Video Event Recognition with Deep Hierarchical Context Model

Xiaoyang Wang and Qiang Ji
Dept. of ECSE, Rensselaer Polytechnic Institute, USA
{wangx16,jiq}@rpi.edu

Abstract

Video event recognition still faces great challenges due to large intra-class variation and low image resolution, in particular for surveillance videos. To mitigate these challenges and to improve the event recognition performance, various context information from the feature level, the semantic level, as well as the prior level is utilized. Different from most existing context approaches that utilize context in one of the three levels through shallow models like support vector machines, or probabilistic models like BN and MRF, we propose a deep hierarchical context model that simultaneously learns and integrates context at all three levels, and holistically utilizes the integrated contexts for event recognition. We first introduce two types of context features describing the event neighborhood, and then utilize the proposed deep model to learn the middle level representations and combine the bottom feature level, middle semantic level and top prior level contexts together for event recognition. The experiments on state of art surveillance video event benchmarks including VIRAT 1.0 Ground Dataset, VIRAT 2.0 Ground Dataset, and the UT-Interaction Dataset demonstrate that the proposed model is quite effective in utilizing the context information for event recognition. It outperforms the existing context approaches that also utilize multiple level contexts on these event benchmarks.

1. Introduction

Video event recognition aims to recognize the spatio-temporal visual patterns of events from videos. In recent years, event recognition has attracted growing interest from both academia and industry [29, 15]. However, recognizing events in surveillance videos is still quite challenging, largely due to the tremendous intra-class variations of events caused by visual appearance differences, target motion variations, viewpoint change and temporal variability. Moreover, the low image resolution, object occlusion, and illumination change in surveillance videos further aggregate the event recognition challenges.

To mitigate these challenges, various work [40, 35] in event recognition turns the focus to context. Context can be regarded as information that is not directly related to event recognition task, but it can be utilized to improve the traditional data-driven and target-centered event recognition. As summarized in [35] recently, existing context approaches generally utilize contexts from feature level [32, 40], semantic level [3, 36], or prior level [28, 33, 34] of the recognition system. From these three levels, context can provide information on the circumstance and environment within which the event occurs, and hence can support the event recognition. Since there is little work in combining different contexts from multiple levels, the work in [35] studies to simultaneously integrate the three levels of context through a Bayesian network (BN) based hierarchical model.

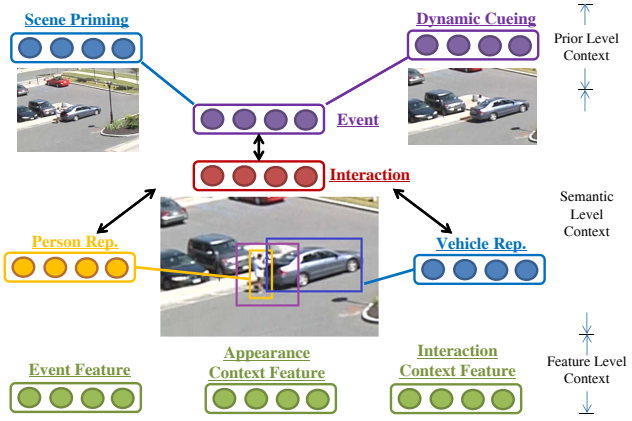


Figure 1. Illustration of our approach for learning and integrating feature level, semantic level and prior level contexts. In the figure, "Person Rep." and "Vehicle Rep." represent the learned middle level representations for person and vehicle respectively.

However, the existing context approaches for event recognition either utilize context directly as feature inputs [32] to classifiers like support vector machines, or incorporate context through traditional probabilistic graphical models like BN [3, 35], Markov random field (MRF) [36], or latent topic model [12]. There is little work studying utilizing probabilistic deep models like deep Boltzmann machines (DBMs) [24] to capture contexts for event recognition. Furthermore, the probabilistic deep models have

great potentials to systematically incorporate multiple levels of contexts because of their multi-level deep structure, the built-in capability of probabilistic reasoning, and the integration of hidden units to synthesize higher level representation than the raw input alone.

Hence, in this work, we propose a DBM based deep hierarchical context model to learn the middle level representation of event semantic components, and to systematically learn and integrate the contexts from the feature level, the semantic level, as well as the prior level, and holistically utilize the integrated contexts for event recognition. To this goal, we first propose two types of context features including the appearance context feature and the interaction context feature at the feature level. These feature level contexts exploit the contextual neighborhood of event instead of the target as [1]. Next, we introduce the deep hierarchical context model that integrates the proposed context features with semantic level context and prior level context. In the proposed model, the semantic level context captures the interactions among the entities of an event (e.g. person and object), and the prior level context includes the scene priming and dynamic curing. Different from the existing three level context model [35] that incorporates existing contexts through a BN model, our proposed method utilizes the deep structure to discover and capture the middle level representations of event components as a novel semantic context, and holistically integrate them with the proposed feature level contexts and the prior level contexts. The proposed model leads to improved performance over state of art context methods [40, 35] that also integrate multiple levels of contexts on several event recognition benchmarks.

In summary, the major contributions of the proposed work can be listed as follows: 1) we propose a deep hierarchical context model to learn middle level context representation and holistically combine the feature, semantic, and prior level contexts; 2) our semantic level context captures the interactions among event entities; 3) we propose two types of context features that capture the appearance of nearby non-target objects and their interactions.

2. Related Work

Contexts in Three Levels. In the event recognition system, contextual information can exist at three levels including *feature level* [32, 40], *semantic level* [3, 36], and *prior level* [28, 33]. At *feature level*, the context features capture feature information regarding the context. Here, Wang et al. [32] propose a contextual feature capturing interactions between interest points in spatio-temporal domains from both local and neighborhood. Also, Zhu et al. [40] propose both the intra-activity and inter-activity context feature descriptors for activity recognition. At *semantic level*, context captures interactions among event and its components. Here, Gupta et al. [3] present a BN based approach for joint action understanding and object perception. Yao et al. [36] utilize an MRF model to capture mutual context of activities, objects and humans poses. At *prior level*, the context captures the prior information of events. Here, the scene prior information [28] is widely used [14, 33] for event recognition.

