FlowWeb: Joint Image Set Alignment by Weaving Consistent, Pixel-wise Correspondences

Tinghui Zhou
UC Berkeley

Yong Jae Lee
UC Davis

Stella X. Yu
UC Berkeley/ICSI

Alexei A. Efros
UC Berkeley

Abstract

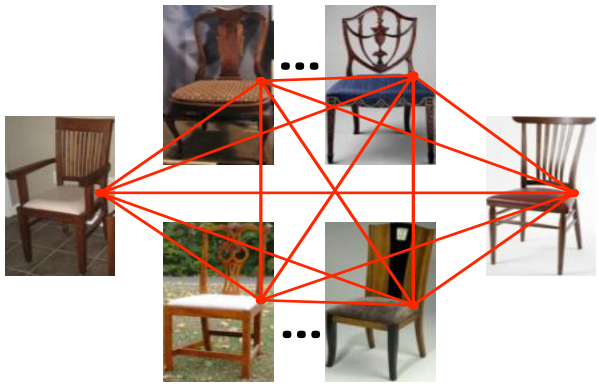
Given a set of poorly aligned images of the same visual concept without any annotations, we propose an algorithm to jointly bring them into pixel-wise correspondence by estimating a **FlowWeb** *representation of the image set. FlowWeb is a fully-connected correspondence flow graph with each node representing an image, and each edge representing the correspondence flow field between a pair of images, i.e. a vector field indicating how each pixel in one image can find a corresponding pixel in the other image. Correspondence flow is related to optical flow but allows for correspondences between visually dissimilar regions if there is evidence they correspond transitively on the graph. Our algorithm starts by initializing all edges of this complete graph with an off-the-shelf, pairwise flow method. We then iteratively update the graph to force it to be more self-consistent. Once the algorithm converges, dense, globally-consistent correspondences can be read off the graph. Our results suggest that FlowWeb improves alignment accuracy over previous pairwise as well as joint alignment methods.*

1. Introduction

Consider a pair of chairs depicted in Fig. 1(a). While the chairs might look similar, locally their features (like the seat corner above) are very different in appearance, so standard pairwise image matching approaches like SIFT Flow [25] have trouble finding correct correspondences. The reason we, human observers, have little trouble spotting visual correspondences between the features of these two chairs is likely because we have been exposed to many hundreds of chairs already, and are able to draw upon this knowledge to bridge the appearance gap. In this paper, we propose to "level the playing field" by starting with a *set of images* and computing correspondences *jointly* over this set in a globally-consistent way, as shown in Fig. 1(b).

Correspondence (also known as alignment or registration) is the task of establishing connections between similar points/regions across different images, either sparsely (e.g. SIFT keypoint matching), or densely at every pixel (e.g. optical flow). Correspondence can be defined either locally, as a pairwise connection between two images, or globally, as a joint label assignment across an image collection.

(a) Example pixel correspondence using SIFT Flow [21]



(b) Example pixel correspondence using our FlowWeb

Figure 1. Finding pixel-wise correspondences between images is difficult even if they depict similar objects: (a) a typical correspondence error using a state-of-the-art pairwise flow estimation algorithm. (b) We propose computing correspondences jointly across an image collection in a globally-consistent way.

One can appreciate the power of joint correspondence by considering faces, a domain where correspondences are readily available, either via human annotation, or via domain-specific detectors. Large-scale face datasets, meticulously annotated with globally-consistent keypoint labels ("right mouth corner", "left ear lower tip", etc) were the catalyst for a plethora of methods in vision and graphics for the representation, analysis, 3D modeling, synthesis, morphing, browsing, etc. of human faces [8, 29, 19, 3, 20]. Of course, faces are a special object class in many ways: they can generally be represented by a linear subspace, are relatively easy to detect in the wild and relatively easy to annotate (i.e. have well-defined keypoints). Nonetheless, we believe that some of the same benefits of having large, jointly registered image collections should generalize beyond faces and ap-

ply more broadly to a range of visual entities, provided we have access to reliable correspondences. Indeed, the recent work of Vicente et al. [39] on reconstructing PASCAL VOC classes using hand-annotated keypoints is an exciting step in this direction. But what about cases when manual keypoint annotation is difficult or infeasible?

The goal of this work is to establish *globally-consistent* pixel-wise correspondences between all images within a given image collection, without any supervision. Just as the face modeling approaches start with a collection of detected faces in coarse alignment (on the level of a bounding box), we start with a collection of coarsely similar images, which could be obtained as a result of an object detector [12], a mid-level discriminative visual element detector [10], or directly from a dataset with labeled bounding boxes.

The key insight is to focus on the correspondence flow fields between the images instead of working with image pixels directly. We achieve this by representing the image collection as a *FlowWeb* – a fully-connected graph with images as nodes and pixel flow fields between pairs of images as edges. We show that, starting with a simple initialization, we are able to force the FlowWeb to become consistent by iteratively updating the flow fields.

2. Prior work

Pairwise Image Flow: The idea of generalizing optical flow to pairs of images that are only semantically related was first proposed in SIFT Flow [25], which adopted the computational framework of optical flow, but with local appearance matching being done on SIFT descriptors instead of raw pixels to add local appearance invariance. Deformable Spatial Pyramid (DSP) Matching [21], a recent follow-up to SIFT Flow, greatly improves the speed of the algorithm, also modestly improving the accuracy. Other works in this space include [2], which generalizes PatchMatch [1] to use feature descriptors instead of pixel patches, and more recently, finding pairwise correspondences using convolutional features [26].

Image graphs for pattern discovery: The vast literature on object discovery and co-segmentation treats the image set as an unordered bag. Recent work exploits the connectivity within an image collection by defining a graph over images (e.g. [15, 27, 47, 13]) or objects (e.g. [28, 20, 6]). More relevant to us, [24, 11, 34] perform joint object discovery and segmentation on a noisy image set, resulting in often excellent region-wise correspondences. However, their main aim is to find and segment a consistent object, whereas we aim to find dense pixel-wise correspondences in an image set.

