Title

Unsupervised Learning of Probabilistic Object Models (POMs) for Object Classification, Segmentation and Recognition using Knowledge Propagation

Authors

Chen, Yuanhao
Zhu, Long Leo
Yuille, Alan
et al.

Publication Date

2008-10-02

Peer reviewed

Unsupervised Learning of Probabilistic Object Models (POMs) for Object Classification, Segmentation and Recognition

Yuanhao Chen
MOE-MS Key Laboratory of MCC
University of Science and Technology of China
yhchen4@ustc.edu

Long (Leo) Zhu
Department of Statistics
University of California, Los Angeles
lzhu@stat.ucla.edu

Alan Yuille
Department of Statistics,
Psychology and Computer Science
University of California, Los Angeles
yuille@stat.ucla.edu

Hongjiang Zhang
Microsoft Advanced Technology Center
hjzhang@microsoft.com

Abstract

We present a new unsupervised method to learn unified probabilistic object models (POMs) which can be applied to classification, segmentation, and recognition. We formulate this as a structure learning task and our strategy is to learn and combine basic POM's that make use of complementary image cues. Each POM has algorithms for inference and parameter learning, but: (i) the structure of each POM is unknown, and (ii) the inference and parameter learning algorithm for a POM may be impractical without additional information. We address these problems by a novel structure induction procedure which uses knowledge propagation *to enable POM's to provide information to other POM's and "teach them" (which greatly reduced the amount of supervision required for training). In particular, we learn a POM-IP defined on Interest Points using weak supervision [1, 2] and use this to train a POM-mask, defined on regional features, which yields a combined POM which performs segmentation/localization. This combined model can be used to train POM-edgelets, defined on edgelets, which gives a full POM with improved performance on classification. We give detailed experimental analysis on large datasets which show that the full POM is invariant to scale and rotation of the object (for learning and inference) and performs inference rapidly. In addition, we show that we can apply POM's to learn objects classes (i.e. when there are several objects and the identity of the object in each image is unknown). We emphasize that these models can match between different objects from the same category and hence enable object recognition.*

1. Introduction

The goal of this paper is to learn unified object models which are able to perform tasks such as classification, segmentation, and recognition (informally – what, where, and who). Unified models are desirable because they allow, for example, improvements in segmentation to enable improvements in classification and vice versa. The models are intended to combine different cues from the images (e.g. instead of relying only on interest points). The models are intended to allow rapid inference and learning, to be invariant to rotation and scaling in the image plane, and to have an unknown number of aspects (e.g. mixture components) to allow for different appearances. We build on recent work [1, 2] which learns similar models for classification using only interest point features.

To place this work in context, we give a brief review of the current literature on classification and localization/segmentation. One part of the literature concentrates on the classification tasks, e.g. [3, 4, 5, 6, 1, 2]. This work usually requires only limited supervision. The training images are known to include an object from a specific class, but the precise localization/segmentation of the object is unknown (this assumption can be weakened [1, 2]). By contrast, the segmentation literature – e.g. [7, 8, 9] – requires that the precise localization/segmentation of the objects are given in the training images. This is considerably more supervision that we use in this paper. Some work does combine classification and localization/segmentation. LOCUS [10] requires only limited supervision, but the approach is only reported on a limited number of images. Leibe *et al* perform localization/segmentation in addition to classification, but has no explicit shape model. Our approach in this

paper is also related to ObjCut [11] which uses motion cues to initialize the grab-cut algorithm [12, 13, 14]. ObjCut was only reported on a limited number of images.

We formulate our models in terms of probabilistic inference and machine learning. From this perspective, learning object models is a structure induction problem where the goal is to learn the structure of the probability model describing the objects as well as the parameters of these distributions. Structure learning is a difficult and topical problem [1] and contrasts with standard learning where the structure of the model is assumed known and only the parameters need to be estimated. Our approach to structure learning involves techniques for growing simple models using proposals obtained by clustering [1, 2] and a new method, which we informally call *knowledge propagation*, which adds new components to the model. These new components allow for new aspects, incorporate new image cues, and enable the models to perform new tasks. More specifically, we will obtain a set of probabilistic object models (POM's) which are used as building blocks and where knowledge is propagated between POM's. Our approach also involves efficient strategies for performing parameter learning and inference for the different structure models. Final inference, detecting, classifying, and segmenting the object in an image is performed in 5 seconds.

