Context Aware Active Learning of Activity Recognition Models

Mahmudul Hasan and Amit K. Roy-Chowdhury
University of California, Riverside
mhasa004@ucr.edu, amitrc@ee.ucr.edu

Abstract

Activity recognition in video has recently benefited from the use of the context e.g., inter-relationships among the activities and objects. However, these approaches require data to be labeled and entirely available at the outset. In contrast, we formulate a continuous learning framework for context aware activity recognition from unlabeled video data which has two distinct advantages over most existing methods. First, we propose a novel active learning technique which not only exploits the informativeness of the individual activity instances but also utilizes their contextual information during the query selection process; this leads to significant reduction in expensive manual annotation effort. Second, the learned models can be adapted online as more data is available. We formulate a conditional random field (CRF) model that encodes the context and devise an information theoretic approach that utilizes entropy and mutual information of the nodes to compute the set of most informative query instances, which need to be labeled by a human. These labels are combined with graphical inference techniques for incrementally updating the model as new videos come in. Experiments on four challenging datasets demonstrate that our framework achieves superior performance with significantly less amount of manual labeling.

1. Introduction

Enormous amount of visual data is being generated continuously from various sources. Learning from these visual data, e.g., learning activity models, should be a continuous process so that the models can be improved with new video observations and adapted to the changes in dynamic environment. However, learning needs labeled data and labeling these large corpus of videos requires expensive and tedious human labor. Continuous manual labeling of these incoming videos in order to train the recognition models is infeasible. Active learning can be used to achieve an effective solution to this problem, since it is a powerful tool for training classifiers from unlabeled data sources with a reduced labeling cost and without compromising performance.

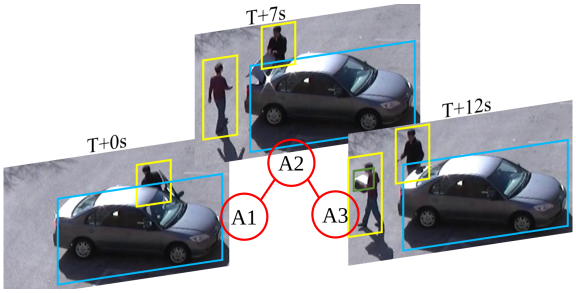


Figure 1. A sequence of a video stream [1] shows three new unlabeled activities - *person getting out of a car* ($A1$) at $T + 0s$, *person opening a car trunk* ($A2$) at $T + 7s$, and *person carrying an object* ($A3$) at $T + 12s$. These activities are spatio-temporally correlated (termed as context). Conventional approaches to active learning for activity recognition do not exploit these relationships in order to select the most informative instances. However, our approach exploits context and actively selects instances (in this case $A2$) that provide maximum information about other neighbors.

Recent successes in visual recognition take advantage of the fact that, in nature, objects and events tend to coexist with each other in a particular configuration, which is often termed as *context* [2]. Similarly, human activities in reality are inter-related and their surroundings can provide significant visual clue for their recognition (Figure 1). Several research works [3–7] considered the use of context from different perspectives to recognize human activities and showed significant performance improvement over the approaches that do not use context. However, these approaches are batch methods that require large amount of manually labeled data and are not able to continuously update their models. Even though few research works such as [8–10] learn human activity models incrementally from streaming videos, they do not utilize contextual information for more efficient recognition. In this work, we formulate a continuous learning framework for context aware activity recognition models that leverages upon a novel active learning technique in order to reduce the required human annotation effort.

Mahmudul Hasan is with the Dept. of CSE and Amit K. Roy-Chowdhury is with the Dept. of ECE at University of California Riverside. This work was supported in part by ONR under grant N00014-12-1-1026, by NSF under grant IIS-1316934, and by US Department of Defense.

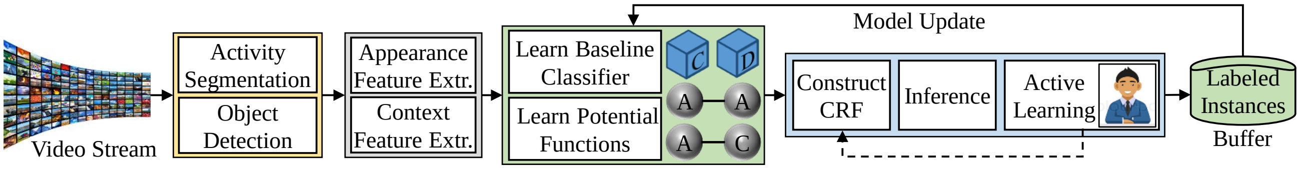


Figure 2. Our proposed framework for learning activity models continuously. Please see the text in Section 1.1 for details.

Active learning has become an important tool for selecting the most informative queries from a large volume of unlabeled data to be labeled by a human annotator, which are then used for training classifiers. During query selection, most of the approaches [11] only exploit informativeness, expected error reduction (EER), etc. of *individual* data instances in a batch or in an online manner assuming that there are no inter-relationships among them. As stated earlier, activities and objects in video show strong inter-relationship, which are generally encoded using graphical models. It would be beneficial to exploit these relationships (i.e., context) during the most informative query selection process as illustrated in Figure 1. Few works, such as [12], exploit link-based dependencies of the networked data, while [13] utilizes the inter-relationship of the data instances in feature space for active learning. Some works [14] perform query selection on CRF model for structured prediction in natural language processing by utilizing only the co-occurrence relationship that exist among the tokens in a sentence, while activities in a video sequence additionally exhibit spatial and temporal relationships as well as interactions with other objects. Hence, it would be a significant contribution to develop a new active learning technique for such applications.

1.1. Main Contributions and Overview

In this work, we propose a novel framework that exploits contextual information which are encoded using a CRF in order to learn activities continuously from videos. The **main contribution** of this work is twofold -

1. A new query selection strategy on a CRF graphical model for inter-related data instances by utilizing entropy and mutual information of the nodes.
2. Continuous learning of both the appearance and the context models simultaneously as new video observations come in so that the models can be adaptive to the changes in dynamic environment.

