Move and the Robot will Learn: Vision-based Autonomous Learning of Object Models

Xiang Li
Department of Computer Science
Texas Tech University, USA
xiang.li@ttu.edu

Mohan Sridharan
Department of Computer Science
Texas Tech University, USA
mohan.sridharan@ttu.edu

***Abstract*—As robots are increasingly deployed in complex real-world domains, visual object recognition continues to be an open problem. Existing algorithms for learning and recognizing objects are predominantly computationally expensive, and require considerable training or domain knowledge. This paper describes an algorithm for robots to use motion cues to identify and focus on a set of interesting objects, automatically extracting appearance-based and contextual cues from a small number of images to efficiently learn representative models of these objects. Learned models exploit complementary strengths of: (a) relative spatial arrangement of gradient features; (b) graph-based models of neighborhoods of gradient features; (c) parts-based models of image segments; (d) color distributions; and (e) mixture models of local context. The learned models are used by an energy minimization algorithm and a generative model of information fusion for reliable and efficient object recognition in novel scenes. The algorithm is evaluated on wheeled robots in indoor and outdoor domains, and on images from benchmark datasets.**

I. Introduction

Sophisticated algorithms developed for representing and recognizing objects using different visual cues are predominantly computationally expensive and require considerable training or prior knowledge to learn object models. However, robot application domains make it challenging to obtain accurate domain knowledge, elaborate human feedback or many labeled samples of relevant objects. Enabling robots to learn object models and recognize objects with minimal human supervision thus continues to be an open problem.

The above-mentioned challenges are offset by some observations. First, many objects possess unique characteristics and distinguishable motion patterns, although these characteristics and patterns are not known in advance and may change over time. Second, images encode information about objects in the form of complementary appearance-based and contextual cues, although different cues may be best suited to represent objects in different situations. Third, in many application domains, robots learn the domain map and do not need to model all domain objects; many tasks require robots to focus on a small set of objects, especially those that move. Our algorithm exploits these observations to achieve the following:

- Learn object models from a small $(3 - 8)$ number of images, efficiently identifying image regions corresponding to moving (i.e., interesting) objects using motion cues.
- Exploit complementary strengths of appearance-based and contextual visual cues to efficiently learn representative models of these objects from relevant image regions.
- Generative models of information fusion and energy minimization algorithms use learned object models for reliable and efficient recognition in novel scenes.

Fig. 1. Local, global and temporal cues extracted from pixels within the yellow boundary represent appearance, while mixture models and relative positions (e.g., "on" and "under") of regions within the red rectangle (outside the yellow polygon) represent context.

These objectives promote incremental learning, enabling robots to acquire and use sensor inputs and human feedback based on need and availability. Object models consist of: spatial arrangements of gradient features, graph-based models of neighborhoods of gradient features, parts-based models of image segments, color distributions, and local context models. Although the underlying visual cues have been used in other algorithms, our representation of these cues fully exploits their complementary strengths, resulting in reliable and efficient learning and recognition in indoor and outdoor domains.

II. Related Work

Many algorithms have been developed for modeling and recognizing objects using scale, rotation and affine-invariant image gradient features [1], and appearance and shape features [2]. Algorithms have modeled global context in the image [3] and local context from regions surrounding the objects of interest [4], extracting adaptive contextual cues from image regions [5]. Researchers have used motion cues in conjunction with other cues for object recognition [6]. Algorithms have also used multiple visual cues and interactions with objects to learn spatial relationships between objects [7], distinguish objects from background [8], and discover groups of related objects [9]. Algorithms have also been developed for unsupervised learning of hierarchical spatial structures using rule-based models [10], and for using a composition system to automatically learn structured, hierarchical object representations without manual segmentation or object localization [11]. However, these algorithms are computationally expensive, requiring many labeled training samples and/or extensive prior knowledge and human supervision.

Existing algorithms support the learning of object models from a small set of images, e.g., using appearance and shape features [12]. Recent research has provided an "objectness" measure based on multiple image cues to automatically identify image windows containing objects from the desired classes [13]. However, this algorithm is computationally expensive and does not fully exploit visual cues (see Section IV).