We now briefly step through the process of structure learning as it occurs in this paper. Firstly, we learn a POM defined on interest points (IP's), POM-IP, using the weakly supervised techniques described in [1, 2]. This POM-IP is able to detect and classify objects, to detect their aspect, deal automatically with scaling and rotation changes, and give a very crude estimate for the segmentation. We now seek to extend this model by incorporating different cues to enable accurate segmentation and to improve classification. We use the POM-IP to train a POM-mask and use regional image cues to perform segmentation with a min-cut/max-flow algorithm [15]. Intuitively, we start by using a version of grab-cut [12, 13, 14] where POM-IP substitutes for human interaction to provide the initial estimate of the segmentation (as motion cues do in ObjCut [11]). We proceed to learn a shape prior for each aspect which yields an integrated POM-IP and POM-mask capable of performing classification and segmentation/localization. To improve classification, we use POM-IP and POM-mask to train a number of POM-edgelets defined in different subregions of the shape, which enables edgelets cues to improve the classification. During the learning process, POM's propagate knowledge in order to train other POM's. It is important to realize that we cannot train these model separately (without adding more supervision). The full model couples the POM-IP, POM-mask, POM-edgelets together (as a regular, though complicated, graphical model) and performs inference on this model.

2. The Image Representation

This section describes the different image features that we use: (i) interest points, (ii) edgelets, and (iii) regional features. These will be used to define POM-IP, POM-edgelets, and POM-mask.

The *edgelet and interest point image features* are represented by triples $x_i = (z_i, \theta_i, A_i)$, where z_i is the location of the feature in the image, θ_i is the orientation of the feature, and A_i is an appearance vector. For edgelets, the appearance vector is not used. The edgelets are extracted by applying the Canny edge detector and estimating the orientation. The interest point features are those reported in [1, 2] which were designed to be relatively independent of scale and photometric properties. The Kadir-Brady procedure [16] is used to detect interest regions. These are described by the SIFT descriptor [17]. Principal Component Analysis (PCA) is used to reduce the description to fifteen dimensions to give the appearance A together with the orientation θ.

The *oriented triplets* were designed [1, 2] to give geometric properties, the *invariant triplet vector* (ITV), which are independent of scale and orientation. (Previous author have used non-oriented triplets [18, 19]). The (ITV) $\vec{l}(z_i, \theta_i, z_j, \theta_j, z_k, \theta_k)$ is a function of the geometry of three features points $(z_i, \theta_i, z_j, \theta_j, z_k, \theta_k)$ which is invariant to scale and rotation.

The *regional image features* are computed by applying a filter $\phi(\cdot)$ to the image I yielding a set of responses $\phi_x(I)$ where $x \in D$ (where D is the image domain). The domain D is split into pixels within the object denoted by $L_x = 1$ and pixels outside the object with $L_x = 0$ (the variable $\{L_x\}$ specifies the location and segmentation of the object, see section (4)). If $\{L_x\}$ is specified, we can compute the histograms of the image statistics inside the object $h_O(., L)$ and in the background $h_B(., L)$:

$$h_O(z, L) = \frac{1}{|D_O|} \sum_{x \in D} \delta_{L_x,1} \delta_{\phi_x(I),z}, \quad (1)$$

$$h_B(z, L) = \frac{1}{|D_B|} \sum_{x \in D_O} \delta_{L_x,0} \delta_{\phi_x(I),z}, \quad (2)$$

where $|D_O| = \sum_{x \in D} \delta_{L_x,1}$ and $|D_B| = \sum_{x \in D} \delta_{L_x,0}$. In this paper, $\phi_x(I)$ is either the colour or grey-scale image intensities. But other choices, including local texture filters are also suitable.

3. POM-IP and POM-edgelets

The POM-IP is defined on sparse interest points and is identical to the probabilistic grammar Markov model (PGMM) described in [1, 2]. The identical formulation is used for the POM-edgelets defined on sparse edgelets. The

