Discovering Multipart Appearance Models from Captioned Images

Michael Jamieson, Yulia Eskin, Afsaneh Fazly,
Suzanne Stevenson, and Sven Dickinson

University of Toronto
{jamieson,yulia,afsaneh,suzanne,sven}@cs.toronto.edu

Abstract. Even a relatively unstructured captioned image set depicting a variety of objects in cluttered scenes contains strong correlations between caption words and repeated visual structures. We exploit these correlations to discover named objects and learn hierarchical models of their appearance. Revising and extending a previous technique for finding small, distinctive configurations of local features, our method assembles these co-occurring parts into graphs with greater spatial extent and flexibility. The resulting multipart appearance models remain scale, translation and rotation invariant, but are more reliable detectors and provide better localization. We demonstrate improved annotation precision and recall on datasets to which the non-hierarchical technique was previously applied and show extended spatial coverage of detected objects.

1 Introduction

Computer vision tasks from image retrieval to object class recognition are based on discovering similarities between images. For all but the simplest tasks, meaningful similarity does not exist at the level of basic pixels, and so system designers create image representations that abstract away irrelevant information. One popular strategy for creating more useful representations is to learn a hierarchy of parts in which parts at one level represent meaningful configurations of subparts at the next level down. Thus salient patterns of pixels are represented by local features, and recurring configurations of features can, in turn, be grouped into higher-level parts, and so on, until ideally the parts represent the objects that compose the scene. The hierarchical representations are inspired by and intended to reflect the compositional appearance of natural objects and artifacts. For instance, each level of the Leaning Tower of Pisa appears as a ring of arches while the tower as a whole is composed of a (nearly) vertical stack of levels.

With this strategy in mind, we build upon the approach of [1] to produce a system with more accurate image annotation and improved object localization. Given images of cluttered scenes, each associated with potentially noisy captions, our previous method [1] can discover configurations of local features that strongly correspond to particular caption words. Our system improves the overall distribution of these local configurations to optimize the overall correspondence with the word. While individual learned parts are often sufficient to indicate the

presence of particular exemplar objects, they have limited spatial extent and it is difficult to know whether a collection of part detections in a particular image are from multiple objects or multiple parts of a single object. Our system learns meaningful configurations of parts wherever possible, allowing us to reduce false annotations due to weak part detections and provide a better indication of the extent of detected objects. Figure 1 illustrates how low-level features are assembled in stages to form a multipart model (MPM) for the Leaning Tower. MPMs are more robust to occlusion, articulation and changes in perspective than a flat configuration of features. While the instantiated system uses exemplar-specific SIFT features, the framework can support more categorical features.

(a) pixels (b) local features (c) parts (d) multipart models

Fig. 1. Object model detection and learning progresses in stages. Gradient patterns in the original image (a) are grouped into local features (b). Configurations of local features with strong word correspondence are captured as part models (c). Finally, we represent meaningful configurations of part models as multipart models (d).

2 Related Work

A number of researchers have studied the problem of automatic image annotation in recent years [2–6, 1]. Given cluttered images of multiple objects paired with noisy captions, these systems can learn meaningful correspondences between caption words and appearance models.

In many automatic annotation systems, the main component of the appearance model is a distribution over colors and textures. This kind of representation is a good fit for relatively structureless materials such as grass, sand or water and is relatively robust to grouping or segmentation errors. However, objects such as buildings and bicycles often lack a distinctive color or texture, and require representations that can capture a particular configuration of individually ambiguous parts. Most of these automatic annotation systems do not focus on learning such feature configurations. Often, appearance is modeled as a mixture of features (*e.g.*, [5, 3, 6]) in which common part configurations are reflected in

co-occurrence statistics but without spatial information. Similarly, the Markov random field model proposed by Carbonetto *et al.* [4] can represent adjacency relationships but not spatial configurations.

In contrast, the broader object recognition literature contains many methods for grouping individual features into meaningful configurations and even arranging features into hierarchies of parts. For instance, Fergus *et al.* [7] and Crandall and Huttenlocher [8] look for features and relationships that recur across a collection of object images in order to learn object appearance models consisting of a distinctive subset of features and their relative positions. A natural strategy to improve the flexibility and robustness of such models is to organize the object representation as a parts hierarchy (*e.g.*, [9–14]). The part hierarchy can be formed by composing low-level features into higher and higher level parts (*e.g.* Kokkinos and Yuille [9], Zhu *et al.* [10]) or by decomposing larger-scale shared structures into recurring parts (*e.g.*, Epshtein and Ullman [13]). The composition and learning method of parts at different levels of the hierarchy may be highly similar (*e.g.*, Bouchard and Triggs [11], Fidler *et al.* [12]) or heterogeneous (*e.g.*, Ommer and Buhmann [14]). Some of these methods can learn an appearance model from training images with cluttered backgrounds, sometimes without relying on bounding boxes. However, unlike most automatic annotation work, they are not designed for images containing multiple objects and multiple annotation words.

