# LEARNING VISUAL GROUPS FROM CO-OCCURRENCES IN SPACE AND TIME

**Phillip Isola**
UC Berkeley
phillipi@berkeley.edu

**Daniel Zoran**
MIT
danielz@mit.edu

**Dilip Krishnan**
Google
dilipkay@google.com

**Edward H. Adelson**
MIT
adelson@mit.edu

## ABSTRACT

We propose a self-supervised framework that learns to group visual entities based on their rate of co-occurrence in space and time. To model statistical dependencies between the entities, we set up a simple binary classification problem in which the goal is to predict if two visual primitives occur in the same spatial or temporal context. We apply this framework to three domains: learning patch affinities from spatial adjacency in images, learning frame affinities from temporal adjacency in videos, and learning photo affinities from geospatial proximity in image collections. We demonstrate that in each case the learned affinities uncover meaningful semantic groupings. From patch affinities we generate object proposals that are competitive with supervised methods. From frame affinities we generate movie scene segmentations that correlate well with DVD chapter structure. Finally, from geospatial affinities we learn groups that relate well to semantic place categories.

## 1 INTRODUCTION

Clown fish live next to sea anemones, lightning is always accompanied by thunder. When looking at the world around us, we constantly notice which things go with which. These associations allow us to segment and organize the world into coherent visual representations.

This paper addresses how representations like "objects" and "scenes" might be learned from natural visual experience. A large body of work has focused on learning these representations as a supervised problem (e.g., by regressing on image labels) and current object and scene classifiers are highly effective. However, in the absence of expert annotations, it remains unclear how we might uncover these representations in the first place. Do "objects" fall directly out of the statistics of the environment, or are they a more subjective, human-specific construct?

Here we probe the former hypothesis. Because the physical world is highly structured, adjacent locations are usually semantically related, whereas far apart locations are more often semantically distinct. By modeling spatial and temporal dependencies, we may therefore learn something about semantic relatedness.

We investigate how these dependences may be learned from unlabeled sensory input. We train a deep neural network to predict whether or not two input images or patches are likely to be found next to each other in space or time. We demonstrate that the network learns dependencies that can be used to uncover meaningful visual groups. We apply the method to generate fast and accurate object proposals that are competitive with recent supervised methods, as well as to automatic movie scene segmentation, and to the grouping of semantically related photographs.

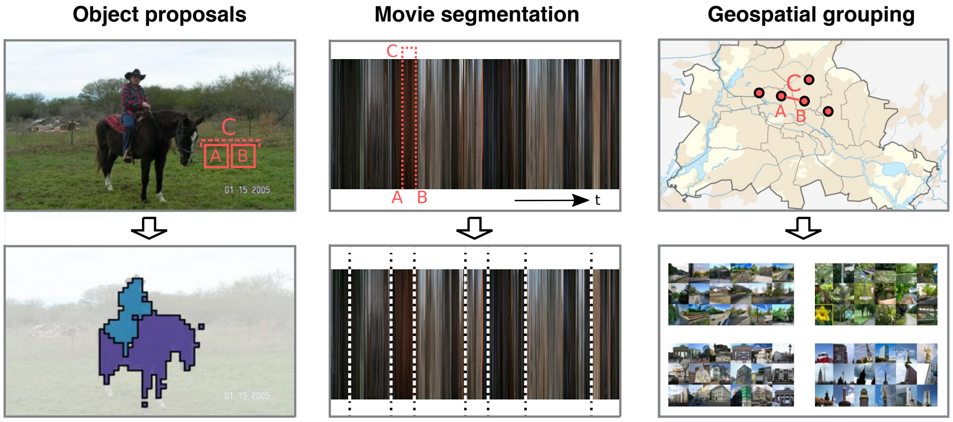

Figure 1: We model statistical dependences in the visual world by learning to predict which visual primitives – patches, frames, or photos – will be likely to co-occur within the same spatial or temporal context. Above, the primitives are labeled $A$ and $B$, and the context is labeled $C$. By clustering primitives that predictably co-occur, we can uncover groupings such as objects (a group of patches; left), movie scenes (a group of frames; middle), and place categories (a group of photos; right).

## 2 Related work

The idea that perceptual groups reflect statistical structure in the environment has deep roots in the perception and cognition literature (Barlow (1985); Wilkin & Tenenbaum (1985); Tenenbaum & Witkin (1983); Lowe (2012); Rock (1983)). Barlow postulated that the brain is constantly on the lookout for events that co-occur much more often than chance, and uses these "suspicious coincidences" to discover underlying structure in the world (Barlow (1985)). Subsequent researchers have argued that infants learn about linguistic groups from phoneme co-occurrence (Saffran et al. (1996)), and that humans may also pick up on visual patterns from co-occurrence statistics alone (Fiser & Aslin (2001); Schapiro et al. (2013)).

Visual grouping is also a central problem in computer vision, showing up in the tasks of edge/contour detection (Canny (1986); Arbelaez et al. (2011); Isola et al. (2014)), (semantic) segmentation (Shi & Malik (2000); Malisiewicz & Efros (2007)), and object proposals (Alexe et al. (2012); Zitnick & Dollár (2014); Krahnenbuhl & Koltun (2015)), among others. Many papers in this field take the approach of first modeling the affinity between visual elements, then grouping elements with high affinity (e.g., Shi & Malik (2000)). Our work follows this approach. However, rather than using a hand-engineered grouping cue (Shi & Malik (2000); Zitnick & Dollár (2014)), or learning to group with direct supervision (Dollár & Zitnick (2013); Krahnenbuhl & Koltun (2015)), we use a measure of spatial and temporal dependence as the affinity.

