Joint Prediction of Activity Labels and Starting Times in Untrimmed Videos

Tahmida Mahmud[1], Mahmudul Hasan[2], Amit K. Roy-Chowdhury[1]
[1]University of California, Riverside, CA-92521, USA
[2]Comcast Labs, Washington, DC-20005, USA
tmahm001@ucr.edu, mahmud.ucr@gmail.com, amitrc@ee.ucr.edu

Abstract

Most of the existing works on human activity analysis focus on recognition or early recognition of the activity labels from complete or partial observations. Predicting the labels of future unobserved activities where no frames of the predicted activities have been observed is a challenging problem, with important applications, which has not been explored much. Associated with the future label prediction problem is the problem of predicting the starting time of the next activity. In this work, we propose a system that is able to infer about the labels and the starting times of future activities. Activities are characterized by the previous activity sequence (which is observed), as well as the objects present in the scene during their occurrence. We propose a network similar to a hybrid Siamese network with three branches to jointly learn both the future label and the starting time. The first branch takes visual features from the objects present in the scene using a fully connected network, the second branch takes previous activity features using a LSTM network to model long-term sequential relationships and the third branch captures the last observed activity features to model the context of inter-activity time using another fully connected network. These concatenated features are used for both label and time prediction. Experiments on two challenging datasets demonstrate that our framework for joint prediction of activity label and starting time improves the performance of both, and outperforms the state-of-the-arts.

1. Introduction

Human activity analysis is a widely studied computer vision problem. The solution to this problem has crucial impact on a wide range of practical applications such as video surveillance, human-computer interaction, autonomous navigation, active sensing, video indexing, active gaming, assisted living, etc. In spite of the enormous amount of research conducted in this area, the problem is still challenging due to the fundamental challenges inherent to the task, such as - the tremendous intra-class variance among the activities, huge spatio-temporal scale variation, target motion variations, etc. Moreover, low image resolution, object occlusion, illumination change and viewpoint change further aggravate these challenges.

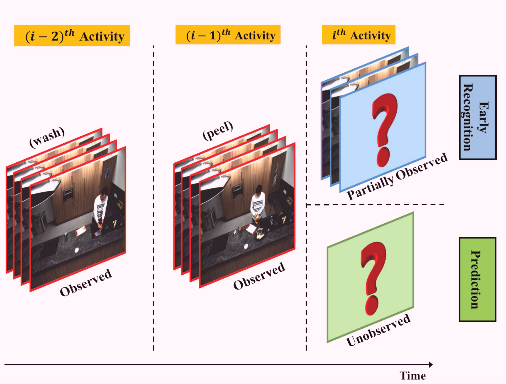

Figure 1. An example sequence of a video stream from MPII-Cooking Dataset [32]. Two related problems are explained here - early recognition of the i^{th} activity from partial observations of it, and prediction of its label from previously observed activities only. In the early recognition problem (top-right), the first few frames of the i^{th} activity (cut slices) have been observed. In the prediction problem (bottom-right), no frame of the i^{th} activity has been observed.

The majority of the existing works focus on the recognition of observed activities or early recognition of partially observed activities. In other words, they try to answer queries like *what happened before* or *what is happening right now*, whereas predicting the labels of future activities which have not yet been observed is a scarcely explored problem. In [2, 23, 24, 33, 42], by using the word 'prediction', these papers basically refer to the early recognition task, i.e., predicting the label of the ongoing activity where the first few frames of that activity have already been observed. However, in the prediction problem we are addressing, no observation is available beforehand. The difference between these two problems is illustrated in Figure 1. Predicting the future activity labels is critical in real life scenarios, where anticipatory response is required such

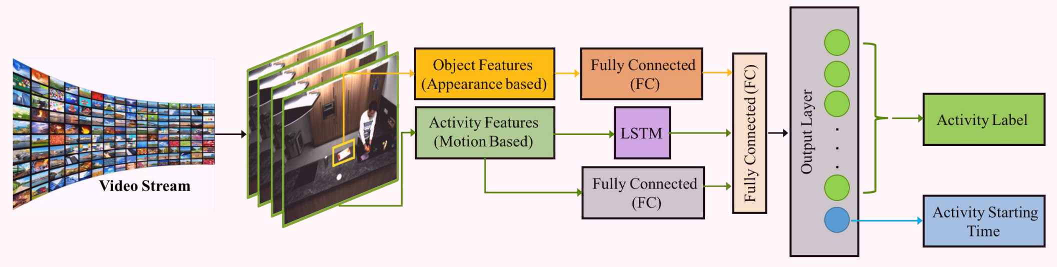

Figure 2. Overview of our approach. For joint prediction, both activity features (motion-based) from previous activities and object featues present in the scene are used for training. Please refer to Section 3.2 for details.

as active sensing and autonomous navigation. For example, it can help autonomous vehicles to decide how to maneuver depending on the next predicted activity and its time of occurrence, or assist robots to make future decisions. There are only a few approaches [3, 19] which perform label prediction on real-life activity datasets like VIRAT [29]. To the best of our knowledge, only one work [26] in the video analysis community addresses the problem of predicting the starting time of future unobserved activities.

1.1. Overview of the Proposed Approach

In this paper, for a video observed up to a particular time, we present an integrated approach that can answer two important questions regarding its unobserved portion: *what will happen next* and *when will it happen*, i.e., we predict the **labels** and the **starting times** of future unobserved activities in both coarse (VIRAT Ground Dataset [29]) and fine grained activity datasets (MPII-Cooking Dataset [32]). We pose this as a joint (label and starting time) prediction task because the problems of predicting the label and the starting time of unobserved activities are closely related and handling them together is intuitive. For example, in MPII-Cooking Dataset, 'cut slices' can be followed by two probable activities: 'spice' or 'take out from drawer'. Usually, 'spice' takes place immediately after 'cut slices'; but if there is a delay, then 'take out from drawer' happens before.

