Key Object Driven Multi-category Object Recognition, Localization and Tracking Using Spatio-temporal Context

Yuan Li and Ram Nevatia

University of Southern California
Institute for Robotics and Intelligent Systems
Los Angeles, CA, USA
{yli8,nevatia}@usc.edu

Abstract. In this paper we address the problem of recognizing, localizing and tracking multiple objects of different categories in meeting room videos. Difficulties such as lack of detail and multi-object co-occurrence make it hard to directly apply traditional object recognition methods. Under such circumstances, we show that incorporating object-level spatio-temporal relationships can lead to significant improvements in inference of object category and state. Contextual relationships are modeled by a dynamic Markov random field, in which recognition, localization and tracking are done simultaneously. Further, we define human as the *key object* of the scene, which can be detected relatively robustly and therefore is used to guide the inference of other objects. Experiments are done on the CHIL meeting video corpus. Performance is evaluated in terms of object detection and false alarm rates, object recognition confusion matrix and pixel-level accuracy of object segmentation.

1 Introduction

Object recognition is a fundamental problem of computer vision. Its significance lies not only in the static image domain but also in video understanding and analysis, *e.g.*, is the man typing on a laptop or writing on a pad of paper? What objects have been put on the table and where are they? What is the motion of the passenger and his luggage if he is carrying any? Answering questions of this kind requires the ability to recognize, localize and even track different categories of objects from videos captured with a camera of usually broad view field.

There are a number of difficulties in this problem: background clutter, lack of image detail, occlusion, multi-object co-occurrence and motion. To enhance purely appearance-based approaches in the hope of overcoming these difficulties, we incorporate contextual information to aid object recognition and localization. There are three key notions in our approach: 1) spatial relationships between different object categories are utilized so that co-inference helps enhance accuracy; 2) temporal context is utilized to accumulate object evidence and to track objects continuously; 3) we borrow techniques from research efforts in single category object recognition to robustly detect *key objects* (such as humans) and use them to reduce inference space for other objects.

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part IV, LNCS 5305, pp. 409–422, 2008.

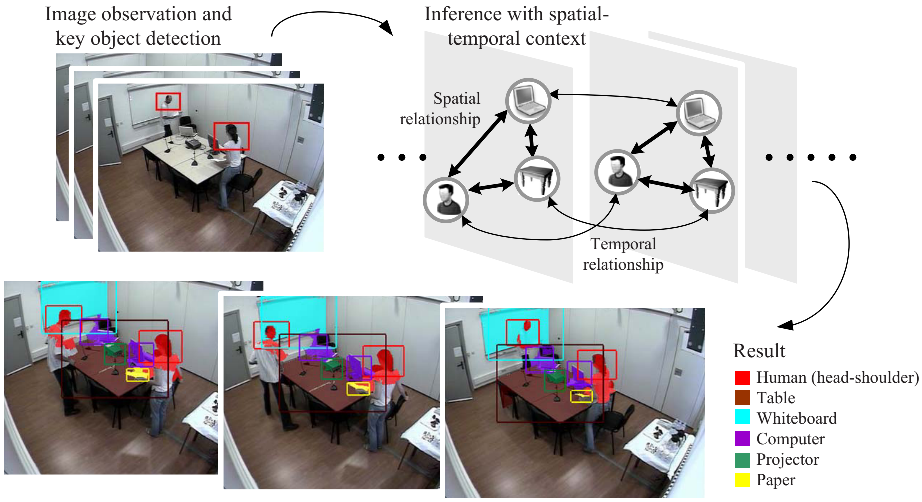


Fig. 1. Finding objects in spatio-temporal context

These concepts are modeled by a dynamic Markov random field (MRF). Figure 1 gives an illustration. Instead of letting each node represent a pixel or image blob in a pre-defined grid, as is commonly done in segmentation, in our model a node represents a hypothetical object in one frame, which enables integration of object-level information during inference. Spatial and temporal relationships are modeled by intra-frame and inter-frame edges respectively. Since objects are recognized on-the-fly and change with time, the structure of the MRF is also dynamic. To avoid building an MRF with excessive false hypothetical object nodes, key objects are detected first and provide contextual guidance for finding other objects. Inference over the resulting MRF gives an estimate of the states of all objects through the sequence. We apply our approach to meeting room scenes with humans as the key objects.

The rest of the paper is organized as follows: Section 2 summarizes related work by categories; Section 3 gives the formulation of the model; Section 4 defines the potential functions of the MRF and Section 5 describes the inference algorithm; Section 6 shows the experimental results; Section 7 discusses about future work and concludes the paper.

2 Related Work

Our approach uses elements from both object recognition and detection. Object recognition focuses on categorization of objects [1][2]; many approaches assume a close-up view of a single object in the input image. Object detection focuses on single category object classification and localization from the background [3][4][5]. Both have received intense research interest recently, bringing forward a large body of literature. While our approach assimilates several established ideas from the two, our emphasis is on integration of spatio-temporal context. We hereby focus on the recent growing effort in tackling object-related problems based on contextual relationships.

Object in the scene. Modeling object-scene relationship enables the use of prior knowledge regarding object category, position, scale and appearance. [6] learns a scene-specific prior distribution of the reference position of each object class to improve classification accuracy of image features. It assumes that a single reference position explains all observed features. [7] proposes a framework for placing local object detection in the 3D scene geometry of an image. Some other work seeks to classify the scene and objects at the same time [8][9]. [8] uses the recognized scene to provide strong prior of object position and scale. Inter-object relationship is not considered. [9] proposes an approach to recognize events and label semantic regions in images, but the focus is not on localizing individual objects.