Grouping based on co-occurrence has received some prior attention in computer vision (Sivic et al. (2005); Faktor & Irani (2012; 2013); Isola et al. (2014)). Sivic et al. (2005) demonstrated that object categories can be discovered and roughly localized using an unsupervised generative model. Isola et al. (2014) showed that statistical dependences between adjacent pixel colors, measured by pointwise mutual information (PMI) can be very effective at localizing object boundaries. Both these methods require modeling generative probability distributions, which restricts their ability to scale to high-dimensional data. Our model, on the other hand, is discriminative and can be easily scaled.

A recent line of work in representation learning has taken a similar tack, training discriminative models to predict one aspect of raw sensory data from another. This work may be termed self-supervised learning and has a number of flavors. The common theme is exploiting spatial and/or temporal structure as supervisory signals. Mobahi et al. (2009) learn a feature embedding such that features adjacent in time are similar and features far apart in time are dissimilar. Srivastava et al. (2015) predict future frames in a video, and rely on strict temporal ordering; extension to spatial or unordered data is unclear. Wang & Gupta (2015) use a siamese triplet loss to learn a representation that can track patches through a video. They rely on training input from a separate tracking algorithm. Agrawal et al. (2015) as well as Jayaraman & Grauman (2015) regress on egomotion sig-

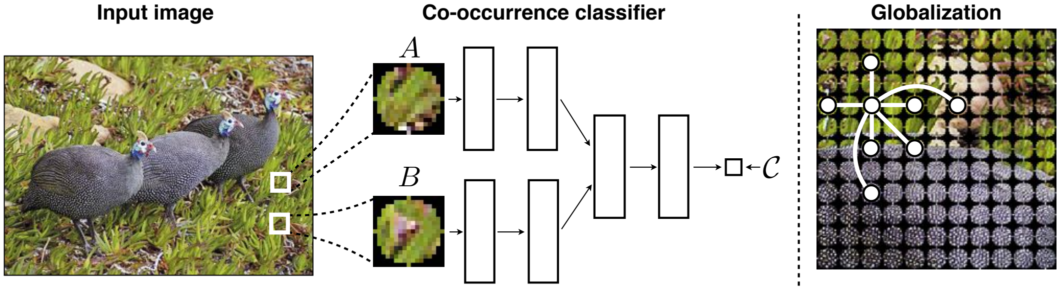

Figure 2: Overview of our approach to learning to group patches. We train a classifier to that takes two isolated patches, $A$ and $B$, and predicts $\mathcal{C}$: whether or not they were taken from nearby locations in an image. We use the output of the classifier, $P(\mathcal{C} = 1|A, B)$, as an affinity measure for grouping. The rightmost panel shows our grouping strategy. We setup a graph in which nodes are image patches, and all nearby nodes are connected with an edge, weighted by the learned affinity (for clarity, only a subset of nodes and edges are shown). We then apply spectral clustering to partition this graph and thereby segment the image. The result on this image is shown in Figure 5.

nals to learn a representation. Finally, Doersch et al. (2015) learn features by predicting the relative orientation between patches within an image.

Each of these works focus on learning good generic features, which may then be applied as pretraining for a supervised model. Our current goal is rather different. Rather than learning a vector space representation of images, we search for more explicit structure, in the form of visual groups. We show that the learned groups are semantically meaningful in and of themselves. This differs from the usual approach in feature learning, where the features are not necessarily interpretable, but are instead used as an intermediate representation on top of which further models can be trained.

## 3 Modeling visual affinities by predicting co-occurrence

We would like to group visual primitives, $A$ and $B$, based on the probability that they belong to the same semantic entity. $A$ and $B$ may, for example, be two image patches, in which case we would want to group them if they belong to the same visual object.

Given object labels, a straightforward approach to this problem would be to train a supervised classifier to predict indicator variable $\mathcal{Q} \in \{0, 1\}$, where $\mathcal{Q} = 1$ iff $A$ and $B$ lie on the same object (Manen et al. (2013)). Throughout this paper, we use $\mathcal{Q}$ to indicate the property that $A$ and $B$ share the same semantic label.

Acquiring training data for $\mathcal{Q}$ may require time-consuming and expensive annotation. We instead will explore an alternative strategy. Instead of training a classifier to predict $\mathcal{Q}$ directly, we train classifiers to predict spatial or temporal proximity, denoted by $\mathcal{C} \in \{0, 1\}$. Because the semantics of the world change slowly over space and time, we hope that $\mathcal{C}$ might serve as a cheap proxy for $\mathcal{Q}$ (c.f. Kayser et al. (2001); Wiskott & Sejnowski (2002)). The degree to which this is true is an empirical question, which we will test below. Throughout the paper, $\mathcal{C} = 1$ iff $A$ and $B$ are nearby each other in space or time.

Formally, we model the affinity between visual primitives $A$ and $B$ as

$$w(A, B) = \frac{P(\mathcal{C} = 1|A, B) + P(\mathcal{C} = 1|B, A)}{2}. \tag{1}$$

In other words, we model affinity as the probability that two primitives will co-occur within some context (with symmetry between the order of $A$ and $B$ enforced). We will then use this affinity metric to cluster primitives, in particular using spectral clustering (Section 4).