Graph consistency: The idea of utilizing consistency constraints within a global graph structure has been applied to various vision and graphics problems, including co-segmentation [40, 41], structure from motion [47, 42], and shape matching [17]. Most related to ours is [17], which formulates the constraint of cycle consistency as positive semi-definiteness of a matrix encoding a collection of pairwise correspondence maps on shapes, and solves for consistent maps via low-rank matrix recovery with convex relaxation. We also employ a cycle consistency constraint, but optimize it completely differently. Our problem complexity is also considerably larger: the number of pixels per image is typically two orders of magnitude greater than the number of sample points per shape.

Joint pixel-wise alignment of image sets: Average images have long been used informally to visualize joint (mis)alignment of image sets (e.g. [37]). However, it was the seminal work of Congealing [23, 16] that established unsupervised joint alignment as a serious research problem. Congealing uses sequential optimization to gradually lower the entropy of the intensity distribution of the entire image set by continuously warping each image via a parametric transformation (e.g. affine). Congealing demonstrates impressive results on the digit dataset and some others, but does not perform as well on more difficult data.

RASL [33] also focuses on modeling a common image intensity structure of the image set; in their case, as a low-rank linear subspace plus sparse distractors specific to each image. Again, parametric transformations are used to align the images to the common subspace. The main difficulty with subspace methods is that they assume that the majority of images are already in good correspondence, else the subspace would end up encoding multiple shifted copies of the data. Collection Flow [18] also uses a low-rank subspace to model the common appearance of the collection, but with a clever twist by using non-parametric transformations (i.e. optical flow) that align between each image and its low-rank projection at each iteration (their application domain is faces, where the coarse alignment is good enough for subspace projections to work well). Mobahi et al. [31] propose a generative image representation that models each image as the composition of three functions: color, appearance, and shape. The appearance and shape functions are assumed to be constructed from a small set of basis functions (i.e., restricted to low-dimensional subspaces) in order to control the composition capacity. The model is used to establish dense correspondences between instances of the same object category.

All the subspace-based methods above share the same basic idea – compute some global representation of the image data, and then try to warp every image to make it more consistent with that representation (one can think of this as a star graph centered at the global representation connecting each image in the set). This works well if the distances between the images and the global representation can be trusted. But what if the image data lives on an articulation manifold [30], where only local distances are reliable? [35] takes this view, modeling the image collection not by some global representation, but using a locally-connected graph.

This method shows very good results for aligning images of the same physical scene under low-dimensional transformations (global rotation, stretching, etc). However, it is not directly applicable for collections of multiple instances of the same object category. Concurrently with our work, Carreira et al. [4] models the image collection with a 'virtual view network', and resolves the difficulty of cross-view image alignment by finding the shortest geodesic path along the network. However, constructing the network requires either human annotations (e.g. keypoints) or pre-trained, category-specific pose predictors, whereas our method is fully unsupervised and does not require any training.

Like Collection Flow [18], our method uses compositions of flow fields to model connections between images. But instead of using a global, centered representation of the data like [43, 18, 31], our representation is defined on pairwise connections in the graph, like [35]. However, we differ from [35] in a number of important ways: 1) [35] represents the image set by a nearest neighbor graph, trusting the optical flow algorithm to be reliable when the flow field magnitude is small. Instead, We take a different perspective, and rely on the "wisdom of crowd", trusting the flow consistency among triplets of images in a fully connected image network. With the complementary information among images, not only can we *fill in the blanks* arising from occlusion and outliers, but also find reliable correspondences between images that do not look alike; 2) [35] explicitly projects the manifold into a lower-dimensional space (3-4D), whereas we keep our correspondence flow graph in high dimension and let it become more self-consistent on its own, controlling its own intrinsic dimensionality.

3. Approach

Given a collection of images $\{I_1, \ldots, I_N\}$ of the same visual concept, we would like to find dense pixel-wise correspondences that are consistent throughout the entire image collection. Our basic idea is that global correspondences emerge from consistent local correspondences in a bootstrap fashion. The quality of pixel-wise matching between two images I_i, I_j can be validated with multiple additional images. For each third image I_k, pixels $p \in I_i$ and $q \in$ image I_j are matched transitively if there is $r \in I_k$, where (p, I_i) matches (r, I_k), and (r, I_k) matches (q, I_j). That is, even when p, q do not have sufficient feature similarity directly, there may be sufficient indirect evidence from their similarity to other images supporting their match.

FlowWeb Representation Given a collection of N images, we build a complete graph of N nodes, where a node denotes an image, and the edge between two nodes (i, j) is associated with flow field T_{ij} between images (I_i, I_j) (see Fig. 3). For M pixels per image, T_{ij} is an $M \times 2$ matrix, each row containing the displacement vector between two matching pixels p and q in images I_i and I_j respectively:

$$T_{ij}^{pq} = x_q - x_p, \quad (p, I_i) \text{ matches } (q, I_j)\ , \tag{1}$$

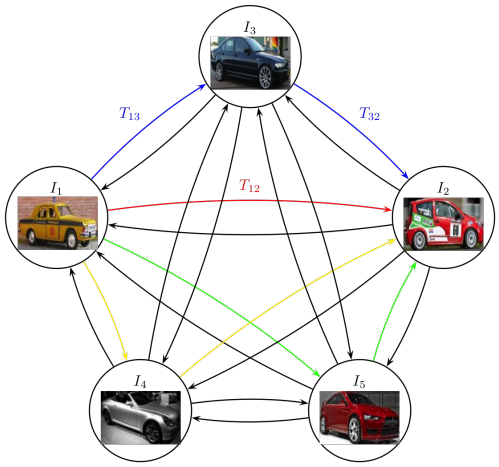


Figure 3. An example of our FlowWeb representation, where a node denotes an image, and each edge represents the flow field between two images.

where x_p denotes the spatial coordinates of pixel p.