Object categorization and segmentation in context. Object segmentation and categorization are often combined to enhance each other. When multiple categories are present, contextual knowledge fits in naturally [10][11][12]. [10] uses Conditional Random Field (CRF) to combine appearance, shape and context. *Shape filters* are used to classify each pixel, based on the appearance of a neighborhood; no object-level relationship is explicitly modeled. By counting the co-occurrence of every object pair, [11] exploits object-level context to refine the category label after each image segment has been categorized independently. While [11] does not model spatial relationship among objects, [12] captures spatial relationship by laying a grid-structured MRF on the image, with each node corresponding to the label of a rectangular image blob. Labeling of one blob is dependent on the labels of its neighbors. However, such relationship is constrained to adjacent image blobs.

Object and human action. There have been several attempts in collaborative recognition of object category and human action [13][14][15]. [13] uses the hand motion to improve the shape-based object classification from the top-down view of a desktop. In [14], objects such as chair, keyboards are recognized from surveillance video of an office scene. Bayesian classification of regions is done completely based on human pose and action signatures. Given estimated human upper body pose, [15] accomplishes human action segmentation and object recognition at the same time. All these approaches require the ability of tracking human poses or recognizing action, which is not a trivial task. But they have reflected the fact that many visual tasks are human centered. Namely the objects of most interest for recognition are those interacting closely with humans. This is also our motivation to choose human as the key object in our framework.

3 Model and Representation

In our approach, a dynamic MRF (Figure 2) is designed to integrate the relationship between the object state and its observation, the spatial relationships between objects, as well as the temporal relationships between the states of one object in successive frames. The MRF has the structure of an undirected graph $\mathcal{G}$, with a set of nodes $\mathcal{V}$ and a set of edges $\mathcal{E}$. Each node $v \in \mathcal{V}$ is associated with an unobserved state variable x_v and a observation y_v. Since we are considering a temporal sequence, each node belongs to exactly one time frame t.

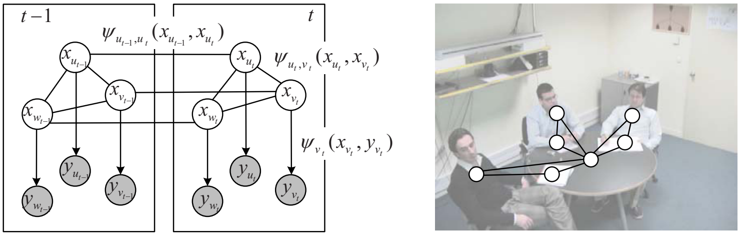


Fig. 2. The MRF defined in our problem (left) and an ideal graph structure for one input frame (right). Section 5 explains how to build such a graph.

We use a node v_t to represent a hypothetical object instance in frame t. Define

$$x_{v_t} = (c_{v_t}, p_{v_t}, s_{v_t}) \tag{1}$$

as the state of the object, where c_{v_t} stands for the object's category label, p_{v_t} for coordinates of its centroid and s_{v_t} for the logarithm of size[1]. y_{v_t} is defined as the image evidence of the object. There are two types of edges: intra-frame edges that represent the spatial relationships between different objects, and inter-frame edges that represent the temporal relationships between states of the same object in adjacent frames. Let the potential functions be pairwise, in which case the distribution of the MRF factorizes as

$$p(\mathbf{x}, \mathbf{y}) = \frac{1}{Z} \prod_{(v,u)\in\mathcal{E}} \psi_{v,u}(x_v, x_u) \prod_{v\in\mathcal{V}} \psi_v(x_v, y_v), \tag{2}$$

where $\mathbf{x} = \{x_v | v \in \mathcal{V}\}$ and $\mathbf{y} = \{y_v | v \in \mathcal{V}\}$, $\psi_{v,u}(x_v, x_u)$ models the spatio-temporal relationship, and $\psi_v(x_v, y_v)$ models the image observation likelihood. Given the structure of the MRF and the potential functions, the states of the objects can be inferred.

Note that rather than letting each node correspond to an image blob in a pre-defined image grid or a pixel, as is commonly done in segmentation literature [12][10], we let each node represent an object, which is similar to some tracking frameworks such as the Markov chain in Particle Filtering and MRF in collaborative tracking proposed by [16]. The reason is twofold: 1) object-based graph enables us to use object-level information during inference, while pixel- or grid-based graph can only model inter-object relationships locally along the boundary of objects; 2) object-based graph has fewer nodes and therefore the complexity of inference is much lower. However, one drawback of object-based graph is that accurate segmentation cannot be directly obtained. One new property of the object-based graph is that its structure is dynamic. In Section 5 we show that the nodes for new objects can be added to the graph online driven by detected key objects. Before that we first give our models for the potential functions.

4 Potential Functions

There are three types of edges in our model, each associated with one kind of potential function representing a specific semantic meaning.

[1] Logarithm is used because scale is multiplicative.

4.1 Observation Potential $\psi_v(x_v, y_v)$

We use two sources of observation evidence. The first is a single-category object detector for the *key objects*. For meeting room applications, we implement a patch-based cascade human upper body detector following the method in [17]. Let c^* stand for the category label of key objects, for each $x_v = (c_v, p_v, s_v)$ with $c_v = c^*$, we define the observation potential to be the likelihood output of the detector: $\psi_v(x_v, y_v) = p(c^*|x_v, y_v) = p(human|p_v, s_v)$. Please refer to [18] for deriving probability of an object class from a boosted classifier.