In [1], we describe an automatic annotation system that can capture explicit spatial configurations of features while retaining the ability to learn from noisy, unstructured collections of captioned images. Guided by correspondence with caption words, the system iteratively constructs appearance graphs in which vertices represent local features and edges represent spatial relationships between them. However, the learned appearance models usually have limited spatial extent, with each model typically describing only a distinctive portion of an object. There is no way to determine whether a set of detections in a given image represents multiple objects or different parts of the same object. Our system addresses these limitations by using the appearance models as parts in larger hierarchical object models.

3 Images, Parts and Multipart Models

Our system learns multipart appearance models (MPMs) by detecting recurring configurations of lower-level 'parts' that together appear to have a strong correspondence with a particular caption word. Though our overall approach could be appropriate for a variety of part features, in this paper our parts are local configurations of interest points as in [1].

In [1], an image is represented as a set of local interest points, $I = \{p_m | m = 1 \ldots |I|\}$. These points are detected using Lowe's SIFT method [15], which defines each point's spatial coordinates, $\mathbf{x}_m$, scale λ_m and orientation θ_m. A PCA-SIFT [16] feature vector ($\mathbf{f}_m$) describes the portion of the image around each point. In addition, a vector of transformation-invariant spatial relationships r_{mn} is defined

between each pair of points, p_m and p_n, including the relative distance between the two points (Δx_{mn}), the relative scale difference between them ($\Delta \lambda_{mn}$) and the relative bearings in each direction ($\Delta \phi_{mn}$, $\Delta \phi_{nm}$).

A part appearance model describes the distinctive appearance of an object part as a graph $G = (V, E)$. Each vertex $v_i \in V$ is composed of a continuous feature vector $\mathbf{f}_i$ and each edge $e_{ij} \in E$ encodes the expected spatial relationship between two vertices, v_i and v_j. Model detections have a confidence score, $\text{Conf}_{detect}(O, G) \in [0, 1]$, based on the relative likelihood of an observed set of points O and the associated spatial relations being generated by the part model G versus unstructured background.

Multipart models are very similar in structure to the local appearance models described in [1]. As shown in Figure 2, a multipart model is a graph $H = (U, D)$ where vertices $u_j, u_k \in U$ are part appearance model detections and each edge $d_{jk} \in D$ encodes the spatial relationships between them, using the same relationships as in the part model: $d_{jk} = (\Delta x_{jk}, \Delta \lambda_{jk}, \Delta \phi_{jk}, \Delta \phi_{kj})$.

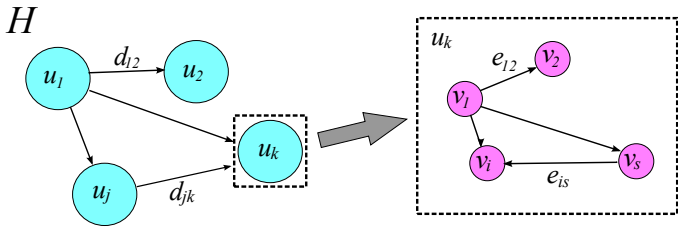


Fig. 2. A multipart model H is a graph with parts $u_j \in U$ and spatial relationships $d_{jk} \in D$, where each part is a graph G with local features $v_i \in V$ and spatial relationships $e_{is} \in E$.

4 Discovering Parts

Multipart models are composed of the same type of individual appearance models that were discovered in [1]. However, models trained to maximize standalone detection performance are generally not ideal as parts of a larger appearance model. Singleton appearance models need to act as high-precision detectors while MPM parts can be individually more ambiguous and rely on the MPM layer to weed out false-positive detections by imposing co-occurrence and spatial constraints. Therefore, when learning MPM parts, we can accept some loss of precision in exchange for better recall and better spatial coverage of the object of interest. We implement this shift toward weaker parts with better coverage by replacing the part initialization process in [1] with our own improved process and by limiting the size of learned part models to eight vertices.

4.1 Model Initialization through Image Pair Sampling

We replace the clustering-based model initialization method of [1] with an approach that makes earlier use of language information. The system in [1] summarizes the visual information within each neighborhood of an image set as a quantized bag-of-features descriptor called a *neighborhood pattern* and then uses clustering to group similar neighborhood patterns. Next, the system checks for promising correspondences between the occurrence patterns of each neighborhood cluster and each word. Finally, clusters with the best correspondences for each word are used to extract initial two-vertex appearance models.

