Map between two shapes.
What happens if you compose these maps?
Q: What do you expect if you compose around a cycle?
Cycle consistency

[saḥy-kuh l ku h n-sis-tuh n-see]:

Composing maps in a cycle yields the identity
Philosophical Point

You should have a good reason if your correspondences are inconsistent.
An Unpleasant Constraint

\[ \phi_1(\phi_2(\phi_3(x))) = \text{Id} \]

Cycle consistency
Contrasting Viewpoint

Many possible pairwise matches!

Additional data should help!
Sampling of methods for consistent correspondence.

- Spanning tree
- Inconsistent cycle detection
- Convex optimization
Simultaneously optimize all maps in a collection.
Joint Matching: Simplest Formulation

- **Input**
  - $N$ shapes
  - $N^2$ maps (see last lecture)

- **Output**
  - Cycle-consistent approximation
Spanning Tree: Original Context

“Automatic Three-Dimensional Modeling from Reality” (Huber, 2002)

Multi-view registration
Given: Model graph $G = (S, E)$
Find: Largest consistent spanning tree

"Automatic Three-Dimensional Modeling from Reality" (Huber, 2002)
Heuristic Algorithm

Extract consistent spanning tree in model graph
Many spanning trees

- Single incorrect match can destroy the maps
Inconsistent Loop Detection

Used to deal with repeating structures like windows!

\[
\begin{align*}
\text{max} & \quad \sum_{L} \rho_L x_L \\
\text{s.t.} & \quad x_L \geq x_e \quad \forall e \in L \\
& \quad x_L \leq \sum_{e \in L} x_e \\
& \quad x_L, x_e \in [0, 1]
\end{align*}
\]

\( x_e = 1 \) for false positive edge

\( x_L = \text{max of } x_e \text{ over loop} \)

“Disambiguating Visual Relations Using Loop Constraints” (Zach et al., CVPR 2010)
Iteratively fix triplets and reweight

Definition 3 Given a collection of maps \( \mathcal{M} \), let \( \mathcal{B}(\mathcal{M}) = \{ m_{i,j} \in \mathcal{M} \mid E_{acc}(m_{i,j}) > 0 \} \) — the collection of inaccurate maps. Then we say that \( \mathcal{M} \) is almost accurate, if there do not exist two maps \( m_1, m_2 \in \mathcal{B}(\mathcal{M}) \), which both belong to the same 3-cycle in \( G_\mathcal{M} \). We call such maps isolated.
Fuzzy Correspondences

Exploring Collections of 3D Models using Fuzzy Correspondences (Kim et al., SIGGRAPH 2012)
Fuzzy Correspondences: Idea

- Compute \( N_k \times N_k \) similarity matrix
  - Same number of samples per surface
  - Align similar shapes

- Compute spectral embedding

- Use as descriptor: Display \( e^{-|d_i - d_j|^2} \)
Consistent Segmentation

Global optimization to choose among many possible segmentations

“Joint Shape Segmentation with Linear Programming”
(Huang, Koltun, Guibas; SIGGRAPH Asia 2011)
Joint Segmentation: Motivation

Structural similarity of segmentations

- Extraneous geometric clues

Single shape segmentation
[Chen et al. 09]

Joint shape segmentation
[Huang et al. 11]

“Joint Shape Segmentation with Linear Programming”
(Huang, Koltun, Guibas; SIGGRAPH Asia 2011; slides provided by authors)
Joint Segmentation: Motivation

Structural similarity of segmentations

- Low saliency

Single shape segmentation
[Chen et al. 09]

Joint shape segmentation
[Huang et al. 11]

"Joint Shape Segmentation with Linear Programming"
(Huang, Koltun, Guibas; SIGGRAPH Asia 2011; slides provided by authors)
Joint Segmentation: Motivation

(Rigid) invariance of segments

- Articulated structures

Single shape segmentation
[Chen et al. 09]

Joint shape segmentation
[Huang et al. 11]
Parameterization

Initial subsets of randomized segmentations

“Joint Shape Segmentation with Linear Programming”
(Huang, Koltun, Guibas; SIGGRAPH Asia 2011; slides provided by authors)
Segmentation Constraint/Score

- Each point covered by one segment
  \[ |\text{cover}(p)| = 1 \quad \forall p \in W \]

- Avoid tiny segments
  \[ \text{score}(S') = \sum_{s \in S} \text{area}(s) \cdot \text{repetitions}_s \]