The second source of observation potential function targets all object categories of interest. We build our object classifier based on the Bag of Features [1] approach and combine it with image region. The motivation of our choice is the proven performance of Bag of Feature and the suggestion in recent literature that classification based on image segments provides better spatial coherence [2][11][14]. These ideas are tailored to our needs. Specifically, interest points are detected with the DoG and Harris corner detectors and at each interest point a 128d SIFT feature is extracted. During training, these features are used to build a code book by clustering. Also every input image is over-segmented by Mean Shift [19]; each segment is associated with interest points. Base on both the point features and the statistics of pixel intensity of the segments, a classifier is built to model $p(c|r_i)$, defined as the likelihood of any given segment r_i belonging to category c. This could be done by standard Bag of Feature categorization, or more sophisticated generative models such as [2]. We build a discriminative model by using AdaBoost to select and weight features from the code book. Given $p(c|r_i)$ for any segment r_i, the observation potential of object v is modeled as:

$$\psi_v(x_v, y_v) = \frac{\sum_{r_i \in \mathcal{R}(x_v)} p(c_v|r_i)\zeta(r_i, x_v)}{\sum_{r_i \in \mathcal{R}(x_v)} \zeta(r_i, x_v)}, \tag{3}$$

where $\mathcal{R}(x_v)$ stands for the set of segments that is associated with the object v; $\zeta(r_i, x_v)$ is a position weight for r_i, which allows the use of object shape prior. In our implementation we let $\mathcal{R}(x_v)$ include all segments which has at least 50% of its area within v's bounding box, and $\zeta(r_i, x_v)$ is defined as a Gaussian centered at p_v. Figure 3 shows an example for the category *paper*. We can see that it is hard to distinguish the paper from a bright computer screen or the white board by appearance (feature point and region).

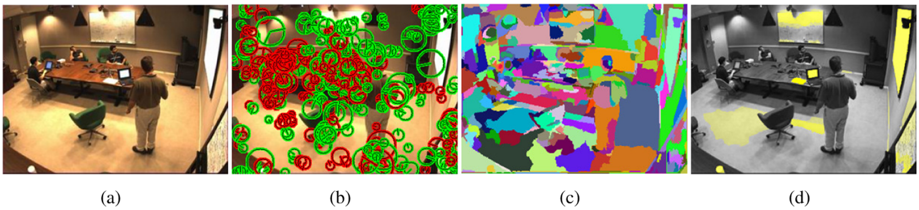


Fig. 3. An example of finding paper based on appearance. (a) Input image; (b) SIFT features (green: feature with positive weight in the classifier, red: feature with negative weight); (c) Segmentation; (d) observation likelihood $p(paper|r_i)$ for each region r_i (yellow: high likelihood).

Note that the observation potential here can be substituted by any object recognition method, possibly with a more complicated model and higher accuracy such as [2]. Here we do not elaborate on this since our emphasis is on the effect of introducing contextual relationship.

4.2 Spatial Potential $\psi_{v_t,u_t}(x_{v_t}, x_{u_t})$

Spatial potential function ψ_{v_t,u_t} is defined on edges between nodes within one frame but of different object categories. The purpose is to model inter-category correlation in terms of position and scale, *e.g.*, a person tends to sit on a chair beside a table and a laptop is often near a person and on a table. Such correlation generalizes well in our experience for the selected scenario.

When defining the form of the potential function, we want to avoid using very complicated models which introduce risk of over-fitting. In practice, we find that a single Gaussian function is sufficient for our amount of training data as well as the problem itself. Denote $N(\mu, \sigma, x)$ as a Gaussian function with mean μ, variance σ and x as the variable. Since nodes involved are from the same time frame, we suppress the subscript t in this subsection. Define

$$\begin{aligned}\psi_{v,u}(x_v, x_u) = N(\mu_p(c_u, c_v), \sigma_p(c_u, c_v), p_v - p_u) \cdot \\ N(\mu_s(c_u, c_v), \sigma_s(c_u, c_v), s_v - s_u),\end{aligned} \tag{4}$$

where $\mu_p(c_u, c_v)$, $\sigma_p(c_u, c_v)$, $\mu_s(c_u, c_v)$ and $\sigma_s(c_u, c_v)$ are the model parameters that describes the relative position and size of two object depending on their category labels c_u and c_v. It is ideal to learn them by maximizing the sum of log likelihoods of all training samples $\{\mathbf{x}^{(i)}\}$. However, this is difficult because $\mathbf{x}^{(i)}$s of different training samples may have different dimensionalities (number of objects varies) and the graph structures also differ. Therefore potential functions are learned independently for each kind of edge in a piecewise manner [20]. The number of different spatial potential functions is $n(n-1)/2$ for n categories. The parameters of the spatial potential function between the categories c_1 and c_2 can be easily learned by maximizing

$$l = \sum_j \log \psi_{v,u}(x_v^{(j)}, x_u^{(j)}), \tag{5}$$

where $\{(x_v^{(j)}, x_u^{(j)})\}$ is the set of all pairs of objects that co-exist in a training sample and satisfy $c_v = c_1$ and $c_u = c_2$.