This affinity can be understood in a variety of ways. First, as described above, it can be seen as a proxy for what we are really after, $P(\mathcal{Q} = 1|A, B)$. Second, it is a measure of the spatial/temporal dependence between $A$ and $B$. Applying Bayes' rule we can see that $P(\mathcal{C} = 1|A, B) \propto \frac{P(A,B|\mathcal{C}=1)}{P(A,B)}$, which factors to $\frac{P(A,B|\mathcal{C}=1)}{P(A)P(B)}$ when $A$ and $B$ are independent. Therefore, if we sample primitives iid

Table 1: Average precision scores of our method (labeled "Co-occurrence classifier") compared to baselines at predicting $\mathcal{C}$ (spatial or temporal adjacency) and $\mathcal{Q}$ (semantic sameness) for three domains: image patches, video frames, and geospatial photos.

| | **Patches** | | **Frames** | | **Photos** | |
|---|---|---|---|---|---|---|
| **Affinity measure** | $\mathcal{C}$ | $\mathcal{Q}$ | $\mathcal{C}$ | $\mathcal{Q}$ | $\mathcal{C}$ | $\mathcal{Q}$ |
| Raw color | 0.83 | 0.73 | 0.77 | 0.58 | 0.58 | 0.58 |
| Mean color | 0.87 | 0.74 | 0.82 | 0.63 | 0.56 | 0.57 |
| Color histogram | 0.95 | **0.80** | 0.90 | 0.64 | 0.63 | 0.62 |
| HOG | 0.67 | 0.67 | 0.77 | 0.61 | 0.63 | 0.75 |
| Co-occurrence classifier | **0.96** | **0.80** | **0.95** | **0.67** | **0.70** | **0.79** |

in space or time[1], our affinity models how much more often $A$ and $B$ appear nearby each other than they would if they were independent (the rate of co-occurrences were they independent would simply be $P(A)P(B)$). This value is closely related to the pointwise mutual information (PMI) between $A$ and $B$ conditioned on $\mathcal{C} = 1$. Previous work on visual grouping (Isola et al. (2014)) and word embeddings (Church & Hanks (1990); Levy et al. (2014)) has found PMI to be an effective measure for these tasks.

### 3.1 Predicting co-occurrences with a CNN

To model $w(A, B)$ we use a Convolutional Neural Net (CNN) with a Siamese-style architecture (Figure 2, Chopra et al. (2005)), which we implement in Caffe (Jia et al. (2014)). The network has two convolutional branches, one to process $A$ and the other to process $B$, with shared weights. These branches can be regarded as feature extractors. The features are then concatenated and fed to a set of fully connected layers that compare the features and try to predict $\mathcal{C}$. We use a logistic loss over $\mathcal{C}$ and train all models with stochastic gradient descent. Our objective can be expressed as

$$E(\mathbf{A}, \mathbf{B}, \mathcal{C}; \theta) = \frac{-1}{N} \sum_{1}^{N} \mathcal{C}_i \log(\sigma(f(\mathbf{A}_i, \mathbf{B}_i; \theta)) + (1 - \mathcal{C}_i) \log(1 - \sigma(f(\mathbf{A}_i, \mathbf{B}_i; \theta)) \quad (2)$$

where $\theta$ are the net parameters we optimize over (weights and biases), $N$ is the number of training examples, $\sigma$ is the logistic function, and $f$ is a neural net. For each of our experiments, $N = 500,000$ training examples, 50% of which are positive ($\mathcal{C} = 1$) and 50% negative ($\mathcal{C} = 0$).

We examine three domains: 1) learning to group patches based on their spatial adjacency in images, 2) learning to group video frames based on their temporal adjacency in movies, and 3) learning to group photos based on their geospatial proximity.

Each task corresponds to a different choice of $A$, $B$, $\mathcal{C}$, and $\mathcal{Q}$. In each case, we analyze performance at predicting $\mathcal{C}$ and at predicting $\mathcal{Q}$, comparing our CNN to baseline grouping cues. Results of this analysis are given in Table 1. As baselines, we consider both color and gradient-based similarity measures, defined as follows. "Raw color" similarity refers to Euclidean distance between all RGB values in $A$ and $B$. "Mean color" similarity is the same but for the mean RGB value in $A$ and $B$. "Color histogram" similarity involves first vector quantizing colors into 256 values (we learn the quantization using k-means on RGB colors in the training set), and then measuring similarity with a histogram intersection kernel. "HOG" similarity is measured as Euclidean distance between HOG vectors (Dalal & Triggs (2005)) given by the implementation in Dollár (2015). Similarity measures like these are commonly used in visual grouping algorithms (Arbelaez et al. (2011); Faktor & Irani (2012)). In all cases, our co-occurrence classifier matches or outperforms the baselines.

### 3.2 Patch affinities from co-occurrence in images

We first examine whether or not predicting patch co-occurrence in images results in an effective affinity measure for object-level grouping. We set $A$ and $B$ to be $17 \times 17$ pixel patches (with circular masks). The context function $\mathcal{C}$ is spatial adjacency. Positive examples ($\mathcal{C} = 1$) are pairs of adjacent patches (with no overlap between them) and negative examples ($\mathcal{C} = 0$) are pairs sampled

[1] In each of our applications we sample 50% positive ($\mathcal{C} = 1$) and 50% negative ($\mathcal{C} = 0$) examples. Under our logistic regression model, this results in a function monotonically related to what we would get if we had sampled iid (see appendix B.4 in King & Zeng (2001))