Detailed overview of our proposed framework is illustrated in Figure 2. We developed a deep network by merging three branches: one with two fully connected layers, another with two LSTM layers and the last one with another two fully connected layers. Finally, we add another fully connected layer to the output of this merged network. The two fully connected layers in the first branch are trained on the features of the objects present in the last observed portion of the scene, the LSTM layers are trained on the visual activity features of the previously observed sequential activities to exploit the context of long term sequential dependency and the two fully connected layers in the third branch are trained on the visual activity features of the last observed activity to model the context of inter-activity time based on the last observed activity label. So, the entire network is trained on both the previous activity features and the features of the objects present in the scene. In the output layer, we use the first few (equal to the number of activity classes) nodes as the logistic regression nodes for label prediction and the last node as a regression node for starting time prediction exploiting the concatenated features. The logistic regression nodes assign different probabilities to the future activity labels from which the label with the highest probability is chosen and the regression node provides the inter-activity time between the future activity and the last observed activity from which the starting time of the future activity is obtained. The motivation behind incorporating different context attributes is explained in Section 3.1 with ablation study provided in Sections 4.3 and 4.4. Our **main contribution** is that we propose a novel architecture that jointly models sequential relationships of the activities, scene context and inter-activity time context in order to predict the future activity labels as well as their starting times.

2. Related Works

Our work involves the following areas of interest: activity recognition, future activity label prediction, future activity starting time prediction, and Long Short-Term Memory (LSTM) network. We will review some relevant papers from these areas.

Activity Recognition. Activity recognition approaches based on hand-crafted visual features can be divided into three categories: low-level local feature based methods leveraged on interest point [22], mid-level feature based methods leveraged on tracking and pose analysis [27], and high-level semantic attribute based methods [34]. We would like to refer to article [17] and [31] for a comprehensive review of the state-of-the-art approaches. Most of the traditional approaches rely on hand-engineered local features (e.g., STIP, SIFT-3D, HOG-3D, iDT). However, supervised and unsupervised learning of meaningful hierarchical features from deep neural networks (i.e., autoencoder, sparse coding, and convolutional neural networks) have shown huge success over hand-engineered features recently. C3D feature learned with 3D Convolutional Networks is now the state-of-the-art spatio-temporal feature for video and has been shown to achieve best recognition accuracy in activity recognition tasks [38]. Moreover, methods which consider visual context, i.e., the relationships between different activities and objects in the scene, have been successful for recognition. In [45], object and human pose were used as context. In [4] and [21], group context was used for collective activity recognition. In [7, 16, 43], contextual information has been incorporated with deep networks to improve recognition accuracy. Context has also been shown to be useful for efficiently learning the models [13].

Future Activity Label Prediction. There have been a few works which predict the label of the future unobserved activity such as approaches using semantic scene labeling [19], Probabilistic Suffix Tree (PST) [23], augmented-Hidden Conditional Random Field (a-HCRF) [44], Markov Random Field (MRF) [3], kernel-based reinforcement learning [15], max-margin learning [20], and deep network [40]. Among these, only [3, 19] perform prediction, without any observation of the activity to be predicted, in the label space. In [40], where visual representation of images are predicted and then recognition algorithm is applied, actions can be anticipated only upto one second in the future.

Future Activity Starting Time Prediction. Predicting the starting times of future unobserved activities is a new research problem in the video understanding community. Although, there are some relevant works [25, 46] in other fields, to the best of the our knowledge, there is only one relevant work [26] in the domain of video analysis which is one of our previous works where we modeled the inter-activity times using a Log-Gaussian Cox Process (LGCP). Our new approach outperforms this baseline model.

Long Short-Term Memory (LSTM) Network. Unlike traditional neural networks, Recurrent Neural Network (RNN) has the capability of allowing information to be passed from one step of the network to the next using the loops inherent to their structure. However, in practice, RNNs cannot handle long-term dependencies, primarily because of the vanishing and exploding gradient problem. To overcome the challenge of handling long-term dependency, a special type of RNN called LSTM (Long Short-Term Memory) was introduced in [14]. LSTMs have achieved impressive performance in different sequence learning problems [8, 11, 30, 36, 39]. Its ability to capture long-range dependencies makes it a perfect tool for long-term context incorporation.

3. Methodology

3.1. Role of Different Context Attributes

In real life scenarios, it is observed that activities follow fixed temporal sequences. Therefore, previous activities can provide useful information about the upcoming ones which can be referred to as **sequential activity context**. Activities are also characterized by the objects present in the scene during the time of their occurrence which can be referred to as **scene context**. For many activities, predicting the future has multiple plausible options. To deal with this specific ambiguity, we take scene context into account along with the sequential information. Thus combining the information obtained from these two different context attributes (temporal sequence and spatial objects), we infer about future unobserved activities. For example, if three sequential activities in a video are ‘wash objects’, ‘peel’ and ‘cut slices’, then there may be two probable choices for the next activity label: ‘spice’ or ‘put in bowl’ (based on two different training instances). But a bowl present in the scene would increase the possibility of the latter choice. Several research works on activity recognition [4, 7, 16, 21, 43, 45, 47] and prediction [3] have shown significant performance improvement by using such context information which are also known as context-aware approaches. Most of the existing works have graphical model based approaches for context incorporation. However, they are not very suitable to handle the context of long-term dependency. As mentioned before, LSTM is a popular choice for sequential context incorporation. LSTM networks are straightforward to fine-tune end-to-end and can handle sequential data of varying lengths. So, we use LSTM to incorporate sequential activity context. However, for including the scene context, there is no need for handling such sequential dependency and fully connected layers can capture this efficiently.

