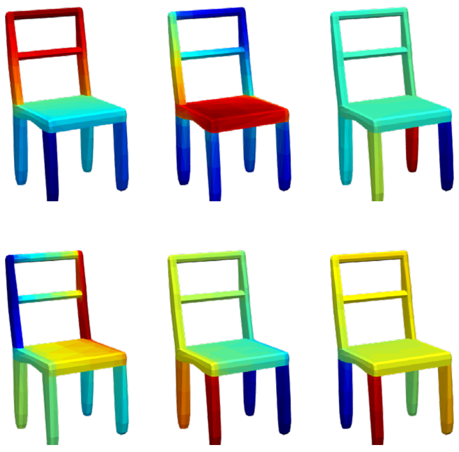


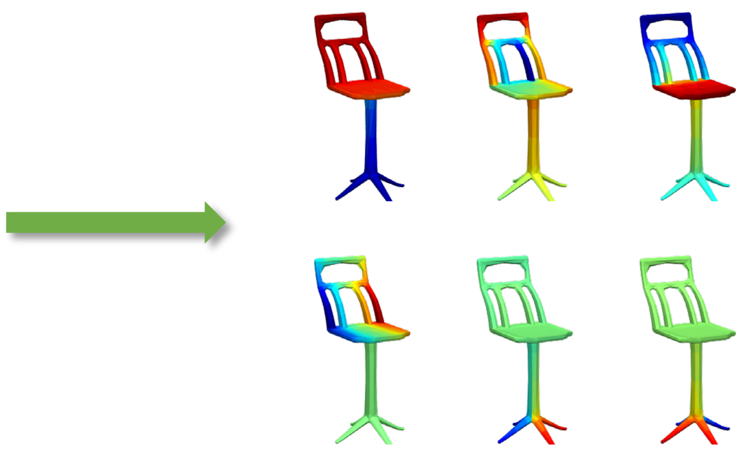
distance histogram

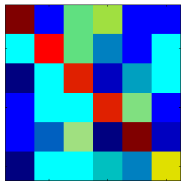
Use the D2 shape descriptor and connect each shape to its nearest neighbors