This clustering approach has several drawbacks. The neighborhood patterns are noisy due to features quantization and detector errors. Therefore a low similarity threshold is needed to reliably group similar appearances. However, this allows unrelated neighborhoods to join the cluster. Especially on large image sets, this can add substantial noise to the cluster occurrence pattern, obscuring its true word correspondences. Therefore recurring visual structure corresponding to rarer object views is often overlooked.

Our initialization method avoids feature quantization and uses word labels early-on in the process. Instead of using a neighborhood pattern, we compare visual features directly. Rather than cluster visual structure across the entire training set, we look for instances of shared appearance between pairs of images with the same word label. For a given word w, the system randomly samples pairs of images I_A and I_B from those with captions containing w and identifies neighborhoods in the two images that share visual structure.

We identify shared neighborhoods in three steps. First, the system looks for uniquely-matching features that are potential anchors for shared neighborhoods. Following [15], we identify matching features that are significantly closer to each other than to either feature's second-best match, *i.e.*, features $\mathbf{f}_m \in I_A$ and $\mathbf{f}_n \in I_B$ that satisfy equations 1 and 2:

$$|\mathbf{f}_m - \mathbf{f}_n|^2 \leq \psi_u |\mathbf{f}_m - \mathbf{f}_k|^2, \forall \mathbf{f}_k \in \{I_B - \mathbf{f}_n\} \quad (1)$$

$$|\mathbf{f}_m - \mathbf{f}_n|^2 \leq \psi_u |\mathbf{f}_l - \mathbf{f}_n|^2, \forall \mathbf{f}_l \in \{I_A - \mathbf{f}_m\} \quad (2)$$

where $\psi_u < 1$ controls degree of uniqueness of anchor matches. For each pair of uniquely-matching features, the system checks for supporting matches in the surrounding neighborhood. These supporting matches aren't required to be unique, so the corresponding uniqueness quantifier $\psi_s > 1$. For each supporting match pair $f_i \in I_A$ and $f_j \in I_B$, the system then verifies that the spatial relationships between the unique feature and the supporting feature in the two images (r_{mi} and r_{nj}) are consistent. A shared neighborhood has a pair of unique matches and at least two spatially consistent supporting matches.

Given this evidence of shared visual structure, we construct a set of two-vertex part models, each with one vertex based on the unique match and the other on a strong supporting match. These two-vertex models represent shared visual structure between two images labeled with word w. To check whether

the models correspond with w, the system detects each model G across the training image set and compares its occurrence pattern with that of w. Below, we explain how we sample image pairs and filter the resulting initial part models to maximize overall coverage of the object.

4.2 Part Coverage Objective

In [1], the system develops the n neighborhood clusters with the best correspondence with w into full appearance models. This approach concentrates parts on the most common views of an object, neglecting less common views and appearances associated with w. Our method instead selects initial part models so that, *as a group*, they have good coverage of w throughout the training set.

Ideally, a set of part models $\mathcal{G}$ would have multiple, non-overlapping detections in every training set image annotated with word w and no detections elsewhere. We represent the distribution of model detections throughout the k training images with the vector $\mathbf{Q}_w = \{Q_{wi} | i = 1, \ldots, k\}$. If n_i is the number of independent model detections in image i, $Q_{wi} = 1 - \nu^{n_i}, \nu < 1$. With multiple detections, Q_{wi} approaches 1, but each successive detection has a smaller effect.

We evaluate how well $\mathcal{G}$ approximates the ideal by evaluating the correspondence between $\mathbf{Q}_w$ and a vector $\mathbf{r}_w$ indicating images with w in the caption using an F-Measure, $F(\mathbf{r}_w, \mathbf{Q}_w)$. The part initialization process greedily grows and modifies a collection of non-overlapping two-vertex part models $\mathcal{G}$ to maximize $F(\mathbf{r}_w, \mathbf{Q}_w)$. At each iteration, it draws a pair of images from the sample distribution $\mathbf{s}_w$ and uses them to generate potential part models. $\mathbf{Q}_w$ influences the sample distribution: $\mathbf{s}_w \sim 1 - \mathbf{Q}_w$. This focuses the search for new models in images that do not already contain several model detections. The algorithm calculates, for each potential model, the effects on the correspondence score F of adding the model to the current part set, of replacing each of the models in the current set and of rejecting the model. The algorithm implements the option which leads to the greatest improvement in correspondence. The process stops once no new models have been accepted in the last N_{pairs} image-pair samples.