In order to achieve these goals, we show how to automatically construct a CRF online that can take care of any number and types of context features. Detailed overview of our proposed framework is illustrated in Figure 2.

Our framework has two phases: initial learning phase and incremental learning phase. During the initial learning phase, with a small amount of annotated videos in hand, we learn a baseline activity classifier and spatio-temporal contextual relationships. During the incremental learning phase, given a set of newly arrived unlabeled activities, we construct a CRF with two types of nodes - activity nodes and context nodes. Probabilities from the baseline classifier are used as the activity node potentials and the object detectors are used to compute context features that are used as the context node potentials. Spatio-temporal contextual relationships are used as the edge potentials. We perform inference on the CRF in order to obtain the marginal probabilities of the activity nodes.

Our active learning system consists of a strong and a weak teacher. We use information theoretic criteria - entropy and mutual information of the activity nodes for selecting the most informative instances to be labeled by a human annotator, which we refer to as the strong teacher. We condition on these newly labeled nodes and run inference again. We retain the highly confident labels obtained from the inference, which we refer to as the weak teacher. Newly labeled examples are stored in a buffer to be used in the next step of incremental update of the baseline classifiers and the contextual relationships.

2. Relation to Existing Works

Our work involves following areas of interest - human activity recognition, active learning, and continuous learning. We will review some relevant papers from these areas.

Activity recognition. Visual feature based activity recognition approaches can be classified into three broad categories such as interest point based low-level local features, human track and pose based mid-level features, and semantic attribute based high-level features based methods. Survey article [15] contains more detailed review on feature based activity recognition. Recently, context has been successfully used for activity recognition. Definition of context may vary based on the problem of interest. For example, [3] used object and human pose as the context for the activity recognition from single images. Collective or group activities was recognized in [5] and [6] using the context in the group. Spatio-temporal contexts among the activities and the surrounding objects were used in [7]. Graphical models was used to predict human activities in [4]. However, as mentioned in Section 1, these approaches are not capable of learning activity models continuously from unlabeled data.

Active learning. It has been successfully applied to many computer vision problems including tracking [16], object detection [17], image [18] and video segmentation [19], and activity recognition [20]. It has also been used on CRF for structured prediction in natural language processing[14,21,22]. They use information theoretic criteria such as entropy of the individual nodes for query selection. We additionally use mutual information because different activities in video are related to each other. It captures the entropy in each activity but subtracts out the conditional entropy of that activity when some other related activities are known. This criteria enables our framework to select the most informative queries from a set of unlabeled data represented by a CRF. Experiment results in Figure 5(column-3) validate our claim that using only entropy is not enough to capture the contextual relationships in videos.

Continuous learning. Among several schemes on continuous learning from streaming data, ensemble of classifiers [23] based methods are most common, where new weak classifiers are trained with the newly available data and then, added to the ensemble. A few methods can be found that learn activity models incrementally. The feature tree based method proposed in [8] grows in size with new training data. The method proposed in [9] uses human tracks and snippets, and the method proposed in [10] is based on active learning and boosted SVM classifiers. However, these methods do not exploit context, which has the ability to enhance the recognition performance.

3. Modeling Contextual Relationships

Prerequisite. We have a set of activities $A = \{a_i\}$ segmented from the video stream. Let $\{x_i\}$ be the visual features extracted from these activity segments. Additionally, we have a baseline activity recognition model $\mathcal{P}$ and a set of object detectors $\mathcal{D}$. We aim to formulate a generalized model that does not depend on any particular choice of feature extraction and classification algorithms in order to perform above mentioned tasks. In Section 5, we describe the specific choices we made during our experiments.

Overview. We model the inter-relationships among the activities and the object attributes using a CRF graphical model as shown in Figure 3. It is an undirected graph $G = (V, E)$ with a set of nodes $V = \{A, C, X, Z\}$, and a set of edges $E = \{A-A, A-C, A-X, C-Z\}$. A are the activity nodes, C are the context features, and X and Z are the observed visual features for the activities and the objects respectively. In Figure 3, $\mathcal{P}$ represents the activity classifier and $\mathcal{D}$ stands for the object detectors. They are used to compute the prior node potentials and to construct the context features respectively. We are interested in computing the posterior of the A nodes. Red edges among the A and C nodes represent spatio-temporal relationship among them. The connections between A and C nodes are fixed but we automatically determine the connectivity among the A nodes along with their potentials. The overall potential function (Φ) of the CRF is shown in Equation 1, where ϕs and ψs are node and edge potentials. We define the potential functions as follows.

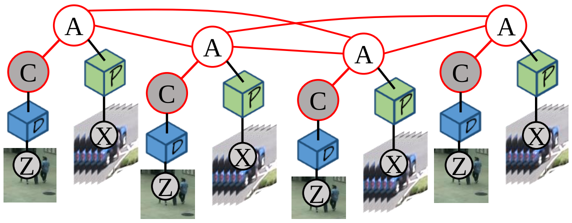


Figure 3. Illustrative example of a CRF for encoding the contextual information. Please see the text in Section 3 for details.

$$\Phi = \prod_{\substack{a_i \in A, c_i \in C \\ x_i \in X, z_i \in Z}} \phi(a_i, x_i)\phi(c_i, z_i) \prod_{\substack{a_i, a_j \in A \\ c_i \in C}} \psi(a_i, a_j)\psi(a_i, c_i) \tag{1}$$

$$\phi(a_i, x_i) = p(a_i | x_i, \mathcal{P}) \tag{2}$$

$$\phi(c_i, z_i) = \phi(c_i^1, z_i) \odot \phi(c_i^2, z_i) \tag{3}$$

$$\phi(c_i^1, z_i) = p(c_i^1 | z_i, \mathcal{D}) \tag{4}$$

$$\phi(c_i^2, z_i) = \text{bin}(c_i^2)\, \mathcal{N}(c_i^2, \mu_{c^2}, \sigma_{c^2}) \tag{5}$$

$$\begin{aligned}\psi(a_i, a_j) = {} & F_a(a_i, a_j)\, \mathcal{N}(\|t_{a_i} - t_{a_j}\|^2, \mu_t, \sigma_t) \\ & \mathcal{N}(\|s_{a_i} - s_{a_j}\|^2, \mu_s, \sigma_s)\end{aligned} \tag{6}$$

$$\psi(a_i, c_i) = \psi(a_i, c_i^1) \otimes \psi(a_i, c_i^2) \tag{7}$$

$$\psi(a_i, c_i^1) = F_{c^1}(a_i, c_i^1)\, \mathcal{N}(\|s_{a_i} - s_{c_i^1}\|^2, \mu_{c^1}, \sigma_{c^1}) \tag{8}$$

$$\psi(a_i, c_i^2) = \sum_{a \in A} \text{bin}(c_i^2)\mathcal{I}(a = a_i)^T\, \mathcal{N}(c_i^2, \mu_{c^2}, \sigma_{c^2}) \tag{9}$$