$$\mathcal{G} = (\mathcal{F}, \mathcal{E})$$

Start From Noisy Shape Descriptor Correspondences

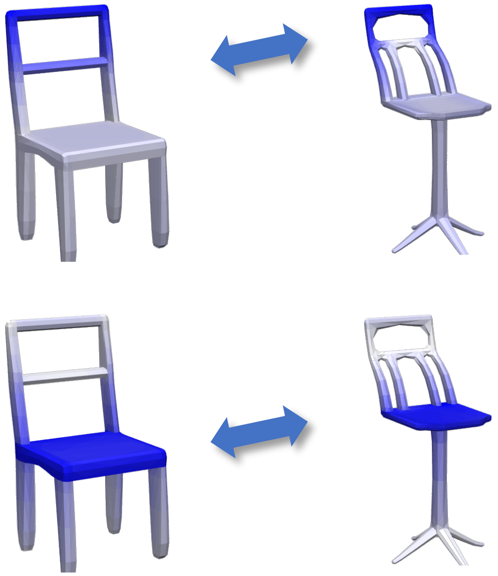






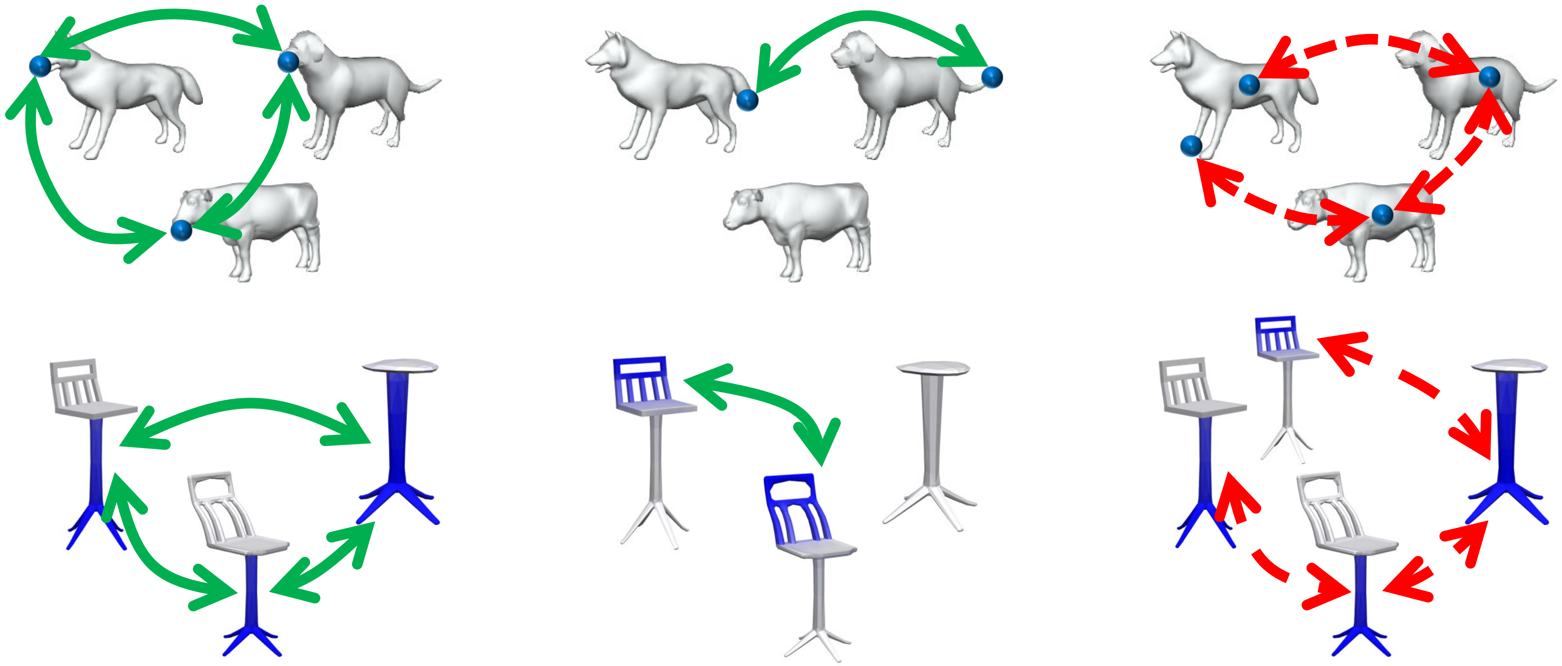
Lift to
functional form

$$C_i X_{ij} \approx D_j$$

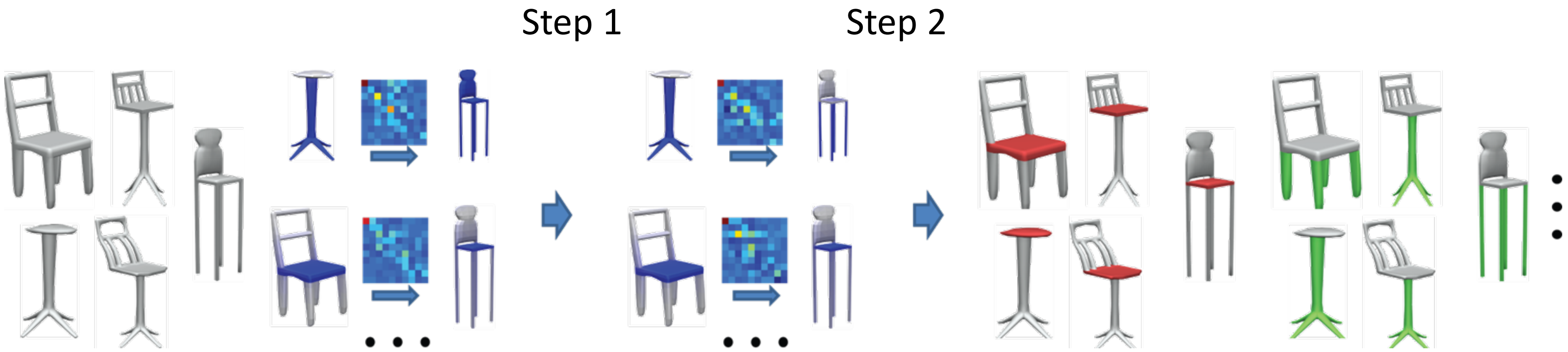


$C_i \bullet \quad \bullet \quad \bullet \; D_i$

Cycle Consistency Under Partial Similarity



The Pipeline



Original shapes
with noisy maps

Cleaned up maps

Consistent basis functions
extracted

Joint Map Optimization

- Step 1: Convex low-rank recovery using robust PCA – we minimize over all *X*

convex!

trace norm
$\|X\|_\star = \sum_i \sigma_i(X)$

$$X^\star = \lambda\|X\|_\star + \min_X \sum_{(i,j)\in\mathcal{G}} \|X_{ij}C_{ij} - D_{ij}\|_{2,1}$$

$\|A\|_{2,1} = \sum_i \|\vec{a}_i\|$

Dual ADMM

- Step 2: Perturb the above *X* to force the factorization

$$\sum_{1\le i,j\le N} \|X^\star_{ij} - Y_j^+ Y_i\|_F^2 + \mu \sum_{i=1}^{N} \sum_{1\le k<l\le L} (\mathbf{y}_{ik}^T \mathbf{y}_{il})^2$$