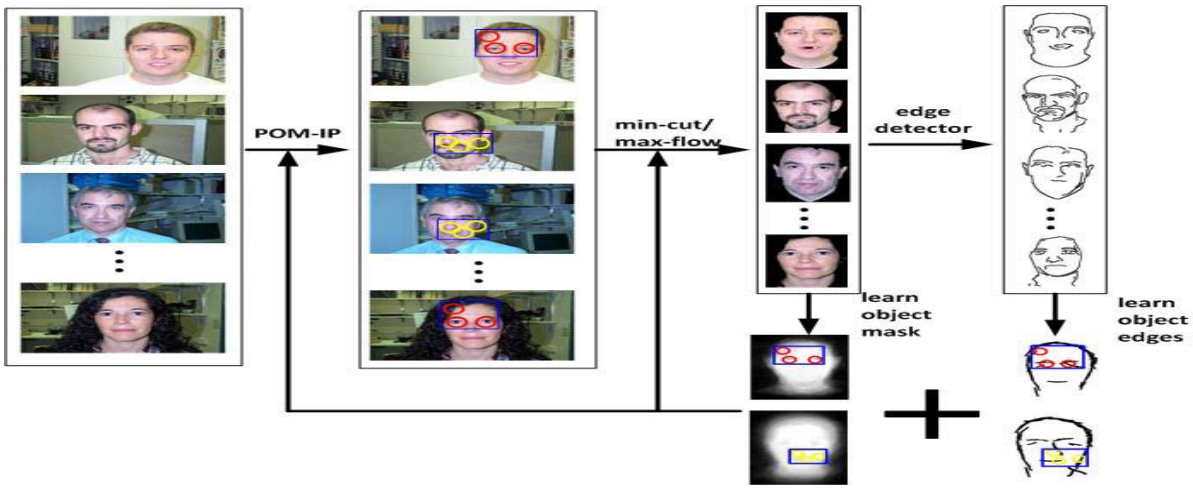

Figure 1. The flow chart of unsupervised learning

input data to the POM-IP is the position, orientation, and apperances $(z, \theta, A) = \{(z_i, \theta_i, A_i) : i = 1, ..., N\}$ of the interest points in the image.

The *POM-IP distribution* is specified by $P(z, \theta, A|V, G, s)P(V|s)P(s)P(G)$. Here s is the aspect of the object. Each aspect consists of a set of attributed nodes $a = 1, .., N_s$ organized into triplet cliques. The correspondence variable $V = \{V_{ai}\}$ specifies the assignment between nodes (labeled a) of the aspect and the interest points (labeled i) (or edgelets) in the image. Each aspect node is required to either precisely match an interest point in the image, or to be unmatched. The term $P(z, \theta, A|V, G, s)$ combines a prior probability on the spatial relationships of the nodes (positions and orientations) together with a model of the appearances. This prior is defined on the ITV's of the oriented triplets ensuring that the model is invariant to scale and rotation. The prior probability $P(G)$ on the pose is the uniform distribution. The *structure* of the POM-IP is specified by the number of aspects and the number of nodes in each aspect. The *parameters* of the POM-IP (which will be learnt) are the parameters of the spatial relationship models (a multivariate Gaussian defined on the ITV's), the parameters of the appearance models (multivariate Gaussian distributions on the appearances), the proportion of background nodes and the probability that aspect nodes will be unobserved. See [1, 2] for more details. The *POM-edgelet distribution* is of the same form but does not include A attributes.

We emphasize two important properties of the POM-IP and POM-edgelets distribution. Firstly, the use of different aspects enables these POM's to capture several different appearances of the object and can be used to directly model hybrid object classes. Secondly, the fact that the spatial relationships are defined on the ITV (of IP's or edgelets) means that the model is invariant to scale and orientation of the object (this builds on the scale invariance of the Kadir-Brady detector [16]).

The *POM-IP inference* estimates V, s, G from each image (the correspondences, the aspect, the pose). This is performed by using dynamic programming (DP) to obtain the correspondence V for each aspect s. Then we maximize over s by enumeration. The pose G is estimated separately (exploiting the scale and orientation invariance of the POM-IP distribution). The *POM-edgelet inference* is similar.

The *POM-IP learning* consists of two parts: (a) determining the structure of the POM-IP, and (b) estimating the parameters of the POM-IP distribution. We perform learning by a structure pursuit strategy which starts with an initial default structure for the POM-IP (i.e. all nodes are background nodes) and grows the POM-IP incrementally by adding new oriented triplets. These can be used either to grow an existing aspect or to create a new aspect consisting of this oriented triplet. Clustering of the training dataset is used to create a vocabulary of oriented triplets which are used as proposals for growing the structure. These proposals come with estimates of their parameters (appearance and spatial relationships). We perform *parameter learning* for the new model by the expectation maximization (EM) algorithm. We use dynamic programming (DP) (sum rule) to help sum out the hidden variables and ensure that the algorithm is computationally efficient. The estimates of the oriented triplet parameters (provided by clustering) help give initial conditions for EM which (empirically) prevent it getting stuck in a bad local minima. We compare the new model to the original by *model selection* which requires us to compare the probability of the training data using either model (the current model and the proposed model). We can again apply dynamic programming (sum rule) to help compute these probabilities efficiently. For more details, see [1, 2]. The POM-edgelet learning is similar but requires information from the POM-mask to restrict the location of the edges, see section (5).

4. POM-mask

The POM-mask uses regional cues to perform segmentation/localization. It is trained using knowledge from the POM-IP giving crude estimates for the segmentation (e.g. the bounding box of the IP's). This training enables POM-mask to learn a shape prior for each aspect of the object. After training, the POM-mask and POM-IP are coupled (see figure 1). During inference, the POM-IP supplies estimates of pose and aspect to help estimate the POM-mask variables.