Activity node potential, $\phi(a_i, x_i)$. These potentials correspond to the A nodes of the CRF. They describe the inherent characteristics of the activities through low level motion features. We extract low level features x_i from the activity segments a_i and train a baseline classifier $\mathcal{P}$. Classification score of a candidate activity segments a_i generated by $\mathcal{P}$ are then used as the node potential as defined in Equation 2.

Context node potential, $\phi(c_i, z_i)$. These potentials correspond to the C nodes of the CRF, which are scene level features and object attributes related to the activity of interest. They are not low level motion features but may provide important and distinctive visual clues. For example, presence of a car may distinguish *unloading a vehicle* activity from *entering a facility* activity. In this work, we employ a semi-automatic technique to learn these contexts by applying a number of detectors on the image observation Z in the activity segment. Number and type of these context features may vary for different applications. For example, we use two context features in an application - objects ($\phi(c_i^1, z_i)$) and person ($\phi(c_i^2, z_i)$) attributes as defined in Equations 4 and 5, where c_i^1 is the object class vector, $c_i^2 = \|L_1 - L_2\|$ is the distance covered by a person in the activity region, bin($\cdot$) is a binning function as in [24], and μ_{c^2} and σ_{c^2} are the mean and variance of the covered distances. We concatenate them in order to compute the context nodes potential (Equation 3 - $\odot$ is the concatenation operation).

Activity-Activity edge potential, $\psi(a_i, a_j)$. This potential models the connectivity among the activities in A. We assume that activities which are within a spatio-temporal distance are related to each other. This potential has three components - association, spatial, and temporal components. The association component is the co-occurrence frequencies of the activities. The spatial (temporal) component models the probability of an activity belonging to a particular category given its spatial (temporal) distance from its neighbors. $\psi(a_i, a_j)$ is defined in Equation 6, where $a_i, a_j \in A$, $F_a(a_i, a_j)$ is the co-occurrence frequency between the activities a_i and a_j, $s_{a_i}, s_{a_j}, t_{a_i},$ and t_{a_j} are the spatial and temporal locations of the activities, and $\mu_t, \sigma_t, \mu_s,$ and σ_s are the parameters of the Gaussian distribution of relative spatial and temporal positions of the activities, given their categories.

Activity-Context edge potential, $\psi(a_i, c_i)$. This potential function models the relationship among the activities and the context features. It corresponds to $A - C$ edges in the CRF. This potential is defined in Equation 7-9. $\psi(a_i, c_i^1)$ models the relationship between the activity and the object attribute and $\psi(a_i, c_i^2)$ models the relationship between the activity and the person attribute. Operator $\otimes$ performs horizontal concatenation of matrices.

Structure Learning. We assume the connection between A and C if the involved person or the objects are detected by the detector $\mathcal{D}$. However, we learn the $A - A$ connections in an online manner because it is hard to predict the number of activities and they might not be related to each other. A recent approach for learning the structure is hill climbing structure search [3], which are not designed for continuous learning. In this work, we utilize an adaptive threshold based approach in order to determine the connections among the nodes in A. At first, we assume all the nodes in A are connected to each other. Then we apply two thresholds - spatial and temporal - on the links. We keep the links whose spatial and temporal distances are below these thresholds, otherwise we delete the links. We learn these two thresholds using a max-margin learning framework.

Suppose, we have a set of training activities $\{(a_i, t_{a_i}, s_{a_i}) : i = 1 \dots m\}$ and we know the pairwise relatedness of these activities. The goal is to learn a function $f_r(d) = w^T d$, that satisfies the constraints in Equation 10, where $d_{ij} = [\text{abs}(t_i - t_j), \|s_i - s_j\|]$.

$$\begin{aligned} f_r(d_{ij}) &= +1, && \forall \text{ related } a_i \text{ and } a_j, \\ f_r(d_{ij}) &= -1, && \text{Otherwise.} \end{aligned} \tag{10}$$

We can formulate this problem as a traditional max-margin learning problem [3]. Solution to this problem will provide us a function to determine the existence of link between two unknown activities.

Inference. In order to compute the posterior probabilities of the A nodes, we choose belief propagation (BP) message passing algorithm. BP does not provide guarantee to convergence to true marginals for a graph with loops but it has proven excellent empirical performance [25]. Its local message passing is consistent with the contextual relationship we model among the nodes. At each iteration, belief of the nodes are updated based on the messages received from their neighbors. Consider a node $a_i \in V$ with a neighborhood $N(a_i)$. The message sent by a_i to its neighbors can be written as, $m_{a_i,a_j}(a_j) = \alpha \int_{a_i} \psi(a_i, a_j)\phi(a_i, x_i) \prod_{a_k \in N(a_i)} m_{a_k,a_i}(a_i) da_i$. The marginal distribution of each node a_i is estimated as, $p'(a_i) = \alpha\phi(a_i, x_i) \prod_{a_j \in N(a_i)} m_{a_j,a_i}(a_i)$. The class label with the highest marginal probability is the predicted class label. We use the publicly available tool [26] to compute the parameters of the CRF and to perform the inference.

4. Context Aware Active Learning

Inference on the CRF $G = (V, E)$ provides the marginal probabilities and pairwise marginal joint distribution of the nodes correspond to the edges. In this section, we use these probabilities to select the most informative set $\mathcal{S}^* \in V$.