Non-linear least squares
Gauss-Newton descent

The Y_i give us the desired latent spaces

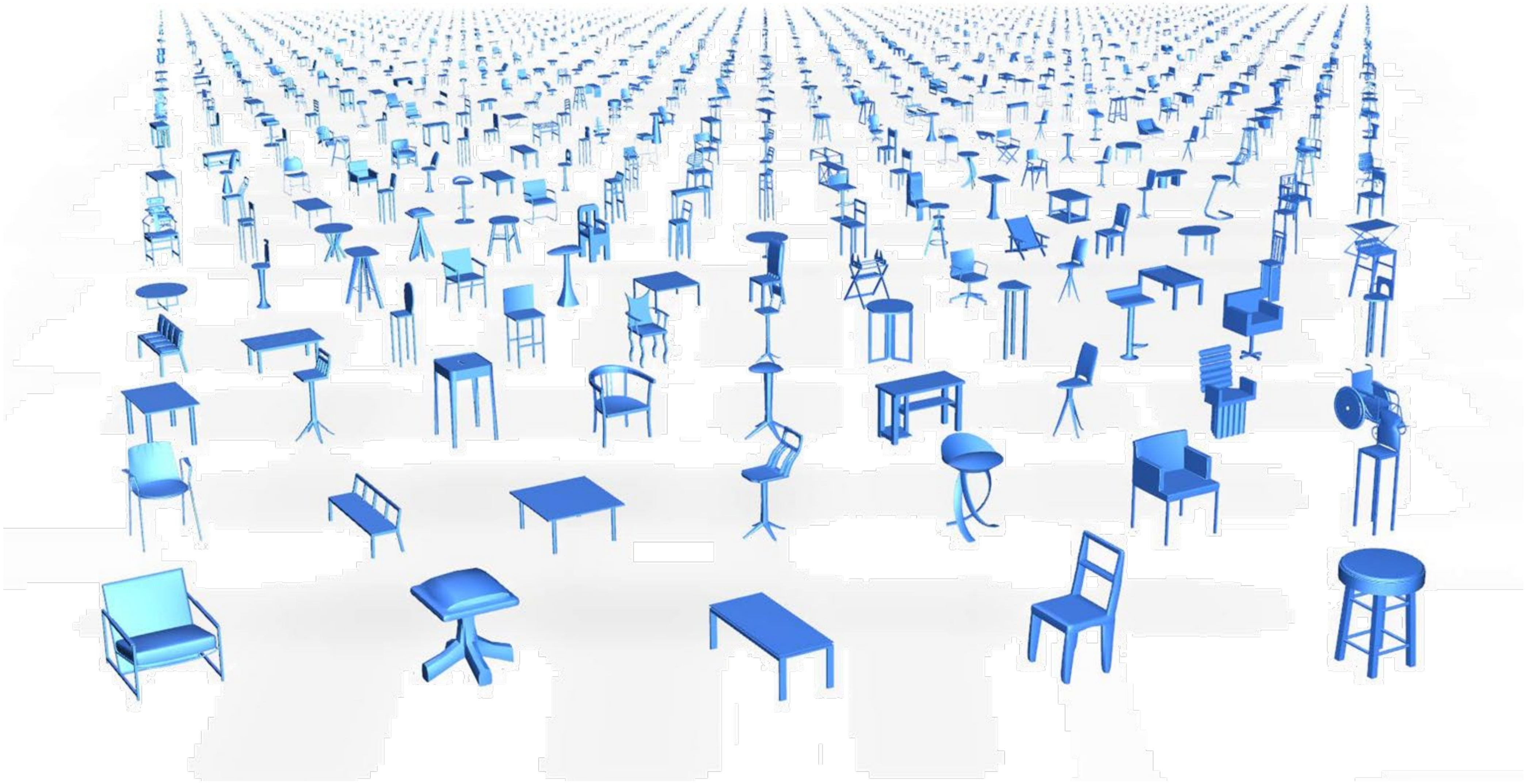
Low-Rank Matrix Factorization

$$X := \begin{pmatrix} X_{11} & \cdots & X_{N1} \\ \vdots & \ddots & \vdots \\ X_{1N} & \cdots & X_{NN} \end{pmatrix} = \begin{pmatrix} Y_1^{+} \\ \vdots \\ Y_N^{+} \end{pmatrix} \begin{pmatrix} Y_1 & \cdots & Y_N \end{pmatrix}$$