3.1. Cycle consistency

Global correspondences in the image collection require the pairwise flow fields to be consistent among different paths connecting two nodes in the graph. Cycle consistency criterion can be expressed as the net displacement along a cycle in the FlowWeb being zero, e.g. for two-image cycle,

$$\begin{aligned} T_{ij}^{pq} + T_{ji}^{qr} &= (x_q - x_p) + (x_r - x_q) \\ &= x_r - x_p = 0, \text{iff } r = p. \end{aligned}$$

Let $T_{ik} \circ T_{kj}$ denote such flow composition from I_i through I_k to I_j. We define:

$$\begin{aligned} &\text{2-cycle consistency:} && T_{ij} \circ T_{ji} = 0 \\ &\text{3-cycle consistency:} && T_{ik} \circ T_{kj} \circ T_{ji} = 0. \end{aligned}$$

While the number of cycles with arbitrary length is exponential in the number of nodes in the graph, [32] shows that considering only 2-cycles and 3-cycles are often sufficient for complete graphs. The concept of cycle consistency has also been explored in joint shape matching [17, 32], co-segmentation [40, 41] as well as SfM [47, 42].

We measure the quality of a matching flow by counting how many consistent 3-cycles go through it in the FlowWeb. If three images form a consistent cycle at a flow T_{ij}^{pq}, it means this flow is validated by a third image I_k, such that

$$T_{ij}^{pq} = T_{ik}^{pr} + T_{kj}^{rq}. \tag{2}$$

Let $\triangle_{ij}^{pq}$ denote the set of image nodes that complete a consistent cycle with flow T_{ij}^{pq}. We define the *single flow cycle*

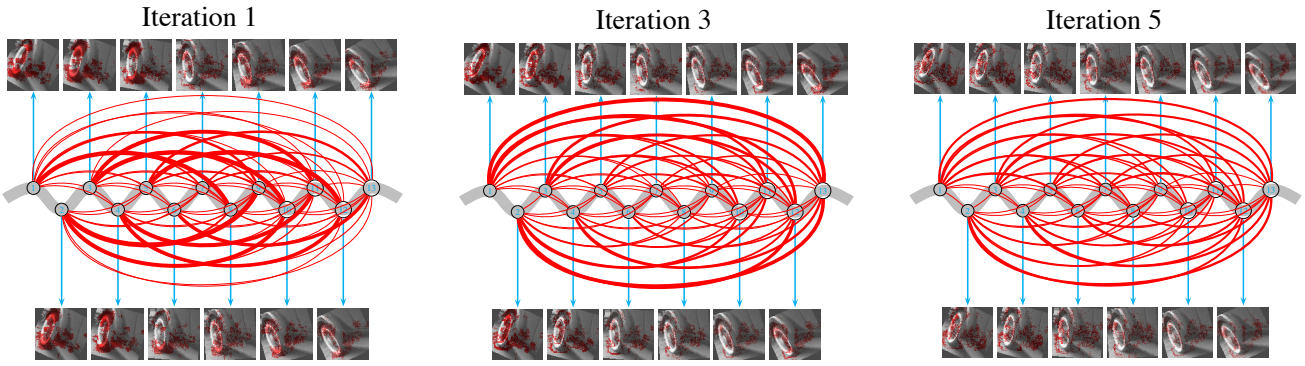


Figure 2. The flow update priority pattern over iterations. Shown here is an image ensemble made of 13 wheel images related by in-plane rotation, i.e. they lie on a 1D manifold (light gray curve) with increasing differences from left to right. The priority score is defined for each flow and it is large if there exists a transitive alternative that achieves better cycle consistency. Each image is shown with a red mask, indicating the sum of priority for all the flows associated with each pixel. The connections between each pair of images show the overall priority summed over all flows between them (thicker means higher). As shown, there are more mid-range connections (high update priority between not so similar images) initially, more long-range connections (high update priority between more distinct images) subsequently, and more even connections throughout the ensemble finally. There are far fewer short-range connections throughout iterations, since nearby images tend to have good correspondences and are cycle-consistent already. These flows thus have low priority.

consistency (SFCC) score as the cardinality of $\triangle$:

$$\mathcal{C}(T_{ij}^{pq}) = |\triangle_{ij}^{pq}|_{card} = \sum_{k=1,k\notin\{i,j\}}^{N} [T_{ij}^{pq} = T_{ik}^{pr} + T_{kj}^{rq}], \quad (3)$$

where $[\cdot]$ is the binary indicator function.

We generalize the SFCC concept to the whole flow set $\mathbf{T} = \{T_{ij}\}$, and define *all flow cycle consistency* (AFCC) that counts the number of consistent 3-cycles in $\mathbf{T}$:

$$\mathcal{C}(\mathbf{T}) = \frac{1}{3} \sum_{i,j=1,i\neq j}^{N} \sum_{p\in I_i} \mathcal{C}(T_{ij}^{pq}) \,. \quad (4)$$

The factor of $1/3$ corrects for the over-counting when summing over SFCC's for the three edges of the same cycle.

3.2. Objective

Our objective has two terms: FlowWeb cycle consistency $\mathcal{C}(\cdot)$, and regularization $\mathcal{R}(\cdot)$ that measures the difference between the current $\mathbf{T} = \{T_{ij}\}$ and the initial flow set $\mathbf{T}_0 = \{S_{ij}\}$ provided by a pairwise flow method (e.g. [21, 25]):

$$\max_{\mathbf{T}} \; \mathcal{C}(\mathbf{T}) - \lambda \mathcal{R}(\mathbf{T}, \mathbf{T}_0) \,, \quad (5)$$

$$\mathcal{R}(\mathbf{T}, \mathbf{T}_0) = \sum_{i,j=1,i\neq j}^{N} \sum_{p\in I_i} \|T_{ij}^{pq} - S_{ij}^{ps}\| \,, \quad (6)$$

where $\lambda > 0$ can be chosen based on the initialization quality, s denotes p's initial correspondence in image j, and $\|\cdot\|$ is the Euclidean norm.

3.3. Optimization

For clarity of exposition, we ignore the regularization term $\mathcal{R}(\cdot)$ for now and focus on optimizing the cycle consistency term alone. Our iterative optimization procedure builds on the following intuition: even when pixels p and q do not have sufficient feature similarity to be matched directly, they should still be matched if there is *sufficient indirect evidence* from 1) their similarity to other images supporting the match (inter-image) and/or 2) proximity to neighboring pixels that have a good match (intra-image). Both are provided by the cycle consistency measure, and exploited alternately at each iteration.