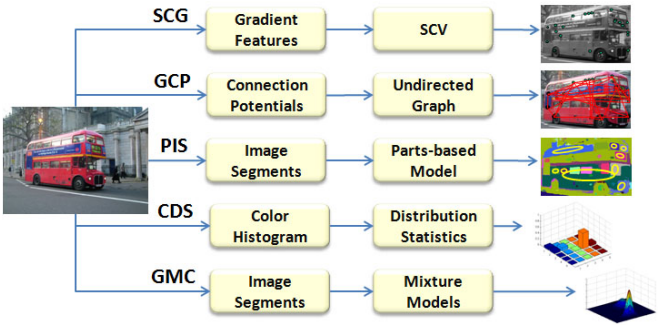

Fig. 2. Learned model uses contextual and appearance-based cues to characterize objects of interest.

Motivated by the limitations of existing algorithms, our recent work provided an overview of an algorithm that exploits the complementary strengths of appearance-based and contextual visual cues [14]. This paper describes and thoroughly evaluates our algorithm that supports incremental learning of representative object models from a small number of images, resulting in reliable and efficient object recognition in novel scenes.

III. Proposed Approach

In our algorithm, robots learn the domain map using range data and consider objects that move to be interesting. Based on the observation that characteristic features of an object have similar relative motion between consecutive images, robots track local gradient features in short sequences ($3-8$ images), identifying regions of interest (ROIs) corresponding to moving objects by clustering features with similar relative motion. The complementary strengths of appearance-based and contextual visual cues extracted from these ROIs are used to learn object models. One underlying assumption (that works in practice) is that object motion has a non-trivial linear component. This section describes the components of learned object models, and the use of learned models for object recognition.

A. Object Model Learning

Figure 2 shows the object model's components for a specific ROI: (1) relative spatial arrangements of gradient features; (2) graph-based model of connection potentials between gradient features; (3) a parts-based model of spatial arrangement of image segments; (4) second-order statistics of color distributions; and (5) Gaussian mixture models of local context. These components are described below.

1) Spatial Coherence of Gradient Features (SCG): Gradient features extracted from the image ROI may not be unique. Our prior work created a *spatial coherence vector* (SCV) to model the relative spatial arrangement of gradient features, which is difficult to duplicate [15]. The SCV is computed along x and y axes for each of the N gradient features in the ROI, e.g., $SCV_{x,i} = \{d^x_{i,1}, d^x_{i,2}, \ldots, d^x_{i,N}\}$ and $SCV_{y,i} = \{d^y_{i,1}, d^y_{i,2}, \ldots, d^y_{i,N}\}$, and if x_i and x_j are the x-coordinates of feature i and j in the image, $d^x_{i,j} = 1, 0$ or -1 for $x_i >, =$ or $< x_j$ respectively; $d^y_{i,j}$ is defined similarly. The object model thus extracts N gradient features from the ROI (each feature is a 128D vector) and a $2(N-1)$-dimensional SCV for each feature. Modeling the relative spatial arrangement of features may make this component sensitive to large changes in orientation; however, it works well in practice.

2) Graph-Based Model of Connection Potentials (GCP): The second component of the object model captures the relationships between neighboring gradient features in the ROI. The *connection potential* for any two gradient features is defined as the distribution of pixels on the line joining the features. The distance between the features is normalized and pixel's color values are collected in a histogram of 100 bins, which is smoothed along each color channel: $C^{new}_n = \alpha C_n + (1-\alpha)C_{n-1}$, where the smoothed value in a bin is a function of the value in previous bin and raw value in the bin. The effect of raw data is controlled by α, while the coarse representation (100 bins) provides computational efficiency. The N gradient features in the ROI are sorted based on distance from the center of the ROI: $\{d_1, ..., d_{k-1}, d_k, d_{k+1}, ...d_N\}, \forall i < j,\ d_i < d_j$. The local neighborhood of each feature includes the four closest neighbors. The object model includes the connection potentials and a undirected graph [16] of local neighborhoods of connection potentials.