4.3 Temporal Potential $\psi_{v_{t-1},v_t}(x_{v_{t-1}}, x_{v_t})$

To build the temporal potential function, feature points used in Section 4.1 are tracked by optical flow through frames. Let the positions of feature points associated with object v be $\{q_t^{(i)}\}_{i=1}^m$ at frame t, x_{v_t} can be estimated from x_{v_t-1} as:

$$\text{Position} : \tilde{p}_{v_t} = p_{v_{t-1}} + \frac{1}{m}\sum_{i=1}^{m}(q_t^{(i)} - q_{t-1}^{(i)}), \tag{6}$$

$$\text{Scale} : \tilde{s}_{v_t} = s_{v_{t-1}} + \log\left(\frac{\sum_{i=1}^m Dist(q_t^{(i)}, \tilde{p}_{v_t})}{\sum_{i=1}^m Dist(q_{t-1}^{(i)}, p_{v_{t-1}})}\right), \tag{7}$$

where $Dist(\cdot)$ is the distance between two points. The temporal potential is defined as a Gaussian distribution centered at the estimated position and scale with fixed variance:

$$\psi_{v_{t-1},v_t}(x_{v_{t-1}}, x_{v_t}) = N(\tilde{p}_{v_t}, \sigma_p, p_{v_t})N(\tilde{s}_{v_t}, \sigma_s, s_{v_t}). \tag{8}$$

5 Integration of Observation and Spatio-temporal Context

Given the graphical model defined above, there are two remaining issues in using it: how to build such a graph on-the-fly and how to do inference. We solve them in a unified manner by belief propagation (BP) [21][22]. *Augmenting nodes* are introduced as nodes that do not correspond to any specific object but are responsible for generating new object nodes by receiving belief messages from nodes of *key objects*. To distinguish *augmenting nodes* from the others, we refer to other nodes as *object nodes*. BP is then applied to compute the distribution $p(x_v|\mathbf{y})$ for all object nodes, from which the state of every object can be estimated. Since message passing in BP is essential for augmenting nodes, we first bring up the inference part and then introduce the augmenting nodes.

5.1 Inference

We choose BP as the inference method because of two main reasons. First, our graph has cycles and the structure is not fixed (due to addition and removal of object nodes, also inference is not done over the whole sequence but over a sliding window). Therefore it is inconvenient to use methods that require rebuilding the graph (such as the junction tree algorithm). While loopy BP is not guaranteed to converge to the true marginal, it has proven excellent empirical performance. Second, BP is based on local message passing and update, which is efficient and more importantly, gives us an explicit representation of the interrelationship between nodes (especially useful for the augmenting nodes).

At each iteration of BP, the message passing and update process is as follows. Define the neighborhood of a node $u \in \mathcal{V}$ as $\Gamma(u) = \{v|(u,v) \in \mathcal{E}\}$, each node u send a message to its neighbor $v \in \Gamma(u)$:

$$m_{u,v}(x_v) = \alpha \int_{x_u} \psi_{u,v}(x_u, x_v)\psi_u(x_u, y_u) \prod_{w \in \Gamma(u) \setminus v} m_{w,u}(x_u) dx_u. \tag{9}$$

The marginal distribution of each object v is estimated by

$$p(x_v|\mathbf{y}) = \alpha\psi_v(x_v, y_v) \prod_{u \in \Gamma(v)} m_{u,v}(x_v). \tag{10}$$

In our problem x_v is a continuous variable whose distribution is non-Gaussian and hard to represent in an analytical form; also, the observation potential function can only be evaluated in a point-wise manner. Therefore we resort to nonparametric version of the BP algorithm [22]. Messages are represented by a nonparametric kernel density estimate. More details of this method can be found in [22]. As a result, a weighted sample set is obtained to approximate the marginal distribution of each object node v: $\{x_v^{(i)}, \omega_v^{(i)}\}_{i=1}^M \sim p(x_v|\mathbf{y})$. The sample set is generated by importance sampling; namely sample $\{x_v^{(i)}\} \sim \prod_{u \in \Gamma(v)} m_{u,v}(x_v)$ and let $\omega_v^{(i)} = \psi_v(x_v^{(i)}, y_v)$. We can then estimate the state of object v (except its category label) by MMSE: $\hat{x}_v = \sum_{i=1}^{M} \omega_v^{(i)} x_v^{(i)} / \sum_{i=1}^{M} \omega_v^{(i)}$.

5.2 Augmenting Nodes

Augmenting nodes find new objects by receiving "hints" (messages) from key object nodes. It is reasonable because we are more interested in finding objects that are closely related to key objects; by combining inter-category spatial relationships with detection techniques specially developed for key objects, other objects can be detected and recognized more robustly and efficiently.

Let the set of key objects in one frame be K, and consider finding new objects of category $c \neq c^*$. The ideal way is: for every subset K' of K, make the hypothesis that there is a new object a which is in context with K'. Based on the NBP paradigm, we estimate a's state by $p(x_a|\mathbf{y}) \propto \psi_u(x_a, y_a) \prod_{v \in K'} m_{v,a}(x_a)$. The number of such hypotheses is exponential of $|K|$, so we simplify it by letting K' contain only one key object (it is reasonable because if a new object is hinted by a set of key objects it is at least hinted by one in some extent). In this case $K' = \{v\}$, the distribution of a's state is estimated as $p(x_a|\mathbf{y}) \propto \psi_a(x_a, y_a) m_{v,a}(x_a)$. This is done for each v in K, each result in a weighted sample set of a hypothetic new object's state.