Besides optimizing the explicit objective function, the initialization system also avoids redundant models with many overlapping detections. Two models are considered to be redundant when their detections overlap nearly as often as they occur separately. When a new two-vertex model is considered, if selected it must replace any models that it makes redundant.

5 Building Multipart Models

After learning distinctive part models, but before assembling them into multipart models, we perform several stages of processing. Algorithm 1 summarizes both the preprocessing steps and the MPM initialization and assembly process, with reference to the subsections below that explain the steps of the algorithm.

Algorithm 1 Uses parts associated with word w to assemble multipart models.

ConstructMPMs(w)

1. For each part G associated with w, find the set $\mathcal{O}_G$ of observations of G in training images.
2. Identify and remove redundant parts (section 5.1).
3. For each G, set the spatial coordinates of each observation $O_G \in \mathcal{O}_G$ (section 5.2):
   - Choose representative vertex v_c to act as center of G.
   - For each $v_i \in \mathbf{v}_G$, find average relationship, $\bar{\mathbf{r}}_{ic}$, between co-occurences of $(v_i, v_c) \in \mathcal{O}_G$.
   - For each $O_G \in \mathcal{O}_G$, and each observed vertex $\mathbf{p_i} \in O_G$ calculate expected position of $\mathbf{x}_c$ based on $(\bar{\mathbf{r}}_{ic}, \mathbf{x}_i)$. Part spatial coordinate $\mathbf{x}_G$ is the average expected center $\bar{\mathbf{x}}_c$.
4. Sort parts by $\text{Conf}_{corr}(G, w)$.
5. For each G:
   - Skip expansion if most $O_G \in \mathcal{O}_G$ are already incorported into existing MPMs (section 5.3).
   - Iteratively expand G into an MPM H using same method as part models (section 5.4):
     - Expand MPM H to H^* by adding new part or spatial relationship.
     - Detect H^* across the training image set (section 5.5).
     - If new MPM–word correspondence, $\text{Conf}_{corr}(H^*, w) > \text{Conf}_{corr}(H, w)$, $H \Leftarrow H^*$.
   - If at least N_{MPM} multipart models have been created, return.

5.1 Detecting Duplicate Parts

Our initialization method avoids excessive overlap of initial part models. However, during model refinement, two distinct part models can converge to cover the same portion of an object's appearance. Near-duplicate parts must be pruned or they could complicate the search for multipart models since they could be interpreted as a pair of strongly co-occurring, independent parts.

Rather than detect near-duplicates by searching for partial isomorphisms between part models, we look for groups of parts that tend to be detected in the same images at overlapping locations. If a vertex v_{Ai} in model G_A maps to the same image point as vertex v_{Bj} in model G_B in more than half of detections, then we draw an equivalence between v_{Ai} and v_{Bj}. If more than half of the vertices in either part are equivalent, we remove the part with the weakest word–model correspondence confidence $\text{Conf}_{corr}(G, w)$.

5.2 Locating Part Detections

The parts described in [1] encode spatial relationships among local interest points; we construct multipart models by discovering spatial relationships between such detected parts. However, while a local interest point detector provides that point's scale, orientation and location, the part detector does not. We therefore set the spatial coordinates for each part detection based on the underlying image points in a way that is robust to occlusion and errors in feature detection.

For each part we select a central vertex and for each detection we estimate the center's spatial coordinates. The center vertex need not be observed in every detection, since each observed vertex contributes to a weighted estimate of the center's coordinates. Figure 3 illustrates this approach. We use the estimated location and orientation of the center and multiply the estimated scale of the center vertex by a part-specific factor so that the detected part scale reflects the normal spread of the part's vertices.

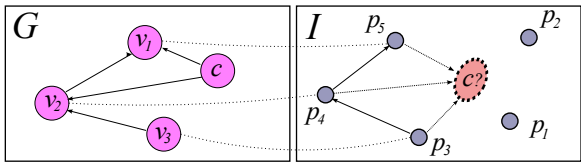


Fig. 3. The spatial coordinates of a part detection are tied to a central vertex c. We estimate c's coordinates based on observed vertices, even if c itself is not observed.

5.3 Choosing Initial Multipart Models

Our system uses the most promising individual part models as seeds for constructing multipart models. Parts that have good correspondence with a word are likely to co-occur with other parts in stable patterns from which large MPMs with good spatial coverage can be constructed. However, if only the strongest part models are expanded, the resulting MPMs may be too clustered around only the most popular views of the object. This would neglect views with weaker individual parts where MPMs can make the biggest difference in precision.