Integrating Multiple Levels of Contexts. Several approaches explore to integrate multiple levels of contexts for event recognition. Specifically, Sun et al. [26] extract the point-level context feature, the intra-trajectory context feature and the inter-trajectory context feature, and combine the features using a multiple kernel learning model. These multiple level contexts are all in the feature level. Li et al. [12] build a Bayesian topic model to capture the semantic relationships among event, scene and objects. This model essentially captures the semantic level context, and incorporates the hierarchical priors in the model. Zhu et al. [40] exploit feature level contexts and semantic level contexts among events simultaneously through the structural linear model. Also, the BN hierarchical context model by Wang and Ji [35] integrates the feature level, semantic level and prior level context simultaneously. Different from the existing multi-level context approaches, we use a deep model to systematically learn and integrate multiple levels of contexts. The hidden units in our model represents a set of middle level presentations for each event component.

Deep Models. In recent years, deep models including probabilistic models like deep belief networks (DBNs) [6] and deep Boltzmann machines (DBMs) [24, 23], as well as non-probabilistic models like the stacked auto-encoders [2, 30] and convolutional neural networks (ConvNets) [11] are used in different applications. For action and activity recognition, the ConvNets [7, 8], convolutional gated restricted Boltzmann machine [27], independent subspace analysis [10], and auto-encoder approaches [4] are developed. However, these action/activity deep models are generally data-driven and target-centered, without explicitly incorporating context information. Comparatively, our proposed deep context model utilizes the deep structure to explicitly capture the prior level, semantic level, and feature level contexts for event recognition.

There is little work studying utilizing deep models to capture contexts for visual recognition tasks. He et al. [5] utilize a RBM model to capture the pixel level interactions for image labeling. Also, Zeng et al. [38] build a multi-stage contextual deep model that uses the score map outputs from multi-stage classifiers as contextual information for the pedestrian detection deep model. However, both these two models are not designed to capture three levels of contexts, and are not for event recognition. As far as we are concerned, there is no existing event recognition research that simultaneously utilizes three levels of contexts through a deep probabilistic model.

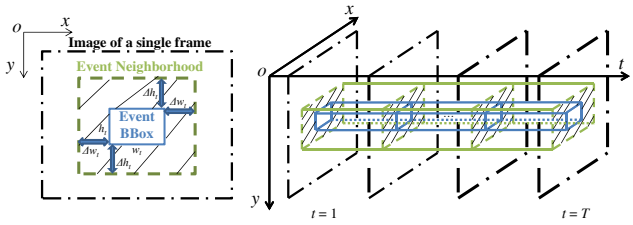
3. Contexts in Three Levels

3.1. Feature Level Contexts

We develop two types of context features including the *appearance context feature* and the *interaction context feature* extracted from the event neighborhood defined below.

3.1.1 Event Neighborhood

Suppose the event bounding box can be denoted as $\{(x_t, y_t, w_t, h_t)_{t=1}^{T}\}$ from frame 1 to T. (x_t, y_t) represents the upper-left corner point. w_t and h_t denote the width and height. As shown in Figure 2(a), we further extend the event bounding box to a larger rectangle by increasing the width with Δw_t on left and right side, and increasing the height with Δh_t on top and bottom side for frame t. The *event neighborhood* of an event in frame t is then the region within the extended rectangle but outside of the event bounding box rectangle, as represented by the shaded region of Figure 2(a). And, Figure 2(b) further illustrates the event neighborhood over T frames.



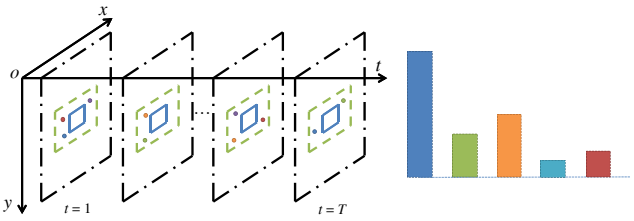
(a) Event Neighborhood (b) Event Neighborhoods over Frames

Figure 2. The definition of event neighborhood, where the blue rectangle indicates the event bounding box, and the dashed green rectangle is the extended rectangle. The shaded region within the extended rectangle but outside of the event bounding is the spatial neighborhood. The event neighborhood is the spatial neighborhoods over frames.

To set the values of Δh_t and Δw_t, we use the ratio λ to determine the relative scope of the event neighborhood with respect to the event bounding box size. Given the width w_t and height h_t of event bounding box rectangle, the ratio satisfies $\lambda = \frac{\Delta h_t}{h_t} = \frac{\Delta w_t}{w_t}$.

3.1.2 Appearance Context Feature

The appearance context feature captures the appearance of contextual objects, which are defined as nearby non-target objects located within the event neighborhood. Since our event neighborhood is a direct spatial extension of the event bounding box, it would naturally contain both the contextual objects and the background. To efficiently extract and capture the contextual objects from the background, we utilize SIFT descriptor [13] to detect the SIFT key points in the event neighborhood for each frame as shown in Figure 3(a).



(a) SIFT key points in Neighborhood (b) BOW based histogram

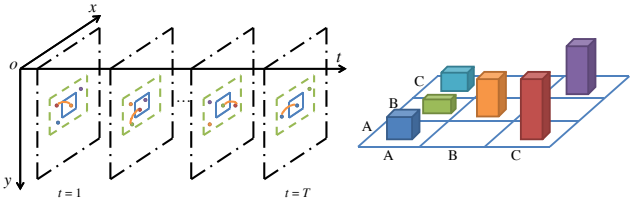
Figure 3. Extracting appearance context feature from the event neighborhood. (a) SIFT key points in the neighborhood of each frame; (b) BOW based histogram feature.

The SIFT descriptor extracts 128 dimensional scale and orientation invariant local textual feature surrounding each detected key point. This feature provides an appearance based description of the contextual objects. For each event sequence, we use bag-of-words (BOW) method to encode these SIFT descriptors into a histogram based context feature with dimension K.

3.1.3 Interaction Context Feature

The interaction context feature captures the interactions between event objects and contextual objects as well as among contextual objects. The contextual objects are represented by the SIFT key points extracted in the event neighborhood as discussed in Section 3.1.2. We use SIFT key points detected within the event bounding box to further represent the event objects.

Then, the k-means clustering is applied to the 128 dimensional features of key points in both within the event bounding box and event neighborhood of all training sequences to generate a joint dictionary matrix $\mathcal{D}_I$ with K' words. With this dictionary, the key points inside and outside the event bounding box can be assigned to the same set of words. As shown in Figure 4, we use a 2D histogram to capture the co-occurrence frequencies of words inside and outside the event bounding box over frames.