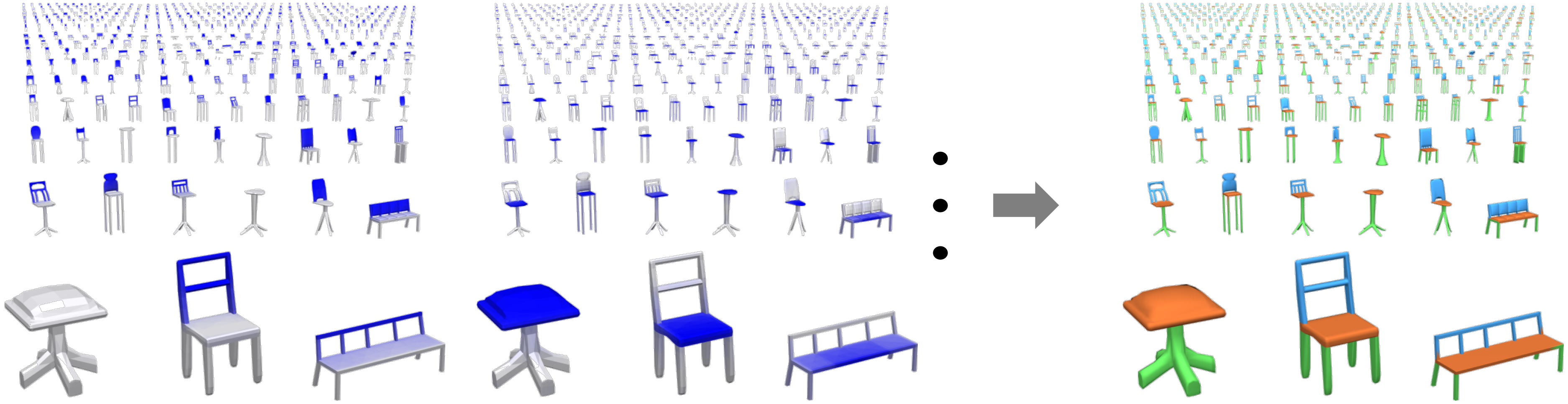
- *Robust computation* --- recover a low-rank matrix from noisy measurements of its entries (initial functional correspondences)
- Structure recovery --- *Y* matrices encode shared structures across the shape collection
- *Efficient encoding* --- We just need to store the *Y* matrices