Suppose, we have a set of labeled data instances $\mathcal{L}$ with c number of classes. We learn a baseline classifier $\mathcal{P}$ and a context model $\mathcal{C}$ with these labeled data $\mathcal{L}$. Now, we receive a set of unlabeled activity instances $\mathcal{U} = \{a_i\}$ with low level visual features $\{x_i\}$ from the video stream. We then construct a CRF G with the activities in $\mathcal{U}$ using $\mathcal{P}$ and $\mathcal{C}$ as discussed in Section 3. Inference on G gives us a probability distribution $\mathcal{P}_\mathcal{G}(a_i)$ for an unlabeled activity a_i. Our goal is to use $\mathcal{U}$ to improve the model $\mathcal{P}$ and $\mathcal{C}$ with least amount of manual labeling.

To begin with, let us assume that no inter-relationships exist among the data instances in $\mathcal{U}$. At first, we apply the current model $\mathcal{P}_\mathcal{G}$ on the instances of $\mathcal{U}$ to obtain a class probability distribution $\mathcal{P}_\mathcal{G}(a_i)$ for each instance. We select the most informative subset $\mathcal{S}$ from the instances in $\mathcal{U}$. An instance is considered informative if the current model $\mathcal{P}_\mathcal{G}$ is uncertain about it. We measure the uncertainty using entropy. This can be formulated using Equation 11, where $\mathcal{H}(\mathcal{S})$ is the sum of entropies of the nodes in $\mathcal{S}$.

$$\mathcal{S}^* = \arg\max_{\mathcal{S} \subset \mathcal{U}} \mathcal{H}(\mathcal{S}) \tag{11}$$

$$\mathcal{H}(\mathcal{S}) = \sum_{a_i \in \mathcal{S}} \mathcal{H}(a_i) = \sum_{a_i \in \mathcal{S}} \sum_{j=1}^{c} \mathcal{P}_\mathcal{G}(a_i = j) \log \frac{1}{\mathcal{P}_\mathcal{G}(a_i = j)}$$

However, in many applications, data instances are interrelated, which can be modeled by a CRF as shown in Figure 3. Related instances are connected by edges, where probability distribution of one instance can influence other neighboring instances. In order to perform active learning on such models, we also have to acknowledge these influences. Intensity of these influences can be computed by mutual information $\mathcal{M}$. The *basic intuition* is that if two instances are connected and can heavily influence each other, we can select only one of them for manual labeling. After getting the label, if we perform inference again on the CRF with conditioning on the newly labeled nodes, neighboring instances will have the chance to receive the correct label with much higher probabilities. Mathematically speaking, we select a set $\mathcal{S}^*$ that maximizes the entropy of the individual instances but minimizes the pairwise mutual information in the set ($\mathcal{M}(\mathcal{S})$) . We also want to select nodes which have more connections with other nodes since they can influence more nodes once they have the correct labels. The overall optimization problem for selecting the activities to be labeled can be formulated using Equation 12, where $Deg(\mathcal{S})$ is the sum of the degrees of the nodes in $\mathcal{S}$.

$$\mathcal{S}^* = \arg\max_{\mathcal{S}\subset\mathcal{U}}[\mathcal{H}(\mathcal{S}) - \mathcal{M}(\mathcal{S}) + \beta Deg(\mathcal{S})] \qquad (12)$$

$$\mathcal{M}(\mathcal{S}) = \sum_{a_i,a_j\in\mathcal{S}} \mathcal{M}(a_i,a_j) = \sum_{a_i,a_j\in\mathcal{S}} \sum_{i,j\in c} \mathcal{P}_{\mathcal{G}}(a_i = i, a_j = j)$$
$$\log \frac{\mathcal{P}_{\mathcal{G}}(a_i = i, a_j = j)}{\mathcal{P}_{\mathcal{G}}(a_i = i)\mathcal{P}_{\mathcal{G}}(a_j = j)}$$

Above mentioned optimization problem will select a subset $\mathcal{S}^*$ that will contain instances with higher entropies and lower pairwise mutual information. However, it is a subset selection problem and NP-hard. We provide a greedy solution to this problem in Algorithm 1 in order to obtain the set $\mathcal{S}^*$, where we set β to 1.

Algorithm 1 Greedy Query Selection (Equation 12)

```
Input: CRF graph $G = (V, E)$, $|V| = N$
       Node probabilities: $N \times c$
       Edge probabilities: $N \times N \times c$
Output: $S \subset V$, $|S| = K$
Compute entropies of the nodes, $\mathcal{H} : N \times 1$
Compute pairwise mutual information, $\mathcal{M} : N \times N$
while $|S| < K$ do
    $v_1 = \arg\max_{v\in V}[\mathcal{H}(v) + \beta Deg(v)]$;
    $S \leftarrow S \cup v_1$;  $V \leftarrow V - v_1$
    $v_2 = \arg\min_{v\in \text{Neigh}(v_1)} \mathcal{M}(v_1, v)$; $S \leftarrow S \cup v_2$; $V \leftarrow V - v_2$
end while
```

We ask a human annotator (strong teacher) to label the instances in $\mathcal{S}^*$. We then perform inference on G again by conditioning on the nodes $a_i \in \mathcal{S}^*$. It provides more accurate labels to the neighbors of $a_i \in \mathcal{S}^*$. Now, for an instance $a_j \in \mathcal{U} - \mathcal{S}^*$, if one of the classes has probability greater than δ (say $\delta = 0.9$), we assume that current model $\mathcal{P}_{\mathcal{G}}$ is highly confident about this instance. We retain this instance along with its label obtained from the inference for incremental training. We refer to this as the weak teacher. Number of instances obtained from the weak teacher actually depends on the value of δ, which we set large for safety so that miss-classified instances are less likely to be used in incremental training. An illustrative example of our active learning system is shown in Figure 4.

4.1. Incremental Updates

We have two models to update - appearance model and context model. These models are responsible for the node and edge potentials of the CRF respectively.