Inter-image phase The first phase of our iterative optimization involves the computation of a *priority* score for each flow in the current flow set. The update priority is high for flows that satisfy two criteria: 1) have low cycle consistency and 2) the consistency of an alternative solution is high. In our case, the alternative solutions to T_{ij}^{pq} are provided by one-hop transitive flows, i.e. $\{T_{ik}^{pr} + T_{kj}^{rt}, \forall k\}$[1]. Essentially, we would like the priority to measure the overall consistency gap between the current solution and some transitive solution. However, exact evaluation of the consistency gap is too expensive, as the change of one flow could potentially affect the consistency of all other flows that involve it in the SFCC computation.

Instead, we compute a lower bound based on the following observation: *if pixels* $< p, r, t >$ *are cycle-consistent, and there exists another pixel* u *such that both* $< p, u, r >$ *and* $< r, u, t >$ *are cycle-consistent, then* $< p, u, t >$ *are also cycle-consistent.* In other words, if we consider the two flows T^{pr} and T^{rt} that comprise a transitive flow between p and t, and denote the set of nodes each is consistent with by $\triangle^{pr}$ and $\triangle^{rt}$ respectively, then the transitive flow $T^{pt} = T^{pr} + T^{rt}$ is guaranteed to be consistent with $\triangle^{pr} \cap \triangle^{rt}$, and $|\triangle^{pr} \cap \triangle^{rt}|_{card}$ is the SFCC lower bound

[1] Note that we use q to denote p's direct correspondence in image j, and t to denote the transitive correspondence.

for T^{pt}, while holding all other flows fixed. In light of this observation, for each pair of images i and j, we compute the update priority of a flow T_{ij}^{pq} by

$$\mathcal{P}(i,j,p) = \max_k |\triangle_{ik}^{pr} \cap \triangle_{kj}^{rt}|_{card} - |\triangle_{ij}^{pq}|_{card}\ , \quad (7)$$

where the first term of the RHS computes the consistency lower bound for each transitive flow and takes the maximum. Intuitively, $\mathcal{P}(i,j,p)$ is the lower bound of cycle consistency improvement if T_{ij}^{pq} is replaced by the transitive flow through image $\hat{k}$, where $\hat{k} = \arg\max_k |\triangle_{ik}^{pr} \cap \triangle_{kj}^{rt}|_{card}$. See Figure 2 for an illustration of the update priority pattern on a set of synthetic examples.

Intra-image phase While the previous phase essentially identifies and updates inconsistent flows to consistent ones through propagation, it is nonetheless unable to deal with cases in which the correct correspondence does not exist in the initial flow set, or simply has low cycle consistency because most of its transitive counterparts are noisy. Consider a set of front-view car images. The hood is typically texture-less while occupying a large image area, and pairwise matching based on low-level features such as SIFT would be highly noisy. As a result, it is likely that all flows emanating from such regions are incorrect and not consistent for propagation with the priority-based update.

The second phase of our iterative optimization addresses this issue by exploiting *consistency-weighted spatial smoothing*, which identifies highly-consistent flows within a pairwise flow field, and utilizes them as *soft* anchor points to guide inconsistent flows to likely better solutions. For the example of front-view cars, one could potentially use flows from headlights or window corners that tend to be more cycle-consistent to guide flows from the hood. Specifically, for each flow field corresponding to a pair of images, we first identify flows that are of relatively low cycle consistency, and then apply a consistency-weighted Gaussian filter to each of them by

$$T_{ij}^{pq} = \frac{1}{Z} \sum_{p' \in I_i} T_{ij}^{p'q'} g_{\sigma_s}(\|x_{p'} - x_p\|) h_{\sigma_c}(\mathcal{C}(T_{ij}^{p'q'}) - \mathcal{C}(T_{ij}^{pq})) \quad (8)$$

where

$$Z = \sum_{p' \in I_i} g_{\sigma_s}(\|x_{p'} - x_p\|) h_{\sigma_c}(\mathcal{C}(T_{ij}^{p'q'}) - \mathcal{C}(T_{ij}^{pq}))\ . \quad (9)$$

$g_{\sigma_s}(\cdot)$ is a zero-mean Gaussian with σ_s controlling the spatial extent of the filter, and

$$h_{\sigma_c}(x) = \begin{cases} \exp(x/\sigma_c) & \text{if } x \geq 0 \\ 0 & \text{Otherwise} \end{cases} \quad (10)$$

determines how much an adjacent flow is weighted according to the gap in cycle consistency. Having $g(\cdot)$ and $h(\cdot)$ together ensures that each filtered flow is only influenced by flows that are both spatially near *and* more cycle-consistent.

Our iterative update pipeline is summarized below:

1. Compute the $SFCC$ score for each T_{ij}^{pq} using Eq. 3.
2. For each T_{ij}^{pq}, compute its update priority by Eq. 7, and record the node $\hat{k}$ that achieves the maximum.
3. Sort flows according to $\mathcal{P}(i,j,p)$, and update top $\beta\%$ flows by their transitive alternatives through image $\hat{k}$.
4. For each image pair i and j, apply Eq. 8 for consistency-weighted filtering.
5. Iterate 1–4 until the improvement of $\mathcal{C}(\mathbf{T})$ is below some threshold.

Regularization Optimizing the regularization term $\mathcal{R}(\cdot)$ can be easily incorporated into both update phases above. For the inter-image phase, the update priority becomes

$$\mathcal{P}(i,j,p) = \max_k |\triangle_{ik}^{pr} \cap \triangle_{kj}^{rt}|_{card} - |\triangle_{ij}^{pq}|_{card} - \lambda(\|T_{ik}^{pr} + T_{kj}^{rt} - S_{ij}^{ps}\| - \|T_{ij}^{pq} - S_{ij}^{ps}\|). \quad (11)$$

Similarly for the intra-image phase, we replace $h_{\sigma_c}(\mathcal{C}(T_{ij}^{p'q'}) - \mathcal{C}(T_{ij}^{pq}))$ with $h_{\sigma_c}(\mathcal{C}(T_{ij}^{p'q'}) - \mathcal{C}(T_{ij}^{pq}) - \lambda(\|T_{ij}^{p'q'} - S_{ij}^{ps}\| - \|T_{ij}^{pq} - S_{ij}^{ps}\|))$.