Large-Scale Data

8K shapes

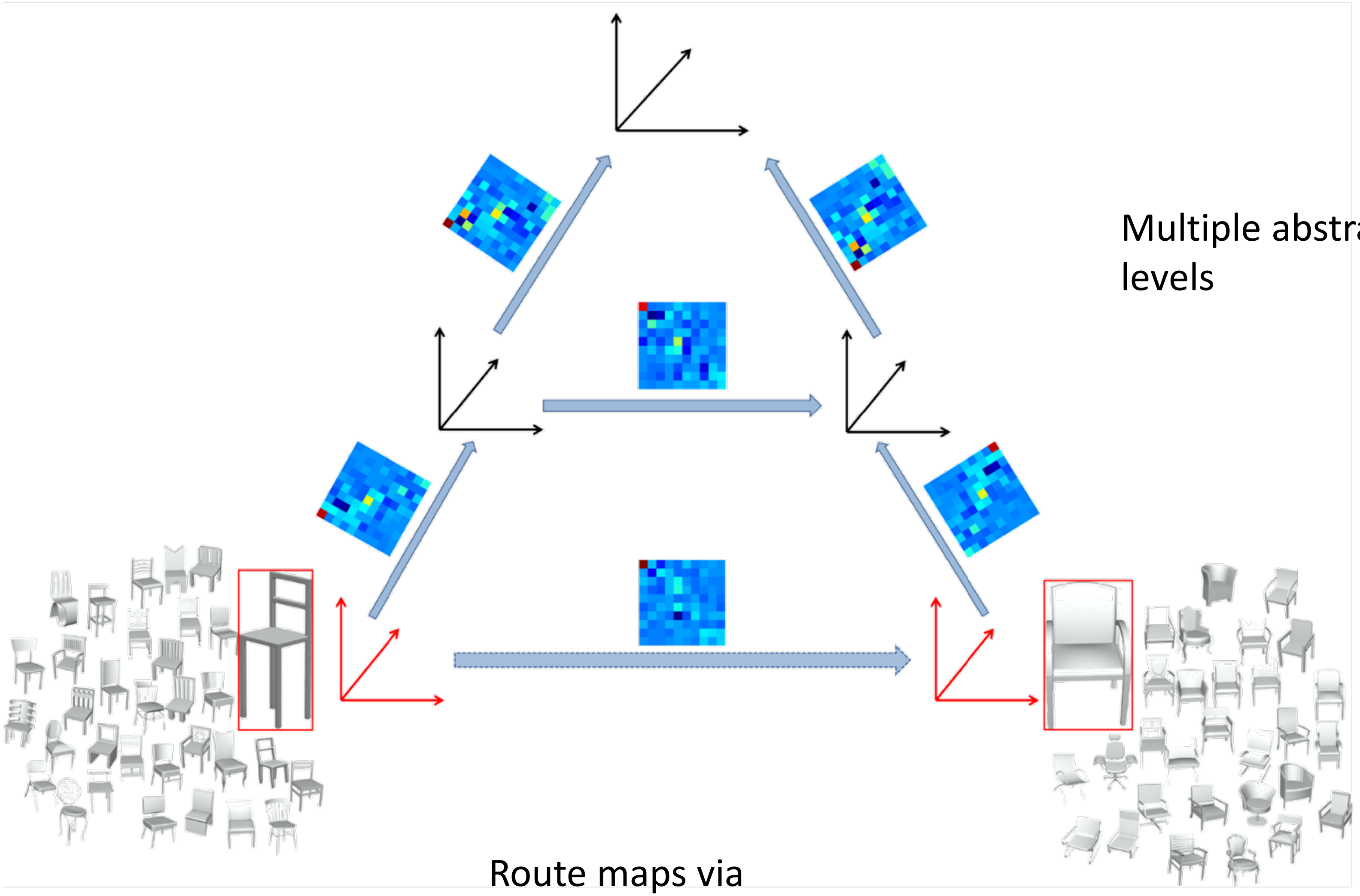


Consistent Shape Segmentation



Via 2nd order MRF on each shape independently

Hierarchical Scaling



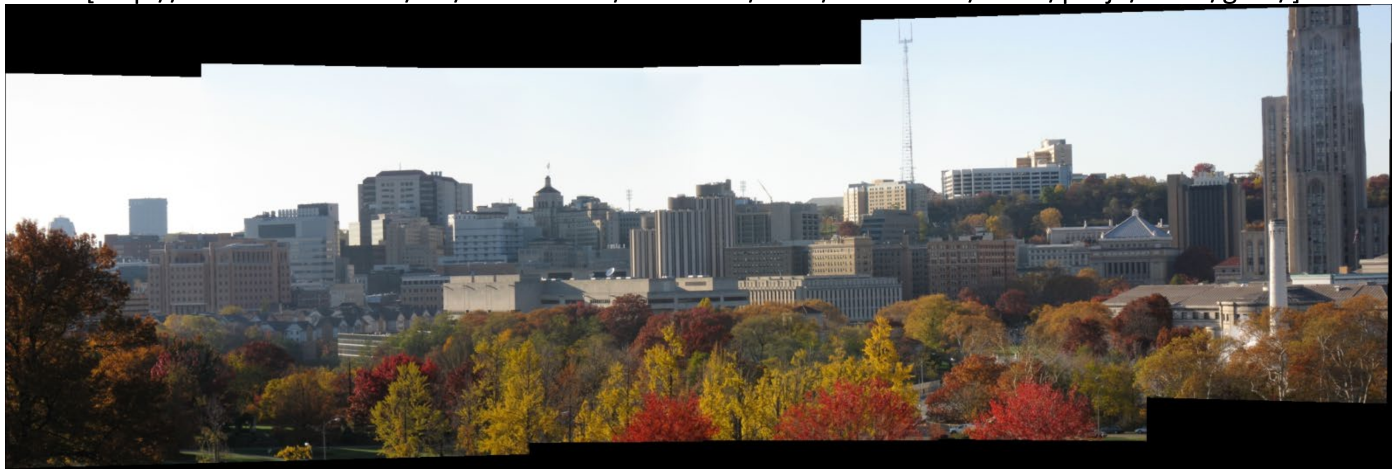
Multiple abstraction levels