The inter-activity time between different activities depends on their labels. For example, it is obvious from our experience that ‘peel’ or ‘cut slices’ takes more time than ‘wash objects’. Thus, by observing the previous activity features we can infer about the difference between the starting time of the observed activity and the future unobserved activity referred to as **inter-activity time context**.

3.2. Network Architecture

Our proposed architecture and the basic idea of the problem are shown in Figure 3. For our case, the LSTM is used to solve a sequential input, static output problem. We use the activity features extracted from three (chosen empirically) previously observed activities as the LSTM input. Increasing the sequence length does not improve the prediction accuracy significantly (see Parameter Sensitivity in Section 4.3 for details). We use a two-layer (chosen empirically) LSTM in the second branch with 256 memory units in each layer. The input of the two (chosen empirically) fully connected layers in the first branch are the visual features extracted from the objects present in the scene with 256 nodes in each layer. The input of the two (chosen empirically) fully connected layers in the third branch are the activity features extracted from the last observed activity with 256 nodes in each layer as well. Finally, the outputs from these three branches are tied together and another fully con-

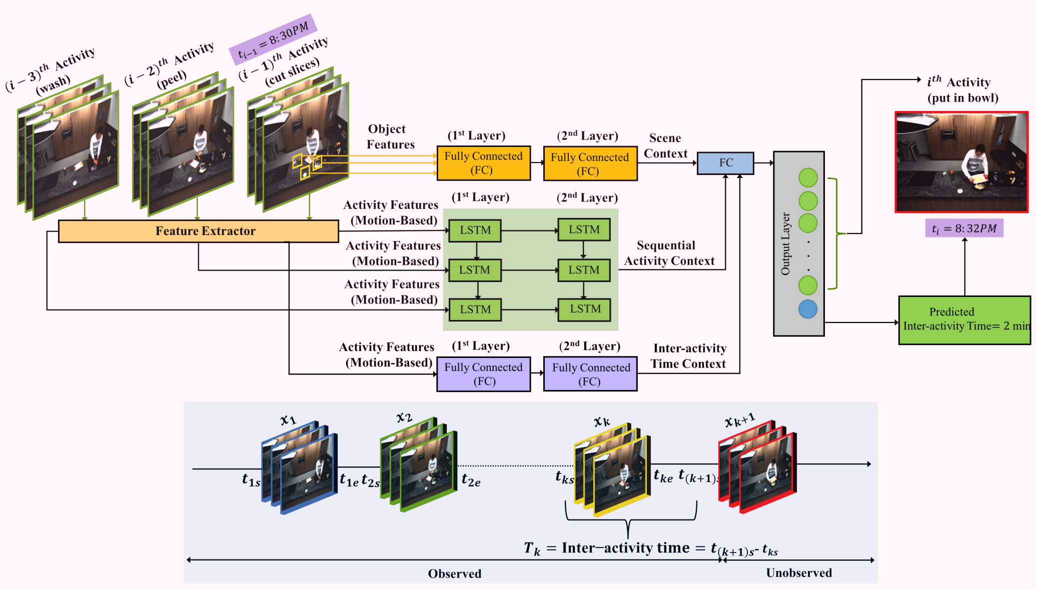
$(i-3)^{th}$ Activity (wash)
$(i-2)^{th}$ Activity (peel)
$t_{i-1} = 8{:}30PM$
$(i-1)^{th}$ Activity (cut slices)
Object Features
(1st Layer)
Fully Connected (FC)
(2nd Layer)
Fully Connected (FC)
Scene Context
FC
Output Layer
i^{th} Activity (put in bowl)
$t_i = 8{:}32PM$
Predicted Inter-activity Time= 2 min
Feature Extractor
Activity Features (Motion-Based)
(1st Layer)
(2nd Layer)
LSTM
LSTM
Activity Features (Motion-Based)
LSTM
LSTM
Activity Features (Motion-Based)
LSTM
LSTM
Sequential Activity Context
Activity Features (Motion-Based)
(1st Layer)
Fully Connected (FC)
(2nd Layer)
Fully Connected (FC)
Inter-activity Time Context
x_1
x_2
x_k
x_{k+1}
t_{1s}
t_{1e} t_{2s}
t_{2e}
t_{ks}
t_{ke} $t_{(k+1)s}$
T_k = Inter−activity time = $t_{(k+1)s}$- t_{ks}
Observed
Unobserved

Figure 3. Proposed architecture for future activity label prediction. The top two fully connected layers (yellow) incorporate the scene context which use object features as input. The two LSTM layers (green) are used to incorporate the sequential activity context which use motion-based features as inputs. The bottom two fully connected layers (purple) are used to incorporate inter-activity time context which use the last observed activity features (motion-based) as input. There is a fully connected layer (blue) where all these layers are merged together. The output layer (gray) performs the final prediction, where the first few nodes (green) are used as the logistic regression nodes for label prediction and the last node (blue) is used as the regression node for starting time prediction. In the problem description figure (bottom), activities have starting times (t_{1s}, t_{2s}, ..., t_{ks}) and ending times (t_{1e}, t_{2e}, ..., t_{ke}). We want to predict the starting time $t_{(k+1)s}$, of the $(k+1)^{th}$ activity by predicting the inter-activity time T_k.

nected layer is added on top of it. The merging combines the effect of different context attributes. In the output layer, the first few (equal to the number of activity classes) nodes are used as the logistic regression nodes for label prediction and the last node is used as a regression node for starting time prediction.