4.1. Overview of the POM-mask

The probability distribution of the POM-mask is defined by:

$$P(I|L,q)P(L|s,G)P(q|s)P(s)P(G). \tag{3}$$

where I is the intensity image, L is a binary mask indicating which pixels belong the inside and the outside of the object, $q = (q_O, q_B)$ is a set of distributions on the image statistics inside and outside the object. $P(L|s,G)$ is a prior on the position of the mask conditioned on the aspect s and the pose G. (We refer to this as the probability mask). The prior $P(q|s)$ is set to be the uniform distribution because our attempts to learn it showed that it was extremely variable for most objects.

The *inference* for the POM-mask requires the POM-IP to make initial estimates for the pose G, the aspect s and the silhouette of the object. Then inference of q and L can be performed by estimating each alternatively until convergence. The L is estimated by the min-cut/max-flow algorithm [15] and q is estimated by computing the histograms within the current estimates of the object and the background (i.e. setting $q_O(.) = h_O(., L)$ and $q_B(.) = h_B(., L)$). Initialization for the algorithm is provided by using the (thresholded) probability mask to make the initial estimate of the object shape (using the pose estimate of G supplied by POM-IP).

Learning the POM-mask is also performed with knowledge propagated from the POM-IP. The main parameter to be learnt is the prior probability of the shape, which we represent by a *probability mask*. Learning proceeds in the EM-style. We initialize the probability mask by the uniform distribution. Then we estimate the shapes of the object in each image keeping account of the aspect (this stage is similar to grab-cut [12, 13, 14] where the initialization segmentation is given by the bounding box of the IP's output by the POM-IP). Then we estimate the probability mask (correcting for the pose). Then we repeat the segmentation process using the new probability mask as the prior.

4.2. Details of the POM-mask probability model

The term $P(I|L,h)$ is given by:

$$\frac{1}{Z}\exp\{\sum_i \phi_1(I_i|L_i,h) + \sum_{i,j\in N(i)} \phi_2(I_i,I_j|L_i,L_j)\} \tag{4}$$

where i is the index of image pixel, j is a neighboring pixel of i and Z is the normalizing constant.

The unary potential terms are given by:

$$\phi_1(I_i|L_i,h) = \begin{cases} \log q_O(I_i) & \text{if } L_i = 1 \\ \log q_B(I_i) & \text{if } L_i = 0 \end{cases}. \tag{5}$$

The binary potentials $\phi_2(I_i, I_j|L_i, L_j)$ is the contrast term [11]:

$$\phi_2(I_i,I_j|L_i,L_j) = \begin{cases} \gamma(i,j) & \text{if } L_i \neq L_j, \\ 0 & \text{if } L_i = L_j \end{cases}. \tag{6}$$

where $\gamma(i,j) = \lambda\exp\{-\frac{g^2(i,j)}{2\gamma^2}\}\frac{1}{dist(i,j)}$, $g(.,.)$ is a distance measure on the colors I_i, I_j and $dist(i,j)$ measures the spatial distance between i and j. For more details, see [12, 13].

The prior probability distribution $P(L|s,G)$ for the labels L is defined as follows:

$$P(L|s,G) = \frac{1}{Z}\exp\{\sum_i \psi_1(L_i;s,G) + \sum_{i,j}\psi_2(L_i,L_j|\zeta)\} \tag{7}$$

The unary potential $\psi_1(L_i;s,G)$ encodes a shape prior (probability mask) given by:

$$\begin{aligned}\psi_1(L_i;s,G) &= L_i\log(T(G)\{M^s\}_i) \\ &+(1-L_i)\log(1-T(G)\{M^s\}_i),\end{aligned} \tag{8}$$

where $M^s = \{M_i^s\}$ is a probability mask, where s indexes the aspect (i.e. there are different probability masks for different aspects). Here $M_i^s \in [0,1]$ is the probability that pixel i is inside the object. We denote $T(G)\{M^s\}_i$ to be the probability that pixel i is inside the object after transforming the mask by position, scale, and orientation (indexed by G).

The binary potential is of Ising form:

$$\psi_2(L_i,L_j|\zeta) = \begin{cases} 0, & \text{if } L_i \neq L_j \\ \zeta, & \text{if } L_i = L_j \end{cases}. \tag{9}$$

where ζ is a parameter of the generic prior.

5. Combining the POM-edgelet Models

Once the POM-mask model has been learnt we can use it to teach POM-edgelets which are defined on sub-regions of the shape (adjusted for our estimates of pose and aspect). The method to *learn* the POM-edgelets is exactly the same

as the one for learning the POM-IP except we do not have appearance attributes and the sub-region where the edgelets appear is fixed to a small part of the image (i.e. the estimate of the shape of the sub-region). (Note that training a POM-edgelet model on the entire image is impractical because the numbers of edgelets in the image is orders of magnitude larger then the number of interest points, and all edgelets have similar appearances).

The *inference* for the POM-edgelets requires an estimate for the pose G and aspect s which is supplied by the POM-IP (the POM-mask is only used in the learning of the POM-edgelets).