Therefore initial model selection proceeds as follows. Part models are evaluated in the order of their correspondence with a word w. A model is expanded if at least half of its 'good' detections (in images labeled with w) have not been incorporated into any of the already-expanded MPMs. Selective expansion continues until the list of part models is exhausted or N_{MPM} distinct multipart models have been trained for a given word.

5.4 Refinement and Expansion of Multipart Models

In order to expand the multipart models, we take an approach very similar to [1], in that we use the correspondence strength $\text{Conf}_{corr}(H, w)$ between a multipart model H and word w to guide the expansion of these two-vertex graphs into larger multipart models. Introduced in [1], the correspondence score reflects the amount of evidence, available in a set of training images, that a word and a part model are generated from a common underlying source object, as opposed to appearing independently.

Each iteration of the expansion algorithm begins by detecting all instances of the current multipart model in the training set (section 5.5) and identifying additional parts that tend to co-occur with a particular spatial relationship relative to the multipart model. We propose an expansion of the MPM H either by adding a new part model and spatial relationship from among these candidates or by adding a new edge between existing vertices. The proposal is accepted if it improves $\text{Conf}_{corr}(H, w)$ (starting a new iteration), and rejected otherwise. The expansion process continues until potential additions to H have been exhausted.

5.5 Detecting Multipart Models

As in part model detection, multipart detection must be robust to changes in viewpoint, occlusion and lighting that can cause individual part detections to be somewhat out of place or missing entirely. We use a simple generative model illustrated in Figure 4 to explain the pattern of part detections both in images that contain a particular multipart model and those that do not.

Each image i has an independent probability $P(h_i = 1)$ of containing the multipart model H. Given h_i, the presence of each model part is determined independently ($P(u_{ij} = 1|h_i)$). The foreground probability of a model part being present is relatively high ($P(u_{ij} = 1|h_i = 1) = 0.95$), while the background probability, $P(u_{ij} = 1|h_i = 0)$, is equal to its normalized frequency across the training image set. If a part is present, it tends to have a higher observed detection confidence, o_{ij} ($p(o_{ij}|u_{ij} = 1) = 2o_{ij}$, $p(o_{ij}|u_{ij} = 0) = 2(1 - o_{ij})$). If the multipart model is present ($h_i = 1$) and contains an edge r_{jk}, and the parts u_{ij} and u_{ik} are present, then the observed spatial relationship s_{ijk} between the two parts has a relatively narrow distribution centered at the edge parameters. Otherwise, all spatial relationships follow a broad background distribution.

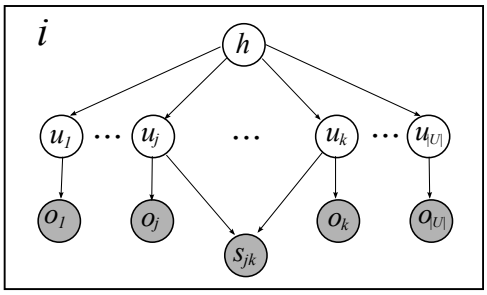


Fig. 4. A graphical model of the generative process with multipart model indicator h, part indicators $\mathbf{u}$, part detection confidences $\mathbf{o}$ and observed spatial relations $\mathbf{s}$.

In any given image, there may be many possible assignments between multipart model vertices and observed part detections. We choose assignments in a greedy fashion in order to maximize $P(h_i = 1|\mathbf{o}_i, \mathbf{s}_i)$. First we choose the best-fit assignment of two linked vertices, then one by one we choose the vertex assignment that makes the largest improvement in $P(h_i = 1|\mathbf{o}_i, \mathbf{s}_i)$ and is consistent with existing assignments.

The prior probability $P(h_i = 1)$ depends on the complexity of the MPM, with more complex multipart models having a lower prior probability. Specifically:

$$P(h_i = 1) = \alpha^{|U|} \cdot \beta^{|D|}. \tag{3}$$

where $\alpha, \beta < 1$ and $|U|$ and $|D|$ are, respectively, the number of vertices and edges in H. The constants α and β were selected based on detection experiments on

random synthetic MPMs with a wide range of sizes in order to prevent large, complex models from being detected when only a tiny fraction of their vertices are present.

6 Results

Once we have discovered a set of individual part models and learned multipart models from configurations of the parts, we can use these learned structures to annotate new images. We begin by detecting all part models in the image (even those that are relatively weakly detected or have relatively low individual correspondence confidence). Based on these part observations, we then evaluate detection confidence for all learned MPMs. Following [1], our annotation confidence for both parts and multipart models is the product of detection confidence, $\text{Conf}_{detect}(i, H)$, and correspondence confidence $\text{Conf}_{corr}(H, w)$. Overall annotation confidence is the maximum annotation confidence over word w's detected models in image i.