Further, if two hypotheses of the same category are close in position and scale, they should be the same new object. So for each category, Agglomerative Clustering is done on the union of the $|K|$ sample sets to avoid creating duplicated nodes. For each

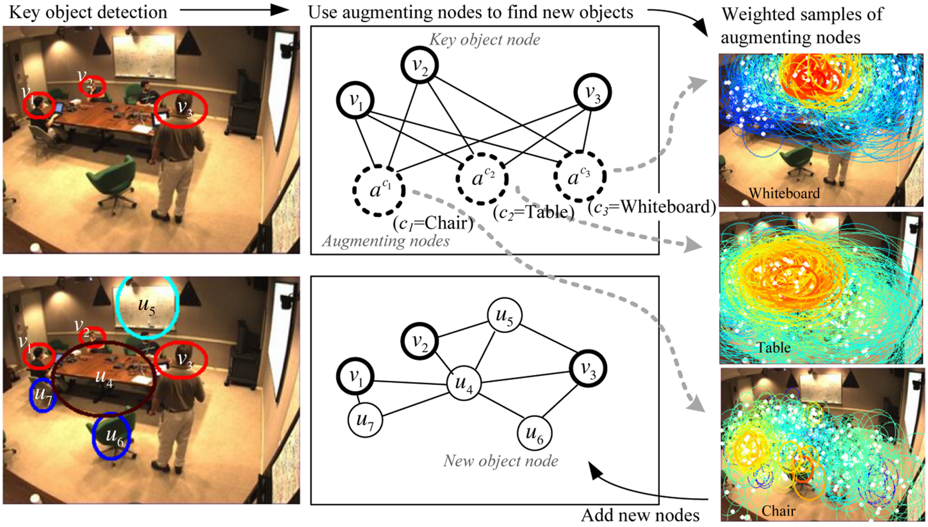


Fig. 4. Use of augmenting nodes to update graph structure. Augmenting nodes for each category are shown as one (dotted circle). For weighted samples, red indicates the highest possible weight, while blue indicates the lowest.

Table 1. Algorithm: inference over a sequence

Denote by $\mathcal{V}_t$ and $\mathcal{E}_t$ the sets of nodes and edges in frame t respectively.
With the graph $\mathcal{G}$ over a L-frame sliding window containing frame $(t - L)$ to $(t - 1)$, proceed as follows with the arrival of a new frame t:

- **Output** the estimated state $\hat{x}_v$ for each object node v of frame $(t - L)$. Remove sub-graph $(\mathcal{V}_{t-L}, \mathcal{E}_{t-L})$ from $\mathcal{G}$ and move the sliding window one frame forward.
- **Add** new sub-graph $(\mathcal{V}_t, \mathcal{E}_t)$ for frame t to $\mathcal{G}$ by algorithm in Table 2.
- **Inference**: perform the nonparametric BP algorithm over $\mathcal{G}$. For each object node v a weighted sample set is obtained: $\{x_v^{(i)}, \omega_v^{(i)}\}_{i=1}^M \sim p(x_v|\mathbf{y})$.
- **Evaluate** confidence of each object v by $W = \sum_{j=t-L+1}^{t} \sum_{i=1}^{M} \omega_{v_j}^{(i)}$. If $W < \gamma$, remove node v_j from frame j for each $j = (t - L + 1) \ldots t$. γ is an empirical threshold.

Table 2. Algorithm: build the sub-graph for a new frame t

Build the sub-graph $(\mathcal{V}_t, \mathcal{E}_t)$ for a new frame t as follows:

- For each object node $v_{t-1} \in \mathcal{V}_{t-1}$, let $\mathcal{V}_t \leftarrow \mathcal{V}_t \cup \{v_t\}$, $\mathcal{E} \leftarrow \mathcal{E} \cup \{(v_{t-1}, v_t)\}$. Pass message forward along edge (v_{t-1}, v_t) to get an approximation of $p(x_{v_t}|\mathbf{y}) \propto \psi_{v_t}(x_{v_t}, y_{v_t}) m_{v_{t-1}, v_t}(x_{v_t})$.
- Detect key object by applying $p(c^*|x)$ to all possible state x in the image. Cluster responses with confidence higher than τ_{c^*}. For each cluster non-overlapping with any existing node, create a new node v_t. Let the initial estimated state $\hat{x}_{v_t}$ be the cluster mean. Denote the set of all key object node as K.
- For each category $c \neq c^*$:
    - Create an augmenting node a for each key object node $v \in K$ and add an edge (v, a) between them.
    - For each such augmenting node and key object node pair $\{a, v\}$, sample $\{x_a^{(i)}, \omega_a^{(i)}\}_{i=1}^M \sim p(x_a|\mathbf{y}) \propto \psi_a(x_a, y_a) m_{v,a}(x_a)$.
    - Define the union of samples $S = \bigcup_a \{x_a^{(i)}, \omega_a^{(i)}\}_{i=1}^M$; let S' be the subset of S with samples whose weight are higher than τ_c.
    - Do clustering on S'; for each cluster non-overlapping with any existing node, create an object node u_t of category c. Let the initial estimated state $\hat{x}_{u_t}$ be the cluster mean.
    - $\mathcal{V}_t \leftarrow \mathcal{V}_t \cup \{u_t\}$. $\mathcal{E}_t \leftarrow \mathcal{E}_t \cup \{(u_t, v_t) | v_t \in \mathcal{V}_t, \psi_{u_t, v_t}(\hat{x}_{u_t}, \hat{x}_{v_t}) > \lambda\}$.
    - Remove augmenting nodes and corresponding edges.

high-weight cluster, a new object node is created. Figure 4 illustrates how to use augmenting nodes to update the graph.

More details of our overall algorithm and the algorithm of building sub-graph for each new frame are shown in Table 1 and Table 2.

6 Experiments

Experiments are done on the CHIL meeting video corpus [23]. Eight categories of objects are of interest: *human, table, chair, computer, projector, paper, cup* and *whiteboard* (or *projection screen*).