(a) Interactions of key points (b) 2D histogram

Figure 4. Extracting interaction context feature with a 2D histogram that captures the co-occurrence frequencies of words of event objects and contextual objects.

We normalize this 2D histogram to ensure all elements in the matrix sum to 1. They then constitute as the interaction context feature we use for event recognition after reshaped into a K'^2 dimensional vector.

3.2. Semantic Level Contexts

The semantic level contexts stand for the semantic interactions among event entities. Since both the person and object are two important entities of an event, the semantic level contexts for this work capture the interactions between event, person and object.

Suppose we have $\mathcal{K}$ types of events to recognize. We use a $\mathcal{K}$ dimensional vector $\mathbf{y}$ with binary units to represent the event label through the 1-of-$\mathcal{K}$ coding scheme, in which the event belonging to class $\mathcal{C}_k$ would be a vector with element k as "1" and all the remaining elements as "0". We use the binary hidden units $\mathbf{h}_p$ and $\mathbf{h}_o$ to represent the middle level representations of person and object.

Semantic context modeling. The model structure shown in Figure 5 is used to capture the semantic level contexts. In this structure, the event label vector $\mathbf{y}$ lies in the top layer, and the hidden units $\mathbf{h}_p$ for person and $\mathbf{h}_o$ for object both lie in the bottom layer. Another set of hidden units $\mathbf{h}_r$ standing in the intermediate layer is incorporated to connect the units of event, person and object. Here, every single hidden unit in $\mathbf{h}_r$ is connected to all the units in $\mathbf{h}_p$, $\mathbf{h}_o$, and $\mathbf{y}$. In such way, the global interactions among units from person, object, as well as the event label are captured through the intermediate hidden layer $\mathbf{h}_r$.

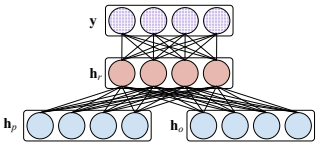


Figure 5. The model capturing semantic level contexts, where $\mathbf{h}_p$ and $\mathbf{h}_o$ are the first layer hidden units representing person and object middle level representation, $\mathbf{h}_r$ is the second layer hidden units capturing interactions, and $\mathbf{y}$ stands for the event class label.

Combining with observations. The observation vectors for the event, person and object can be further added to the semantic level context model in Figure 5. It results in the context model as shown in Figure 6. The vectors $\mathbf{p}$ and $\mathbf{o}$ denote the person and object observation vectors as continuous STIP features. Both the person observation vector $\mathbf{p}$ and the object observation vector $\mathbf{o}$ are connected only to their corresponding hidden units $\mathbf{h}_p$ and $\mathbf{h}_o$ respectively. In this way, the middle level representations for person and object can be obtained from their corresponding observations. In addition, the event observation $\mathbf{e}$ as STIP event feature, and the context feature $\mathbf{c}$ introduced in Section 3.1 are directly connected to the event label $\mathbf{y}$.

The model in Figure 6 combines semantic contexts in middle level with context feature $\mathbf{c}$ in bottom level. This model is called the **Model-BM** context model, and is compared in the experiment section.

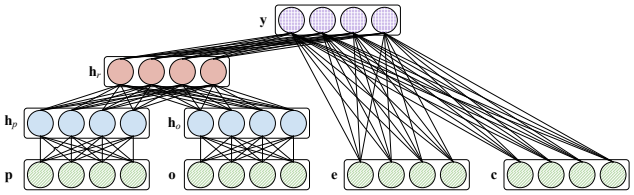


Figure 6. The model combining semantic level contexts with observations, where vectors $\mathbf{p}$ and $\mathbf{o}$ denote the person and object observations, $\mathbf{e}$ and $\mathbf{c}$ represent the event and context observations.

3.3. Prior Level Contexts

The prior level contexts capture the prior information of events. We utilize two types of prior contexts: the scene priming [28] and the dynamic cueing. The model can also be applied to other prior level contexts.

Scene priming. The scene priming context refers to the scene information obtained from the global image. It reflects the environment such as location (e.g. parking lot, shop entrance) and time (e.g. noon, dark) that can serve as prior to dictate whether certain events would occur. To capture the scene context as prior, we utilize the hidden units $\mathbf{h}_s$ to represent different possible scene states. As shown in Figure 7, each hidden unit in $\mathbf{h}_s$ is connected to all the elements within the event label vector $\mathbf{y}$. In this way, the state of the scene would have a direct impact to the event label. The observation vector $\mathbf{s}$ represents the GIST feature extracted from the global scene image. Elements in $\mathbf{s}$ is connected to each unit in $\mathbf{h}_s$ to provide global observation to the hidden scene states.

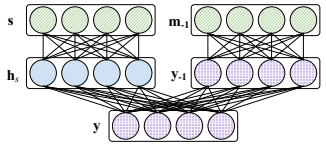


Figure 7. The model capturing prior level contexts, where $\mathbf{s}$ represents the global scene observation, $\mathbf{m}_{-1}$ denotes the recognition measurement of the previous event, $\mathbf{h}_s$ denotes the hidden units representing different possible scene states, $\mathbf{y}_{-1}$ denotes the previous event, and $\mathbf{y}$ stands for the current event class.

Dynamic cueing. The dynamic cueing context provides temporal support for the prediction of the current event given previous event. In this work, the previous event is represented by the $\mathcal{K}$ dimensional binary vector $\mathbf{y}_{-1}$ in the 1-of-$\mathcal{K}$ coding scheme. Moreover, $\mathbf{y}_{-1}$ is further connected to previous event measurement vector $\mathbf{m}_{-1}$ which denotes the recognition measurement of the previous event.

As shown in Figure 7, both $\mathbf{h}_s$ and $\mathbf{y}_{-1}$ provide top-down prior information for the inference of current event.

4. Deep Context Model Formulation

Given the contexts in three levels as introduced previously, we now discuss about the formulation of the proposed

deep hierarchical context model for integrating them.