6. Results

We now give results for a variety of different tasks and scenarios. We compare performance of the default PGMM [1] and the full POM. We collect 14 classes (see figure 2) from Caltech 101 [20]. In all experiments, we learnt the full POM on a *training set* consisting of half the set of images (randomly selected) and evaluated the full POM on the remaining images, or *testing set*. The images in the dataset were required to have at least fifty images to ensure that there was sufficient data in the training set to learn the POM's.

The speed for inference is less than 5 seconds on a 450×450 image. This breaks down into 1 second for interest-point detector and SIFT descriptor, 1 second for edge detection, 1 second for the graph cut algorithm, and 1 second for the image parsing. The training time for 250 images is approximately 4 hours.

6.1. The Tasks

We tested on three tasks: (I) The *classification* task is to determine whether the image contains the object or is simply background. (II) The *segmentation* task is evaluated by *precision and recall*. The precision $|\mathrm{R} \cap \mathrm{GT}|/|\mathrm{R}|$ is the proportion of pixels in the estimated shape region R that are in the ground-truth shape region GT. The recall $|\mathrm{R} \cap \mathrm{GT}|/|\mathrm{GT}|$ is the proportion of pixels in the ground-truth shape region that are in the estimated shape region. (III) The *recognition* task which we illustrate by showing matches.

We performed these tests for three scenarios: (I) *Single object category* when the training and testing images containing an instance of the object with unknown background. Due to the nature of the datasets we used there is little variation in orientation and scaling of the object, so the invariance of our learning and inference was not tested. (II) *Single object category with variation* where we had manipulated the training and testing data to ensure significant variations in object orientation and scale. (III) *Hybrid object category* where the training and testing images contain an instance of one of three objects (face, motorbike, or airplane).

Table 1. Classification

Dataset	Ours	[1]	[3]	[21]
Faces	98.0	98.0	96.4	96.7
Airplane	91.8	90.9	90.2	98.4
Motorbikes	94.6	92.6	92.5	92.0
Faces(Scaled)	96.5	-	-	-
Faces(Rotated)	96.7	94.8	-	-
Faces(Scale+Rotated)	94.6	92.3	-	-
Hybrid	87.8	84.6	-	-

Table 2. Classification Results on 14 classes

Dataset	POM-IP	POM-Edge
14-class Average	86.4	89.4

Table 3. The comparisons of segmentation. The precision and recall measure is reported. "S": scale. "R": rotation. "S+R": scale plus rotation. "Average": the average performance on 14 classes.

Dataset	[1]	POM-IP	POM-Mask	POM-Edge
Average	**67 / 62**	**75 / 65**	**80 / 65**	**80 / 69**
S	83 / 63	71 / 90	76 / 87	76 / 89
R	80 / 61	62 / 90	70 / 88	70 / 90
S+R	81 / 57	63 / 84	68 / 85	68 / 87
Hybrid	60 / 61	69 / 72	77 / 65	73 / 73

Table 4. The comparisons of segmentation. The measure of segmentation accuracy in pixels is used.

	POM	[22]
Faces easy	86.0%	78.0%
Motorbikes	79.0%	77.0%
Grand Piano	84.8%	78.0%
Starfish	85.9%	69.0%
Sunflower	86.2%	86.0%
Watch	75.5%	60.0%

6.2. Scenario 1: Classification and Segmentation for Single object category

In this experiment, the training and testing images come from one object class. The experimental results, see tables 1 and 2, show improvement in *classification* when we use the full POM (compared to the POM-IP/PGMM). These improvements are due entirely to the edgelets in the full POM. We note that the regional features from POM-mask supply no information for object classification because the appearance model is very weak (i.e. the q_0 distribution has uniform prior). The improvements are biggest for those objects where the edgelets give more information compared to the interest points (e.g. the football, motorbike, and grand piano).

Observe that *segmentation* (see table 3) is extremely improved by using the full POM compared to the POM-IP.