3.3. Model Training Approach

We use the popular open source deep learning package Keras [5] with TensorFlow [1] in the backend which has ready-to-use implementations of LSTM and fully connected layers. The network is trained on a NVIDIA Tesla K40 GPU. The input sequences for the LSTM are chosen in a sliding window manner with a stride of one for data augmentation. For example, to predict the i^{th} activity label, activity features extracted from the $(i-1)^{th}$, $(i-2)^{th}$ and $(i-3)^{th}$ activities are used and for predicting the $(i+1)^{th}$ activity label, activity features extracted from the i^{th}, $(i-1)^{th}$ and $(i-2)^{th}$ activities are used and so on. We use ReLU activation function for all the fully connected layers. In output layer, we use softmax activation function in the logistic regression nodes for label prediction and ReLU activation function in the regression node for starting time prediction. The parameters of the entire network (both of the LSTM and the fully connected layers) are jointly optimized.

We take the summation of the following two losses to compute the final loss. One is the cross-entropy loss function which is defined as follows:

$$\begin{aligned} \mathcal{L}(\mathbf{X},\mathbf{Y}) \quad &= -\tfrac{1}{n}\textstyle\sum_{i=1}^{n}\sum_{j=1}^{c}\mathbf{1}(y^{(i)}=j) \\ &\quad \times \log p(y^{(i)}=j|\mathbf{x}^{(i)}) \end{aligned} \tag{1}$$

Here, $\mathbf{X} = \{\mathbf{x}^{(1)}, ..., \mathbf{x}^{(n)}\}$ is the set of input feature vectors in the training dataset, $\mathbf{Y} = \{y^{(1)}, ..., y^{(n)}\}$ is the corresponding set of labels for those input features, and $j = \{1, ..., c\}$ is the set of class labels. $\mathbf{1}(.)$ is an identity function. For a particular training instance, $\mathbf{x}^{(i)}$ represents the sequential activity features extracted from the previous three activities and the object features from the last observed portion of the scene.

Another is the mean squared loss function which is defined as follows:

$$\mathcal{L}(\mathbf{P},\mathbf{Q}) = \frac{1}{n}\sum_{i=1}^{n}(q^{(i)} - \hat{q}^{(i)})^2 \tag{2}$$

Here, $\mathbf{P} = \{\mathbf{p}^{(1)}, ..., \mathbf{p}^{(n)}\}$ is the set of input feature vectors in the training dataset, and $\mathbf{Q} = \{q^{(1)}, ..., q^{(n)}\}$ is the corresponding set of inter-activity times for those input features. $\hat{q}^{(i)}$ represents the predicted inter-activity time given input $p^{(i)}$ where the ground truth inter-activity time is $q^{(i)}$. For a particular training instance, $\mathbf{p}^{(i)}$ represents the activity features extracted from the last observed activity.

To optimize the network, we use a stochastic gradient descent with an adaptive sub-gradient method (Adam) [18] which is popular for its strong theoretical convergence guarantee and impressive history of empirical success. We also tested with Adagrad [10], Adamax [18], Nadam [9] and RMSProp [37] but empirically chose Adam. We use Dropout layer [35] with a probability of 0.2 after each layer to prevent overfitting. We use a batch size of 128 and a learning rate of 0.001. Our network converges roughly at 60 epochs.

4. Experiments

We conduct experiments on two challenging datasets: MPII-Cooking Dataset [32] (fine grained indoor activities) and VIRAT Ground Dataset [29] (coarse outdoor activities) to evaluate the performance of our proposed framework.

4.1. Datasets

MPII-Cooking Dataset. MPII-Cooking Dataset is a fine grained complex activity dataset where the participants interact with different tools, ingredients and containers to complete a recipe. It has 65 different cooking activities recorded from 12 participants. In total there are 44 videos with a length of more than 8 hours. The dataset contains a total of $5,609$ annotations [32].

VIRAT Ground Dataset. VIRAT Ground Dataset is a challenging human activity dataset which consists of 11 different activities recorded in natural outdoor scenes with background clutter. There are total 329 videos with a length of around 5 hours [29]. However, we use only 275 of them as some videos have incomplete annotations.

Detailed description of these datasets is available in the supplementary material. These datasets are untrimmed and have context information unlike the trimmed datasets popularly used for recognition tasks in activity analysis.

4.2. Features

For MPII-Cooking Dataset, we use the bag-of-word based Motion Boundary Histograms (MBH) [6] as activity features. According to [41], these features are extracted around densely sampled points and a codebook is generated using k-means clustering for these 4000 words long features. Scene context features (dimension of 212: 41 for tools, 117 for ingredients and 54 for containers) naturally exist in the dataset. For VIRAT Ground Dataset, we use C3D features [38] as activity features. Scene context features naturally exist in VIRAT Ground Dataset too. We use MBH features for MPII-Cooking Dataset as these features come with the dataset. For VIRAT Ground Dataset, we extract the C3D features as it does not come with any features. We report results for MPII-Cooking Dataset using C3D features as well.

4.3. Label Prediction Results

Objective. The main objective of these experiments is to analyze how well our framework can predict the labels of future unobserved activities.