4.1. Deep Model

In this section, we introduce the deep context model to systematically incorporate three levels of contexts. As shown in Figure 8, the model consists of six layers. From bottom to top, the first layer at bottom includes the target and contextual measurement vectors $\mathbf{p}$, $\mathbf{o}$, $\mathbf{e}$, and $\mathbf{c}$ that are visible in both learning and testing. The vectors $\mathbf{p}$ and $\mathbf{o}$ denote the person and object observations. And, the vectors $\mathbf{e}$ and $\mathbf{c}$ denote the event and context features. The second layer includes binary hidden units $\mathbf{h}_p$ and $\mathbf{h}_o$ representing middle level representations for person and object. On the third layer, the binary hidden units $\mathbf{h}_r$ are incorporated as an intermediate layer to capture the interactions between event, person and object. The fourth layer denoted by vector $\mathbf{y}$ represents the event label through the 1-of-$\mathcal{K}$ coding scheme. On the top two layers, the hidden units $\mathbf{h}_s$ represent the scene states, and vector $\mathbf{s}$ is the scene observation. Also, $\mathbf{y}_{-1}$ represents the previous event state, with its measurement as $\mathbf{m}_{-1}$. This model is essentially the combination of Model-BM and prior model in Figure 6 and 7 respectively.

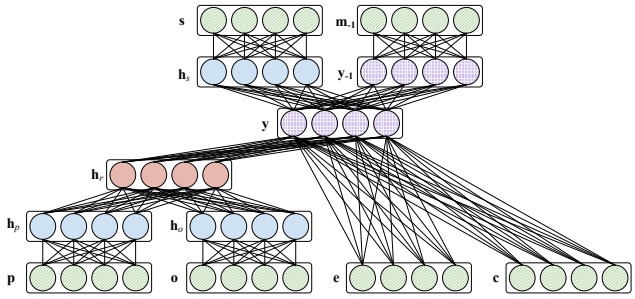


Figure 8. The proposed deep context model integrating feature level, semantic level and prior level contexts, where the shaded units represent the hidden units, the striped units represent the observed units that would be available both in training and testing, and the units in grid are event label units which are available in training and not available in testing.

The proposed model is an undirected model. With the given structure in Figure 8, the model energy function is:

$$
\begin{aligned}
&E(\mathbf{y},\mathbf{h}_r,\mathbf{h}_p,\mathbf{h}_o,\mathbf{p},\mathbf{o},\mathbf{e},\mathbf{c},\mathbf{y}_{-1},\mathbf{m}_{-1},\mathbf{h}_s,\mathbf{s};\theta) = -\tilde{\mathbf{p}}^\top\mathbf{W}^1\mathbf{h}_p \\
&-\tilde{\mathbf{o}}^\top\mathbf{W}^2\mathbf{h}_o - \mathbf{h}_p^\top\mathbf{Q}^1\mathbf{h}_r - \mathbf{h}_o^\top\mathbf{Q}^2\mathbf{h}_r - \mathbf{h}_r^\top\mathbf{L}\mathbf{y} - \tilde{\mathbf{e}}^\top\mathbf{U}^1\mathbf{y} - \\
&\tilde{\mathbf{c}}^\top\mathbf{U}^2\mathbf{y} - \mathbf{y}_{-1}^\top\mathbf{D}\mathbf{y} - \mathbf{h}_s^\top\mathbf{T}\mathbf{y} - \mathbf{m}_{-1}^\top\mathbf{F}\mathbf{y}_{-1} - \tilde{\mathbf{s}}^\top\mathbf{G}\mathbf{h}_s - \mathbf{b}_{\mathbf{h}_p}^\top\mathbf{h}_p \\
&-\mathbf{b}_{\mathbf{h}_o}^\top\mathbf{h}_o - \mathbf{b}_{\mathbf{h}_r}^\top\mathbf{h}_r - \mathbf{b}_{\mathbf{y}}^\top\mathbf{y} - \mathbf{b}_{\mathbf{h}_s}^\top\mathbf{h}_s - \mathbf{b}_{\mathbf{y}_{-1}}^\top\mathbf{y}_{-1} - \mathbf{b}_{\mathbf{m}_{-1}}^\top\mathbf{m}_{-1} \\
&+\sum_i \frac{(p_i - b_{\mathbf{p}i})^2}{2\sigma_{\mathbf{p}i}^2} + \sum_j \frac{(o_j - b_{\mathbf{o}j})^2}{2\sigma_{\mathbf{o}j}^2} + \sum_k \frac{(e_k - b_{\mathbf{e}k})^2}{2\sigma_{\mathbf{e}k}^2} \\
&+\sum_{i'} \frac{(c_{i'} - b_{\mathbf{c}i'})^2}{2\sigma_{\mathbf{c}i'}^2} + \sum_{j'} \frac{(s_{j'} - b_{\mathbf{s}j'})^2}{2\sigma_{\mathbf{s}j'}^2} \qquad (1)
\end{aligned}
$$

where $\mathbf{W}^1$, $\mathbf{W}^2$, $\mathbf{Q}^1$, $\mathbf{Q}^2$, $\mathbf{L}$, $\mathbf{U}^1$, $\mathbf{U}^2$, $\mathbf{T}$, $\mathbf{D}$, $\mathbf{F}$ and $\mathbf{G}$ are the weight matrices between the groups of visible or hidden units. Also, $\mathbf{b}_{\mathbf{h}_p}$, $\mathbf{b}_{\mathbf{h}_o}$, $\mathbf{b}_{\mathbf{h}_r}$, $\mathbf{b}_{\mathbf{y}}$, $\mathbf{b}_{\mathbf{h}_s}$, $\mathbf{b}_{\mathbf{y}_{-1}}$ and $\mathbf{b}_{\mathbf{m}_{-1}}$ are the bias terms for the discrete units. And, $\mathbf{b}_{\mathbf{p}}$, $\boldsymbol{\sigma}_{\mathbf{p}}$, $\mathbf{b}_{\mathbf{o}}$, $\boldsymbol{\sigma}_{\mathbf{o}}$, $\mathbf{b}_{\mathbf{e}}$, $\boldsymbol{\sigma}_{\mathbf{e}}$, $\mathbf{b}_{\mathbf{c}}$, $\boldsymbol{\sigma}_{\mathbf{c}}$ and $\mathbf{b}_{\mathbf{s}}$, $\boldsymbol{\sigma}_{\mathbf{s}}$ are the parameters for the continuous units, similar to those in Gaussian-Bernoulli RBM. We use θ to represent the whole model parameter set that includes all the parameters in the weight matrices and the bias terms.