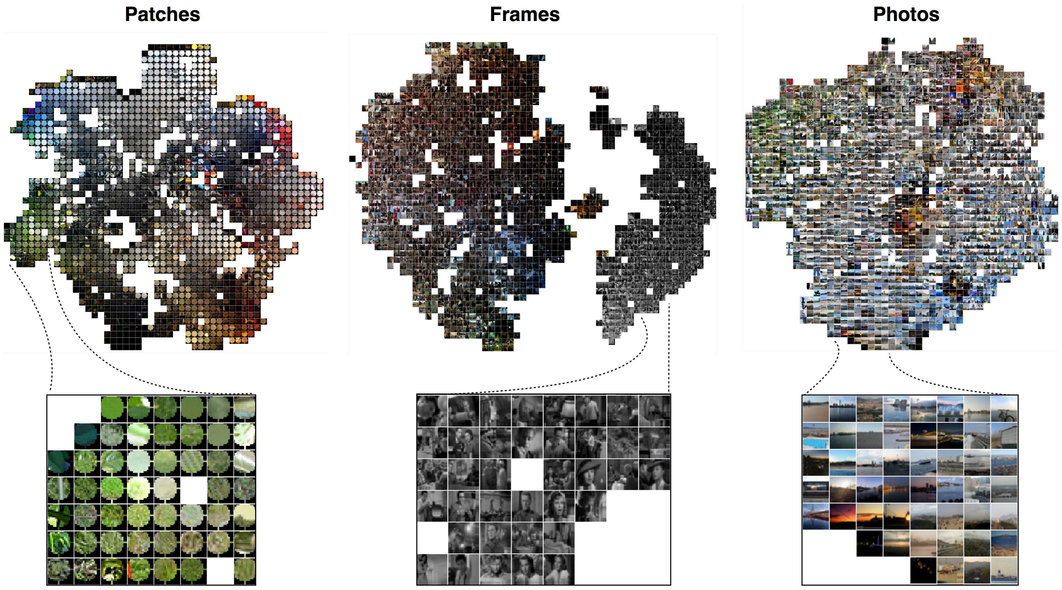

Figure 3: t-SNE visualizations of the learned affinities in each domain. We construct an affinity matrix between 3000 randomly sampled primitives to create each visualization, using $w(A, B)$ as the affinity measure. We then apply t-SNE on this matrix (van der Maaten & Hinton (2009)). To avoid clutter, we visualize the embedded primitives snapped to the nearest point on a grid. The learned affinities pick up on different kinds of similarity in each domain. Patches are arranged largely according to color, while the geo-photo affinities are less dependent on color, as can be seen in the inset where day and night waterfronts map to nearby points in the t-SNE embedding.

from random locations across the dataset. We sample training patches from the Pascal VOC 2012 training set.

We train a CNN (two convolutional layers, two fully connected layers; Figure 2) to model $P(\mathcal{C} = 1|A, B)$. Figure 3 shows a t-SNE visualization of the learned affinities (van der Maaten & Hinton (2009)). As can be seen, the network learns to associate patches with different kinds of structure such as texture, local features, and color similarities.

To evaluate performance, we sample 10,000 patches from the Pascal VOC 2012 validation set, $50\%$ with $\mathcal{C} = 1$ and $50\%$ with $\mathcal{C} = 0$. In Table 1 we measure the Average Precision of using several affinity measures as a binary classifier of either $\mathcal{C}$ or $\mathcal{Q}$. In this case, we defined $\mathcal{Q}$ to indicate whether or not the center pixel of the two patches lies on the same labeled object instance. To test $\mathcal{Q}$ independently from $\mathcal{C}$ we create the $\mathcal{Q}$ test set by only sampling from patch pairs for which $\mathcal{C} = 1$ (so the net cannot do well at predicting $\mathcal{Q}$ simply by doing well at predicting $\mathcal{C}$). Our network performs well relative to the baseline affinity metrics, although color histogram similarity does reach a similar performance on predicting $\mathcal{Q}$.

Even though it was only trained to predict $\mathcal{C}$, our method is effective at predicting $\mathcal{Q}$ as well, achieving an average precision (AP) of $0.80$. This validates that spatial proximity, $\mathcal{C}$, is a good surrogate for “same object”, $\mathcal{Q}$. This raises the question, would we do any better if we directly trained on $\mathcal{Q}$? We tested this, training a new network on 50% patches with $\mathcal{Q} = 1$ and 50% with $\mathcal{Q} = 0$. This net achieves higher performance on predicting $\mathcal{Q}$ (AP = $0.85$) and lower performance at predicting $\mathcal{C}$ (AP = $0.92$), than our net trained to predict $\mathcal{C}$. Therefore, although predicting co-occurrence may be a decent proxy for predicting semantic sameness, there is still a gap in performance compared to directly training on $\mathcal{Q}$. Designing better context functions, $\mathcal{C}$, that narrow this gap is an important direction for future research.

Table 2: Probing the learned affinities by transforming $B$ while leaving $A$ unmodified. Each number reports the mean output, $w(A, B)$, from each network after the specified transformation has been applied. Transformations applied to one example patch are shown at the top of each column. Comparison should be made with respect to the unmodified input, given in the left-most column.

| | No transformation | Rotated 90° | Vertical mirror | Horizontal mirror | Color removed | Luminance darkened |
|---|---|---|---|---|---|---|
| | | | | | | |
| **Patches** | 0.819 | 0.818 | 0.818 | 0.819 | 0.382 | 0.523 |
| **Frames** | 0.817 | 0.794 | 0.772 | 0.813 | 0.264 | 0.608 |
| **Photos** | 0.550 | 0.488 | 0.499 | 0.546 | 0.520 | 0.516 |

### 3.3 FRAME AFFINITIES FROM CO-OCCURRENCE IN MOVIES

Our framework can also be applied to learning temporal associations. To test this, we set $A$ and $B$ to be frames, cropped and down sampled to $33 \times 33$ pixels, from a set of 96 movies sampled from the top 100 rated movies on IMDB[2]. In this setting, $\mathcal{C}$ indicates temporal adjacency – specifically, two frames are assigned $\mathcal{C} = 1$ if they are at least 3 seconds from each other and not more than 10 seconds apart. $\mathcal{C} = 0$ otherwise.

Again we train a CNN to model $P(\mathcal{C} = 1|A, B)$ (three convolutional layers, two fully connected layers). To evaluate predicting $\mathcal{C}$, we train on half the movies and test on the remaining half. Our method can learn to predict $\mathcal{C}$ quite effectively, reaching an Average Precision of $0.95$ on the test set.