Performance Measures. The evaluation metrics we use are: 1. multi-class precision (Pr), 2. multi-class recall (Rc), and 3. overall accuracy for top-1 match, top-2 matches and top-3 matches. For all these metrics, the higher value indicates better prediction performance.

Compared Methods. We compare our approach to different state-of-the-art methods. There is no existing method for predicting future activity labels for MPII-Cooking Dataset. Therefore, we compare with a recent recognition approach which estimates the labels of the *observed* activities using a combination of CNN and LSTM [28]. For VIRAT Ground Dataset, there is an existing graphical model based approach [3] and a semantic scene labeling based approach [19]. We compare our method with [3] but cannot compare with [19] because they use scene specific customized set of labels which are not annotated in the original dataset. We also compare with a state-of-the-art active learning based recognition approach which uses sparse autoencoder [12] and achieve higher accuracy.

Experimental Setup. For experiments on MPII-Cooking Dataset, we use five fold leave-one-person-out cross validation approach for the training-testing split and average our results over these five combinations. Among 12 subjects, we use 7 for training and 5 for testing. For each of the five training instances, we use 7 training subjects and 4 testing subjects for training, leaving 1 from that set for testing. This is done 5 times leaving 1 testing subject out and then the results are averaged. For experiments on VIRAT Ground Dataset, we use the first 170 videos for training and the rest of them for testing.

Results for MPII-Cooking Dataset. Comparison of our label prediction results on MPII-Cooking Dataset with state-of-the-art method is shown in Table 1. The method we compare to did not report all of the evaluation metrics we use- hence the missing values. It is seen that our method outperforms the recognition method proposed in [28]. This is not surprising because in recognition problems the network has to decide among all the activity classes whereas in the sequence learning based prediction task, the network needs to consider only a subset of classes which occurred in the training phase after that particular sequence. Using C3D features, we achieve Top-1 accuracy of 79.9%. The coherence in Top-1 accuracies using both MBH and C3D features indicates that our method is independent of any particular choice of feature.

Results for VIRAT Ground Dataset. Comparison of our label prediction results on VIRAT Ground Dataset with state-of-the-art methods is shown in Table 1. It is seen that our method outperforms the prediction method proposed

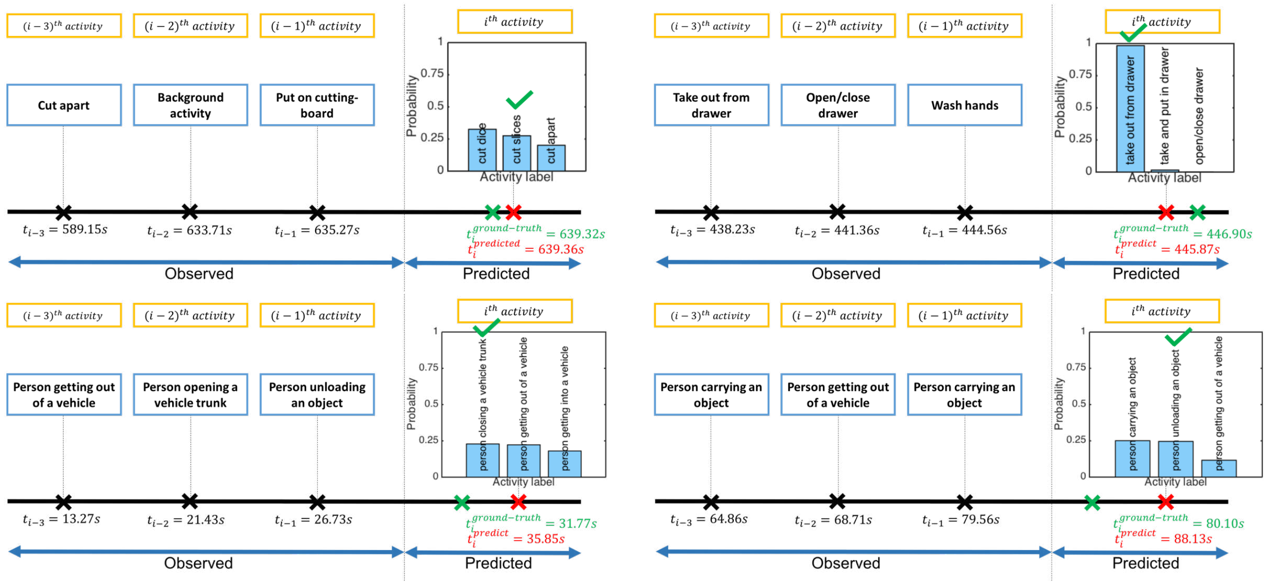

Figure 4. Four example activity sequences showing our label prediction results and time prediction results on MPII-Cooking Dataset (top row) and VIRAT Ground Dataset (bottom row). For time prediction, green × marks the ground truth starting time of the activity we are trying to predict, and red × marks the predicted time. For label prediction, top-3 matches are shown here and in most of the cases our top-1 match corresponds to the activity that actually happened (green tick).

MPII-Cooking Dataset [32]	Goal	Precision	Recall	Accuracy % (Top-1)	Accuracy % (Top-2)	Accuracy % (Top-3)
CNN + LSTM [28]	Recognition	34.8	51.7	-	-	-
Proposed Method	Prediction	70.7	66.5	80.1	90.0	93.7
VIRAT Ground Dataset [29]	Goal	Precision	Recall	Accuracy % (Top-1)	Accuracy % (Top-2)	Accuracy % (Top-3)
Sparse Autoencoder [12]	Recognition	-	-	54.2	-	-
Graphical Model [3]	Prediction	-	-	68.5	-	-
Proposed Method	Prediction	49.6	22.2	71.8	79.8	86.4

Table 1. Label prediction performance comparisons for MPII-Cooking Dataset and VIRAT Ground Dataset.

in [3]. We also achieve higher accuracy than the recognition method proposed by [12]. The intuition behind prediction accuracy being higher than recognition accuracy is explained above. However, for datasets like VIRAT Ground Dataset, where the number of classes is small, prediction accuracy is closer to recognition accuracy. Figure 4 depicts some example sequences showing both of our label prediction results and time prediction results on the two datasets.