For convenience, Equation 1 utilize vectors $\tilde{\mathbf{p}}, \tilde{\mathbf{o}}, \tilde{\mathbf{e}}, \tilde{\mathbf{c}}, \tilde{\mathbf{s}}$, which are the original observation vectors $\mathbf{p}, \mathbf{o}, \mathbf{e}, \mathbf{c}, \mathbf{s}$ divided by $\boldsymbol{\sigma}_{\mathbf{p}}, \boldsymbol{\sigma}_{\mathbf{o}}, \boldsymbol{\sigma}_{\mathbf{e}}, \boldsymbol{\sigma}_{\mathbf{c}}, \boldsymbol{\sigma}_{\mathbf{s}}$ respectively in each dimension. For instance, $\tilde{p}_i = p_i/\sigma_{\mathbf{p}i}$.

Given the energy function, the joint probability of all the variables $\mathbf{y}$, $\mathbf{h}_r$, $\mathbf{h}_p$, $\mathbf{h}_o$, $\mathbf{p}$, $\mathbf{o}$, $\mathbf{e}$, $\mathbf{c}$, $\mathbf{y}_{-1}$, $\mathbf{m}_{-1}$, $\mathbf{h}_s$, and $\mathbf{s}$ can be written as:

$$
\begin{aligned}
&P(\mathbf{y},\mathbf{h}_r,\mathbf{h}_p,\mathbf{h}_o,\mathbf{p},\mathbf{o},\mathbf{e},\mathbf{c},\mathbf{y}_{-1},\mathbf{m}_{-1},\mathbf{h}_s,\mathbf{s};\theta) = \frac{1}{Z(\theta)}\cdot \\
&\exp(-E(\mathbf{y},\mathbf{h}_r,\mathbf{h}_p,\mathbf{h}_o,\mathbf{p},\mathbf{o},\mathbf{e},\mathbf{c},\mathbf{y}_{-1},\mathbf{m}_{-1},\mathbf{h}_s,\mathbf{s};\theta)) \qquad (2)
\end{aligned}
$$

4.2. Model Learning

The model learning process learns the model parameter set θ which includes all the weight matrices and bias terms in Equation 1. With the training data $\{\mathbf{y}_i, \mathbf{y}_{-1,i}, \mathbf{p}_i, \mathbf{o}_i, \mathbf{e}_i, \mathbf{c}_i, \mathbf{s}_i, \mathbf{m}_{-1,i}\}_{i=1}^N$, these parameters can be learned by maximizing the log likelihood $L(\theta)$ as:

$$
\begin{aligned}
&\theta^* = \arg\max_\theta L(\theta) \\
&L(\theta) = \sum_{i=1}^N \log P(\mathbf{y}_i, \mathbf{y}_{-1,i}, \mathbf{p}_i, \mathbf{o}_i, \mathbf{e}_i, \mathbf{c}_i, \mathbf{s}_i, \mathbf{m}_{-1,i}; \theta) \qquad (3)
\end{aligned}
$$

The optimization in Equation 3 can be solved with stochastic gradient ascent method in which the gradients are calculated as:

$$
\frac{\partial L(\theta)}{\partial \theta} = \langle \frac{\partial E}{\partial \theta} \rangle_{P_{\text{data}}} - \langle \frac{\partial E}{\partial \theta} \rangle_{P_{\text{model}}} \qquad (4)
$$

where E is the model energy function defined in Equation 1. The operator $\langle \cdot \rangle_{P_{\text{data}}}$ denotes the data-dependent expectation, and $\langle \cdot \rangle_{P_{\text{model}}}$ denotes the model's expectation.

Since the exact computation of both expectations takes the time that is exponential to the number of hidden units, the exact maximum likelihood learning for this model is intractable. Here, we use the approximate learning methods to learn the model parameters. The learning process starts with the greedy layer-wise pretraining by learning a stack of RBMs [24] to initialize the model parameters. Then, the mean-field based variational inference approach is used to estimate the data-dependent expectation, and the Markov chain Monte Carlo (MCMC) based stochastic approximation procedure is used to estimate the model's expectation. The computational complexity of learning is $\mathcal{O}(RNM^2)$, where R is the learning iteration number, N is the training sample number, and M is number of nodes in the model.

4.3. Model Inference

Given a query event sequence with event observation vector $\mathbf{e}$, context observation vector $\mathbf{c}$, person observation vector $\mathbf{p}$, object observation vector $\mathbf{o}$, the global scene observation vector $\mathbf{s}$, and the previous event measurement $\mathbf{m}_{-1}$, the model can recognize the event category k^* by maximizing its posterior probability given all the observation vectors through Equation 5.

$$k^* = \arg\max_k P(y_k = 1|\mathbf{e}, \mathbf{c}, \mathbf{p}, \mathbf{o}, \mathbf{s}, \mathbf{m}_{-1}; \theta) \quad (5)$$

Computing this posterior probability requires marginalizing over all the hidden units in $\mathbf{h}_p$, $\mathbf{h}_o$, $\mathbf{h}_r$ and $\mathbf{h}_s$. Its exact calculation is intractable. However, the inference can be efficiently performed using the Gibbs sampling method. The method randomly initializes $\mathbf{h}_p$, $\mathbf{h}_o$ and $\mathbf{y}$, and then iteratively samples $\mathbf{h}_r$, $\mathbf{h}_p$, $\mathbf{h}_o$, $\mathbf{h}_s$, $\mathbf{y}_{-1}$ and $\mathbf{y}$ given the adjacent hidden or visible units. The sampled $\mathbf{y}$ instances are then used to calculate the corresponding marginal probability. More inference details are in supplementary material. The computational complexity for the model inference is $\mathcal{O}(CTM^2)$, where C is the Markov chain number, T is the chain length, and M is number of nodes in the model.

5. Experiments

To demonstrate the effectiveness of the proposed approach, we experiment on three event recognition benchmark datasets. The first two datasets are the VIRAT 1.0 Ground Dataset and VIRAT 2.0 Ground Dataset [15]. These two datasets are state of the art real world surveillance video datasets focusing on surveillance video events which include interactions between persons and objects.

The **VIRAT 1.0 Ground Dataset** includes around 3 hours of videos with six types of person-vehicle interaction events including *Loading a Vehicle* (LAV), *Unloading a Vehicle* (UAV), *Opening a Trunk* (OAT), *Closing a Trunk* (CAT), *Getting into a Vehicle* (GIV), and *Getting out of a Vehicle* (GOV). Videos in this dataset are recoded from different school parking lots. Here, we use half of the event sequences for training, and the rest sequences for testing.