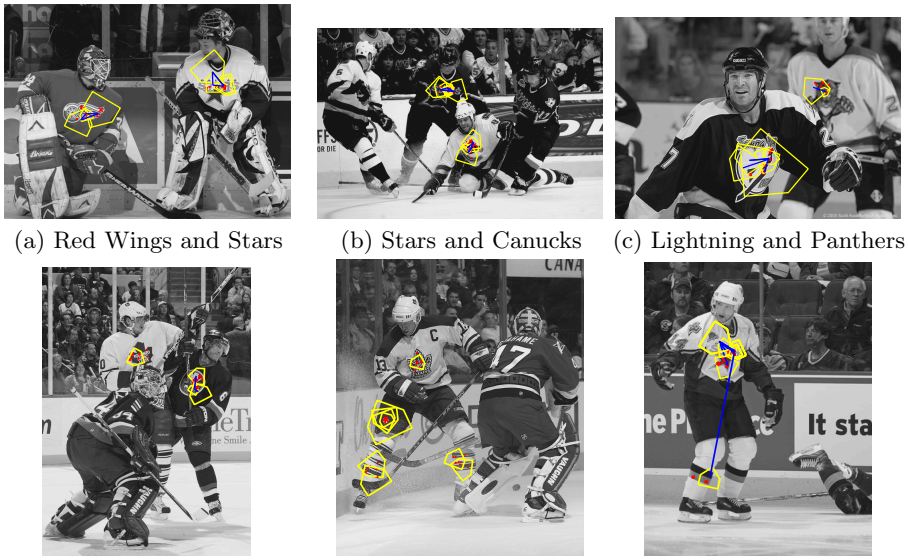
For ease of comparison, we ran our system on three image sets described in [1]. In all three cases, the changes to part initialization combined with the addition of MPM models improve the precision and recall of annotation on new images compared to the system in [1]. The degree of improvement seems to depend on the scale and degree of articulation of named objects.

In experimentation on the small TOYS image set, we find that the particular values of our system parameters do not have a significant effect on our results. The same parameter values chosen based on the TOYS set results are carried over to the two larger and more significant sets without modification. We set uniqueness factors $\psi_u = 0.9$ and $\psi_s = 1.2$. $N_{pairs} = 50$ allows a large number of failed pair samples before ending initial model search. $\nu = 0.75$ allows Q_{wi} to build gradually. We set the maximum number of MPMs per word, $N_{MPM} = 25$, more than the number of distinct views available for individual objects in these image collections. Finally, we choose MPM detection parameters $\alpha = 0.25$ and $\beta = 0.33$ based on experiments on synthetic data.

The first set, TOYS, is a small collection of 228 images of arrangements of children's toys captured and annotated by the authors of [1]. For the sake of completeness, we report our results on this set while focusing on the larger and more natural HOCKEY and LANDMARK sets. Without MPMs, our new model initialization method modestly improves recall on the TOYS set while slightly lowering overall precision. Including MPMs corrects precision, resulting in a net improvement in recall of about 3% at 95% precision.

6.1 Experiments on the HOCKEY Data Set

The HOCKEY set includes 2526 images of National Hockey League (NHL) players and games, with associated captions, downloaded from a variety of sports websites. It contains examples of all 30 NHL teams and is divided into 2026 training and 500 test image–caption pairs. About two-thirds of the captions are



(a) Red Wings and Stars (b) Stars and Canucks (c) Lightning and Panthers

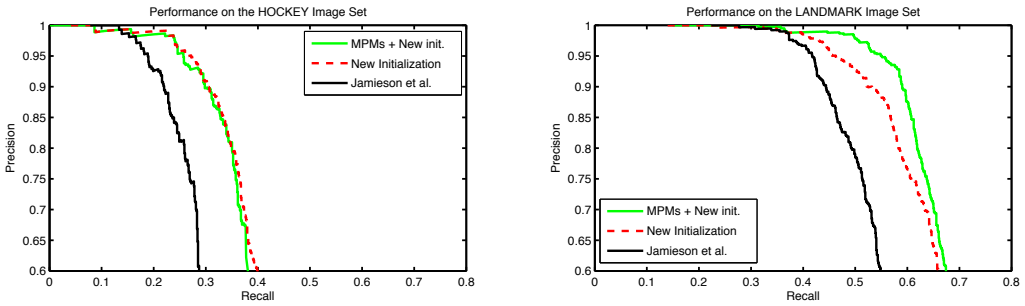
(d) Maple Leafs and Islanders (e) Maple Leafs (f) Stars

Fig. 5. Sample detections of objects in the HOCKEY test set. Part detections are drawn in yellow, supporting interest points in red and spatial relationships in blue.

full sentence descriptions, whereas the remainder simply name the two teams involved in the game.