How do the learned temporal associations relate to semantic visual scenes? To test this, we compared against DVD chapter annotations, setting $\mathcal{Q}$ to be "do these two frames occur in the same DVD chapter?" We sample 10,000 frame pairs, 50% with $\mathcal{Q} = 1$ and 50% with $\mathcal{Q} = 0$, while holding $\mathcal{C}$ constant (so that good performance at predicting $\mathcal{Q}$ cannot be achieved simply by doing well at predicting $\mathcal{C}$). Our network achieves an AP of $0.67$ on this task. Similar to above, we can then see that temporal adjacency, $\mathcal{C}$, is an effective surrogate for learning about semantic sameness, $\mathcal{Q}$.

### 3.4 PHOTO AFFINITIES FROM GEOSPATIAL CO-OCCURRENCE

Just as an object is a collection of associated patches, and a movie scene is a collection of associated frames, a visual *place* can be viewed a collection of associated photographs. Here we set $A$ and $B$ to be geotagged photos, cropped and down sampled to $33 \times 33$ pixels, and $\mathcal{C}$ indicates whether or not $A$ and $B$ are taken within 11 meters of one another (we exclude exact duplicate locations).

Using the same CNN architecture as for the movie frame network, we again learn $P(\mathcal{C} = 1|A, B)$, but for this new setting of the variables. We train on five cities selected from the MIT City Database (Zhou et al. (2014)) and test predicting $\mathcal{C}$ on a held out set of three more cities from that dataset. We also test how well the network predicts place semantics. For this, we define $\mathcal{Q}$ as "do these two photos belong to the same place category?" We test this task on the LabelMe Outdoors dataset (Liu et al. (2009)) for which each photo was assigned to one of eight place categories (e.g., "coast", "highway", "tall building"). Our network shows promising performance on this task, reaching $0.79$ AP on predicting $\mathcal{Q}$. HOG similarity reaches the same performance, which corroborates past findings that HOG is effective at grouping related photos (Dalal & Triggs (2005)).

Notice that while HOG does well on associating photographs, it does not do well at associating movie frames nor image patches. On the other hand, color histogram similarity does well on associating image patches and movie frames, but fails at grouping everyday photographs – while patches on an object, or frames in a movie scene, may tend to all use a consistent color palette, tourist photos of the same location will have high color variance, due to seasonal and lighting variations. Different grouping rules will be effective at different tasks. Our learning based approach has the advantage that it automatically figures out the appropriate grouping cue for each new domain, and thereby achieves good performance on all our tasks.

[2] http://www.imdb.com/

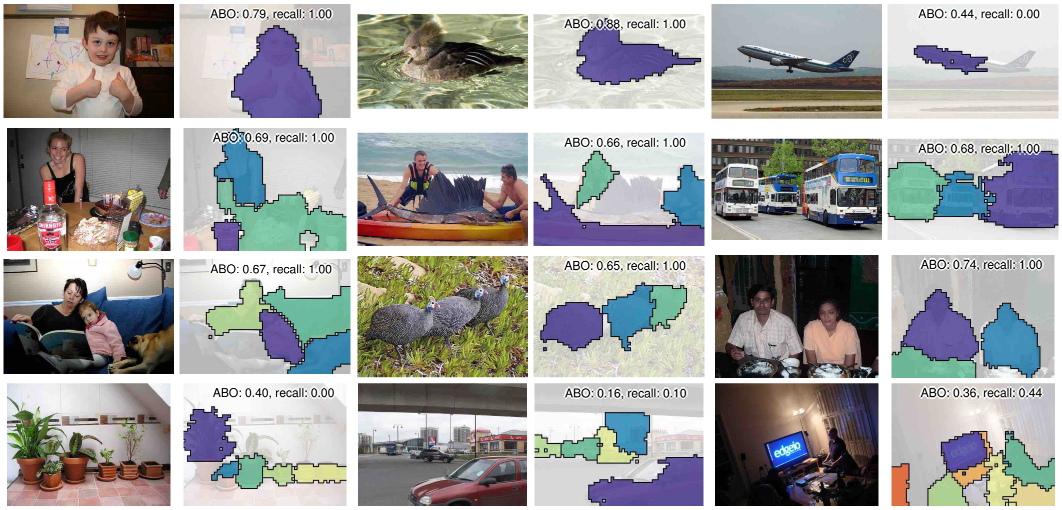

Figure 4: Example object proposals. Out of 100 proposals per image, we show those that best overlap the ground truth object masks. Average best overlap (defined in Krahenbuhl & Koltun (2014)) and recall at a Jaccard index of 0.5 are superimposed over each result.

### 3.5 Which cues did the networks learn to use?

In each domain tested above, the grouping rules may be very different. Here we study them by probing the trained networks with controlled stimuli. Similar to how a psychophysicist might experiment on human perception, we show our networks specially made stimuli. For each test, we feed the networks many pairs $\{A, B\}$, sampled from locations such that $\mathcal{C} = 1$. We leave $A$ unaltered, but modify $B$ in a controlled way. This allows us to test what kinds of transformations of $B$ will change the network's prediction as to wether or not $\mathcal{C} = 1$ (c.f. Lenc & Vedaldi (2014)). We consider the following modifications: rotation by 90°, mirroring vertically and horizontally, removal of color (by replacing each pixel with its mean over color channels), and a darkening transformation in which we multiply each color channel by 0.5.