For testing we use 16 videos captured from three sites (IBM, AIT and UPC) and three camera views for each site. Each sequence has about 400 frames. One frame out of every

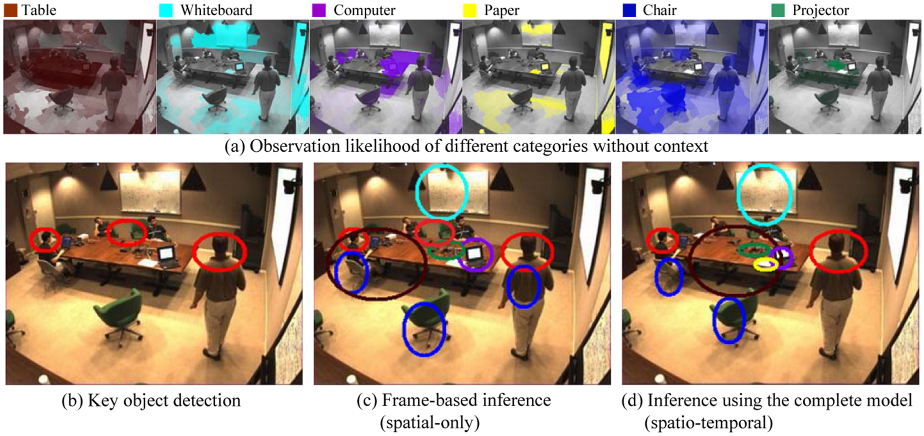
60 is fully annotated for evaluation. For training the parameters of spatial potential function, we selected 200 images from two views of the IBM and UPC site (no intersection between training images and test videos), and manually annotated the object size and position. Observation models for objects are trained with object instances from images of various meeting room and office scenes including a training part from CHIL.

We design our experiments to compare three methods with different levels of context: 1) no context, *i.e.* object observation model is directly applied to each frame; 2) spatial context only, *i.e.* a MRF without the temporal edges is applied in a frame-by-frame manner; 3) spatio-temporal context, *i.e.* the full model with both spatial and temporal edges is applied to the sequence.

6.1 Quantitative Analysis

Quantitative analysis is performed with metrics focusing on three different aspects: object detection and tracking, image segment categorization and pixel-level segmentation accuracy.

Object-level detection and tracking. The overall object detection rate and false alarm rate is shown in Figure 6(left). Two methods are compared: frame-based method with only spatial context, and the spatio-temporal method using the complete model. For the spatial-only method, an ROC curve is obtained by changing the threshold τ_c for creating new object nodes. The result shows that integrating temporal information helps improve detection rate and reduce false alarms, which is the effect of temporal smoothing and evidence accumulation. In object-level evaluation we do not include the non-contextual method, because the object observation model is based on classifying image segments, and we find that applying exhaustive search using such a model does not give a meaningful result. Some visual results of these methods can be found in Figure 5(a)(c)(d) respectively.



(a) Observation likelihood of different categories without context

(b) Key object detection

(c) Frame-based inference (spatial-only)

(d) Inference using the complete model (spatio-temporal)

Fig. 5. Comparison among observation with no context, inference using spatial relationship only and inference using spatio-temporal relationship

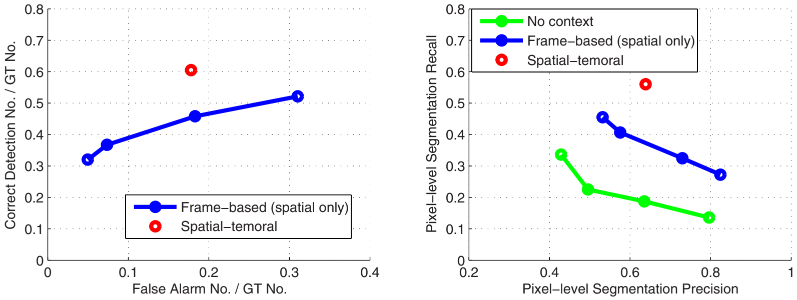


Fig. 6. Object detection rate and false alarm rate (left); pixel-level segmentation precision and recall (right)

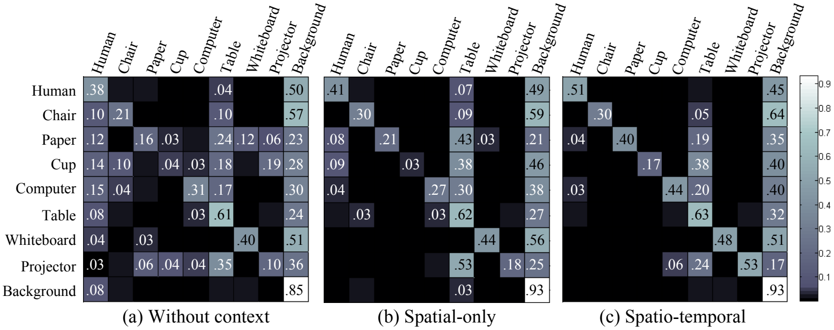
Table 3. Object tracking evaluation

Category	Ground truth trajectories	Mostly tracked trajectories (%GT)	Partially tracked trajectories (%GT)	Fragments
Human	64	46 (71.9%)	12 (18.8%)	12
Chair	30	10 (33.3%)	4 (13.3%)	2
Paper	40	21 (52.5%)	7 (17.5%)	0
Cup	11	2 (18.2%)	0 (0%)	0
Computer	24	10 (41.7%)	3 (12.5%)	2
Table	16	14 (87.5%)	0 (0%)	0
Screen	14	12 (85.7%)	1 (7.1%)	2
Projector	13	7 (53.8%)	2 (15.4%)	0
All	212	122 (57.5%)	29 (13.7%)	8

For the spatio-temporal method, we further evaluate its performance by the number of objects that are consistently tracked through the sequence, as shown in Table 3. All the numbers stand for trajectories, where *mostly tracked* is defined as at least 80% of the trajectory is tracked, and *partially tracked* defined as at least 50% is tracked. When a trajectory is broken into two, a *fragment* is counted. We can see that small objects such as *cups* and *computers* are harder to detect and track. *Paper* has a high false alarm rate, probably due to lack of distinct interior features (Figure 8(h) shows segments of human clothes detected as *paper*). Most fragments belong to human trajectories, because humans exhibit much more motion than other objects.