Figure 5 shows sample multipart model detections on test-set images and the associated team names. Compared to MPMs in the TOY and LANDMARK sets, most MPMs in the HOCKEY set are relatively simple. They typically consist of 2 to 4 parts clustered around the team's chest logo. Since the chest logos are already reasonably well covered by individual part models, there is little reward for developing extensive MPMs. In principle, MPMs could tie together parts that describe other sections of the uniform (socks, pants, shoulder insignia) like those shown in Figure 5(e), but this type of MPM (seen in Figure 5(f)) is quite rare. There may be too much articulation and too few instances of co-occurrence of these parts in the training set to support such MPMs.

Figure 6(a) indicates that our new approach for initializing part models leads to about a 12% improvement in recall. Considering the barriers to achieving high recall on the HOCKEY set (discussed in [1]), this represents a substantial gain. Our initialization system is better able to identify regions of distinctive appearance than the approach in [1]. For instance, one of the best-recognized NHL teams using our method was completely undetected in [1]. On the other hand, the addition of MPMs does not improve annotation performance at all. This is probably due to the relatively small size of distinctive regions in the HOCKEY images combined with a degree of articulation and occlusion that make larger models unreliable.



(a) HOCKEY set (b) LANDMARK set

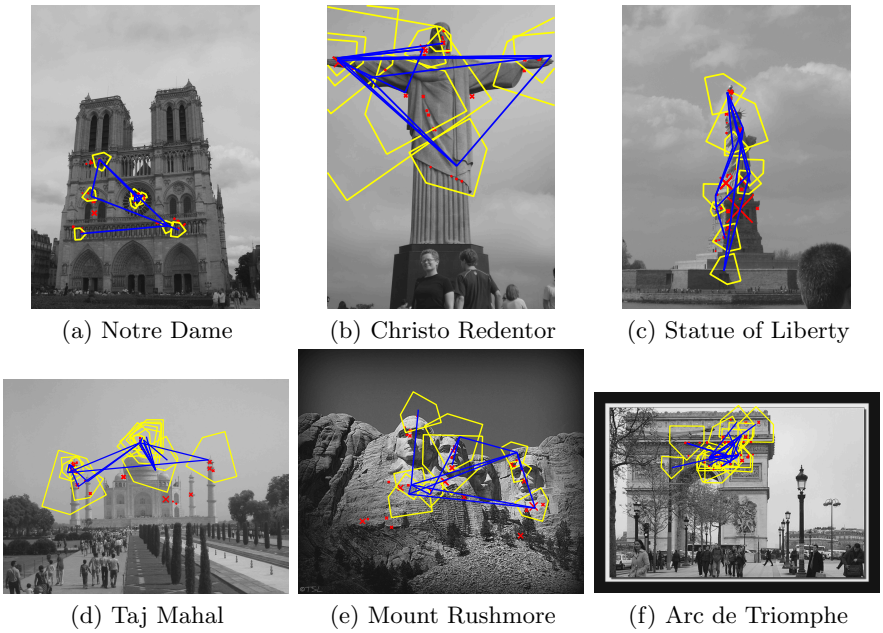
Fig. 6. A comparison of precision–recall curves over the HOCKEY (a) and LANDMARK test sets, for three systems: MPMs with our new initialization, our new initialization alone and the system described in [1]. Our initialization system substantially improves overall recall in both image sets. MPMs have little effect in the HOCKEY set, where the distinctive portions of a player's appearance are of limited size and do not tend to co-occur in repeating patterns. In contrast, MPMs significantly improve precision for the LANDMARK set, perhaps because distinctive portions of landmarks more often co-occur with stable spatial relationships.

6.2 Experiments on the LANDMARK Data Set

The LANDMARK data set includes images of 27 famous buildings and locations with some associated tags downloaded from the Flickr website, and randomly divided into 2172 training and 1086 test image–caption pairs. Like the NHL logos, each landmark appears in a variety of perspectives and scales. Compared to the hockey logos, the landmarks usually cover more of the image and have more textured regions in a more stable configuration. On the other hand, the appearance of the landmarks can vary greatly with viewpoint and lighting, and many of the landmarks feature interior as well as exterior views.

Figure 7 provides some sample detections of multipart models in the LANDMARK test set. The MPMs can integrate widely-separated part detections, thereby improving detection confidence and localization. However, many of the models still display a high degree of part overlap, especially on objects such as the Arc de Triomphe with a dense underlying array of distinctive features. In addition, MPM coverage of the object, while better than individual parts, is not as extensive as it could be. For instance, the system detects many more parts on the western face of Notre Dame than are incorporated into the displayed MPM. In the future, we may wish to modify the MPM training routine to explicitly reward spatial coverage improvements. Finally, MPMs often seem to have one or two key parts with a large number of long-range edges. This edge structure may unnecessarily hamper robustness to occlusion.