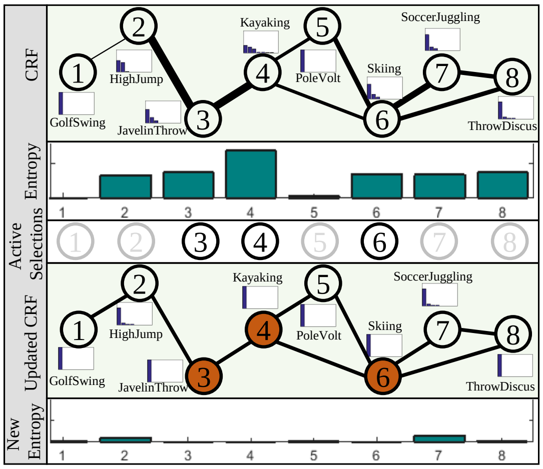


Figure 4. Inference on the CRF (top) gives us marginal probability distribution of the nodes and edges. We use these distributions to compute entropy and mutual information. Relative mutual information is shown by the thickness of the edges, whereas entropy of the nodes are plotted below the top CRF. Algorithm 1 exploits entropy and mutual information criteria in order to select the most informative nodes (3, 4, and 6). We condition upon these nodes (filled) and perform inference again, which provides us more accurate recognition and a system with lower entropy (bottom plot).

Updating appearance model. We use a multinomial logistic regression model as the baseline activity classifier. In this model, the probability of label y^i of x^i belongs to class j is written as $p(j|x_i;\theta) = \exp(\theta_j^T x_i)/\sum_{l=1}^{c}\exp(\theta_l^T x_i)$, where, $j \in \{1, \ldots, c\}$ is the set of class labels, θ_j^T is the weight vector corresponds to class j, and the superscript T denotes transpose operation. The cost function is given by, $\arg\min_\theta J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{c} 1\{y_i = j\}\log p\left(y_i = j|x_i;\theta\right)$. This is a convex optimization problem and we solve this using gradient descent method, which provides a globally optimal solution. The gradient equation can be written as, $\nabla_{\theta_j} J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\left[x_i\left(1\{y_i = j\} - p(y_i = j|x_i;\theta)\right)\right]$.

For updating this model, we obtain the newly labeled instances from the active learner and store them in a buffer. When the buffer is full, we use all of these instances to compute the change of gradient $\nabla_{\theta_j} J(\theta)$ of the model. Then we update the model parameters using gradient descent as follows, $\theta_j^{t+1} = \theta_j^t - \alpha\nabla_{\theta_j^t} J(\theta)$, where, α is the learning rate. This technique is known as the mini-batch training in literature [27], where model parameter changes are accumulated over some number of instances before actually updating the parameters. We use [28] for incrementally training the SVM when we use it as the baseline classifier.

Updating context model. Updating the context model is actually recomputing the parameters of the Equations 5, 6, 8, and 9. The parameters are mainly co-occurrence frequencies and means and variances of the Gaussian distributions. The parameters of the Gaussians can be updated using the method in [29], wheres the co-occurance frequency matrices can be updated as follows, $F_{ij} = F_{ij} + \text{sum}([(L = i).(L = j)^T]. * Adj)$, where, $i, j \in \{1, \ldots, c\}$, L is the set of labels of the instances in $\mathcal{U}$ obtained after the inference, Adj is the adjacency matrix of the CRF G of size $|L| \times |L|$, $\text{sum}(.)$ is the sum of the elements in the matrix, and $.*$ is the element wise matrix multiplication.

5. Experiments

The objectives of the experiments is to analyze the performance of our framework and to compare it with the state-of-the-art batch and incremental methods as well as with the recent active learning techniques. It is desirable that the performance improves with the availability of new training instances and eventually converges to the performance of the batch methods after all of the training instances are seen. We conduct experiments on four challenging datasets - VIRAT [1], UCLA-Office [30], MPII-Cooking [31], and UCF50 [32] - to evaluate the performance of our proposed continuous learning framework. Detailed description of these datasets are available in the supplementary material.

Experiment setup. We conduct five fold cross validation on each of the datasets. Four folds are used as the training and remaining one is used as the testing set. We divide the training set into five or six batches. First batch is used to train prior appearance and context models. Rest of the batches are used to update the models sequentially. Instances in the first batch are manually labeled, whereas we perform active learning on other batches and use the obtained labels for incremental training of the models. After finishing incremental training with a batch of data, we evaluate the resultant models on the testing set and report these results as shown in Figure 5. Each row corresponds to one dataset and each column corresponds to one experiment scenario, which we describe below.

Activity segmentation. For VIRAT and UCLA-Office, we use an adaptive background subtraction algorithm to identify motion regions. We detect moving persons around these motion regions using [33] and use them to initialize a tracking method in order to obtain local trajectories of the moving persons. We collect STIP features [34] from these local trajectories and use them as the model observation in the method proposed in [35] to identify candidate activity segments from these motion regions. Activities are already segmented in UCF50, whereas for MPII-Cooking we use the segmentation provided with the dataset.

Appearance feature. We extract STIP [34] features from the activity segments. We use a spatio-temporal pyramid and average pooling based technique similar to [36] to compute an uniform representation using these STIP features. For MPII-Cooking dataset, we use bag-of-word based MBH [37] feature that comes with the dataset.