“Joint Shape Segmentation with Linear Programming”  
(Huang, Koltun, Guibas; SIGGRAPH Asia 2011; slides provided by authors)
Consistency Term

- Defined in terms of mappings
  - Oriented
  - Partial

Many-to-one correspondences

Partial similarity

“Joint Shape Segmentation with Linear Programming”
(Huang, Koltun, Guibas; SIGGRAPH Asia 2011; slides provided by authors)
Multi-Way Joint Segmentation

- Objective function

\[
\sum_{i=1}^{n} \text{score}(S_i) + \sum_{(S_i,S_j) \in \mathcal{E}} \text{consistency}(S_i, S_j)
\]

See paper: Linear program relaxation
Q:

Can you extract consistent maps in a globally optimal way?
Basic Setup

Map as a permutation matrix

$$
\begin{pmatrix}
1 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & \cdots & 1 \\
0 & 1 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 1 & \cdots & 0
\end{pmatrix}
$$
What is the **inverse** of a permutation matrix?
Discrete Relaxation

Map as a doubly-stochastic matrix

\[
\begin{pmatrix}
1 & 0 & 0 & \ldots & 0 \\
0 & 0 & 0 & \ldots & 1 \\
0 & 1 & 0 & \ldots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 1 & \ldots & 0 \\
\end{pmatrix}
\]

Sums to 1
Basic Setting

- Given \( n \) objects
- Each object sampled with \( m \) points

“Consistent Shape Maps via Semidefinite Programming” (Huang & Guibas, SGP 2013)
Map Collection: Matrix Representation

\[ X_{12} = \begin{bmatrix}
0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
\end{bmatrix} \]

\[ X = \begin{bmatrix}
I_m & X_{12} & \cdots & \cdots & X_{1n} \\
X_{12}^T & I_m & \cdots & \cdots & \vdots \\
\vdots & \vdots & \ddots & \cdots & \cdots \\
X_{1n}^T & X_{(n-1),n}^T & \cdots & I_m \\
\end{bmatrix} \]

- Diagonal blocks are identity matrices
- Off diagonal blocks are permutation matrices
- Symmetric
Q: What is the rank of a consistent map collection matrix?
Hint: “Urshape” Factorization

\[ X = \begin{bmatrix} I_m & X_{12} & \cdots & X_{1n} \\ X_{12}^T & I_m & \cdots & \vdots \\ \vdots & \vdots & \ddots & X_{(n-1),n} \\ X_{1n}^T & X_{(n-1),n}^T & \cdots & I_m \end{bmatrix} \]

- Diagonal blocks are identity matrices
- Off diagonal blocks are permutation matrices
- Symmetric
Rank $m$, Number of Samples

\[ X_{ij} = X_{j1}^T X_{i1} \iff X = \begin{pmatrix} I_m \\ \vdots \\ X_{n1}^T \end{pmatrix} \begin{pmatrix} I_m & \cdots & X_{n1} \end{pmatrix} \]
Many Equivalent Conditions

Definition 2.1 Given a shape collection $\mathcal{S} = \{S_1, \ldots, S_n\}$ of $n$ shapes where each shape consists of the same number of samples, we say a map collection $\Phi = \{\phi_{ij} : S_i \rightarrow S_j \mid 1 \leq i, j \leq n\}$ of maps between all pairs of shapes is cycle-consistent if and only if the following equalities are satisfied:

\[
\begin{align*}
\phi_{ii} &= \text{id}_{S_i}, \quad 1 \leq i \leq n, \quad \text{(1-cycle)} \\
\phi_{ij} \circ \phi_{ij} &= \text{id}_{S_i}, \quad 1 \leq i < j \leq n, \quad \text{(2-cycle)} \\
\phi_{ik} \circ \phi_{jk} \circ \phi_{ij} &= \text{id}_{S_i}, \quad 1 \leq i < j < k \leq n, \quad \text{(3-cycle)}
\end{align*}
\]

where $\text{id}_{S_i}$ denotes the identity self-map on $S_i$.