Route maps via the abstractions

Co-Limits:
The Network is the Abstraction

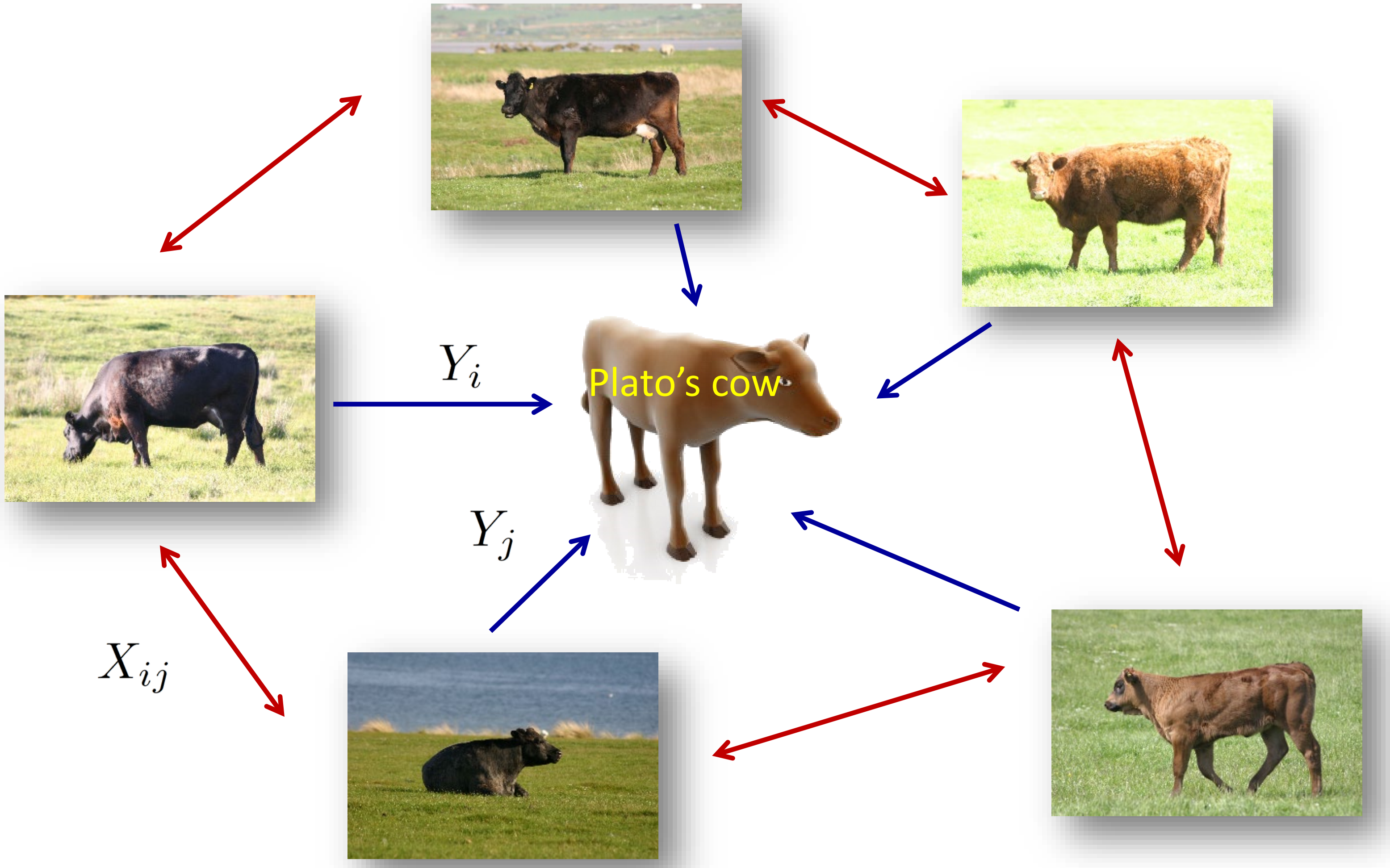
Mosaicking or SLAM at the Level of Functions

[http://www.cs.cmu.edu/afs/cs.cmu.edu/academic/class/15463-f08/www/proj4/www/gme/]