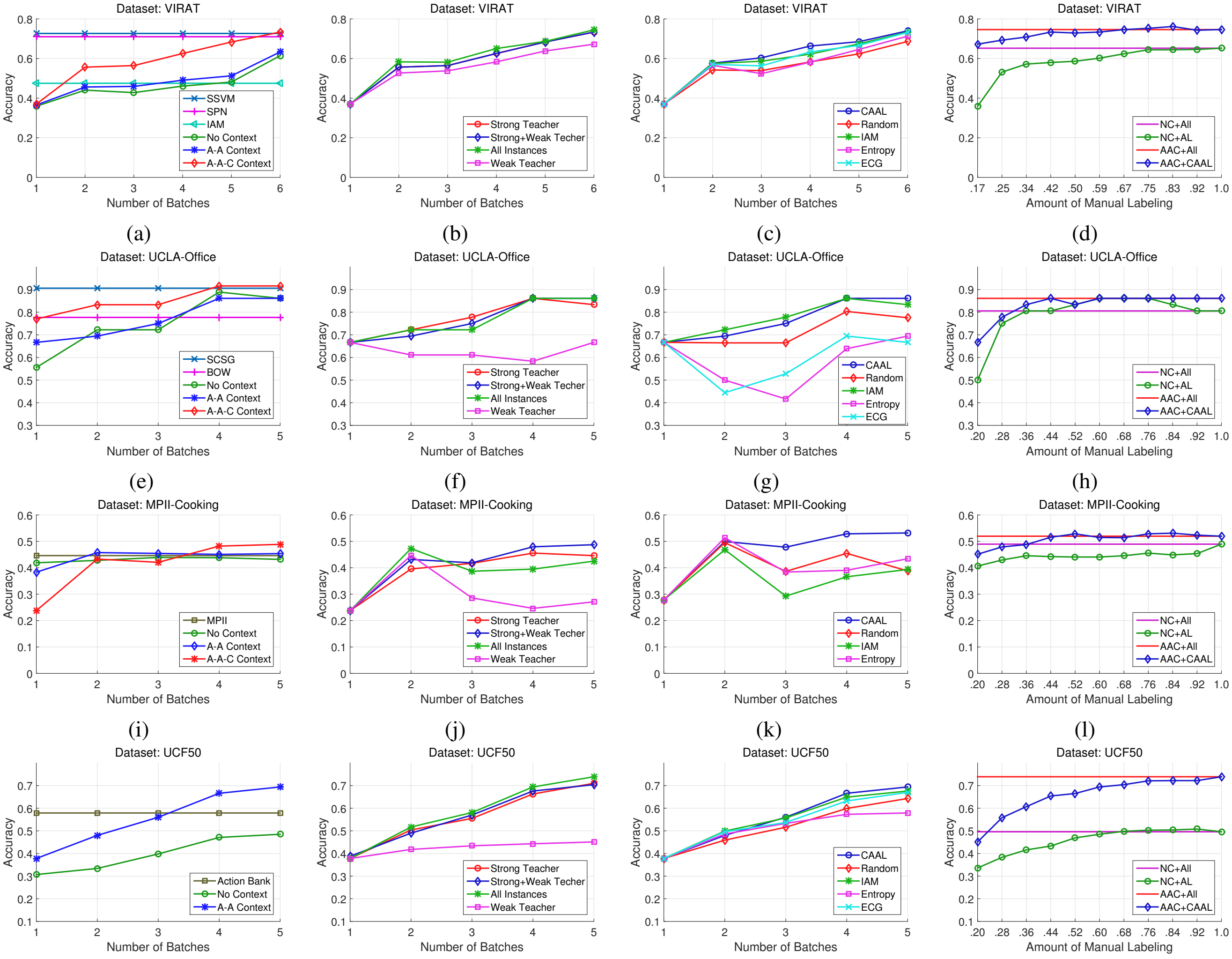
Context features. Number of context features and their types may vary based on the datasets. Our generalized CRF formulation can take care of any number and type of context features. We use co-occurrence frequency of the activities and the objects, their relative spatial and temporal distances, movement of the objects and persons in the activity region, etc. as the context feature. Some of the features were described in Section 3. Context features naturally exist in VIRAT, UCLA-Office, and MPII-Cooking datasets, whereas for UCF50 we improvise a context feature by assuming that similar types of activities tend to co-occur in the nearby spatial vicinity. Dataset specific detailed description of these features can be found in the supplementary material.

Baseline Classifier. We use multinomial logistic regression or softmax as the baseline classifier for VIRAT, UCLA-Office, and UCF50 datasets, whereas we use linear SVM for the MPII-Cooking dataset.

Result Analysis. We conduct four different experiments for each dataset - 1) comparison with other batch and incremental methods against three different variants (based on the use of context) of our approach, 2) performance evaluation of the four variants (based on the use of strong and weak teachers) of our proposed active learning system, 3) comparison against other state-of-the-art active learning techniques, and 4) the accuracy vs. the percentage of manual labeling plot. We show these plots in the first, second, third, and fourth column of Figure 5 respectively. We analyze these plots in the subsequent paragraphs.

Comparison with state-of-the-arts. Plots in Figure 5(a, e, i, m) illustrate the comparisons of our three test cases - no context, A-A context, and A-A-C context against state-of-the-art batch and incremental methods for four datasets. The definitions of these test cases are as follows. No context means we apply the appearance model $\mathcal{P}$ independently on the activity segments without exploiting any spatio-temporal contextual information. A-A context means we only utilize the inter-relationship among the activities, which are only the A nodes and corresponding red edges in Figure 3. A-A-C context means we exploit the object and person attribute context along with the A-A context. In all these three test cases, we use active learning with both of the weak and the strong teachers. We apply several object detectors based on HOG features and SVM in order to construct the A-C part of the A-A-C context for VIRAT and UCLA-Office datasets. These datasets have five and four different object classes respectively. We directly use the context features provided with the MPII-Cooking dataset. However, we do not use A-C context for UCF50, where each activity is associated with a specific object like foodball, piano, etc. Use of A-C context would have been produced better results from a over-fitted model that would not reflect the true contribution of this work.