Equivalence for binary map matrix $\Phi$:

1. $\Phi$ is cycle-consistent

2. $X = Y_i^\top Y_i$, where $Y_i = (X_{i1}, \ldots, X_{in})$

3. $X \succeq 0$
Approximation by Consistent Maps

\[
\max_X \quad \sum_{i,j \in E} \langle X_{ij}^{in}, X_{ij} \rangle \\
\text{s.t.} \quad X \in \{0, 1\}^{nm \times nm} \\
X \succeq 0 \\
X_{ii} = I_m \\
X_{ij} 1 = 1 \\
X_{ij}^T 1 = 1
\]
Approximation by Consistent Maps

\[
\begin{align*}
\max X & \quad \sum_{i,j \in E} \langle X_{ij}^{in}, X_{ij} \rangle \\
\text{s.t.} & \quad X \in \{0, 1\}^{nm \times nm} \\
& \quad X \geq 0 \\
& \quad X_{ii} = I_m \\
& \quad X_{ij} 1 = 1 \\
& \quad X_{ij}^T 1 = 1
\end{align*}
\]

Maximize number of preserved matches
Approximation by Consistent Maps

\[ \max_X \sum_{i,j \in E} \langle X_{ij}^\text{in}, X_{ij} \rangle \]

s.t. \[ X \in \{0, 1\}^{nm \times nm} \]
\[ X \succeq 0 \]
\[ X_{ii} = I_m \]
\[ X_{ij}1 = 1 \]
\[ X_{ij}^\top 1 = 1 \]
Approximation by Consistent Maps

\[
\max_X \sum_{i,j \in E} \langle X_{ij}^{\text{in}}, X_{ij} \rangle \\
\text{s.t.} \quad X \in \{0, 1\}^{nm \times nm} \\
X \succeq 0 \\
X_{ii} = I_m \\
X_{ij} \mathbf{1} = 1 \\
X_{ij}^\top \mathbf{1} = 1
\]

Every block is a permutation
Approximation by Consistent Maps

$$\max_{X} \sum_{i,j \in E} \langle X_{ij}^{in}, X_{ij} \rangle$$

s.t. 

$$X \in \{0, 1\}^{nm \times nm}$$

$$X \succeq 0$$

$$X_{ii} = I_m$$

Self maps are identity

$$X_{ij} \cdot 1 = 1$$

$$X_{ij}^\top \cdot 1 = 1$$
Approximation by Consistent Maps

\[
\max_X \sum_{i,j \in E} \langle X_{ij}^{in}, X_{ij} \rangle \\
\text{s.t.} \quad X \in \{0, 1\}^{nm \times nm} \\
X \succeq 0 \\
X_{ii} = I_m \\
X_{ij}1 = 1 \\
X_{ij}^T 1 = 1
\]

Already showed: Equivalent to low-rank
Approximation by Consistent Maps

\[
\max_X \sum_{i,j \in E} \langle X_{ij}^{\text{in}}, X_{ij} \rangle \\
\text{s.t.} \quad X \in \{0, 1\}^{nm \times nm} \\
X \succeq 0 \\
X_{ii} = 1_m \\
X_{ij} 1 = 1 \\
X_{ij}^T 1 = 1
\]
Convex Relaxation

\[
\max_X \quad \sum_{i,j \in E} \langle X_{ij}^{in}, X_{ij} \rangle \\
\text{s.t.} \quad X \succeq 0 \\
X \succeq 0 \\
X_{ii} = I_m \\
X_{ij} \mathbf{1} = \mathbf{1} \\
X_{i \cdot j}^\top \mathbf{1} = \mathbf{1}
\]
**Rounding Procedure**

\[
\begin{align*}
\max_X & \quad \langle X, X_0 \rangle \\
\text{s.t.} & \quad X \geq 0 \\
& \quad X1 = 1 \\
& \quad X^\top 1 = 1
\end{align*}
\]

*Guaranteed to give permutation*

**Linear assignment problem**
Recovery Theorem

Can tolerate $\lambda_2 / 4(n - 1)$ incorrect correspondences from each sample on one shape.

$\lambda_2$ is algebraic connectivity; bounded above by two times maximum degree.
Recovery Theorem: Complete Graph

Can tolerate 25% incorrect correspondences from each sample on one shape.