Results of these tests are given in Table 2. Each number is the mean output of the network for a given test case. The left-most column provides the mean output without any transformation applied. Interestingly, the patch network is almost entirely invariant to 90°rotations and mirror flips – according to the network, sharing a common orientation does not increase the probability of two patches being nearby. On the other hand, the patch network's output depends dramatically on color similarity. The frame network behaves similarly, but shows some sensitivity to geometric transformations. On the other hand, the geo-photo network exhibits the opposite pattern of behavior: it's output is changed more when geometric transformationsare applied than when color transformations are applied. According to the geo-network, two photos may have different color compositions and still be likely to be nearby.

## 4 From predicting co-occurrence to forming visual groups

We apply the following general approach to learning visual groups:

1. Define $A$, $B$, and $\mathcal{C}$ based on the domain.
2. Learn $P(\mathcal{C}|A, B)$ using a CNN.
3. Setup a graph in which $\mathbf{A}_{i=1}^{N}$ and $\mathbf{B}_{i=1}^{N}$ are nodes and edge weights are given by $w(\mathbf{A}_i, \mathbf{B}_i)$ (Eqn. 1). Then partition the graph into visual groups using spectral clustering.

### 4.1 Finding objects

As demonstrated in Section 4, patches that predictably co-occur usually belong to the same object. This suggests that we can localize objects in an image by grouping patches using our co-occurrence

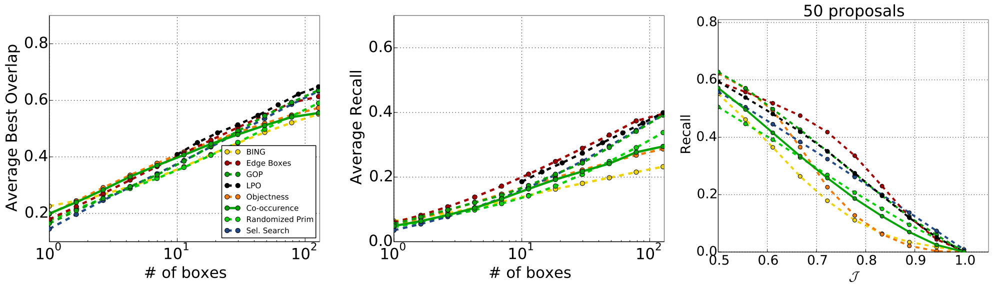

Figure 5: Object proposal results, evaluated on bounding boxes. Our unsupervised method (labeled "Co-occurrence") is competitive with recent supervised algorithms at proposing up to around 100 objects. ABO is the average best overlap metric from (Krahenbuhl & Koltun (2014)), $\mathcal{J}$ is Jaccard index. The papers compared to are: BING (Cheng et al. (2014)), EdgeBoxes Zitnick & Dollár (2014), LPO (Krahenbuhl & Koltun (2015)), Objectness (Alexe et al. (2012)), GOP (Krahenbuhl & Koltun (2014)), Randomized Prim (Manen et al. (2013)), Sel. Search (Uijlings et al. (2013)).

affinity. We focus on the specific problem of "object proposals" (Zitnick & Dollár (2014); Krahnenbuhl & Koltun (2015)), where the goal is to localize all objects in an image.

We use the patch associations $P(\mathcal{C} = 1|A, B)$ defined in Section 4, trained on the Pascal VOC 2012 training set. Follow previous benchmarks, we test on the validation set. Given a test image, we sample all $17 \times 17$ patches at a stride of 8 pixels. We construct a graph in which each patch is a node and nodes are connected by an edge if the spatial distance between the patch centers is at least 17 pixels and no more than 33 pixels. Each patch is multiplied by a circular mask so that no two patches connected by an edge see any overlapping pixels (see Figure 2(right)). Each edge, indexed by $i, j$, is weighted by $\mathbf{W}_{i,j} = w(\mathbf{A}_i, \mathbf{B}_i)^{\alpha}$, resulting in the affinity matrix $\mathbf{W}$, where we use the value $\alpha = 20$ in our experiments.

To globalize the associations, we apply spectral clustering to the matrix $\mathbf{W}$. First we create the Laplacian eigenmap $L$ for $\mathbf{W}$, using the 2nd through 16th eigenvectors with largest eigenvalues. Each eigenvector is scaled by $\lambda^{-\frac{1}{2}}$ where $\lambda$ is the corresponding eigenvalue. We then generate object proposals simply by applying k-means to the Laplacian eigenmap. To generate more than a few proposals, we run k-means multiple times with random restarts and for values of k from 5 to 16. Finally, we prune redundant proposals and sort proposals to achieve diversity throughout the ranking (by encouraging proposals to be made from different values of k before giving proposals from a random restart at the same value of k).

Qualitative results from our method are shown in Figure 4. In each case, we show the proposals that have best overlap with the ground truth object masks for 100 proposals. We quantitatively compare against other recent methods in Figure 5. Even though our method is not trained on labeled images, it reaches performance comparable to recent supervised methods at proposing up to 100 objects per image. Our implementation runs in about 4 seconds per image on a 2015 Macbook Pro.

### 4.2 Segmenting movies

Just as objects are composed of associated patches, scenes in a movie are composed of associated frames. Here we show how our learned frame affinities can be used to break a movie into coherent scenes, a problem that has received some prior attention (Chen et al. (2008); Zhai & Shah (2006)).

To segment a movie, we build a graph in which each frame is a node and all frames within ten seconds of one another are connected by an edge. We then weight the edges using the frame-associations $P(\mathcal{C} = 1|A, B)$ (Section 4), and partition the graph using spectral clustering.

