3D Object Modeling by Sharing Visual Attributes across Poses and Scales

Liliana Lo Presti and Marco La Cascia
DICGIM - Universitá degli Studi di Palermo (Italy)
V.le delle Scienze, Ed. 6
liliana.lopresti@unipa.it, marco.lacascia@unipa.it

1. Introduction

Scene parsing aims at understanding a scene and the arrangements of the objects in it. While this is a task human beings are pretty good at [7], a machine needs to: recognize the kind of scene (indoor vs outdoor, bedroom vs. living room etc.)[4], detect and recognize 3D objects across multiple poses and scales [8, 5], infer the geometrical arrangement of the objects in the scene [2, 1], etc..

In the proposed framework, a 3D object is modeled as a graph. Each node in the graph represents a visual attribute automatically discovered by considering features that are consistently and repeatedly present across different poses and scales. Such visual attributes are different from "parts" [5], which are referred in literature as regions the object may be segmented in and related together by geometrical properties or transformations. Differently than other methods [5], which learn the appearance of the object given the pose, we aim at building a model that jointly represents local appearance, pose and scale.

To discover visual attributes, we track key-points across multiple views and cluster the set of trajectories to get the nodes in the graph. In practice, each node is a visual attribute and is described by the clustered key-point expected appearance, the pose probability distribution representing how likely is that the visual attribute may be detected in a discretized set of poses, the scale probability distribution representing the probability that the visual attribute is visible in a discretized set of scales.

Edges in the graph represent connectivity among the visual attributes across poses: two visual attributes are connected if they can be detected together at least in a pose.

Figure 1.a and 1.b show the set of trajectories detected across a subset of views for two objects belonging to the same category. The set of trajectories across multiple objects is used to learn the visual attribute graph for the category (1.c).

2. Visual Attribute Graph Learning

Finding correspondences across multiple views is done by ordering the images by camera point of view, and matching key-points in pairs of subsequent images. The problem is similar to that of tracking points across multiple views, but it has to deal with abrupt changes in camera point of view and scales. This procedure yields a set of key-point trajectories that may be clustered to compute the visual attribute graph.

We define a visual attribute n_i as a set of trajectories $\{\tau_i^1, \tau_i^2, \ldots, \tau_i^k\}$ of local discriminative feature points with similar appearance, spatially close, and consistently detected across multiple near poses and scales.

The visual attribute graph represents connected discriminative regions that are detectable given a certain pose/scale.

2.1. Features Extraction and Robust Matching

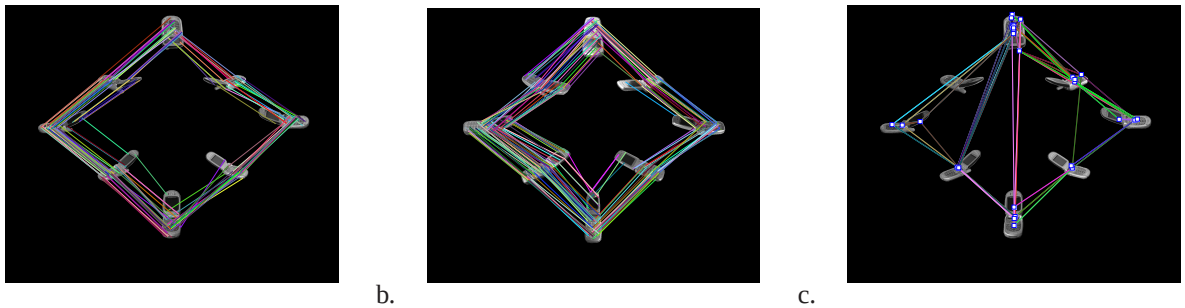
For the sake of demonstrating our method, we have employed SIFT[3] as feature points. SIFTs have been extensively used for image classification, scene categorization, object recognition; the SIFT descriptor is almost invariant to rotation and scale changes, but suffers from abrupt illumination changes.

To establish matches in pair of images, we consider an image registration problem where we look for the 2D transformation that better aligns two given images. Given an initial set of correspondences (computed with Lowe's method [3]), we iteratively estimate the affine transformation that better fits the set of correspondences. Associations of similar points within local corresponding regions in the pair of images are found by applying the Hungarian Algorithm.After few iterations the procedure provides a reliable number of correspondences. We use RANSAC to estimate the fundamental matrix on the set of correspondences and filter out outliers.