$\lambda_2$ is algebraic connectivity; bounded above by two times maximum degree
Phase Transition

Always recovers / Never recovers
Solving the multi-way matching problem by permutation synchronization

Deepak Pachauri, Rishi Kondor, and Vikas Singh
1Dept. of Computer Sciences, University of Wisconsin–Madison
2Dept. of Biostatistics & Medical Informatics, University of Wisconsin–Madison
3Dept. of Computer Science and Dept. of Statistics, The University of Chicago
pachauri@cs.wisc.edu  rishi@uchicago.edu  vsingh@biostat.wisc.edu

Abstract

The problem of matching not just two, but \( n \) different sets of objects to each other arises in many contexts, including finding the correspondence between feature points across multiple images in computer vision. At present it is usually solved by matching the sets pairwise, in series. In contrast, we propose a new method, Permutation Synchronization, which finds all the matchings jointly, in one shot, via a relaxation to eigenvector decomposition. The resulting algorithm is both computationally efficient, and, as we demonstrate with theoretical arguments as well as experimental results, much more stable to noise than previous methods.

1 Introduction

Finding the correct bijection between \( \{x_1', x_2', \ldots, x_n'\} \) is a fundamental problem in computer vision [1]. In this paper, we consider its permutation synchronization [2].

Our primary motivation is that matching feature points across many images is crucial for image registration [2] and is used to build scene structure from motion (SFM) [8, 9]. How to solve this problem efficiently remains an open question.

Presently, multi-matching is usually solved by matching \( X_1 \) to \( X_2 \), then a permutation of \( X_2 \) is found that minimizes a cost function. One can conceive of various strategies, but data are noisy, a single error in the set of pairwise matches [12, 13, 14]. In contrast, in this paper we describe a new method, Permutation Synchronization, which finds all the matchings jointly, in one shot, via a relaxation to eigenvector decomposition. The resulting algorithm is both computationally efficient, and, as we demonstrate with theoretical arguments as well as experimental results, much more stable to noise than previous methods.
Q: Where do the pairwise input maps come from?
Possible Extension with Guarantees

Tight Relaxation of Quadratic Matching

Ray Kezurer†, Shahar Z. Kovalsky‡, Ronen Basri, Yaron Lipman
Weizmann Institute of Science

Abstract
Establishing point correspondences between shapes is extremely challenging as it involves both finding sets of semantically persistent feature points, as well as their combinatorial matching. We focus on the latter and consider the Quadratic Assignment Matching (QAM) model. We suggest a novel convex relaxation for this NP-hard problem that builds upon a rank-one reformulation of the problem in a higher dimension, followed by relaxation into a semidefinite program (SDP). Our method is shown to be a certain hybrid of the popular spectral and doubly-stochastic relaxations of QAM and in particular we prove that it is tighter than both.

Experimental evaluation shows that the proposed relaxation is extremely tight: in the majority of our experiments it achieved the certified global optimum solution for the problem, while other relaxations tend to produce sub-optimal solutions. Thus, however, comes at the price of solving an SDP in a higher dimension.

Our work is closely related to the problem of Consistent Shape Matching (CSM), where we solve

\[
\begin{align*}
\max_Y \quad & \text{tr}(WY) \\
\text{s.t.} \quad & Y \succeq [X][X]^T \\
& X \in \text{conv} \mathcal{P}_n \\
& \text{tr}Y = k \\
& Y \succeq 0 \\
& \sum_{qrst} Y_{qrst} = k^2 \\
& Y_{qrst} \leq \begin{cases} 
0, & \text{if } q = s, r \neq t \\
0, & \text{if } r = t, q \neq s \\
\min \{X_{qr}, X_{st}\}, & \text{otherwise}
\end{cases} \\
\end{align*}
\]

Figure 1: Consistent Collection Matching. Results of the proposed one-stage procedure for finding consistent correspondence between shapes in a collection showing strong variability and non-rigid deformations.
Approximate Methods

Consistent Partial Matching of Shape Collections via Sparse Modeling

L. Comuni, E. Rodolà, A. Albarelli, F. Memoli, D. Cremers

1 University of Venice, Italy 2 TU Munich, Germany 3 Ohio State University, U.S.

Abstract
Recent efforts in the area of joint object-matching approach the problem by taking as input a set of shapes which are then jointly optimized across the whole collection so that certain accuracy requirements are satisfied. Our natural requirement is cycle-consistency -- namely the fact that maps in the same result regardless of the path taken in the shape collection. In this paper, we develop a new method to obtain consistent matches without requiring initial pairwise solutions to be given as input. Our approach uses a joint measure of metric distortion directly over the space of cycle-consistent maps. We formulate the problem as a series of quadratic programs, making our technique a natural candidate for analyzing collections with large data sets. The particular form of the problem allows us to leverage results and tools from the Euclidean distance geometry theory. This enables a highly efficient optimization procedure which accounts for large numbers of shapes in a matter of minutes in collections with hundreds of shapes.