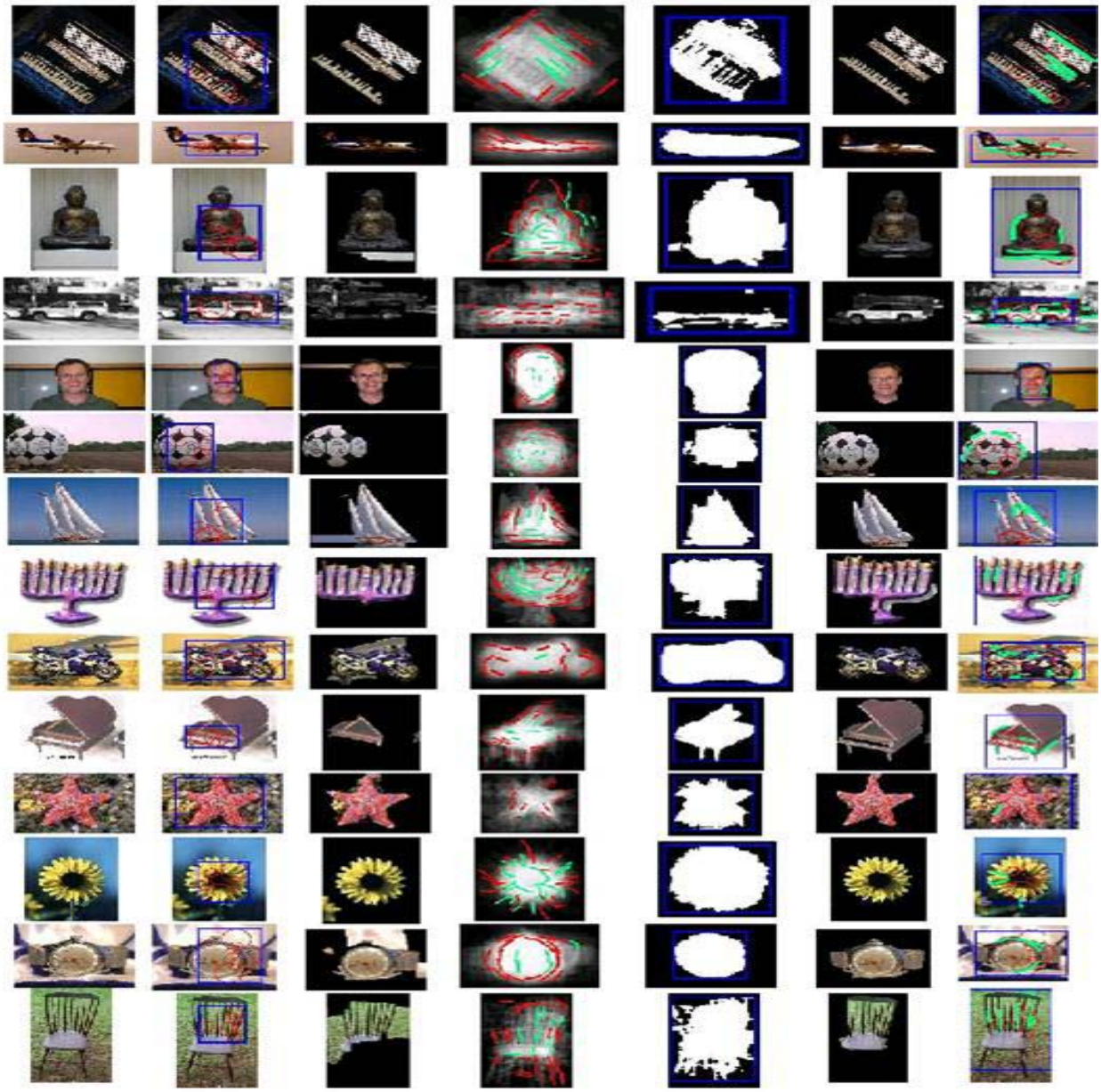

Figure 2. The columns show the fourteen objects that we used. The seven columns are labelled left to right as follows: (1) Original Image, (2) the Bounding Box for the Interest-Point Model, (3) the GraphCut segmentation with the features estimating using the Bounding Box, (4) the probability object-mask with the edgelets (green means features within the object, red means on the boundary), (5) the thresholded probability mask,(6) the new segmentation using the probability object-mask, (7) the parsed result.

To evaluate these comparisons we show improvements between using the PGMM model, the POM-IP model (with grab-cut), the POM-IP combined with the POM-mask, and the full POM.. The main observation is that the bounding box round the interest-points is only partially successful. There is a bigger improvement when we use the interest-points to initialize a grab-cut algorithm. But the best performance occurs when we use the edgelets. We also compare our method with [22] for segmentation. See the comparisons in table 4.

To get better understanding of segmentation results, and the relative importance of the different components of the full POM, consider figure (2) where we show examples for each object category. The first column shows the input image and the second column gives the bounding box of the interest points of POM-IP. Observe that this bounding box only gives a crude segmentation and can lie entirely inside the object (e.g. face, football), or encompass the object (e.g. car, starfish), or only capture a part of the object (e.g. accordion, airplane, grand piano, windsor chair). The third column shows the results of using grab-cut initialized by the POM-IP. This gives reasonable segmentations for some objects (e.g. accordion, football) but has significant errors for others (e.g. car, face, clock, windsor chair) sometimes cap-