2.2. Visual Attribute Discovery

Given the correspondences in pairs of images, simple book-keeping is used to reconstruct the trajectories of the visible key-points across multiple views. Each trajectory may span multiple views but, in general, not all due to the object self-occlusions.

Within the same category, we transform the trajectories of all the objects on the same reference system; this is



a. b. c.

Figure 1. Images (a) and (b) show the set of trajectories detected across a subset of views for two objects belonging to the same category. The set of trajectories across multiple objects is used to learn the visual attribute graph for the category (c). Edges represent overlapping over poses, not visual similarity of the nodes. Only the strongest connections are showed. Node locations are computed as average of the clustered key-points.

achieved by picking up an object as reference, and registering the images taken with similar camera point of views and scales of the remaining objects to the corresponding images of the reference object.

Given the set of registered trajectories, we apply spectral clustering to group trajectories that look similar with respect to appearance, location, spanned poses. A trajectory τ is represented by means of the average SIFT descriptor d_τ, the set of key-point locations l_τ, scales and poses spanned by the trajectories that is s_τ and p_τ respectively. We assume two trajectories τ and γ, are overlapping if the intersection between their set of poses p_τ and p_γ is not empty. We also represent p_τ as a binary vector whose components represent an object pose/camera view point. The distance $D(\tau, \gamma)$ is a weighted sum of distances measuring appearance similarity, spatial proximity and overlap across poses and scales:

$$D(\tau,\gamma) = \alpha \cdot E(d_\tau, d_\gamma) + \beta \cdot \min_i E(l^i_\tau, l^i_\gamma) + \delta \cdot H(p_\tau, p_\gamma)$$

where $E(\cdot,\cdot)$ is the Euclidean distance and $H(\cdot,\cdot)$ is the Hamming distance. As for the locations, we compare only points detected in similar views.

The weight matrix W associated to the visual attribute graph may be computed as $W = e^{-D}$.

Given the weight matrix, we compute the Laplacian matrix for the graph and apply spectral clustering [6]. We set the number of clusters as the number of eigenvalues under a fixed threshold.

3. Object Detection

Given a test image, we build a graph by applying spectral clustering to the SIFT points.We adopt a sliding window approach and establish a match among the visual attribute graph and the graph in the test window by inferring a co-path, that is a pair of matching paths on the two graphs (the object visual attribute graph and the test image graph). The problem is similar to dynamic warping where the set of reachable nodes in the two graphs changes over time due to the different connectivities among the nodes in the two graphs. We use dynamic programming to infer the co-path for each possible pair of scale and pose. Among the set of inferred paths, the one providing the minimum distance is used to detect the object on the image.The inference procedure is repeated across categories and the object is classified as the category yielding the minimum cost path.

While still under testing, the proposed framework shows promising results in object recognition on a subset of the very challenging 3D object dataset [5] and in comparison to the Hungarian Algorithm.

However, still problems arise due to missing key-points, illumination changes and cluttered background that may affect the clustering and therefore the structure of the graph on the test image. In future works we will try to improve our method by using the object visual attribute graph to optimize the test image graph structure learning given the object label.

References

[1] V. Hedau, D. Hoiem, and D. Forsyth. Thinking inside the box: Using appearance models and context based on room geometry. In *ECCV*. Springer, 2010.

[2] D. Lee, A. Gupta, M. Hebert, and T. Kanade. Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces. In *NIPS*, 2010.

[3] D. Lowe. Object recognition from local scale-invariant features. In *ICCV*. IEEE, 1999.

[4] A. Oliva and A. Torralba. Building the gist of a scene: The role of global image features in recognition. *Progress in brain research*, 2006.

[5] S. Savarese and F.-F. Li. 3d generic object categorization, localization and pose estimation. In *ICCV*, 2007.

[6] U. Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 2007.

[7] J. Xiao, J. Hays, K. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*. IEEE, 2010.

[8] M. Zia, M. Stark, B. Schiele, and K. Schindler. Detailed 3d representations for object recognition and modeling. 2013.