3) Parts-based Models of Image Segments (PIS): The third component uses a graph-based segmentation algorithm to extract segments from the ROI such that RGB values within a segment are similar and significantly different from pixels in neighboring segments [17]. Valid segments are modeled as 2D Gaussians that represent spatial locations in the ROI: $\mathcal{N}(\mu_k, \Sigma_k), k = 1, ..., M$ and constitute "parts" of the object. Each pixel n in the ROI is assigned membership in one of M parts based on Gaussian density functions of the parts: $\text{argmax}_j\, p(n\,|\mu_j, \Sigma_j)$. Then, each pixel's similarity with pixels in the same part and dissimilarity with pixels in neighboring parts are computed, weighted by the probability that these pixels belong to the same part or different parts. Similarity and dissimilarity measures for each part ($PartSimM_k$, $PartDiffM_k$) are defined as the logarithm of sum of contributions of all pixels in that part. To capture local variations in part positions, the envelope around the extracted parts is displaced a few times and the corresponding values of $PartSimM$ and $PartDiffM$ are modeled as gamma (Γ) distributions for each part. The object model includes image segments, parts-based model, and these similarity and dissimilarity measures.

4) Color Distribution Statistics (CDS): The fourth component captures color information [15]. The ROI's pixels are used to learn normalized histograms (pdfs) in the HSV color space. Each pdf is learned in (h, v) with ten bins in each dimension. Since color distributions are not a unique representation, distances are computed between every pair of pdfs, using the Jensen-Shannon (JS) measure [18]. The fourth component consists of the pdfs and incrementally-learned distribution of distances between the pdfs.

5) Gaussian Mixture Model of Context (GMC): The fifth component models the object's *local context* using image segments (extracted for PIS) that share a boundary with the ROI, i.e., segments within the red rectangle but outside the yellow boundary in Figure 1. The pixels in each such segment are used to learn a 2D Gaussian in the normalized HSV color space (using h, v). The relative spatial arrangement of each segment with respect to the ROI is used to assign labels "on", "under" and "beside" to the segment; segments can have more than one label. Segments with the same label are used to learn a Gaussian mixture model (GMM), e.g., each of the K 2D Gaussians with label "on" is assigned a mixing factor π_k that

is the ratio of number of pixels in the corresponding segment divided by the number of pixels in all K segments. Each GMM is also assigned a weight that is the ratio of number of pixels in segments with the corresponding label to the number of pixels in all segments used to model context. The object model includes GMMs, and their relative positions and sizes with respect to the ROI's center and size.

B. Information Fusion for Recognition

The learned models are used for object recognition in images of novel scenes, *irrespective of whether the objects are stationary or moving*. Energy minimization is used to select ROIs in test images, and generative models merge evidence from components of learned models. Consider the analysis of a specific test image ROI using a specific object model.

1) SCG-Based Matching: The SCVs of gradient features in the learned model and the matched features in the test image ROI are used to obtain the probability of occurrence of corresponding object in the ROI:

$$p_{scg} = \frac{x_{correct} + y_{correct}}{2 * M}, \quad p_{scg} \in [0,1] \tag{1}$$

$$x_{correct} = \sum_{m=1}^{M} \frac{Nx_{m\_correct}}{N-1},\ y_{correct} = \sum_{m=1}^{M} \frac{Ny_{m\_correct}}{N-1}$$

where $Nx_{m\_correct}$ and $Ny_{m\_correct}$ are the number of values in the ROI's SCV that match the learned model's SCV along x and y axes respectively; M and N are the number of gradient features in the learned model and ROI respectively. This computation is repeated with each learned object model.

2) GCP-Based Matching: Next, the neighborhood of connection potentials between features in the learned model is compared with the neighborhood of connection potentials between matched ROI features. The similarity between two connection potentials i and j is:

$$con(i,j) = \sum_{n=1}^{100} f(C_n^i, C_n^j),\ \ f(a,b) = \begin{cases} 1 & |a-b| > \beta \\ 0 & otherwise \end{cases}$$

where parameter β is used to identify significant changes in entries of connection potentials. The probability of occurrence of the learned object in the ROI is:

$$p_{gcp} = \frac{1}{Z} \sum_{k \in \{1,\ldots,M\}} \sum_{i \in N_k, j \in N_{k_m}} con(i,j) \tag{2}$$

where M gradient features in the object model match features in the ROI, N_{k_m} and N_k are the connected neighborhoods of feature k_m and matched feature k in the object model and ROI respectively, and Z is a normalizer. This computation is repeated with each learned object model.