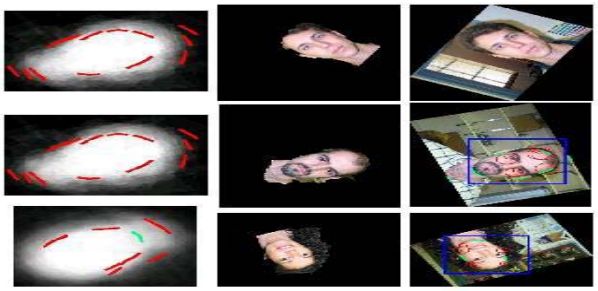

Figure 3. POM can be learnt while training images are randomly scaled and rotated.

turing large parts of the background while missing significant parts of the object (e.g. windsor chair). The fourth column shows the POM-mask learns good shape priors (probability masks) for all objects despite the poorness of some of the initial segmentation results. This column also shows the positions of the edgelet features learn by the POM-edgelets. The thresholded probability mask is shown in the fifth column and we see that it takes reasonable forms even for the windsor chair. The sixth column show the results of using the full POM model to segment these objects (i.e. using the probability mask as a shape prior) and we observe that the segmentations are good and significantly better than those obtained using grab-cut only. Observe that the background is almost entirely removed and we now recover the missing parts, such as the legs of the chair and the rest of the grand piano. Finally, the seventh column illustrates the locations of the feature points (interest points and edgelets) and shows that the few errors occur for the edgelets at the boundaries of the objects.

6.3. Scenario 2: Varying the scale and orientation of the objects

The full POM is designed so that it is invariant to scale and rotation for both learning and inference. This advantage was not exploited in scenario 1, since the objects tended to have similar orientations and sizes. To emphasize and test this invariance, we learnt the full POM for a data-set of faces where we scaled, translated, and rotated the objects, see figure (3). The scaling was from 0.6 to 1.5 (i.e. by a factor of 2.5) and the rotation was uniformly sampled from 0 to 360 degrees. We considered three cases where we varied the scale only, the rotation only, and scale and rotation. The results, see the the bottom rows of see table (1), show only slight degradation in performance for the tasks.

6.4. Scenario 3: Hybrid Object Models

We now make the learning and inference tasks even harder by allowing the training images to contain several different types of objects (extending work in [1] for the PGMM). More specifically, each image will contain either a face, a motorbike, or an airplane (but we do not know which). The full POM will be able to successfully learn a hybrid model because the different objects will correspond to different aspects. It is important to realize that we can identify the individual objects as different aspects of the full POM, see figure (4). In other words, the POM does not only learn the hybrid class, it also learns the individual object classes in an unsupervised way.

Table 5. Confusion Matrix obtained by hybrid model. The mean of diagonal is 89.8% which is comparable with 92.9% [21].

	Face	Motorbikes	Airplanes
Face	96.0%	0.0%	4.0%
Motorbikes	2.2%	85.4%	10.4%
Airplanes	2.0%	10.0%	88.0%

The performance of learning this hybrid class is shown in table (3). We see that the performance degrades very little, despite the fact that we are giving the system even less supervision. The confusion matrix between faces, motobikes and airplanes is shown in table 5. Our result is slightly worse than [21].

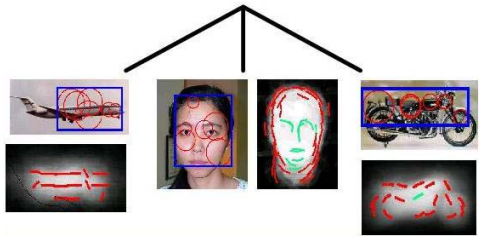

Figure 4. Hybrid Model. Training images consist of faces, motorbikes and airplanes where we don't know the identity of the class for each image.

6.5. Scenario 4: Matching and Recognition

The POM is capable of performing matching and recognition. Figure 5 shows an example of correspondence between two images. For recognition, we use 200 images containing 23 persons. Given a query of a image containing a face, we output the top three candidates from the 200 images. The similarity between two images is measured by the differences of intensity of the corresponding interest points. The recognition results are illustrated in figure 6.

7. Discussion

This paper is part of a research program where the goal is to learn object models for all object-related visual tasks. In this paper we built on previous work [1, 2] which used

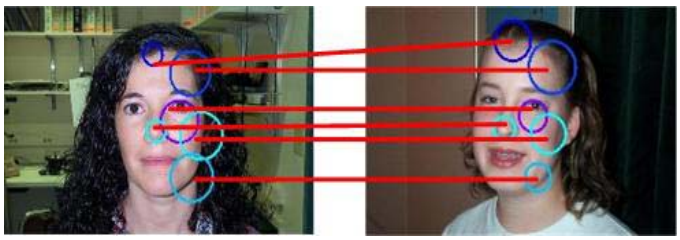

Figure 5. An example of correspondence obtained by POM.