Categories and Subject Descriptors (according to ACM CCS): 1.3.5 [Computational Geometry and Object Modeling—Shape Analysis]

1. Introduction
Finding matches among multiple objects is a research topic that has generated a large amount of interest. In this work, we focus on the problem of finding correspondences among shapes from a collection of objects. We present a novel approach to solving this problem, which we call Consistent Partial Matching (CPM). CPM is designed to handle large datasets efficiently and to provide cycle-consistent matches. The key idea behind CPM is to formulate the matching problem as a series of quadratic programs, which allows us to leverage existing optimization techniques to find solutions at a processing cost of just a few minutes for large datasets.

Figure 1: A partial multi-way correspondence obtained with our approach on a heterogeneous collection of shapes. Our method does not require initial pairwise maps as input, as it actively seeks a reliable correspondence in the joint, cycle-consistent space of joint, cycle-consistent matches. Partially similar as well as outlier shapes are accounted for by adopting a sparse model for the joint correspondence. A subset of all matches is shown.

Sequence of quadratic programs; based on metric distortion and WKS descriptor match

Figure 6: Our matching pipeline. First sub-problem (from left): Given a collection of shapes as input, a set $Q$ of queries are generated (e.g., by farthest point sampling in the joint WKS space); we then compute distance maps (shown here as heat maps over the shapes) in descriptor space from each shape point to each query $q_i \in Q$, and keep the vertices having distance smaller than a threshold; finally, a single multi-way match is extracted by solving problem (13). Second sub-problem: The multi-way matches extracted by iterating the previous step are compared using a measure of metric distortion; the final solution (in orange) is obtained by solving problem (13) over the reduced feasible set.
Approximate Methods

Multiplicative updates for nonconvex nonnegative matrix factorization

Entropic Metric Alignment for Correspondence Problems

Justin Solomon*  
MIT

Gabriel Peyré  
CNRS & Univ. Paris-Dauphine

Vladimir G. Kim  
Adobe Research

Suvin Sra  
MIT

Abstract

Many shape and image processing tools rely on computation of correspondences between geometric domains. Efficient methods that stably extract “soft” matches in the presence of diverse geometric structures have grown to be valuable for shape retrieval and transfer of geometric information. With these applications in mind, we propose a fully probabilistic correspondence optimization formulation that can be expressed as a nonnegative matrix factorization. This allows the use of modern optimization methods that are simple and numerically robust. Our algorithm can be applied to any domain expressible as a metric measure space (see §2). Conceptually, only distance matrices are required as input, and hence the method can be applied to many classes of domains including meshes, point clouds, graphs, and even more.

Figure 1: Entropic GW can find correspondences between a source surface (left) and a target surface with similar structure, a surface with shared semantic structure, a noisy 3D point cloud, an icon, and a hand drawing. Each fuzzy map was computed using the same code.

The probability of an entropic GW matching expression is a “fuzzy” correspondence matrix in the style of (Kim et al. 2012; Solomon et al. 2012); we control the sharpness of the correspondence via the weight of an entropic regularization.

Although [Mohlo et al. 2011] and subsequent work identified the possibility of using GW distances for geometric correspondence, computational challenges hindered their practical application. To overcome these challenges, we build upon recent methods for regularized optimal transportation introduced in [Benamou et al. 2015; Solomon et al. 2015]. While optimal transportation is a fundamentally different optimization problem from regularized GW computation (linear versus quadratic matching), the core of our method relies upon solving a sequence of regularized optimal transport problems.

Our algorithm can be applied to any domain expressible as a metric measure space (see §2). Conceptually, only distance matrices are required as input, and hence the method can be applied to many classes of domains including meshes, point clouds, graphs, and even more.

\[
\min_A KL(G|AA^T)
\]
Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks

Jun-Yan Zhu*   Taesung Park*   Phillip Isola   Alexei A. Efros
Berkeley AI Research (BAIR) laboratory, UC Berkeley

Abstract

Image-to-image translation is a class of vision and graphics problems where the goal is to learn the mapping between two domains or categories of images.