3) PIS-based Matching: Next, different relative arrangements of the learned model's parts are compared with pixels in the test image ROI. For pixels in the overlapping regions (for any arrangement), the similarity of pixels within a learned model part and the dissimilarity of pixels in neighboring parts are computed. The learned Γ distributions of these measures (for each part) compute the likelihood of this arrangement:

$$p_{pis} = \sum_j \{w_j \cdot f(PartSimM_j) \cdot f(PartDiffM_j)\}$$

$$f(x_j) = \Gamma\Big(|\overline{x_j} - x_j| - (k-1)\theta, k, \theta\Big) \tag{3}$$

where, for the learned object's j^{th} part, $(k-1)\theta$ is the stationary point of the learned Γ pdf, x_j is the similarity or dissimilarity computed using ROI pixels in the part ($PartSimM_j$, $PartDiffM_j$), and $\overline{x_j}$ is the mean of the Γ pdf. The match probability of this arrangement is the sum of product of these measures for each part, weighted (w_j) by the ratio of number of ROI pixels in a part and number of ROI pixels in all parts of object model. The arrangement that maximizes p_{pis} is chosen. This computation is repeated with each learned object model.

4) CDS-Based Matching: Next, the average distance d_{avg} is computed between the ROI's color space pdf and the pdfs in the learned object model, using the JS measure. A comparison with the expected (Gaussian) distribution of distances (in the object model) provides the probability of occurrence of the learned object (p_{cds}). This computation is repeated with each learned object model. When second-order statistics of object models are being learned, relative values of average distances between the ROI's color space pdf and learned pdfs of object models are used to obtain the probability of occurrence of learned objects in the ROI.

5) GMC-Based Matching: Next, each GMM in the learned model is scaled and positioned with respect to the ROI. A matching score is computed using each GMM, considering the pixels around the convex boundary of the ROI that fall within the spatial scope of the GMM (N_{lbc}). The probability of occurrence of the learned object is the weighted sum of individual scores:

$$p_{gmc} = \sum_{lbc \in \{on,under,beside\}} w_{lbc} \cdot \Gamma\Big(f(\boldsymbol{x}_{lbc}), k, \theta\Big)$$

$$f(\boldsymbol{x}_{lbc}) = \frac{1}{N_{lbc}} \sum_{l=1}^{N_{lbc}} \sum_{j=1}^{N_{lbc}^{gmm}} \pi_j\, e^{-\frac{1}{2}(\boldsymbol{x}_l - \boldsymbol{\mu}_j)^T \Sigma_j^{-1} (\boldsymbol{x}_l - \boldsymbol{\mu}_j)} \tag{4}$$

where N_{lbc}^{gmm} is the number of 2D Gaussians in the GMM with label $lbc \in \{on, under, beside\}$. Each ROI pixel $\boldsymbol{x}$ is a 2D vector in the normalized (h,v) space. The value of $f(\boldsymbol{x}_{lbc})$ is scaled by a Γ distribution and weighted (w_{lbc}) by the ratio of number of pixels that fall within the corresponding GMM and number of pixels that fall within all GMMs in the learned model—π_j, $\boldsymbol{\mu}_j$ and Σ_j are obtained from the learned model. This computation is repeated with each learned object model.

6) Information Fusion: For ease of explanation, assume that any ROI contains no more than one object—*the algorithm can detect multiple objects in an image or ROI*. If a test image sequence contains a moving object, the corresponding ROI is identified by (as during learning) tracking and clustering gradient features; the probability of occurrence of a learned object in this ROI is then the product of probabilities provided by components of the object model.