Implementation details: For better robustness to noisy initial pairwise matching, we use a relaxed threshold for determining cycle completeness in Eq. 3. In particular, we replace $[T_{ik}^{pr} + T_{kj}^{rq} = T_{ij}^{pq}]$ with $[\|T_{ik}^{pr} + T_{kj}^{rq} - T_{ij}^{pq}\| \leq \epsilon]$, where $\epsilon = 0.05 \cdot \max(h,w)$ (h and w are image height and width). $\beta = 20, \sigma_c = 0.05, \sigma_s = \epsilon$, and $\lambda = 0.01$ for all our experiments. The code will be available on our website.

4. Experiments

We compare our alignment performance with Congealing [23] (using SIFT), Collection Flow [18], DSP [21], and RASL [33]. All the baseline algorithms perform joint alignment across the whole image collection, except DSP, which is the state-of-the-art pairwise image matching algorithm and also used by us to initialize $\mathbf{T}_0$. We use publicly available code for all baselines except Collection Flow, for which we implement our own version in Matlab. All baselines are run with default parameters.

The image sets we use are sampled from the PASCAL-Part dataset [5]. To parse the images of each category into sets that are meaningful to align (a counter example would be aligning front-view cars to side-view cars), we run K-means clustering ($K = 10$) on the provided part visibility labels and coarse viewpoint annotations from the original VOC 2010 dataset, and select three representative clusters with largest sizes to evaluate for each category. A cluster is pruned if it has less than 10 images since joint alignment has little effect with few samples. The total number of image sets remaining is 47. In the interest of time, we limit

	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mean
Congealing	0.26	0.40	**0.24**	0.48	0.68	0.46	0.39	0.19	0.49	0.30	0.42	**0.15**	0.26	0.32	0.18	0.38	0.35	0.71	0.45	0.58	0.38
RASL	0.26	0.40	0.22	0.49	0.70	0.46	0.42	0.19	0.51	0.30	**0.43**	**0.15**	0.25	0.33	0.18	0.38	0.34	0.72	0.47	0.64	0.39
CollectionFlow	0.29	0.40	0.22	0.49	0.69	0.46	0.41	**0.20**	0.51	0.28	0.35	**0.15**	0.25	0.28	0.18	0.36	0.34	0.66	0.44	0.59	0.38
DSP	0.25	0.46	0.21	0.48	0.63	0.50	0.45	0.19	0.48	0.30	0.37	0.14	0.26	0.35	0.13	0.40	0.37	0.66	0.48	0.62	0.39
Ours	**0.33**	**0.53**	**0.24**	**0.51**	**0.72**	**0.54**	**0.51**	**0.20**	**0.52**	**0.32**	0.41	**0.15**	**0.29**	**0.45**	**0.19**	**0.41**	**0.39**	**0.73**	**0.51**	**0.68**	**0.43**

Table 1. Weighted intersection over union (IOU) for part segment matching on 20 PASCAL VOC categories. Higher is better.

	aero	bike	boat	bottle	bus	car	chair	table	mbike	sofa	train	tv	mean
Congealing	0.12	0.23	0.03	0.22	0.19	0.14	0.06	**0.04**	0.12	0.07	0.08	0.06	0.11
RASL	0.18	0.17	0.04	0.33	0.31	0.17	0.09	**0.04**	0.12	0.10	0.11	0.23	0.16
CollectionFlow	0.16	0.17	0.04	0.31	0.25	0.16	0.09	0.02	0.08	0.07	0.06	0.09	0.12
DSP	0.17	0.30	**0.05**	0.19	0.33	0.34	0.09	0.03	0.17	0.12	0.12	0.18	0.17
Ours	**0.29**	**0.41**	**0.05**	**0.34**	**0.54**	**0.50**	**0.14**	**0.04**	**0.21**	**0.16**	**0.15**	**0.33**	**0.26**

Table 2. Keypoint matching accuracy (PCK) on 12 rigid PASCAL VOC categories. Higher is better.

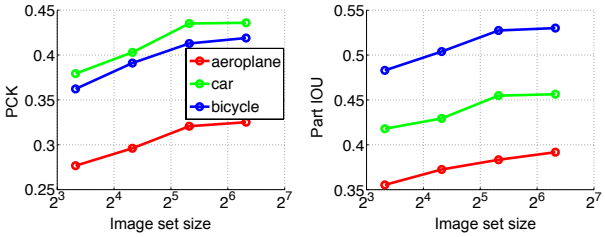


Figure 6. Alignment accuracy as a function of image set size using our method. The test set remains fixed as more images are included for joint alignment. Left: PCK. Right: Part segment IOU. Overall, more images leads to more accurate correspondences.

the largest size of each set to 100. Images within each set are further resized to the average aspect ratio and maximum dimension of 150.

4.1. Part segment matching

We first evaluate alignment quality using human-annotated part segments. For quantitative evaluation, we use weighted intersection over union (IOU) with weights determined by the pixel area of each part, and report the mean performance over all sets for each category in Table 1. For categories without part annotations (boat, chair, table, and sofa) we simply use silhouette annotations for evaluation. We outperform all baselines on almost all categories.

We also visualize the part matching results in Fig. 4. Overall, our method is able to produce substantially more accurate correspondences than the baselines. The fact that many of the mistakes made by the initial DSP matching are corrected in our final output highlights the effectiveness of our joint alignment procedure.