The **VIRAT 2.0 Ground Dataset** includes over 8 hours of surveillance videos from school parking lot, shop entrance, outdoor dining area and construction sites. For this dataset, we also focus on the six types of person-vehicle interaction events as in VIRAT 1.0 Ground Dataset. Half of these event sequences are used for training, and the remaining sequences are used for testing.

The third dataset is the **UT-Interaction Dataset** [22]. This is a surveillance video dataset with person-person interaction events. It consists of six person-person interaction events including: *hand shaking*, *hugging*, *kicking*, *pointing*, *punching* and *pushing*. The dataset includes two sets, each with 10 video sequences in the length of around 1 minute. To compare with state of art, we use the standard 10-fold leave-one-out cross validation for evaluation on set 1.

5.1. Experiments on Proposed Feature Context

We first evaluate the effectiveness of the proposed appearance and interaction context features discussed in Section 3.1. The experiment is performed on the VIRAT 2.0 Ground Dataset [15] with the six person-vehicle interaction events. The baseline event feature is the STIP [9] extracted from event bounding box.

Given the baseline event feature, we test the proposed context features by combining both the appearance and interaction context features each with the baseline event feature. For neighborhood size, we set $\lambda = 0.35$. Both context features are in 100 dimensions. Also, we further test the performance of combining both the appearance and interaction context features with the baseline event feature. We test these configurations using the standard SVM classifier. The overall performance comparison is shown in Table 1.

Table 1. Average recognition accuracies of combining two types of context features with the baseline event features (i.e. STIP) on six events of VIRAT 2.0 dataset for event recognition.

%	STIP	STIP + App.	STIP + Int.	STIP + App. & Int.
Accuracy	41.74	47.87	47.54	**51.91**

App.: appearance context feature; Int.: interaction context feature.

From Table 1, we can see that combining either the appearance or the interaction context feature can already improve the performance of baseline event feature for event recognition. Combining two context features with the baseline event feature can further improve the recognition accuracy. In all, combining our proposed context features can improve the event recognition performance by over 10%.

5.2. Experiments on Proposed Context Model

After discussing the performances of the proposed appearance context feature and the interaction context feature, we proceed to demonstrate the effectiveness of the proposed context model that integrates the three levels of contexts. For this model, $\mathbf{h}_p$, $\mathbf{h}_o$, $\mathbf{h}_r$, and $\mathbf{h}_s$ has 50, 50, 100, and 20 hidden units respectively. These experiments are performed on VIRAT 1.0, VIRAT 2.0, and the UT-Interaction Datasets.

5.2.1 Baselines and State of Arts Compared

Three baseline approaches are used in our experiments to evaluate the effectiveness of the proposed context model. The first baseline uses the SVM classifier with the STIP event feature, and is denoted as **SVM-STIP**. This approach does not use any contexts for event recognition. The second

baseline denoted as **SVM-Context** concatenates the event feature with both the appearance and interaction context features, and also uses SVM as the classifier. It hence evaluates the effectiveness of the proposed context features. The third baseline is the **Model-BM** model in Figure 6 that simultaneously integrates feature level contexts and semantic level contexts. These three baseline approaches are compared with our proposed model that systematically integrates the feature, semantic and prior level contexts.

We also compare our results with state of art performances in VIRAT 1.0 and 2.0 Ground Datasets, as well as in the UT-Interaction Dataset. On the VIRAT 1.0 and VRIAT 2.0 Ground Datasets, performances of the following state of art methods are compared: the approach by Reddy et al. [19] utilizing the histogram of spatiotemporal gradients as event feature, the approach by Zhu et al. [40] that integrates feature and semantic level contexts through the structural event recognition model, and the approach by Wang and Ji [35] utilizing Bayesian network (BN) based hierarchical context model for integrating contexts.

On the UT-Interaction Dataset, we compare with several types of state of art approaches including the spatial-temporal relationship matching method by Ryoo and Aggarwal [21], the integral histogram of spatio-temporal feature approach by Ryoo [20], the Hough-voting approach by Waltisberg et al. [31], the propagative Hough voting method by Yu et al. [37], the segmental alignment based method by Shariat and Pavlovic [25], and the bag of spatio-temporal phrase based approach by Zhang et al. [39]. These approaches are generally target-centered event recognition approaches with efforts mainly focused on improving the event descriptors. Also, we compare with the poselet keyframing approach by Raptis and Sigal [18]. This approach learns a set of discriminative keyframes of the videos and capture the local temporal context between them.

5.2.2 Performance on VIRAT 1.0 Ground Dataset

We first compare our proposed model with our three baselines (SVM-STIP, SVM-Context and Model-BM) approaches on VIRAT 1.0 Ground Dataset. In Figure 9, we show the recognition accuracy for each event, and the average recognition accuracy over the six events.

In this comparison, the Model-BM baseline performs better than the SVM-Context baseline for over 8%. This result indicates that incorporating the semantic level context between event and the middle level representations of person and object can obviously improve the event recognition performance. More importantly, our proposed deep context model outperforms the three baselines for five of the six events. For the average recognition accuracy of the six events, the SVM-STIP reaches 39.91%, the SVM-Context reaches 53.21%, and the Model-BM reaches 62.15%. And, our proposed model performs the best at 69.88%. This is a 29% absolute improvement over the SVM-STIP, and a 16% absolute improvement over the SVM-Context.

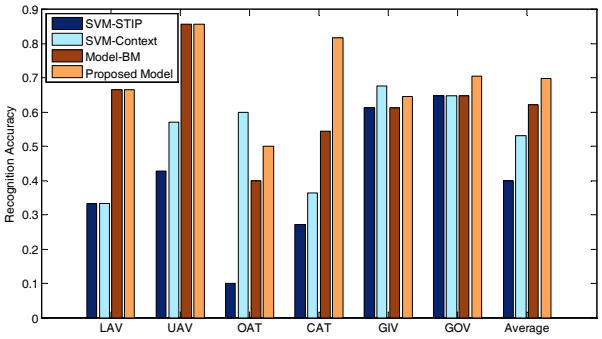


Figure 9. Results compared with baseline approaches on VIRAT 1.0 Ground Dataset. For the average recognition accuracy of six events, the SVM-STIP reaches 39.91%, the SVM-Context reaches 53.21%, and the Model-BM reaches 62.15%. Our proposed model performs the best at 69.88%.