When test images are individual snapshots of objects, ROIs are identified by matching gradient features in the images with gradient features in the object models, e.g., to compute the probability of occurrence of the i^{th} learned object in a test image, K nearest neighbors are found in the test image for each of the M gradient features in the learned model. Candidate ROIs are created by selecting M matched features in the test image from the (at most) $K * M$ features, using the iterated conditional modes (ICM) energy minimization algorithm [19]. Since this algorithm can be sensitive to the

choice of initial estimates in high-dimensional spaces, the nearest neighbors of the learned object's gradient features provide the initial ROI estimate. For a set of M matched features, the probability of occurrence of the i^{th} learned object (p_{O_i}) considers evidence from each feature:

$$p_{O_i} = \prod_{j \in \{1,...,M\}} p\Big(g_j | O_i, \{g_n | n = 1, ..., M, n \neq j\}\Big) = \prod_{j \in \{1,...,M\}} p(g_j | O_i) \quad (5)$$

where $\{g_n | n = 1, ..., M, n \neq j\}$ is the subset of M matched gradient features excluding the j^{th} feature under consideration. This term is dropped in the following equations since this information is always available. The probability that each matched feature comes from object O_i is formulated as a generative model over components of the object model:

$$p(g_j | O_i) = \sum_{Lbg_j \in \{fg,bg\}} p(g_j | Lb_{g_j}, O_i) \cdot p(Lb_{g_j} | O_i) \quad (6)$$

where $Lb_{g_j} \in \{fg, bg\}$ indicates whether the j^{th} feature belongs to the foreground (i.e., part of the target object) or the background (i.e., not part of the target).

When specific labels (fg, bg) are assigned to candidate matched features, the ROI is defined by minimal convex set containing the foreground features. Generative models thus consider multiple arrangements to refine the initial choice made by feature matching and energy minimization. Equation 6 is decomposed using the independence relationships:

$$p(g_j | O_i) = \sum_{Lbg_j \in \{fg,bg\}} p(g_j | Lb_{g_j}, O_i) \cdot p(Lb_{g_j} | O_i) \quad (7)$$
$$= \sum_{Lb_{g_j} \in \{fg,bg\}} p(g_j | Lb_{g_j}, scg_{O_i}) \cdot p(g_j | Lb_{g_j}, gcp_{O_i}) \cdot p(Lb_{g_j} | pis_{O_i}) \cdot p(Lb_{g_j} | cds_{O_i}) \cdot p(Lb_{g_j} | gmc_{O_i})$$

Since parts-based models (PIS), color statistics (CDS) and context-based models (GMC) capture visual cues that are not evaluated based on relative arrangements of local cues, they are used to evaluate the relative likelihoods of labels (fg, bg) for the feature under consideration. The other components, i.e., SCG and GCP, evaluate the probability of occurrence of the gradient features given the specific labels. The probabilities in Equation 7 are provided by Equations 1-4 and these independence assumptions work well in practice. The ROI that maximizes Equation 7 and thus Equation 5 is the best estimate of the corresponding object's location in the test image.

Finally, the probability distribution of occurrence of the L learned objects in a test image ROI is normalized: $\overline{p}_{O_i}, i \in [1, L]$ and used to recognize objects, and to detect novel objects when none of the learned objects has a match probability significantly larger than others. The robot thus concurrently and incrementally learns object models and recognizes objects while revising the domain map and planning navigation.

IV. Experimental Setup and Results

This section describes the robot platform and the experimental results of evaluating our algorithm.

A. Robot Platform

The robot platform is a $40cm \times 41cm \times 15cm$ wheeled base equipped with a stereo camera, monocular camera, laser range finder and pan-tilt unit. The experiments used 640×480 images from one of the cameras of the stereo unit. Input from the laser range finder is used to learn the domain map. All experiments were performed on-board using a 2GHz processor and 1GB RAM. Trials were conducted in indoor and outdoor settings.

B. Experimental Setup and Results

Experimental trials used 20 object categories, with separate models learned for different objects in a category, e.g., different boxes or books, resulting in 60 subcategories. Objects were placed in complex backgrounds that made learning and recognition challenging. Some objects (e.g., humans and cars) moved on their own, while some (e.g., boxes) were moved on trolleys. It is difficult to obtain an image database of objects with well-defined motion. Experiments used ≈ 2000 images, including short sequences and snapshots, ≈ 700 of which were captured by the robot. To establish applicability to different domains, ≈ 1300 images of motorbikes, buses, some cars and airplanes were chosen from the *Pascal VOC2006* and *Caltech-256* benchmark datasets, which include ROIs for objects in the images—suitable ROIs and neighboring segments were selected when these images were used for learning object models. Each object model is learned from $3-8$ images, with ≈ 250 images used for learning all object models; remaining images are used for testing. The robot processes $3-5$ frames/second to identify ROIs, learn models and recognize objects while performing other operations. The images used for learning and recognition were chosen randomly (in repeated trials) to obtain the results below.