4.2. Keypoint matching

We next compare alignment accuracy using keypoint annotations for the 12 rigid PASCAL categories provided by [44]. We use the same set of images sampled in the previous experiment. The matching accuracy is assessed by the standard PCK measure [45], which defines a keypoint matching to be correct if the prediction falls within $\alpha \cdot \max(h, w)$ pixels of the ground-truth (h and w are image height and width respectively). For each category, we report the mean PCK

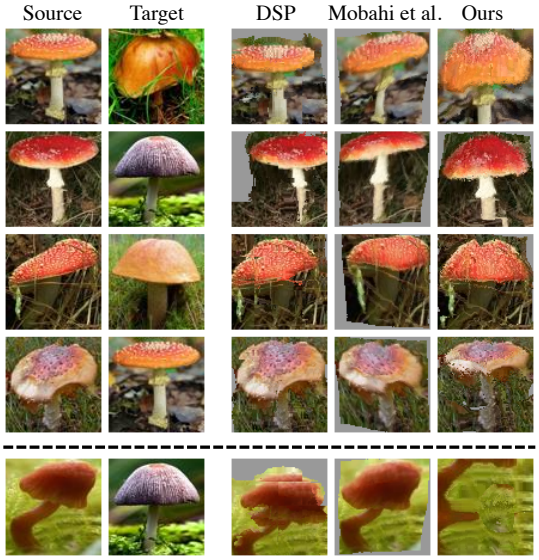


Figure 7. Comparison with the compositional model of [31]. Rows 1–4/5 are success/failure examples of our method.

over all sampled sets with different methods in Table 2. Again, our method substantially outperforms all baselines.

Fig. 5 compares the keypoint correspondence tracks between DSP (pairwise matching used for our initialization) and ours. DSP tracks tend to drift more as the path becomes longer, while our tracks are relatively stable and cycle-consistent along the graph (note that the first and the last image is the same for all examples).

4.3. Effect of image collection size

We hypothesize that the more images in the set, the better correspondences our method would produce as the cycle consistency measure becomes more robust. To verify this, we plot alignment accuracy as a function of image set size. Specifically, for car, aeroplane, and bicycle categories, we randomly sample 10 images as the test set for evaluation, and progressively add more images to construct the alignment set together with the 10 test images. As shown in Fig. 6, both keypoint and part-based matching accuracies indeed improve as more images become available.

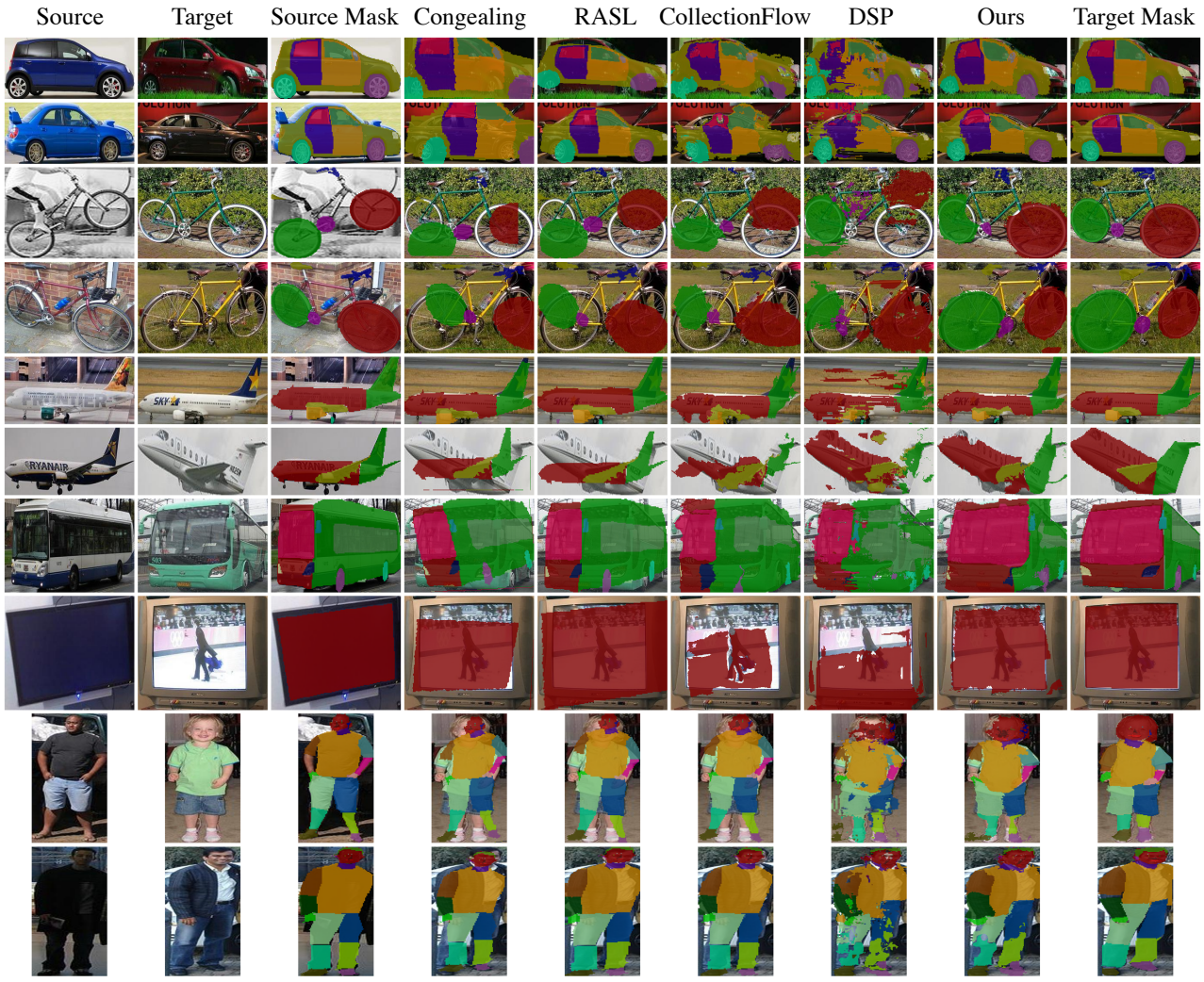


Figure 4. Correspondence visualization for different methods with color-coded part segments. Columns 1–2: source and target images. Column 3: annotated part segments for the source image. Column 4–8: predicted part correspondences on the target image using different methods. Column 9: annotated part segments for the target image (i.e. ground-truth). Overall, our correspondence output improves significantly over the initial DSP matching, and align part segments in greater precision than all baselines. (Best viewed in pdf.)