We compare the results on the VIRAT dataset against structural SVM (SSVM) [7], sum product network (SPN) [38], and incremental activity modeling (IAM) [10]. We compare the results on UCLA office dataset against stochas-



(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)

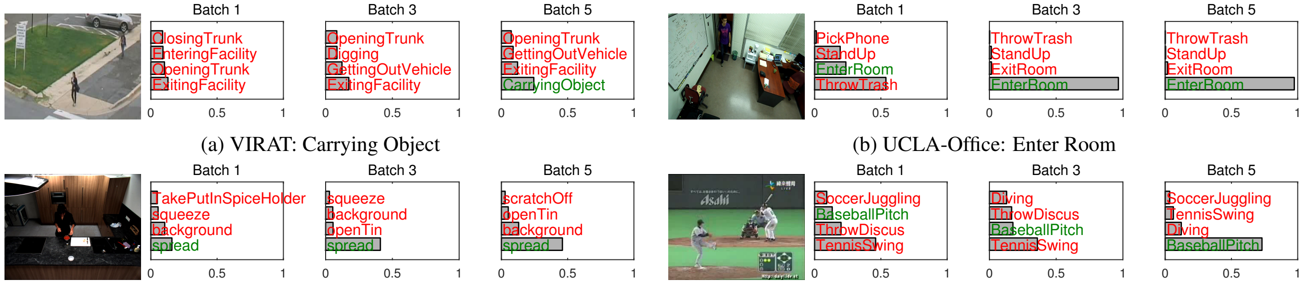
(m) (n) (o) (p)

Figure 5. Plots (a, e, i, and m) show the performance comparisons of our frameworks against state-of-the-art methods on VIRAT, UCLA-Office, MPII-Cooking, and UCF50 datasets respectively. Plots (b, f, j, and n) show the performances of weak and strong teachers. Plots (c, g, k, and o) show the performance comparisons of our proposed active learning system (CAAL) against state-of-the-art active learning and semi-supervised methods. Plots (d, h, l, and p) show accuracy vs. percentage of manual labeling for our methods and batch methods. Please see the text for the explanation of the plots. For clarity, please see on a color monitor. The plots can be zoomed in.

tic context sensitive grammar (STSG) [30], and SVM based bag-of-word. We compare the results on MPII-Cooking and UCF50 datasets against MPII [31] and action bank [39] respectively. Since these are the batch methods, we report only the final performances of these methods when they finish using all the training instances. Hence, plots of accuracies of these methods are horizontal straight lines.

Followings are the analysis of the plots - i) All of the four plots for four different datasets show similar asymptotic characteristics. Performance improves with new batches of training instances. ii) Performance improves when we use more contextual information. A-A-C performs better than A-A. A-A performs better than no context. iii) Our methods outperform other state-of-the-art batch and incremental method with far less amount of manually labeled data. In these plots our method uses around forty to fifty percent manually labeled data depending on the datasets, whereas all other methods use all the instances to train their models except IAM. IAM does not report amount of manual labeling for VIRAT. iv) Our no context and A-A test cases also outperform other methods that do not use context features.

Performances of four variants. Plots in Figure 5(b, f, j, n) illustrate the comparisons among the four test cases based on the use of weak and strong teachers. These test cases are defined as follows. Weak teacher - for incremental training, we only use the highly confident labels provided by the model after the inference. No manually labeled instances are used in this test case. Strong teacher - we label a portion of the incoming instances manually. This portion is determined by Algorithm 1. Strong+Weak teacher - we use both of the above mentioned teachers. All instances - we label all the incoming instances manually and use all of them to



(a) VIRAT: Carrying Object

(b) UCLA-Office: Enter Room

(d) MPII-Cooking: Spread

(c) UCF50 - Baseball Pitch

Figure 6. Evaluation of continuous learning on individual activities. Activity with green color means the ground truth class, whereas activities with red color means false predictions. Grey bars represent probability scores. Here, we show the results obtained after the arrival of batch 1, 3, and 5 data. In each of these examples, continuous learning helps to obtain the correct label with a higher probability even though some of them were miss-classified initially. Best viewable in color.

incrementally update the models.

Followings are the analysis of the plots - i) Performance of all of the test cases improves as more training instances are seen except the weak teacher. ii) Strong+weak teacher test case uses around forty percent of manually labeled instances. However, its performance is very similar to all instance test case that uses hundred percent manually labeled instances. It proves the efficiency of our method for selecting the most informative queries. iii) Performances of Strong+weak teacher and strong teacher are almost overlapped. It means that weakly labeled instances don't posses useful information for training because they are already confidently classified by the model. iv) Performance of weak teacher is not as good as other because it does not manually label the instances except in the first batch. Its performance tends to diverge due to the fact that some of the initial labels provided by the classifier are not correct.

Comparison with other active learning methods. Plots in Figure 5(c, g, k, o). illustrate the comparisons of our context aware active learning (CAAL) method against random sampling and three other state-of-the-art active learning techniques such as IAM [10], Entropy [21], and expected change of gradients (ECG) [11]. IAM selects a query by utilizing classifier's decision ambiguity over an unlabeled instance and takes advantages of both weak and strong teachers. Entropy [21] selects a query if the classifier is highly uncertain about it based on entropy measure. ECG [11] considers an instance informative if it brings significant change to the cost function. We obtained the codes from the authors of IAM, while we implemented Entropy and ECG by ourselves. We follow the same conventions and parameter setup for these experiments for ensuring fairness. Our method outperforms other active learning methods and random sampling for all datasets. This is because our method can utilize the interrelationships of the instances. The margin of improvement is large for UCLA-Office, MPII-Cooking, and UCF50 datasets. IAM performs better than Random, Entropy, and ECG because it is benefited from the weak teacher.

Accuracy vs. manual labeling. Plots in Figure 5(d, h, l, p). illustrate the accuracy vs. the percentage of manual labeling for four different test cases - no context + all manual label (NC+All), no context + active learning (NC+AL), A-A context + all manual label (AAC+All), and A-A context + context aware active learning (AAC+CAAL). A-A context and no context test cases have been defined above. Additionally, no context and A-A context use strong+weak teacher active learning. All of the reported accuracies in this plot are after the final batch. All manual label test case manually labels all the instances. It has only one accuracy, which is a straight line in this plot. It is evident in the plot that A-A-C begins to achieve accuracy similar to all instance test case with only forty to fifty percent manually labeled data. Performance of no context is far below than A-A-C context. We conducted experiments on challenging natural video datasets where intra-class variance is very high. We achieve performance similar to the state-of-the-art batch methods on such challenging datasets by utilizing roughly forty to fifty percent manual labeled data. It proves the robustness of the framework for selecting the most informative queries. Only UCF50 is the exception. For UCF50, there are short clips, so context does not help much and that is the reason we do not see a similar reduction in labeling effort. So the main impact of our work is in natural videos, which have not been pre-processed into short clips so that the relationships between the activities can be exploited.