Regardless of their limitations, Figure 6(b) indicates that MPMs can significantly improve annotation precision. The new initialization system improves overall recall by about 10%, and the addition of MPMs lifts the precision of the



(a) Notre Dame (b) Christo Redentor (c) Statue of Liberty

(d) Taj Mahal (e) Mount Rushmore (f) Arc de Triomphe

Fig. 7. Sample detections of objects in the LANDMARKS test set.

curve towards the 100% boundary. The structures on which our system achieved the poorest results were St. Peter's Basilica, Chichen Itza and the Sydney Opera House. The first two of these suffer from a multiplicity of viewpoints, with training and test sets dominated by a variety of interior viewpoints and zoomed images of different parts of the structure. The Sydney Opera House's expressionist design has relatively little texture and is therefore harder to recognize using local appearance features.

7 Conclusions

Our initialization method and multipart models are designed to work together to improve annotation accuracy and object localization over the approach in [1]. Our initialization mechanism boosts recall and part coverage by detecting potential parts that would have been overlooked by the system in [1], providing for a better distribution of parts over the image set and including more individually ambiguous parts. The MPM layer boosts precision and localization by integrating parts that may be individually ambiguous into models that can cover an entire view of an object.

Together, our new methods significantly improve annotation accuracy over previous results on the experimental data sets, with the amount of improvement strongly dependent on the image set. Our improvements to part initialization and training have significantly increased recall, though sometimes at the expense of

precision. For objects with recurring patterns of distinctive parts, the MPM layer can filter out bad detections, resulting in a substantially improved precision–recall curve.

Our initialization mechanism and the development of multipart models also improves object localization. Parts have less spatial overlap than in [1], they cover portions of the object that are less individually distinctive and they are better-distributed across object views. MPMs tie together recurring patterns of parts, allowing us to distinguish between the presence of multiple parts and multiple objects. Future work could further improve localization by ensuring that MPMs use all available parts to maximize spatial coverage and are themselves well-distributed across object views.

References

1. Jamieson, M., Fazly, A., Dickinson, S., Stevenson, S., Wachsmuth, S.: Using language to learn structured appearance models for image annotation. IEEE PAMI **32** (2010) 148–164
2. Barnard, K., Duygulu, P., Forsyth, D., de Freitas, N., Blei, D., Jordan, M.: Matching words and pictures. Journal of Machine Learning Research **3** (2003) 1107–1135
3. Carneiro, G., Chan, A., Moreno, P., Vasconcelos, N.: Supervised learning of semantic classes for image annotation and retrieval. IEEE PAMI **29** (2007) 394–410
4. Carbonetto, P., de Freitas, N., Barnard, K.: A statistical model for general contextual object recognition. In: ECCV. (2004)
5. Monay, F., Gatica-Perez, D.: Modeling semantic aspects for cross-media image indexing. IEEE PAMI **29** (2007) 1802–1817
6. Quattoni, A., Collins, M., Darrell, T.: Learning visual representations using images with captions. In: CVPR. (2007)
7. Fergus, R., Fei-Fei, L., Perona, P., Zisserman, A.: Learning object categories from Google's image search. In: CVPR. (2005)
8. Crandall, D.J., Huttenlocher, D.P.: Weakly supervised learning of part-based spatial models for visual object recognition. In: ECCV. (2006)
9. Kokkinos, I., Yuille, A.: HOP: Hierarchical object parsing. In: CVPR. (2009)
10. Zhu, L., Lin, C., Huang, H., Chen, Y., Yuille, A.: Unsupervised structure learning: Hierarchical recursive composition, suspicious coincidence and competitive exclusion. In: ECCV. (2008)
11. Bouchard, G., Triggs, B.: Hierarchical part-based visual object categorization. In: CVPR. (2005)
12. Fidler, S., Boben, M., Leonardis, A.: Similarity-based cross-layered hierarchical representation for object categorization. In: CVPR. (2008)
13. Epshtein, B., Ullman, S.: Feature hierarchies for object classification. In: ICCV. (2005)
14. Ommer, B., Buhmann, J.: Learning the compositional nature of visual object categories for recognition. IEEE PAMI **32** (2010) 501–516
15. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV **60** (2004) 91–110
16. Ke, Y., Sukthankar, R.: PCA-SIFT: A more distinctive representation for local image descriptors. In: CVPR. (2004)