Image segment-level categorization. To compare with the result of applying object observation without contextual information, we compute the categorization accuracy of all the image segments in the form of a confusion matrix (Figure 7). The matrix shows that incorporating context helps reduce the confusion between different object categories, such as *paper* versus *whiteboard*. It is also observed that many objects are easily confused with *table*, mainly because they are often on top of or adjacent to the *table*.

Pixel-level segmentation. We obtain segmentation of each object based on the likelihood $p(c|r_i)$ of each segment r_i classified as category c. Pixel-level precision and recall rates of the three methods are shown in Figure 6(right). Similar to the previous two evaluations, the spatio-temporal method gives the best result. The segmentation



	Human	Chair	Paper	Cup	Computer	Table	Whiteboard	Projector	Background
Human	.38					.04			.50
Chair	.10	.21				.10			.57
Paper	.12		.16	.03		.24	.12	.06	.23
Cup	.14	.10		.04	.03	.18		.19	.28
Computer	.15	.04			.31	.17			.30
Table	.08				.03	.61			.24
Whiteboard	.04		.03				.40		.51
Projector	.03		.06	.04	.04	.35		.10	.36
Background	.08								.85

(a) Without context

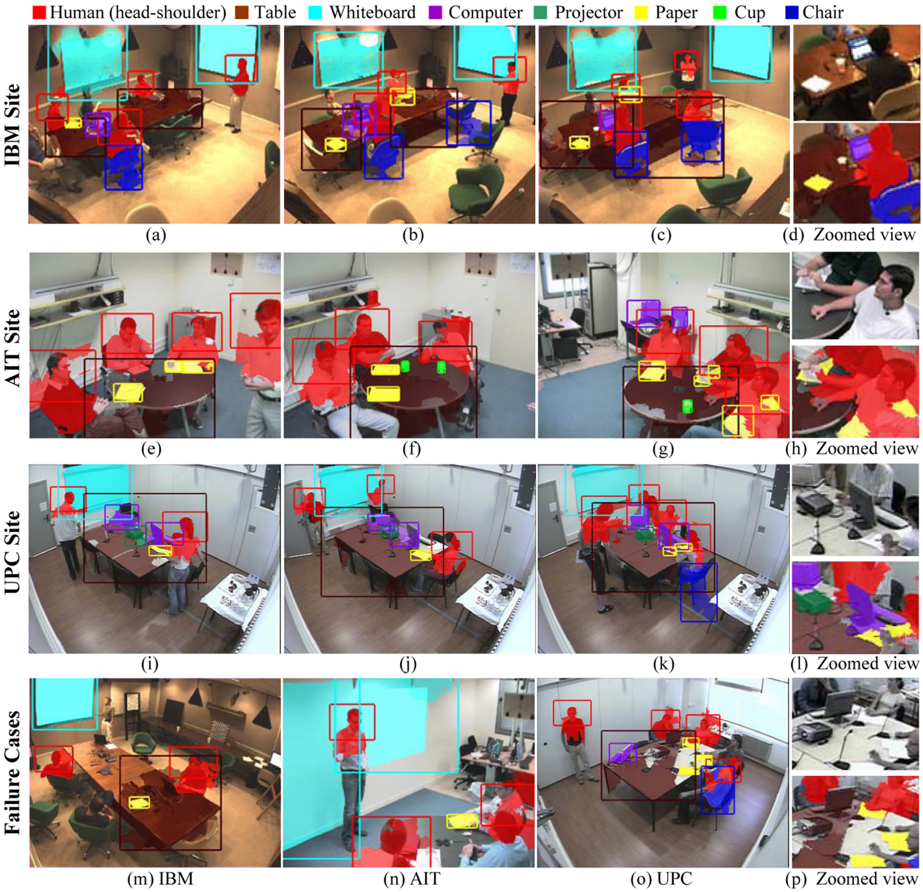
	Human	Chair	Paper	Cup	Computer	Table	Whiteboard	Projector	Background
Human	.41					.07			.49
Chair		.30				.09			.59
Paper	.08		.21			.43	.03		.21
Cup	.09			.03		.38			.46
Computer	.04				.27	.30			.38
Table		.03			.03	.62			.27
Whiteboard							.44		.56
Projector						.53		.18	.25
Background						.03			.93

(b) Spatial-only

	Human	Chair	Paper	Cup	Computer	Table	Whiteboard	Projector	Background
Human	.51								.45
Chair		.30				.05			.64
Paper	.04		.40			.19			.35
Cup				.17		.38			.40
Computer	.03				.44	.20			.40
Table						.63			.32
Whiteboard							.48		.51
Projector					.06	.24		.53	.17
Background									.93

(c) Spatio-temporal

Fig. 7. Confusion matrix of image region categorization by different methods. The value at (i, j) stands for the proportion of segments of category i classified as category j.