4.4. Comparison with Mobahi et al. [31]

To compare with Mobahi et al. [31], we use their Mushroom dataset [31], comprised of 120 mushroom images and ground-truth foreground region and boundary masks for evaluation. After joint alignment, for each image pair, we compute both region and boundary matching scores as defined in [31]. The region score measures the fraction of foreground pixels in the warped source image that coincide with the foreground pixels in the target image (perfect alignment would result in a region score of 1; so higher is better). The boundary matching score measures the boundary displacement error (in pixels) between the warped source image and the target image (perfect alignment would result in a boundary score of 0; so lower is better). We average these scores computed for every pair of images in the dataset.

We obtain 0.84 and 6.44 for region and boundary alignment, respectively, compared to Mobahi et al.'s 0.73 and 5.69. Upon closer examination of why we perform worse in the boundary measure, we find our alignment to be more deformable than [31]. This can lead to highly accurate results (top four rows in Fig. 7) but also to very poor results if the deformation of the object is completely wrong (bottom row in Fig. 7). Such behavior could greatly affect boundary matching score as it is very sensitive to outliers.

4.5. Annotation-free Active Appearance Models

Training Active Appearance Models (AAM) [8] typically requires extensive human labeling of landmark keypoints. We show that it is possible to bypass the keypoint annotation step by using the cycle-consistency measure to

Figure 5. Comparison of keypoint correspondence tracks along a cycle in the graph (the first and the last image is the same for all examples) between DSP (initialization to our method) and ours. The keypoint correspondences become much more accurate and cycle-consistent after our joint alignment procedure.

identify keypoint surrogates. In particular, we can sum over the $SFCC$ score for all the flows coming out of a pixel p in image i by $\sum_{j=1}^{N}\sum_{k=1,k\notin\{i,j\}}^{N}[T_{ij}^{pq} = T_{ik}^{pr} + T_{kj}^{rq}]$, and use it to guide keypoint selection. Here is a simple pipeline: 1) Compute per-pixel consistency score using the above equation; 2) Pick a seed alignment image with the highest overall consistency; 3) Run max pooling to select a sparse set of candidate keypoints; 4) Do thresholding to select a final high-quality set of keypoints; 5) Obtain the keypoint correspondences for the rest of the image set according to the flows from the seed image to the target. Once the keypoints are established, standard AAM can be applied (we used the package from [38]). Fig. 8 shows sample results on cars.

4.6. Runtime complexity

For 50 images of size 150×150, our algorithm typically takes about 10 iterations to converge, and each iteration takes about 10 minutes on a 3GHz, 16GB machine using a Matlab implementation. For 100 images, each iteration takes about an hour. There are two major computational bottlenecks: 1) The computation of priority is $\mathcal{O}(MN^4)$; 2) Consistency-weighted filtering is $\mathcal{O}(N^2M^2)$. One way to speed up the alignment process is to first break down the fully-connected graph into sub-clusters (to reduce N) and optimize the flows within each cluster, and then bring them together by connecting the closest matches between clusters. Our preliminary experiments show that the overall alignment accuracy won't be compromised much with such approximation as long as the size of each cluster is still considerably large. We plan to explore more options for efficiency improvement in the future.

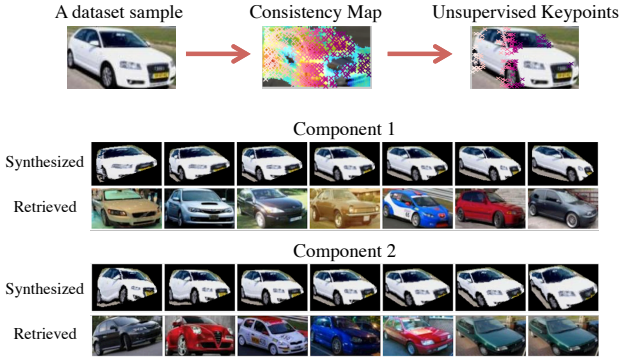


Figure 8. Visualization of unsupervised keypoint selection using cycle-consistency and its application to AAM (see Sec. 4.5 for more details). By varying the coefficients for AAM shape components, one can synthesize new instances that pertain to the variations within the image collection.

5. Discussion

Now that object detection and retrieval is finally starting to work, it's possible to go from a very large, unorganized image collection to a relatively small set of coarsely-aligned images, e.g. using CNNs [22]. But going from coarse to fine-grained pixel-wise correspondence is still very much an open problem, the solution to which could benefit many vision and graphics tasks, including image edit propagation [14, 46], co-segmentation [34, 7], structure-from-motion [36], 3D object reconstruction [4, 39] unsupervised object discovery [9, 34, 7], etc.

Acknowledgements We thank Philipp Krähenbühl and Jun-Yan Zhu for insightful discussions. This work is partially sponsored by ONR MURI N000141010934. The authors are also grateful to the young Jessica for not being in a hurry.

References

[1] C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *SIGGRAPH*, 28(3), 2009. 2

[2] C. Barnes, E. Shechtman, D. B. Goldman, and A. Finkelstein. The generalized patchmatch correspondence algorithm. In *ECCV*, 2010. 2

[3] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *SIGGRAPH*, pages 187–194, 1999. 1

[4] J. Carreira, A. Kar, S. Tulsiani, and J. Malik. Virtual view networks for object reconstruction. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*. IEEE, 2015. 3, 8

[5] X. Chen, R. Mottaghi, X. Liu, S. Fidler, R. Urtasun, and A. Yuille. Detect what you can: Detecting and representing objects using holistic models and body parts. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 5

[6] X. Chen, A. Shrivastava, and A. Gupta. Neil: Extracting visual knowledge from web data. In *ICCV*, 2013. 2

[7] X. Chen, A. Shrivastava, and A. Gupta. Enriching visual knowledge bases via object discovery and segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 2035–2042. IEEE, 2014. 8