The average classification accuracy over all 60 subcategories is: 0.8860 ± 0.0432, which is promising given the small number of images used for learning. Table I shows accuracy for a subset of (ten) object categories, averaged over subcategories in each category; off-diagonal terms represent errors. Accurate classification requires an object to be matched to the correct model—matching an object in *car-class1* to model *car-class2* is an error. One reason for errors is the learning of object models with non-unique features, e.g., long shots of humans cause features to be extracted from clothes, resulting in non-unique object models and lower recognition accuracy. Some errors correspond to an insufficient number of test image features being matched with the learned models due to occlusions, motion blur or a large difference in viewpoint. Revision of object models over time further improves recognition accuracy. Some errors also occur when test image ROIs are assigned the label of the object model with the maximum match probability, even if this value is similar to match probabilities of other objects—these errors are eliminated by requiring that the maximum match probability be substantially higher than match probabilities of other object classes. Furthermore, errors are less frequent in sequences of objects in motion because correctly identifying the ROI enables some subset of the components of learned models to provide high match probabilities for the appropriate object.

Our algorithm and existing vision algorithms have disparate objectives; our algorithm efficiently learns representative models of relevant objects using $3-8$ images (each),

Fig. 3. Robot recognizes objects from different categories, multiple objects and multiple instances of an object in cluttered backgrounds. Last column shows an incorrect envelope (top) and an incorrect classification due to occlusion (bottom).

	Box	Car	Human	Robot	Book	Airplane	Bus	Motorbike	Fire Truck	Firehydrant
Box	**0.941**	0	0.017	0.025	0	0	0	0	0	0.017
Car	0.010	**0.917**	0	0.021	0	0	0	0.042	0	0.010
Human	0.080	0.024	**0.820**	0.060	0.016	0	0	0	0	0
Robot	0.027	0	0.042	**0.899**	0.027	0	0	0.005	0	0
Book	0.016	0	0	0.042	**0.942**	0	0	0	0	0
Airplane	0.029	0.051	0	0.023	0.009	**0.888**	0	0	0	0
Bus	0	0	0	0	0	0	**0.856**	0.036	0.108	0
Motorbike	0	0.073	0	0.010	0.016	0	0.062	**0.839**	0	0
Fire Truck	0	0.032	0	0	0	0	0.080	0.016	**0.872**	0
Firehydrant	0.029	0.029	0	0	0	0	0	0	0.058	**0.884**

TABLE I. RECOGNITION ACCURACY AVERAGED OVER DIFFERENT MODELS (I.E., SUBCATEGORIES) IN A SUBSET OF (TEN) OBJECT CATEGORIES.

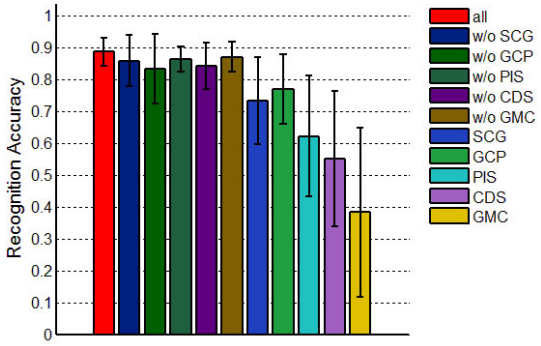

Fig. 4. Our algorithm provides higher accuracy than any individual component or any four of the components; *results are statistically significant*.

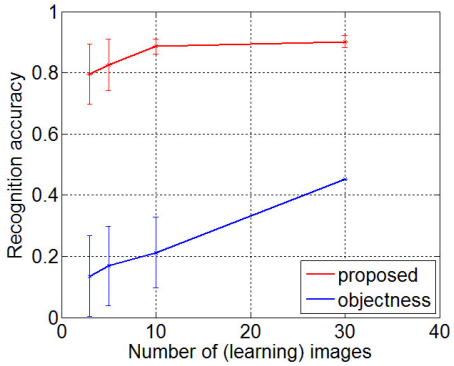

Fig. 5. Our algorithm provides higher accuracy than the *objectness* measure using a much smaller number of image for learning the object models.

while existing algorithms typically focus on modeling a large number of objects using a much larger number of images of each object. Although finding a common frame of reference is challenging, the following experiments were conducted.