To evaluate, we use DVD chapter annotations as ground truth. Following a standard evaluation procedure in image boundary detection (Arbelaez et al. (2011)), we measure performance on the retrieval task of finding all ground truth boundaries. In Figure 6(right), we compare against the baseline affinity metrics from Section 4. In each case, we apply the same spectral clustering pipeline as for our method, except with edge weights given by the baseline metrics instead of by our net. It is important to properly scale each similarity metric or else spectral clustering will fail. To provide fair comparison, we sweep over a range of scale factors $\alpha$, setting edge weights as $\exp(\frac{w(A,B)}{\alpha^2})$,

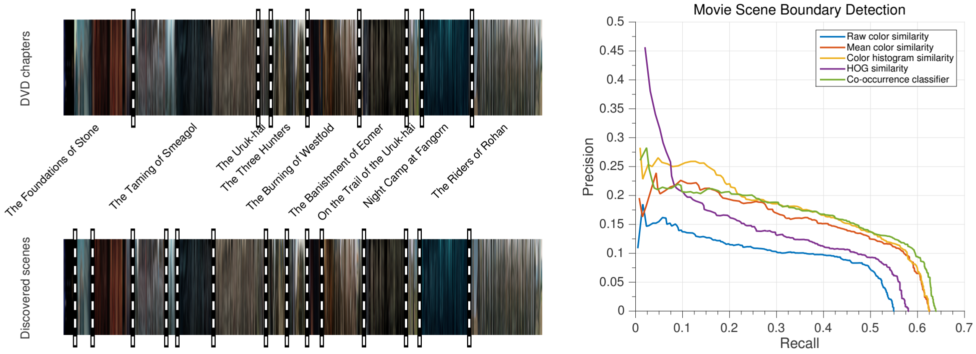

Figure 6: Movie scene segmentation results. On the left, we show a "movie barcode" for *The Two Towers*, in which each frame of the movie is resized into a signal column of the visualization; the top shows the DVD chapters and the bottom our recovered scene segmentation. Notice that some DVD chapters contain multiple different scenes. Our method tends to detect these sub-chapter scenes, resulting in an over-segmentation compared to the chapters. On the right, we quantify our performance on this scene segmentation task; see text for details.

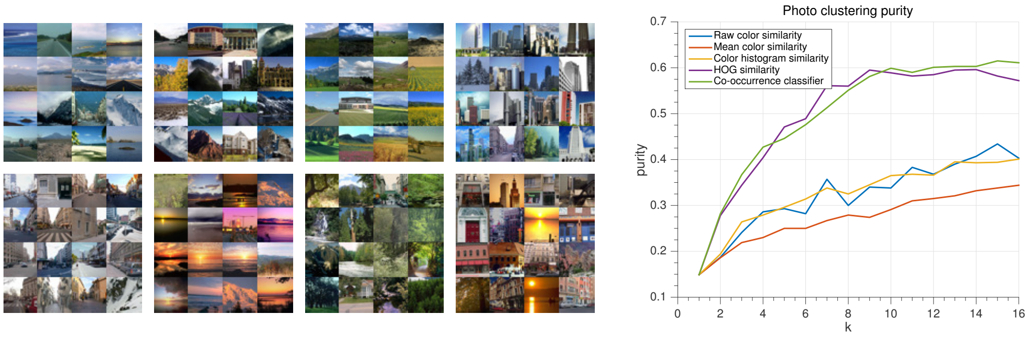

Figure 7: Left: Clustering the LabelMe Outdoor dataset (Liu et al. (2009)) into 8 groups using our learned affinities. Random sample images are shown from each group. Right: Photo cluster purity versus number of clusters $k$. Note that we trained our model on an independent dataset, the MIT City dataset (Zhou et al. (2014)).

where $w$ is the affinity measure. In Figure 6(right) we show results selecting the optimal $\alpha$ for each method.

Our approach finds more boundaries with higher precision than these baselines, except for color histogram similarity, which reaches a similar performance. Figure 6 (left) shows an example segmentation of a section of *The Two Towers*. The movie is displayed as a "movie barcode"[3] in which each frame is squished into a single column and time advances to the right. On top are the DVD chapter annotations, and on the bottom are our inferred boundaries.

### 4.3 Discovering place categories

Taking the geospatial-associations model from Section 4, we cluster photos into coherent types of places. Here we create a fully connected graph between all photos in a given collection, weight the edges with $P(\mathcal{C} = 1|A, B)$ and then apply spectral clustering to partition the collection. We test the purity of the clusters on LabelMe Outdoors dataset (Liu et al. (2009)). Clustering purity versus number of clusters $k$ is given in Figure 7 (right), showing that our method is effective at discovering semantic place categories. As in our movie segmentation experiments, we select the optimal $\alpha$ to scale the affinity of our method as well each baseline. Figure 7 (left) shows random sample images from each cluster after clustering into 8 categories. This clustering has $59\%$ purity.

[3] http://moviebarcode.tumblr.com/

## 5 Conclusion

We have presented a simple and general approach to learning visual groupings, which requires no pre-defined labels. Instead our framework uses co-occurrence in space or time as a supervisory signal. By doing so, we learn different clustering mechanisms for a variety of tasks. Our approach achieves competitive results on object proposal generation, even when compared to methods trained on labeled data. Additionally, we demonstrated that the same method can be used to segment movies into scenes and to uncover semantic place categories. The principles underlying the framework are quite general and may be applicable to data in other domains, when there are natural co-occurrence signals and groupings.

### Acknowledgments

We thank William T. Freeman, Joshua B. Tenenbaum, and Alexei A. Efros for helpful feedback and discussions. Thanks to Andrew Owens for helping collect the movie dataset. This work is supported by NSF award 1161731, Understanding Translucency, and by Shell Research.

## References

Pulkit Agrawal, Joao Carreira, and Jitendra Malik. Learning to see by moving. *arXiv preprint arXiv:1505.01596*, 2015.

Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. Measuring the objectness of image windows. *PAMI*, 2012.

Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *PAMI*, 33(5):898–916, 2011.

Horace Barlow. Cerebral cortex as model builder. *Models of the visual cortex*, pp. 37–46, 1985.

John Canny. A computational approach to edge detection. *PAMI*, (6):679–698, 1986.

Liang-Hua Chen, Yu-Chun Lai, and Hong-Yuan Mark Liao. Movie scene segmentation using background information. *Pattern Recognition*, 41(3):1056–1065, 2008.

Ming-Ming Cheng, Ziming Zhang, Wen-Yan Lin, and Philip Torr. Bing: Binarized normed gradients for objectness estimation at 300fps. In *CVPR*, 2014.

Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pp. 539–546. IEEE, 2005.

Kenneth Ward Church and Patrick Hanks. Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1):22–29, 1990.

Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction. *CoRR*, abs/1505.05192, 2015. URL `http://arxiv.org/abs/1505.05192`.

Piotr Dollár. Piotr's Computer Vision Matlab Toolbox (PMT). `http://vision.ucsd.edu/~pdollar/toolbox/doc/index.html`, 2015.

Piotr Dollár and C Lawrence Zitnick. Structured forests for fast edge detection. In *ICCV*, 2013.

Alon Faktor and Michal Irani. Clustering by composition–unsupervised discovery of image categories. In *ECCV*. 2012.

Alon Faktor and Michal Irani. Co-segmentation by composition. In *ICCV*, 2013.

József Fiser and Richard N Aslin. Unsupervised statistical learning of higher-order spatial structures from visual scenes. *Psychological science*, 12(6):499–504, 2001.

Phillip Isola, Daniel Zoran, Dilip Krishnan, and Edward H. Adelson. Crisp boundary detection using pointwise mutual information. In *ECCV*, 2014.

Dinesh Jayaraman and Kristen Grauman. Learning image representations equivariant to ego-motion. *arXiv preprint arXiv:1505.02206*, 2015.

Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the ACM International Conference on Multimedia*, pp. 675–678. ACM, 2014.

Christoph Kayser, Wolfgang Einhäuser, Olaf Dümmer, Peter König, and Konrad Körding. Extracting slow subspaces from natural videos leads to complex cells. In *Artificial Neural Networks?ICANN 2001*, pp. 1075–1080. Springer, 2001.

Gary King and Langche Zeng. Logistic regression in rare events data. *Political analysis*, 9(2):137–163, 2001.

Philipp Krahenbuhl and Vladlen Koltun. Geodesic object proposals. In *ECCV*. 2014.

Phillip Krahnenbuhl and Vladlen Koltun. Learning to propose objects. In *CVPR*, 2015.

Karel Lenc and Andrea Vedaldi. Understanding image representations by measuring their equivariance and equivalence. *arXiv preprint arXiv:1411.5908*, 2014.

Omer Levy, Yoav Goldberg, and Israel Ramat-Gan. Linguistic regularities in sparse and explicit word representations. *CoNLL-2014*, pp. 171, 2014.

Ce Liu, Jenny Yuen, and Antonio Torralba. Nonparametric scene parsing: Label transfer via dense scene alignment. In *CVPR*, 2009.

David Lowe. *Perceptual organization and visual recognition*, volume 5. Springer Science & Business Media, 2012.

Tomasz Malisiewicz and Alexei A Efros. Improving spatial support for objects via multiple segmentations. In *BMVC*, 2007.

Santiago Manen, Matthieu Guillaumin, and Luc Van Gool. Prime object proposals with randomized prim's algorithm. In *ICCV*, 2013.

Hossein Mobahi, Ronan Collobert, and Jason Weston. Deep learning from temporal coherence in video. In Léon Bottou and Michael Littman (eds.), *ICML*, pp. 737–744, Montreal, June 2009. Omnipress.

Irvin Rock. The logic of perception. 1983.

Jenny R Saffran, Richard N Aslin, and Elissa L Newport. Statistical learning by 8-month-old infants. *Science*, 274(5294):1926–1928, 1996.

Anna C Schapiro, Timothy T Rogers, Natalia I Cordova, Nicholas B Turk-Browne, and Matthew M Botvinick. Neural representations of events arise from temporal community structure. *Nature Neuroscience*, 16(4): 486–492, 2013.

Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *PAMI*, 22(8):888–905, 2000.

Josef Sivic, Bryan C Russell, Alexei Efros, Andrew Zisserman, William T Freeman, et al. Discovering objects and their location in images. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pp. 370–377. IEEE, 2005.

Nitish Srivastava, Elman Mansimov, and Ruslan Salakhutdinov. Unsupervised learning of video representations using lstms. *arXiv preprint arXiv:1502.04681*, 2015.

Jay M Tenenbaum and AP Witkin. On the role of structure in vision. *Human and machine vision*, pp. 481–543, 1983.

Jasper RR Uijlings, Koen EA van de Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *IJCV*, 104(2):154–171, 2013.

L.J.P. van der Maaten and G.E. Hinton. Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 2009.

Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. *arXiv preprint arXiv:1505.00687*, 2015.

Andrew P Wilkin and Jay M Tenenbaum. What is perceptual organization for? *From Pixels to Predicates*, 1985.

Laurenz Wiskott and Terrence J Sejnowski. Slow feature analysis: Unsupervised learning of invariances. *Neural computation*, 14(4):715–770, 2002.

Yun Zhai and Mubarak Shah. Video scene segmentation using markov chain monte carlo. *Multimedia, IEEE Transactions on*, 8(4):686–697, 2006.

B. Zhou, Liu. Liu, A. Oliva, and A. Torralba. Recognizing City Identity via Attribute Analysis of Geo-tagged Images. *ECCV*, 2014.

C Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*. 2014.