1. Introduction

What did Claude Monet see as he placed paints on his palette on the banks of the Seine River in Argenteuil? How did his imagination prompt him to render that image? What did our algorithm learn to render as it places paints on its palette on the banks of the Seine River in Argenteuil? How did its imagination prompt it to render that image? Figure 1: Given any two unordered image collections \( X \) and \( Y \), our algorithm learns to automatically “translate” an image from one into the other and vice versa: (left) Monet paintings and landscape photos from Flickr; (center) zebras and horses from ImageNet; (right) summer and winter Yosemite photos from Flickr. Example application (bottom): using a collection of paintings of famous artists, our method learns to render natural photographs into the respective styles.

Slides courtesy the authors
https://junyanz.github.io/CycleGAN/
Paired vs. Unpaired Problems

Paired

\[ x_i, y_i \]

\[
\begin{array}{c}
\text{\{ } \\
\text{\{ } \\
\text{\{ } \\
\text{\{ } \\
\end{array}
\]

Unpaired

\[ X, Y \]

\[
\begin{array}{c}
\text{\{ } \\
\text{\{ } \\
\text{\{ } \\
\end{array}
\]

⋯
Adversarial Networks: Problem

Discriminator $D(x) G(x)$

Generator $G(x)$

Real!

Real too!
Mode collapse
Cycle-Consistent Adversarial Networks

\[ x \xrightarrow{G} \hat{Y} \xrightarrow{F} \hat{x} \]

 Reconstruction error

\[ \|F(G(x)) - x\|_1 \]

Large cycle loss

Small cycle loss

[Zhu*, Park*, Isola, and Efros, ICCV 2017]
Cycle Consistency Loss

\[ x \xrightarrow{G} \hat{y} \xrightarrow{F} \hat{x} \]
\[ y \xrightarrow{F} \hat{x} \xrightarrow{G} \hat{y} \]

\[ D_Y(G(x)) \]
\[ D_G(F(x)) \]

Reconstruction error

\[ \|F(G(x)) - x\|_1 \]
\[ \|G(F(y)) - y\|_1 \]

See similar formulations [Yi et al. 2017], [Kim et al. 2017]

[Zhu*, Park*, Isola, and Efros, ICCV 2017]
<table>
<thead>
<tr>
<th>Input</th>
<th>Monet</th>
<th>Van Gogh</th>
<th>Cezanne</th>
<th>Ukiyo-e</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Input Image" /></td>
<td><img src="image2" alt="Monet Image" /></td>
<td><img src="image3" alt="Van Gogh Image" /></td>
<td><img src="image4" alt="Cezanne Image" /></td>
<td><img src="image5" alt="Ukiyo-e Image" /></td>
</tr>
<tr>
<td><img src="image6" alt="Input Image" /></td>
<td><img src="image7" alt="Monet Image" /></td>
<td><img src="image8" alt="Van Gogh Image" /></td>
<td><img src="image9" alt="Cezanne Image" /></td>
<td><img src="image10" alt="Ukiyo-e Image" /></td>
</tr>
<tr>
<td><img src="image11" alt="Input Image" /></td>
<td><img src="image12" alt="Monet Image" /></td>
<td><img src="image13" alt="Van Gogh Image" /></td>
<td><img src="image14" alt="Cezanne Image" /></td>
<td><img src="image15" alt="Ukiyo-e Image" /></td>
</tr>
<tr>
<td><img src="image16" alt="Input Image" /></td>
<td><img src="image17" alt="Monet Image" /></td>
<td><img src="image18" alt="Van Gogh Image" /></td>
<td><img src="image19" alt="Cezanne Image" /></td>
<td><img src="image20" alt="Ukiyo-e Image" /></td>
</tr>
<tr>
<td><img src="image21" alt="Input Image" /></td>
<td><img src="image22" alt="Monet Image" /></td>
<td><img src="image23" alt="Van Gogh Image" /></td>
<td><img src="image24" alt="Cezanne Image" /></td>
<td><img src="image25" alt="Ukiyo-e Image" /></td>
</tr>
<tr>
<td><img src="image26" alt="Input Image" /></td>
<td><img src="image27" alt="Monet Image" /></td>
<td><img src="image28" alt="Van Gogh Image" /></td>
<td><img src="image29" alt="Cezanne Image" /></td>
<td><img src="image30" alt="Ukiyo-e Image" /></td>
</tr>
</tbody>
</table>
More Than Two Domains?

(a) Cross-domain models

(b) StarGAN

Choi et al., CVPR 2018
Extra: Angular Synchronization

Justin Solomon

6.8410: Shape Analysis
Spring 2023