(a) (b) (c) (d) Zoomed view

(e) (f) (g) (h) Zoomed view

(i) (j) (k) (l) Zoomed view

(m) IBM (n) AIT (o) UPC (p) Zoomed view

Fig. 8. Sample results of our method

accuracy is not high, since this is only a simple post-process. But from the sample results of the spatio-temporal method in Figure 8 we can see that most detected objects are reasonably segmented when the object position and scale are correctly inferred.

6.2 Scenario Analysis

Figure 8 shows some sample results of our method on data from different meeting room sites and views. Objects that are in close interaction with key objects (humans) are detected more accurately. The method also has a tolerance to missed detection of key objects, *e.g.*, for the IBM site, although human detection rate is not high due to complex background, most objects are reasonably detected (Figure 8(a)-(c)). However such tolerance is to certain extent: Figure 8(m) shows a case when missed detections of key objects cause failure in detecting other objects.

Partial occlusions are frequently encountered and handled, such as occlusions of tables, whiteboards and laptops. But there is a bigger chance of failure when only a small part of an object is visible, such as in Figure 8(a)-(d) the table is broken into two; in Figure 8(n)(o) the table or part of it is missing from detection. This is also true for small objects, *e.g.* in Figure 8(g)(h) the paper occluded by the hand is broken into two. But in such case the result is still correct in the image segment level.

The bottleneck of performance is the observation model for objects other than key objects. As in Figure 8(p) the computer and projector are missing simply because observation likelihood is low. Although contextual information improves the overall result, the observation model in our current implementation is relatively simple compared with the complexity of the object recognition problem.

7 Conclusion

In this paper we address the problem of recognizing, localizing and tracking multiple categories of objects in a certain type of scenes. Specifically, we consider eight categories of common objects in meeting room videos. Given the difficulty of approaching this problem by purely appearance-based methods, we propose the integration of spatio-temporal context through a dynamic MRF, in which each node represents an object and the edges represent inter-object relationships. New object hypotheses are proposed online by adding *augmenting nodes*, which receive belief messages from the detected *key objects* of the scene (humans in our case). Experimental results show that the performance is greatly enhanced by incorporating contextual information.

There are many open problems and promising directions regarding the topic of object analysis in video. First, a stronger object observation model is needed, and our current training and testing sets are very limited. Second, we made no assumption of a fixed camera, but it can be a strong cue for inference, *e.g.* the position and scale of the stationary objects (such as tables) can be inferred from the activity area of the moving objects (such as humans). Third, 3D geometry of the scene or depth information should be useful for modeling occlusions. Last but not least, object recognition and tracking can be combined with action recognition [14][15] so as to better understand the semantics of human activities.

Acknowledgments. This research is supported, in part, by the U.S. Government VACE program. Yuan Li is funded, in part, by a Provost's Fellowship from USC.

References

1. Fergus, R., Perona, P., Zisserman, A.: Object class recognition by unsupervised scale-invariant learning. In: CVPR (2003)
2. Cao, L., Fei-Fei, L.: Spatially coherent latent topic model for concurrent object segmentation and classification. In: ICCV (2007)
3. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: CVPR (2001)
4. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR (2005)
5. Wu, B., Nevatia, R.: Cluster boosted tree classifier for multi-view, multi-pose object detection. In: ICCV (2007)
6. Sudderth, E.B., Torralba, A., Freeman, W.T., Willsky, A.S.: Learning hierarchical models of scenes, objects, and parts. In: ICCV (2005)
7. Hoiem, D., Efros, A.A., Hebert, M.: Putting objects in perspective. In: CVPR (2006)
8. Torralba, A., Murphy, K., Freeman, W., Rubin, M.: Context-based vision system for place and object recognition. In: ICCV (2003)
9. Li, L.-J., Fei-Fei, L.: What, where and who? classifying events by scene and object recognition. In: ICCV (2007)
10. Shotton, J., Winn, J., Rother, C., Criminisi, A.: Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: ECCV (2006)
11. Rabinovich, A., Vedaldi, A., Galleguillos, C., Wiewiora, E., Belongie, S.: Objects in context. In: ICCV (2007)
12. Carbonetto, P., de Freitas, N., Barnard, K.: A statistical model for general contextual object recognition. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3021, pp. 350–362. Springer, Heidelberg (2004)
13. Moore, D.J., Essa, I.A., Heyes, M.H.: Exploiting human actions and object context for recognition tasks. In: ICCV (1999)
14. Peursum, P., West, G., Venkatesh, S.: Combining image regions and human activity for indirect object recognition in indoor wide-angle views. In: ICCV (2005)
15. Gupta, A., Davis, L.S.: Objects in action: an approach for combining action understanding and object perception. In: CVPR (2007)
16. Yu, T., Wu, Y.: Collaborative tracking of multiple targets. In: CVPR (2004)
17. Wu, B., Nevatia, R.: Tracking of multiple humans in meetings. In: V4HCI (2006)
18. Friedman, J., Hastie, T., Tibshirani, R.: Additive logistic regression: a statistical view of boosting. Annals of Statistics 28(2), 337–407 (2000)
19. Comaniciu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis. IEEE Transaction on Pattern Analysis and Machine Intelligence 24(5), 603–619 (2002)
20. Sutton, C., McCallum, A.: Piecewise training for undirected models. In: Conference on Uncertainty in Artificial Intelligence (2005)
21. Pearl, J.: Probabilistic Reasoning in Intelligent Systems. Morgan Kaufman, San Mateo (1988)
22. Sudderth, E.B., Ihler, A.T., Freeman, W.T., Willsky, A.S.: Nonparametric belief propagation. In: CVPR (2003)
23. CHIL: The chil project, `http://chil.server.de/`