Table 2 gives the comparison of our proposed approach with state of art performances on VIRAT 1.0 Ground Dataset. Here, our approach performs the best for three of the six events, and outperforms the BN based hierarchical context model approach [35] for over 4% in the overall performance. This result demonstrates that our proposed model is more effective than traditional BN based hierarchical context model in integrating three levels of context information for event recognition.

Table 2. The comparison of our proposed model with state of the art approaches on VIRAT 1.0 Ground Dataset.

Accuracy %	Reddy et al. [19]	Zhu et al. [40]	BN [35]	Proposed Model
LAV	10.0	52.1	**100**	66.67
UAV	16.3	57.5	71.4	**85.71**
OAT	20.0	**69.1**	50.0	50.00
CAT	34.4	72.8	54.5	**81.82**
GIV	38.1	61.3	45.2	**64.52**
GOV	61.3	64.6	**73.5**	70.59
Average	35.6	62.9	65.8	**69.88**

5.2.3 Performance on VIRAT 2.0 Ground Dataset

We further compare the performances of the proposed deep context model with the three baselines SVM-STIP, SVM-Context, Model-BM on VIRAT 2.0 ground dataset for the recognition of six person-vehicle interaction events. As shown in Figure 10, our proposed model can consistently outperform the baseline approaches for each event, and improves the average recognition accuracy from 41.74% (SVM-STIP), 51.91% (SVM-Context), 58.75% (Model-BM) to 66.45% (Proposed Model). This is close to 25% absolute improvement from the SVM-STIP, and close to 15% absolute improvement from the SVM-Context.

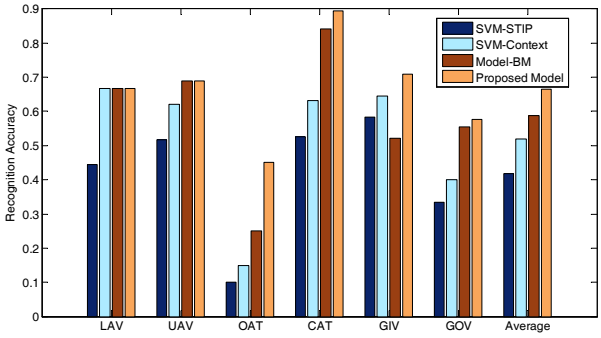


Figure 10. Results compared with baseline approaches on VIRAT 2.0 Ground Dataset. For the average recognition accuracy of six events, the SVM-STIP reaches 41.74%, the SVM-Context reaches 51.91%, and the Model-BM reaches 58.75%. Our proposed model performs the best at 66.45%.

In Table 3, we compare the recognition accuracies of the threes baselines, and the performance of [35] that also incorporates three levels of contexts through a BN model, as well as our proposed context model. From Table 3, we can find that our proposed context model, by incorporating three levels of contexts through a deep structure, can outperform the BN based three level context model in [35] by over 7%. This result shows the strength of the proposed model for event recognition.

Table 3. Average recognition accuracies of SVM-STIP, SVM-Context, Model-BM baselines and the proposed three level model compared with the state of the art three level context model in [35] for the recognition of six events on VIRAT 2.0 dataset.

Accuracy %	SVM-STIP	SVM-Context	Model-BM	Our Model	BN [35]
LAV	44.44	66.67	66.67	66.67	**77.78**
UAV	51.72	62.07	68.97	**68.97**	58.62
OAT	10.00	15.00	25.00	**45.00**	35.00
CAT	52.63	63.16	84.21	**89.47**	63.16
GIV	58.33	64.58	52.08	**70.83**	68.75
GOV	33.33	40.00	55.56	**57.78**	48.89
Average	41.74	51.91	58.75	**66.45**	58.70

In VIRAT 2.0 dataset, we also experiment with the baseline model excluding all hidden layers in Figure 8. This model reaches 52.54% average accuracy, which is slightly better than SVM-Context, but is around 14% worse than our proposed model. This result suggests that, with the introduction of hidden layers, the proposed deep model can effectively learn the salient representations from the input and improve recognition performance.

5.2.4 Performance on UT-Interaction Dataset

UT-Interaction Dataset is a surveillance video dataset with person-person interaction events. For this dataset, we first utilize the HOG feature based person detectors to detect the two persons within the event bounding box of the video. The STIP features for each of the two persons are then extracted accordingly. To compare with state of art performances on this dataset, we use the Fisher Vector encoding method [16, 17] for the STIP event feature. We experiment on the set 1 of this dataset. The overall performances of the SVM-STIP and our proposed model, as well as different state of the art performances are listed in Table 4.

Table 4. Overall recognition accuracies compared with state of art methods on set 1 of UT-Interaction dataset.

Method	Overall Accuracy
Ryoo and Aggarwal [21]	70.8%
Ryoo [20]	85.0%
Waltisberg et al. [31]	88%
Yu et al. [37]	93.3%
Raptis and Sigal [18]	93.3%
Shariat and Pavlovic [25]	91.57%
Zhang et al. [39]	95%
Our SVM-STIP	85.00%
Our Proposed Model	**95.00%**

The state of the art performances listed in Table 4 are mainly target-centered descriptor based approaches. Our SVM-STIP baseline, which is the most standard target-centered descriptor based approach, performs not as well as many of these state of the art approaches. However, our proposed context model can further improve the SVM-STIP baseline, and reaches the stat of art performance. This model also outperforms our baselines SVM-Context (88.33%) and Model-BM (93.33%). In addition, our approach outperforms the approach by Raptis and Sigal [18], which captures the temporal context between key frames.

6. Conclusion

In this paper, we propose a deep Boltzmann machine based context model to integrate the feature level, semantic level and prior level contexts. We first introduce new context features. Then, we introduce a deep context model that can learn the semantic context and to systematically incorporate contexts at different levels through learning and inference. The model is trained with mean-field based approximate learning method, and can be directly used to infer event classes through Gibbs sampling. We evaluate our model performance on VIRAT 1.0 Ground Dataset, VIRAT 2.0 Ground Dataset and the UT-Interaction Dataset for recognizing the real world surveillance video events with complex backgrounds. The results with the proposed deep context model show significant improvements over the baseline approaches that also utilize multiple levels of contexts. In addition, the proposed model also outperforms state of the art methods on benchmark datasets.