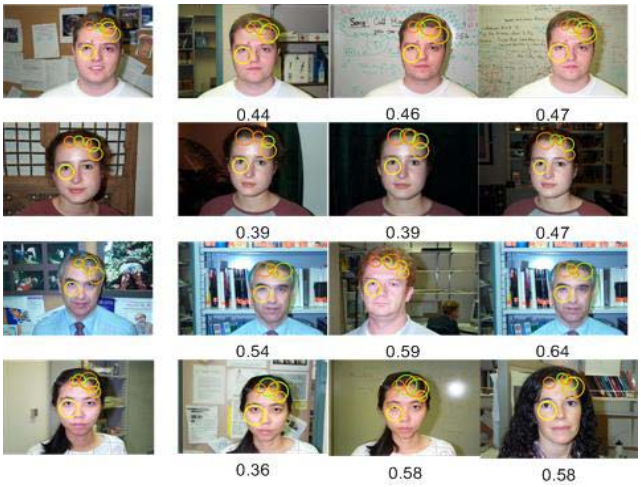

Figure 6. Recognition Examples. The first column is the prototype. The next three columns show the top three rankings. A distance to the prototype is shown under each image

weak supervision to learn a probabilistic grammar Markov model (PGMM) which used interest point features and performed classification. Our extension is based on combining elementary probabilistic object models (POM's) which use different visual cues and can combine to perform a variety of visual tasks. The POM's cooperate to learn and do inference by *knowledge propagation*. In this paper, the POM-IP (or PGMM) was able to train a POM-mask model so that the combination could perform localization/segmentation. In turn, the POM-mask was able to train a set of POM-edgelets which when combined into a full POM can use edgelet features to improve the classification. We demonstrated this approach on large numbers of images of different objects. We also showed the ability of our approach to learn and perform inference when the scale and rotation of objects is unknown. We showed its ability to learn a hybrid model containing several different objects. The inference is performed in seconds, and the learning in hours.

8. Acknowledgments

This research was supported by NSF grant 0413214.

References

[1] L. Zhu, Y. Chen, and A. L. Yuille, "Unsupervised learning of a probabilistic grammar for object detection and parsing," in *NIPS*, 2006, pp. 1617–1624.

[2] ——, "Unsupervised learning of probabilistic grammar-markov models for object categories," in *Technical Report*, 2007.

[3] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *CVPR (2)*, 2003, pp. 264–271.

[4] ——, "A sparse object category model for efficient learning and exhaustive recognition," in *CVPR (1)*, 2005, pp. 380–387.

[5] D. J. Crandall and D. P. Huttenlocher, "Weakly supervised learning of part-based spatial models for visual object recognition," in *ECCV (1)*, 2006, pp. 16–29.

[6] B. Leibe, A. Leonardis, and B. Schiele, "Combined object categorization and segmentation with an implicit shape model," in *ECCV'04 Workshop on Statistical Learning in Computer Vision*, Prague, Czech Republic, May 2004, pp. 17–32.

[7] E. Borenstein and S. Ullman, "Learning to segment," in *ECCV (3)*, 2004, pp. 315–328.

[8] A. Levin and Y. Weiss, "Learning to combine bottom-up and top-down segmentation," in *ECCV (4)*, 2006, pp. 581–594.

[9] X. Ren, C. Fowlkes, and J. Malik, "Cue integration for figure/ground labeling," in *NIPS*, 2005.

[10] J. M. Winn and N. Jojic, "Locus: Learning object classes with unsupervised segmentation," in *ICCV*, 2005, pp. 756–763.

[11] M. P. Kumar, P. H. S. Torr, and A. Zisserman, "Obj cut," in *CVPR (1)*, 2005, pp. 18–25.

[12] A. Blake, C. Rother, M. Brown, P. Pérez, and P. H. S. Torr, "Interactive image segmentation using an adaptive gmmrf model," in *ECCV (1)*, 2004, pp. 428–441.

[13] Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images," in *ICCV*, 2001, pp. 105–112.

[14] C. Rother, V. Kolmogorov, and A. Blake, ""grabcut": interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, 2004.

[15] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," in *EMMCVPR*, 2001, pp. 359–374.

[16] T. Kadir and M. Brady, "Saliency, scale and image description," *International Journal of Computer Vision*, vol. 45, no. 2, pp. 83–105, 2001.

[17] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[18] Y. Amit and D. Geman, "A computational model for visual selection," *Neural Computation*, vol. 11, no. 7, pp. 1691–1715, 1999.

[19] S. Lazebnik, C. Schmid, and J. Ponce, "A sparse texture representation using local affine regions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1265–1278, 2005.

[20] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," *Comput. Vis. Image Underst.*, vol. 106, no. 1, pp. 59–70, 2007.

[21] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman, "Discovering objects and their localization in images," in *ICCV*, 2005, pp. 370–377.

[22] L. Cao and L. Fei-Fei, "Spatially coherent latent topic model for concurrent object segmentation and classification," in *ICCV*, 2007.