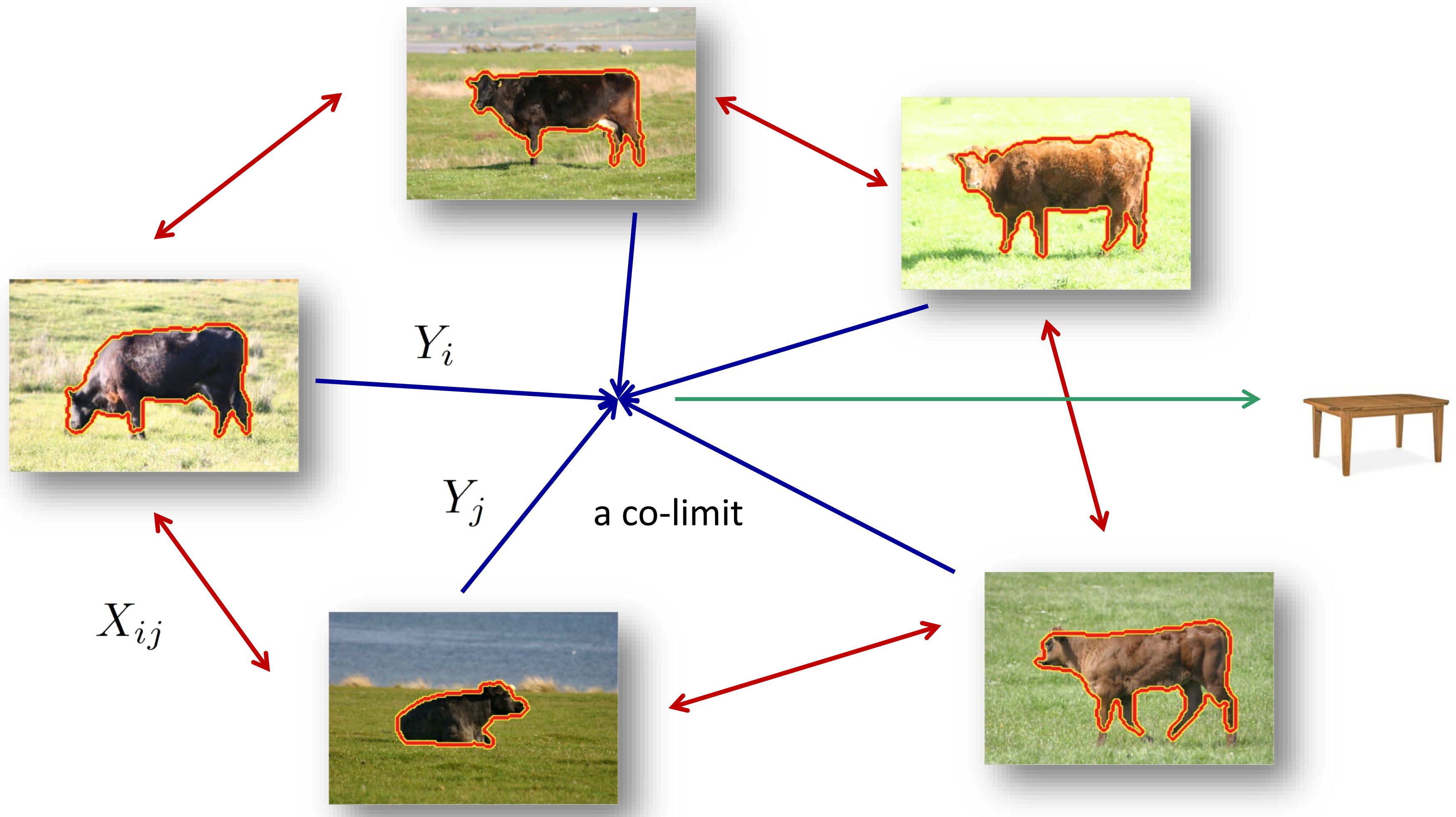


[robotics.ait.kyushu-u.ac.jp]

The Network is the Abstraction

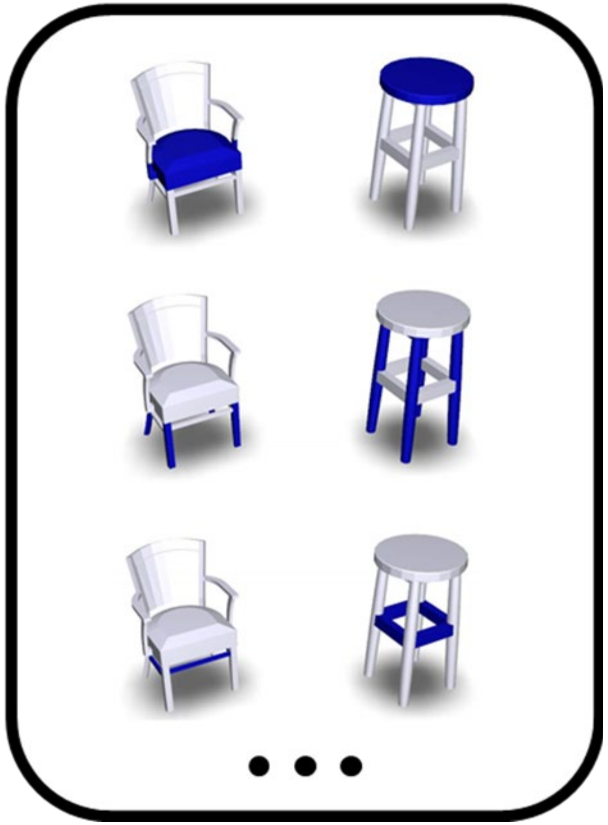


The Network is the Abstraction



Conclusions

- Functional maps generalize normal maps and provide a compact representation of correspondences between data sets
- They can act as powerful information semantic transporters
- When linked into functional map networks, path invariance or cycle closure provides additional supervision that enables to purify and de-noise the maps
- Out of such consistent functional map networks shared latent spaces naturally emerge that help explain the structure of the data



The End