Acknowledgments

This work is funded in part by US Defense Advanced Research Projects Agency under grants HR0011-08-C-0135-S8 and HR0011-10-C-0112, and by the Army Research Office under grant W911NF-13-1-0395.

References

[1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, 2002.

[2] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle. Greedy layer-wise training of deep networks. In *Advances in Neural Information Processing Systems (NIPS)*, volume 19, pages 153–160, 2007.

[3] A. Gupta and L. Davis. Objects in action: An approach for combining action understanding and object perception. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2007.

[4] M. Hasan and A. K. Roy-Chowdhury. Continuous learning of human activity models using deep nets. In *European Conference on Computer Vision (ECCV)*, pages 705–720, 2014.

[5] X. He, R. S. Zemel, and M. A. Carreira-Perpinán. Multiscale conditional random fields for image labeling. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 695–702, 2004.

[6] G. Hinton, S. Osindero, and Y.-W. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554, 2006.

[7] S. Ji, W. Xu, M. Yang, and K. Yu. 3d convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):221–231, 2013.

[8] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1725–1732, June 2014.

[9] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2008.

[10] Q. V. Le, W. Y. Zou, S. Y. Yeung, and A. Y. Ng. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3361–3368, 2011.

[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[12] L.-J. Li and L. Fei-Fei. What, where and who? classifying events by scene and object recognition. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1–8, Oct. 2007.

[13] D. G. Lowe. Object recognition from local scale-invariant features. In *IEEE International Conference on Computer Vision (ICCV)*, volume 2, pages 1150–1157, 1999.

[14] S. Oh and A. Hoogs. Unsupervised learning of activities in video using scene context. In *International Conference on Pattern Recognition (ICPR)*, pages 3579–3582, Aug. 2010.

[15] S. Oh, A. Hoogs, A. Perera, N. Cuntoor, C.-C. Chen, J. T. Lee, S. Mukherjee, J. Aggarwal, H. Lee, L. Davis, E. Swears, X. Wang, Q. Ji, K. Reddy, M. Shah, C. Vondrick, H. Pirsiavash, D. Ramanan, J. Yuen, A. Torralba, B. Song, A. Fong, A. Roy-Chowdhury, and M. Desai. A large-scale benchmark dataset for event recognition in surveillance video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3153–3160, 2011.

[16] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2007.

[17] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *European Conference on Computer Vision (ECCV)*, pages 143–156, 2010.

[18] M. Raptis and L. Sigal. Poselet key-framing: A model for human activity recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2650–2657, 2013.

[19] K. K. Reddy, N. Cuntoor, A. Perera, and A. Hoogs. Human action recognition in large-scale datasets using histogram of spatiotemporal gradients. In *IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance (AVSS)*, pages 106–111, 2012.

[20] M. Ryoo. Human activity prediction: Early recognition of ongoing activities from streaming videos. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1036–1043, 2011.

[21] M. S. Ryoo and J. K. Aggarwal. Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1593–1600, 2009.

[22] M. S. Ryoo and J. K. Aggarwal. UT-Interaction Dataset, ICPR contest on Semantic Description of Human Activities (SDHA). `http://cvrc.ece.utexas.edu/SDHA2010/Human_Interaction.html`, 2010.

[23] R. Salakhutdinov and G. Hinton. An efficient learning procedure for deep boltzmann machines. *Neural Computation*, 24(8):1967–2006, 2012.

[24] R. Salakhutdinov and G. E. Hinton. Deep boltzmann machines. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 448–455, 2009.

[25] S. Shariat and V. Pavlovic. A new adaptive segmental matching measure for human activity recognition. In *IEEE International Conference on Computer Vision (ICCV)*, pages 3583–3590, 2013.

[26] J. Sun, X. Wu, S. Yan, L.-F. Cheong, T.-S. Chua, and J. Li. Hierarchical spatio-temporal context modeling for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2004–2011, 2009.

[27] G. W. Taylor, R. Fergus, Y. LeCun, and C. Bregler. Convolutional learning of spatio-temporal features. In *European Conference on Computer Vision (ECCV)*, pages 140–153, 2010.

[28] A. Torralba. Contextual priming for object detection. *International Journal of Computer Vision*, 53(2):169–191, 2003.

[29] P. Turaga, R. Chellappa, V. Subrahmanian, and O. Udrea. Machine recognition of human activities: A survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(11):1473–1488, 2008.

[30] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and composing robust features with denoising autoencoders. In *International conference on Machine learning (ICML)*, pages 1096–1103, 2008.

[31] D. Waltisberg, A. Yao, J. Gall, and L. Van Gool. Variations of a hough-voting action recognition system. In *Recognizing Patterns in Signals, Speech, Images and Videos*, pages 306–312. Springer, Berlin Heidelberg, 2010.

[32] J. Wang, Z. Chen, and Y. Wu. Action recognition with multiscale spatio-temporal contexts. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3185–3192, June 2011.

[33] X. Wang and Q. Ji. Incorporating contextual knowledge to dynamic bayesian networks for event recognition. In *International Conference on Pattern Recognition (ICPR)*, pages 3378–3381, 2012.

[34] X. Wang and Q. Ji. Context augmented dynamic bayesian networks for event recognition. *Pattern Recognition Letters*, 43:62–70, 2014.

[35] X. Wang and Q. Ji. A hierarchical context model for event recognition in surveillance video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2561–2568, 2014.

[36] B. Yao and L. Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17–24, June 2010.

[37] G. Yu, J. Yuan, and Z. Liu. Propagative hough voting for human activity recognition. In *European Conference on Computer Vision (ECCV)*, pages 693–706, 2012.

[38] X. Zeng, W. Ouyang, and X. Wang. Multi-stage contextual deep learning for pedestrian detection. In *IEEE International Conference on Computer Vision (ICCV)*, pages 121–128, 2013.

[39] Y. Zhang, X. Liu, M.-C. Chang, W. Ge, and T. Chen. Spatio-temporal phrases for activity recognition. In *European Conference on Computer Vision (ECCV)*, pages 707–721, 2012.

[40] Y. Zhu, N. Nayak, and A. Roy-Chowdhury. Context-aware modeling and recognition of activities in video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2491–2498, June 2013.