Multiple Possibilities for Future Activity Label. One particular activity sequence can have multiple possible outcomes. For example, 'wash objects' and 'peel' can be followed by either 'cut apart' and 'cut slices'. As the network has been trained on both of these possible sequences (in one case the network has probably seen 'cut apart' as the next activity and in another case 'cut slices' as the next activity), it is hard to say precisely which is the next activity. Earlier we mentioned that in case of multiple possibilities, such as while choosing between 'spice' or 'put in bowl' after 'wash objects', 'peel' and 'cut slices', a bowl in the scene increases the probability of the activity label being the latter one. But in these types of closely related activities ('cut apart' and 'cut slices'), scene context cannot contribute much as both of the activities require a knife. This is why we present the top-3 choices with the associated probabilities for each of them. We did not go beyond top-3 because after that the probabilities become much lower as we found empirically. This is shown in the first example of Figure 4 where our network assigns almost equal probability to all of the possible future activities ('cut dice', 'cut slices', 'cut apart') but the activity which actually happened ('cut slices') is the one with the second highest probability. In spite of having these closely related ambiguous activities in the dataset, our top-1 match outperforms the baseline in terms of accuracy. Our method can also handle the case of predicting an unknown label (never seen in training) when the probability of none of the predicted future activities crosses a threshold.

Parameter Sensitivity. We empirically choose a sequence length of 3 for preceding activity features as sequence length of 2, 5, 7 and 9 give relatively lower accuracy for MPII-Cooking Dataset as shown in Table 2.

Top-1 Accuracy %				
Sequence Length 2	Sequence Length 3	Sequence Length 5	Sequence Length 7	Sequence Length 9
78.8	80.1	79.2	77.8	77.2

Table 2. Parameter sensitivity analysis for MPII-Cooking Dataset.

Ablation Study. Using only sequential activity context and scene context (eliminating inter-activity time context), we get relatively lower label prediction accuracy for MPII-Cooking Dataset than that of our proposed network. Similarly, using only sequential activity context and inter-activity time context (eliminating scene context), we get lower label prediction accuracy than that of our proposed network for MPII-Cooking Dataset. These ablation study results shown in Table 3 justifies the integration of label and time prediction.

Top-1 Accuracy %			
Dataset	Proposed Network	Removing Inter-activity Time Context	Removing Scene Context
MPII-Cooking [32]	80.1	75.1	33.1
VIRAT Ground [29]	71.8	69.2	61.0

Table 3. Ablation study for label prediction for both of the datasets.

4.4. Starting Time Prediction Results

Objective. The main objective of these experiments is to analyze how well our framework can predict the starting times of future unobserved activities.

Performance Measures. We use Root-Mean-Square Error (RMSE) as our evaluation metric. The lower the value, the better is the prediction performance.

Compared Method. We compare our approach to state-of-the-art starting time prediction method (a statistical model) [26]. In [26], there is an underlying assumption of exponential distribution for the inter-activity time. Our new approach is free from this assumption.

Experimental Setup. For experiments on MPII-Cooking Dataset, we use five fold leave-one-person-out cross validation approach for the training-testing split and average our results over these five combinations. For experiments on VIRAT Ground Dataset, we use the first 210 videos for training and the rest of them for testing.

Results for MPII-Cooking Dataset. Comparison of our starting time prediction results on MPII-Cooking Dataset with state-of-the-art method is shown in Table 4. It is seen that our method outperforms [26]. We also analyze our time prediction result as a function of the last observed activity label and as a function of the label of the activity being predicted. Figure 5 shows the RMSE values based on the label of the last observed activity (top) and the label of the predicted activity (bottom) for MPII-Cooking Dataset. It is seen that only one of the observed activity labels (28) (top) and some of the predicted activity labels (bottom) are contributing to a higher amount of error. We found that if the last observed activity is a relatively longer one by nature, such as ‘make puree’ (label 28 in Figure 5 (top)), then the predicted starting time of the next unobserved activity is relatively more erroneous.

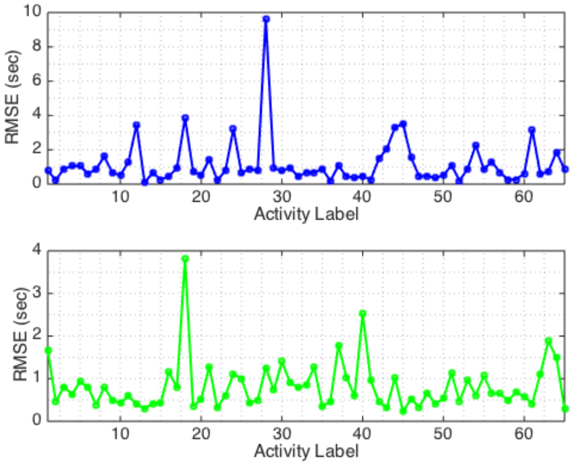

Figure 5. RMSE values based on the label of the observed activity (top) and the label of the predicted activity (bottom) for MPII-Cooking Dataset.