Figure 6 shows the incremental performance of our framework on four individual activities from four datasets. Due to space limitation in the main paper, we will provide more results in the supplementary materials.

6. Conclusion

We presented a continuous learning framework for context aware activity recognition. We formulated a new active learning technique that utilize the contextual information among the activities and objects. We utilized entropy and mutual information of the nodes in active learning to account for the inter-relationships between them. We also showed how to incrementally update the models using the newly labeled data. Finally, we presented experimental results to demonstrate the robustness of our method.

References

[1] S. Oh, A. Hoogs, and et. al., “A large-scale benchmark dataset for event recognition in surveillance video,” in *CVPR*, 2011.

[2] A. Oliva and A. Torralba, “The role of context in object recognition,” *Trends in Cognitive Science*, 2007.

[3] B. Yao and L. Fei-Fei, “Modeling mutual context of object and human pose in human-object interaction activities,” in *CVPR*, 2010.

[4] Z. Wang, Q. Shi, and C. Shen, “Bilinear programming for human activity recognition with unknown mrf graphs,” in *CVPR*, 2013.

[5] T. Lan, W. Yang, Y. Wang, and G. Mori, “Beyond actions: Discriminative models for contextual group activities,” in *NIPS*, 2010.

[6] W. Choi, K. Shahid, and S. Savarese, “Learning context for collective activity recognition,” in *CVPR*, 2011.

[7] Y. Zhu, N. M. Nayak, and A. K. Roy-Chowdhury, “Context-aware modeling and recognition of activities in video,” in *CVPR*, 2013.

[8] K. Reddy, J. Liu, and M. Shah, “Incremental action recognition using feature-tree,” in *ICCV*, 2009.

[9] R. Minhas, A. Mohammed, and Q. Wu, “Incremental learning in human action recognition based on snippets,” *IEEE TCSVT*, 2012.

[10] M. Hasan and A. Roy-Chowdhury, “Incremental activity modeling and recognition in streaming videos,” in *CVPR*, 2014.

[11] B. Settles, “Active learning,” *Morgan & Claypool*, 2012.

[12] L. Shi, Y. Zhao, and J. Tang, “Batch mode active learning for networked data,” *ACM TIST*, 2012.

[13] O. Mac Aodha, N. D. Campbell, J. Kautz, and G. J. Brostow, “Hierarchical subquery evaluation for active learning on a graph,” in *CVPR*, 2014.

[14] B. Settles and M. Craven, “An analysis of active learning strategies for sequence labeling tasks,” in *EMNLP*, 2008.

[15] R. Poppe, “A survey on vision-based human action recognition,” *Image and Vision Computing*, 2010.

[16] C. Vondrick and D. Ramanan, “Video annotation and tracking with active learning,” in *NIPS*, 2011.

[17] S. Vijayanarasimhan and K. Grauman, “Large-scale live active learning: Training object detectors with crawled data and crowds,” *International Journal of Computer Vision*, vol. 108, no. 1-2, pp. 97–114, 2014.

[18] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen, “icoseg: Interactive co-segmentation with intelligent scribble guidance,” in *CVPR*, 2010.

[19] A. Fathi, M. F. Balcan, X. Ren, and J. M. Rehg, “Combining self training and active learning for video segmentation,” in *BMVC 2011*, 2011.

[20] X. Liu and J. Zhang, “Active learning for human action recognition with gaussian processes,” in *ICIP*, 2011.

[21] G. Druck, B. Settles, and A. McCallum, “Active learning by labeling features,” in *EMNLP*, 2009.

[22] C. T. Symons, N. F. Samatova, R. Krishnamurthy, B.-H. Park, T. Umar, D. Buttler, T. Critchlow, and D. Hysom, “Multi-criterion active learning in conditional random fields,” in *ICTAI*, 2006.

[23] H. He, S. Chen, K. Li, and X. Xu, “Incremental learning from stream data,” *IEEE TNN*, 2011.

[24] D. Ramanan, “Learning to parse images of articulated objects,” in *NIPS*, 2006.

[25] Y. Li and R. Nevatia, “Key object driven multi-category object recognition, localization and tracking using spatio-temporal context.” in *ECCV*, 2008.

[26] M. Schmidt. Ugm: Matlab code for undirected graphical models. [Online]. Available: http://www.di.ens.fr/mschmidt/Software/UGM.html

[27] W. S. Sarle. (2002) ftp://ftp.sas.com/pub/neural/faq2.html.

[28] G. C. Poggio, “Incremental and decremental support vector machine learning,” in *NIPS*, 2001.

[29] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, “Incremental learning for robust visual tracking,” *IJCV*, 2008.

[30] M. Pei, Y. Jia, and S.-C. Zhu, “Parsing video events with goal inference and intent prediction,” in *ICCV*, 2011.

[31] M. Rohrbach, S. Amin, M. Andriluka, and B. Schiele, “A database for fine grained activity detection of cooking activities,” in *CVPR*, 2012.

[32] K. K. Reddy and M. Shah, “Recognizing 50 human action categories of web videos,” *MVAP*, 2012.

[33] P. F. Felzenszwalb, R. B. Girshic, and D. McAllester. Discriminatively trained deformable part models, release 4. [Online]. Available: http://people.cs.uchicago.edu/pff/latent-release4/

[34] I. Laptev, “On space-time interest points,” *IJCV*, 2005.

[35] R. Chaudhry, A. Ravichandran, G. Hager, and R. Vidal., “Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions,” in *CVPR*, 2009.

[36] J. Yang, K. Yu, Y. Gong, and T. Huang, “Linear spatial pyramid matching using sparse coding for image classification,” in *CVPR*, 2009.

[37] N. Dalal, B. Triggs, and C. Schmid, “Human detection using oriented histograms of flow and appearance,” in *ECCV*, 2006.

[38] M. Amer and S. Todorovic, “Sum-product networks for modeling activities with stochastic structure,” in *CVPR*, 2012.

[39] S. Sadanand and J. Corso, “Action bank: A high-level representation of activity in video,” in *CVPR*, 2012.