When we increase the number of images used for learning, the recognition accuracy increases, e.g., 0.90 ± 0.05 with 400 images (total) for learning, and approaches reported accuracies of state of the art algorithms on benchmark datasets. However, existing algorithms are much more (computationally) expensive for learning and/or recognition, and only some algorithms support incremental learning. Furthermore, it is difficult for existing algorithms to learn good models from a small number of images because they do not fully exploit the complementary strengths of (and dependencies between) different cues.

Next, Figure 4 compares the average recognition accuracy of our algorithm with that of each component and different subsets of four components. None of the individual components provide high recognition accuracy and there is large variance, especially with components that primarily use color. At the same time, each component contributes to the overall accuracy—the accuracy of our algorithm is better than that of different subsets of four components. These results indicate that although each component uses visual cues widely used by other algorithms, our representation exploits their complementary strengths to learn representative object models. Figure 3 shows examples of the robot recognizing objects from different categories, and multiple objects or multiple instances of objects in different scenes. The last column of Figure 3 also shows an instance where (a) top: the object boundary is incorrect (although object label is correct) due to incorrectly matched features; and (b) bottom: occlusion leads to incorrect classification, e.g., object of *bus-class1* matched with *car-class2*. We hypothesize that including a component that matches shapes will minimize these errors; the computational efficiency of our algorithm supports the inclusion of such components.

We also compared the recognition accuracy and efficiency of our algorithm with state of the art algorithms that use gradient features, e.g., SURF [20] and BRIEF [21]. SURF and BRIEF were provided images with labeled ROIs to learn object models, which were used for recognition. During learning, these algorithms extract local gradient features from the ROIs to create suitable object models. During recognition, features in the learned models are matched with features extracted in the test images. Table II shows that our algorithm provides much higher accuracy than these algorithms, primarily because it exploits the complementary strengths and dependencies between local, global, temporal and contextual visual cues. The use of multiple components does increase the computational cost—see Table III. We believe that this trade-off is justified

	Box	Car	Human	Robot	Book	Airplane	Bus	Motorbike	Fire Truck	Firehydrant
Proposed	**0.941**	**0.917**	**0.820**	**0.899**	**0.942**	**0.888**	**0.856**	**0.839**	**0.872**	**0.884**
SURF	0.804	0.784	0.706	0.822	0.832	0.742	0.713	0.772	0.754	0.793
BRIEF	0.843	0.822	0.743	0.855	0.843	0.772	0.733	0.813	0.782	0.834

TABLE II. OUR ALGORITHM PROVIDES HIGHER ACCURACY THAN SURF AND BRIEF USING THE SAME NUMBER OF IMAGES FOR LEARNING OBJECT MODELS.

	SURF	BRIEF	**Proposed**	Objectness
Learning	0.1	0.005	**0.3**	360
Testing	0.12	0.01	**0.25**	5

TABLE III. COMPUTATION TIME IN SECONDS TO PROCESS ONE IMAGE.

since it supports incremental learning of representative object models from a small number of images.

Finally, we compared our algorithm with the algorithm based on the *objectness* measure, which automatically identifies image windows containing objects from desired classes [13]. Compared with the objectness-based algorithm, our algorithm is significantly more efficient—see last column in Table III. Figure 5 compares the recognition accuracy of the two algorithms as a function of the number of images used for learning object models. Our algorithm provides much better recognition accuracy using a much smaller number of images because the objectness measure-based algorithm does not fully exploit all visual cues.

V. CONCLUSIONS AND FUTURE WORK

This paper described an algorithm for robots to identify interesting objects based on motion cues, automatically and efficiently learning representative models of these objects using appearance-based and contextual visual cues extracted from a small number of images. The learned models support reliable and efficient object recognition in novel scenes.

The images used in the experimental trials reported in this paper had a small set of moving objects in any given image. Future research will investigate the extension to image sequences with many moving objects and consider images with substantial occlusions. In parallel, computational efficiency will be improved by using sampling-based algorithms and better energy minimization algorithms [16], [22]. Furthermore, we will develop algorithms that automatically determine the most informative subset of components to represent each object. The long-term goal is to enable robots to automatically and incrementally learn object models with minimal human supervision in complex application domains.