[8] T. F. Cootes, G. J. Edwards, C. J. Taylor, et al. Active appearance models. *TPAMI*, 23(6):681–685, 2001. 1, 7

[9] C. Doersch, A. Gupta, and A. A. Efros. Context as supervisory signal: Discovering objects with predictable context. In *Computer Vision–ECCV 2014*, pages 362–377. Springer, 2014. 8

[10] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. A. Efros. What makes paris look like paris? *SIGGRAPH*, 31(4):101, 2012. 2

[11] A. Faktor and M. Irani. "Clustering by Composition"–Unsupervised discovery of image categories. In *ECCV*. 2012. 2

[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 2

[13] Y. HaCohen, E. Shechtman, D. B. Goldman, and D. Lischinski. Optimizing color consistency in photo collections. *SIGGRAPH*, 32(4):85:1 – 85:9, 2013. 2

[14] S. W. Hasinoff, M. Jwiak, F. Durand, and W. T. Freeman. Search-and-replace editing for personal photo collections. In *ICCP*, 2010. 8

[15] K. Heath, N. Gelfand, M. Ovsjanikov, M. Aanjaneya, and L. J. Guibas. Image webs: Computing and exploiting connectivity in image collections. In *CVPR*, 2010. 2

[16] G. B. Huang, V. Jain, and E. Learned-Miller. Unsupervised joint alignment of complex images. In *ICCV*, 2007. 2

[17] Q. Huang and L. Guibas. Consistent shape maps via semidefinite programming. In *SGP*, 2013. 2, 3

[18] I. Kemelmacher-Shlizerman and S. Seitz. Collection flow. In *CVPR*, 2012. 2, 3, 5

[19] I. Kemelmacher-Shlizerman and S. M. Seitz. Face reconstruction in the wild. In *ICCV*, 2011. 1

[20] I. Kemelmacher-Shlizerman, E. Shechtman, R. Garg, and S. M. Seitz. Exploring photobios. *SIGGRAPH*, 30(4):61, 2011. 1, 2

[21] J. Kim, C. Liu, F. Sha, and K. Grauman. Deformable spatial pyramid matching for fast dense correspondences. In *CVPR*, 2013. 1, 2, 4, 5

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 8

[23] E. Learned-Miller. Data driven image models through continuous joint alignment. *TPAMI*, 28(2):236–250, 2005. 2, 5

[24] Y. J. Lee and K. Grauman. Collect-Cut: Segmentation with Top-Down Cues Discovered in Multi-Object Images. In *CVPR*, 2010. 2

[25] C. Liu, J. Yuen, and A. Torralba. Sift flow: Dense correspondence across scenes and its applications. *TPAMI*, 33(5):978–994, 2011. 1, 2, 4

[26] J. L. Long, N. Zhang, and T. Darrell. Do convnets learn correspondence? In *Advances in Neural Information Processing Systems*, pages 1601–1609, 2014. 2

[27] Y. Lou, N. Snavely, and J. Gehrke. Matchminer: Efficient spanning structure mining in large image collections. In *ECCV*, 2012. 2

[28] T. Malisiewicz and A. A. Efros. Beyond categories: The visual memex model for reasoning about object relationships. In *NIPS*, 2009. 2

[29] I. Matthews and S. Baker. Active appearance models revisited. *IJCV*, 60(2):135–164, 2004. 1

[30] X. Miao and R. P. N. Rao. Learning the lie groups of visual invariance. *Neural Computation*, 2007. 2

[31] H. Mobahi, C. Liu, and W. T. Freeman. A compositional model for low-dimensional image set representation. In *CVPR*, 2014. 2, 3, 6, 7

[32] A. Nguyen, M. Ben-Chen, K. Welnicka, Y. Ye, and L. Guibas. An optimization approach to improving collections of shape maps. In *SGP*, 2011. 3

[33] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma. RASL: Robust Alignment by Sparse and Low-rank Decomposition for Linearly Correlated Images. *TPAMI*, 34(11), November 2012. 2, 5

[34] M. Rubinstein, A. Joulin, J. Kopf, and C. Liu. Unsupervised joint object discovery and segmentation in internet images. In *CVPR*, 2013. 2, 8

[35] A. C. Sankaranarayanan, C. Hegde, S. Nagaraj, and R. G. Baraniuk. Go with the flow: Optical flow-based transport operators for image manifolds. In *Annual Allerton Conference on Communication, Control, and Computing*, 2011. 2, 3

[36] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM transactions on graphics (TOG)*, volume 25, pages 835–846. ACM, 2006. 8

[37] A. Torralba. http://people.csail.mit.edu/torralba/gallery/, 2001. 2

[38] G. Tzimiropoulos and M. Pantic. Optimization problems for fast aam fitting in-the-wild. In *ICCV*, 2013. 8

[39] S. Vicente, J. Carreira, L. Agapito, and J. Batista. Reconstructing pascal voc. In *CVPR*, 2014. 2, 8

[40] F. Wang, Q. Huang, and L. Guibas. Image co-segmentation via consistent functional maps image co-segmentation via consistent functional maps. In *ICCV*, 2013. 2, 3

[41] F. Wang, Q. Huang, M. Ovsjanikov, and L. Guibas. Unsupervised multi-class joint image segmentation. In *CVPR*, 2014. 2, 3

[42] K. Wilson and N. Snavely. Network principles for sfm: Disambiguating repeated structures with local context. In *ICCV*, 2013. 2, 3

[43] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma. Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. In *NIPS*, 2009. 3

[44] Y. Xiang, R. Mottaghi, and S. Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *WACV*. 2014. 6

[45] Y. Yang and D. Ramanan. Articulated human detection with flexible mixtures of parts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(12):2878–2890, 2013. 6

[46] K. Ycer, A. Jacobson, A. Sorkine-Hornung, and O. Sorkine-Hornung. Transfusive image manipulation. *SIGGRAPH Asia*, 31(6), 2012. 8

[47] C. Zach, M. Klopschitz, and M. Pollefeys. Disambiguating visual relations using loop constraints. In *CVPR*, 2010. 2, 3