Results for VIRAT Ground Dataset. Our starting time prediction result on VIRAT Ground dataset is shown in Table 4. The state-of-the-art starting time prediction method [26] does not have results on this dataset. For VIRAT Ground Dataset, there are randomly occurring artificial gaps between many activities. There is no way to train a system to predict the starting time of the next activity with such gaps, since there is no underlying structure in them. (Note that label prediction still works because there is structure in what an actor does next, just not when). Thus, we identify activity sequences where there is a regular pattern of activities happening one after another and show results only on them. For example, labels like ‘person loading an object’, ‘person unloading an object’, ‘person opening a vehicle trunk’, ‘person closing a vehicle trunk’ belong to natural sequences where we can predict when the next activity will happen. As explained above, while suitable for the label prediction problem given the continuous nature of the data, this dataset is not ideal for activity starting time prediction analysis, which, we believe, is making the error higher here.

Ablation Study. Using only inter-activity time context (eliminating sequential activity context and scene context), we get a higher RMSE for starting time prediction than that of our proposed network for MPII-Cooking Dataset. This ablation study result shown in Table 5 justifies the integra-

MPII-Cooking Dataset [32]	Goal	Average Inter-activity Time (sec)	Average RMSE (sec)
Statistical Model [26]	Prediction	5.3426	3.9431
Proposed Method	Prediction	5.3426	1.2454
VIRAT Ground Dataset [29]	Goal	Average Inter-activity Time (sec)	Average RMSE (sec)
Proposed Method	Prediction	13.9567	10.4560

Table 4. Starting prediction performance comparisons for MPII-Cooking Dataset and VIRAT Ground Dataset.

tion of label and time prediction.

Average RMSE (sec)	
Proposed Network	Removing Activity Context & Scene Context
1.2454	1.4872

Table 5. Ablation study for starting time prediction for MPII-Cooking Dataset.

4.5. Effect on Prediction Horizon

For label prediction, we perform multi-step prediction where we predict the next-to-next activity i.e., 2-step prediction (using activity features from the $(i-3)^{th}$, $(i-2)^{th}$ and $(i-1)^{th}$ activities, we predict the label of the $(i+1)^{th}$ activity) and the next-to-next-to-next activity (3-step prediction). As expected, the accuracy decreases as the prediction horizon increases. For starting time prediction, we also perform multi-step prediction. For example, for 2-step prediction, we train our model using the features of the $(i-1)^{th}$ activity, and its inter-activity time with the $(i+1)^{th}$ activity. During the operational phase, we use the observed features to predict the starting times of the next-to-next activities. As the prediction horizon increases, there is a gradual accumulation of error. The decrease in accuracy for multi-step label prediction for both of the datasets and the increase in RMSE for multi-step starting time prediction for MPII-Cooking Dataset are shown in Figure 6.

We did not perform multi-step starting time prediction on VIRAT Ground Dataset because of the random gaps between activities as explained earlier. We did not go beyond 3-step for joint prediction as the RMSE error for starting time prediction is already quite high for 3-step prediction shown in Figure 6. However, when we do label prediction separately as an ablation study for prediction horizon, i.e., using a network with only sequential activity context and scene context, the label prediction results upto 5-step prediction for both of the datasets are shown in Figure 7 averaged across all of the activity labels. The above analysis demonstrates that joint estimation of activity label and starting time leads to higher accuracy, but comes at the cost of a shorter forecasting horizon.

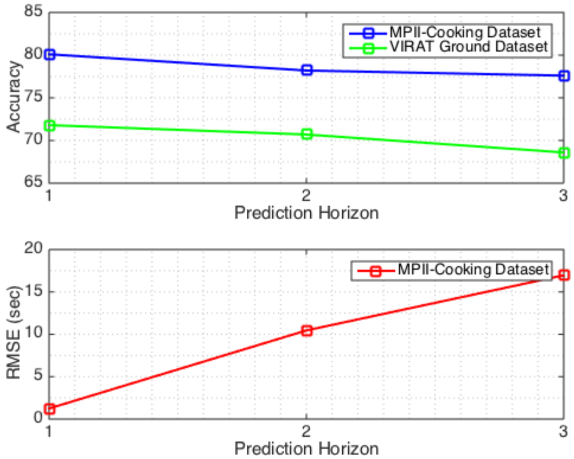

Figure 6. Accuracy of the predicted labels (top) and RMSE of the predicted starting times (bottom) for multi-step prediction. For both of the datasets, the label prediction accuracy decreases and for MPII-Cooking Dataset, the RMSE for predicted times increases with the increasing forecasting horizon as expected.

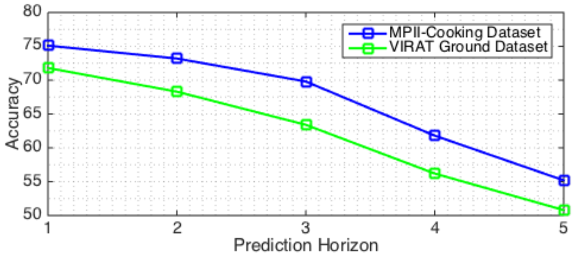

Figure 7. Accuracy of the predicted labels for multi-step prediction without inter-activity time context. For both of the datasets, the label prediction accuracy decreases as we try to predict further ahead as expected.

5. Conclusion

In this work, we propose a framework for jointly predicting the label and the starting time of future unobserved activity by taking advantage of the combination of LSTM and fully connected layers to exploit the contextual relationship among activities and objects. Rigorous experimental analysis on two challenging datasets proves the robustness of our framework. Our approach is capable of both multi-step label prediction and multi-step time prediction with reasonable error. In future, we plan to extend our prediction method for multi-camera environment and investigate how to predict new unseen activity classes.