ACKNOWLEDGMENT

This work was supported in part by the ONR Science of Autonomy award N00014-09-1-0658.

REFERENCES

[1] K. Mikolajczyk and C. Schmid, "Scale and Affine Invariant Interest Point Detectors," *International Journal of Computer Vision*, vol. 60, no. 1, pp. 63–86, 2004.

[2] R. Fergus, P. Perona, and A. Zisserman, "Object Class Recognition by Unsupervised Scale-Invariant Learning," in *International Conference on Computer Vision and Pattern Recognition*, 2003, pp. 264–271.

[3] B. Siddiquie and A. Gupta, "Beyond Active Noun Tagging: Modeling Contextual Interactions for Multi-class Active Learning," in *Computer Vision and Pattern Recognition*, 2010, pp. 2979–2986.

[4] J. Shotton, J. M. Winn, C. Rother, and A. Criminisi, "TextonBoost: Joint Appearance, Shape and Context Modeling for Multi-class Object Recognition and Segmentation," in *European Conference on Computer Vision*, 2006, pp. 1–15.

[5] C. Li, D. Parikh, and T. Chen, "Extracting Adaptive Contextual Cues from Unlabeled Regions," in *International Conference on Computer Vision*, 2011.

[6] A. Murarka, M. Sridharan, and B. Kuipers, "Detecting Obstacles and Drop-offs using Stereo and Motion Cues for Safe Local Motion," in *International Conference on Intelligent Robots and Systems*, 2008.

[7] B. Rosman and S. Ramamoorthy, "Learning Spatial Relationships Between Objects," *International Journal of Robotics Research, Semantic Perception for Robots in Indoor Environments*, vol. 30, no. 11, pp. 1328–1342, 2011.

[8] D. Schiebener, A. Ude, J. Morimotot, T. Asfour, and R. Dillmann, "Segmentation and Learning of Unknown Objects through Physical Interaction," in *International Conference on Humanoid Robots*, 2011.

[9] C. Li, D. Parikh, and T. Chen, "Automatic Discovery of Groups of Objects for Scene Understanding," in *International Conference on Computer Vision and Pattern Recognition*, June 16-21, 2012.

[10] D. Parikh, C. L. Zitnick, and T. Chen, "Unsupervised Learning of Hierarchical Spatial Structures in Images," in *Computer Vision and Pattern Recognition*, 2009.

[11] B. Ommer and J. Buhmann, "Learning the compositional nature of visual object categories for recognition," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 3, pp. 501–516, 2010.

[12] 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," *Computer Vision and Image Understanding*, vol. 106, no. 1, pp. 59–70, 2007.

[13] B. Alexe, T. Deselaers, and V. Ferrari, "Measuring the objectness of image windows," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 11, pp. 2189–2202, Nov.2012.

[14] X. Li, M. Sridharan, and C. Meador, "Extended Abstract: Autonomous Learning of Visual Object Models on a Robot Using Context and Appearance Cues," in *International Conference on Autonomous Agents and Multiagent Systems*, Saint Paul, USA, May 6-10, 2013.

[15] X. Li and M. Sridharan, "Autonomous Learning of Object Models on a Mobile Robot using Visual Cues," in *International Conference on Robotics and Automation*, 2011.

[16] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2010.

[17] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient Graph-Based Image Segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167–181, 2004.

[18] T. M. Cover and J. A. Thomas, *Elements of Information Theory, 2nd Edition*. Wiley-Interscience, 2006.

[19] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother, "A Comparative Study of Energy Minimization Methods for Markov Random Fields with Smoothness-Based Priors," *Pattern Analysis and Machine Intelligence*, vol. 30, no. 6, 2008.

[20] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[21] M. Calonder, V. Lepetit, C.Strecha, and P. Fua, "BRIEF: Binary Robust Independent Elementary Features," in *European Conference on Computer Vision*, 2010.

[22] V. Kolmogorov, "Convergent Tree-Reweighted Message Passing for Energy Minimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, pp. 1568–1583, 2006.