6. Acknowledgements

This work was partially supported by NSF grant IIS-1316934 and ONR contract N00014-15-C-5113 through a sub-contract from Mayachitra Inc.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[2] S. Cao, K. Chen, and R. Nevatia. Activity recognition and prediction with pose based discriminative patch model. In *WACV*, pages 1–9, 2016.

[3] A. Chakraborty and A. Roy-Chowdhury. Context-aware activity forecasting. In *ACCV*, pages 21–36, 2014.

[4] W. Choi, K. Shahid, and S. Savarese. Learning context for collective activity recognition. In *CVPR*, pages 3273–3280, 2011.

[5] F. Chollet. Keras. https://github.com/fchollet/keras, 2015.

[6] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *ECCV*, pages 428–441, 2006.

[7] Z. Deng, M. Zhai, L. Chen, Y. Liu, S. Muralidharan, M. Roshtkhari, J. Mehrsan, and G. Mori. Deep structured models for group activity recognition. *arXiv preprint arXiv:1506.04191*, 2015.

[8] J. Donahue, L. A. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, pages 2625–2634, 2015.

[9] T. Dozat. Incorporating Nesterov momentum into adam. Technical report, Stanford University, Tech. Rep., 2015.[Online]. Available: http://cs229. stanford. edu/proj2015/054 report. pdf, 2015.

[10] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

[11] A. Graves and N. Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *ICML*, volume 14, pages 1764–1772, 2014.

[12] M. Hasan and A. Roy-Chowdhury. Continuous learning of human activity models using deep nets. In *ECCV*, pages 705–720, 2014.

[13] M. Hasan and A. Roy-Chowdhury. Context aware active learning of activity recognition models. In *ICCV*, pages 4543–4551, 2015.

[14] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[15] D. A. Huang and K. M. Kitani. Action-reaction: Forecasting the dynamics of human interaction. In *ECCV*, pages 489–504, 2014.

[16] M. Ibrahim, S. Muralidharan, Z. Deng, A. Vahdat, and G. Mori. A hierarchical deep temporal model for group activity recognition. In *CVPR*, pages 1971–1980, 2016.

[17] S. Ke, H. L. U. Thuc, Y. Lee, J. Hwang, J. Yoo, and K. Choi. A review on video-based human activity recognition. *Computers*, 2(2):88–131, 2013.

[18] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[19] K. M. Kitani, B. D. Ziebart, J. A. Bagnell, and M. Hebert. Activity forecasting. In *ECCV*, pages 201–214, 2012.

[20] T. Lan, T. C. Chen, and S. Savarese. A hierarchical representation for future action prediction. In *ECCV*, pages 689–704, 2014.

[21] T. Lan, Y. Wang, W. Yang, and G. Mori. Beyond actions: Discriminative models for contextual group activities. In *NIPS*, pages 1216–1224, 2010.

[22] I. Laptev. On space-time interest points. *International Journal of Computer Vision*, 64(2-3):107–123, 2005.

[23] K. Li and Y. Fu. Prediction of human activity by discovering temporal sequence patterns. *IEEE TPAMI*, 36(8):1644–1657, 2014.

[24] W. Li and M. Fritz. Recognition of ongoing complex activities by sequence prediction over a hierarchical label space. In *WACV*, pages 1–9, 2016.

[25] M. Lukasik, P. K. Srijith, T. Cohn, and K. Bontcheva. Modeling tweet arrival times using log-Gaussian cox processes. In *EMNLP*, pages 250–255, 2015.

[26] T. Mahmud, M. Hasan, A. Chakraborty, and A. Roy-Chowdhury. A Poisson process model for activity forecasting. In *ICIP*, pages 3339–3343, 2016.

[27] R. Minhas, A. A. Mohammed, and Q. J. Wu. Incremental learning in human action recognition based on snippets. *IEEE TCSVT*, 22(11):1529–1541, 2012.

[28] B. Ni, X. Yang, and S. Gao. Progressively parsing interactional objects for fine grained action detection. In *CVPR*, pages 1020–1028, 2016.

[29] S. Oh, A. Hoogs, A. Perera, N. Cuntoor, C. Chen, J. T. Lee, S. Mukherjee, J. Aggarwal, H. Lee, L. Davis, et al. A large-scale benchmark dataset for event recognition in surveillance video. In *CVPR*, pages 3153–3160, 2011.

[30] P. H. Pinheiro and R. Collobert. Recurrent convolutional neural networks for scene labeling. In *ICML*, pages 82–90, 2014.

[31] R. Poppe. A survey on vision-based human action recognition. *Image and vision computing*, 28(6):976–990, 2010.

[32] M. Rohrbach, S. Amin, M. Andriluka, and B. Schiele. A database for fine grained activity detection of cooking activities. In *CVPR*, pages 1194–1201, 2012.

[33] M. S. Ryoo. Human activity prediction: Early recognition of ongoing activities from streaming videos. In *ICCV*, pages 1036–1043, 2011.

[34] S. Sadanand and J. J. Corso. Action bank: A high-level representation of activity in video. In *CVPR*, pages 1234–1241, 2012.

[35] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[36] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112, 2014.

[37] T. Tieleman and G. Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. Technical report, 2012.

[38] D. Tran, L. Bourdev, R. F. L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015.

[39] F. Visin, M. Ciccone, A. Romero, K. Kastner, K. Cho, Y. Bengio, M. Matteucci, and A. Courville. Reseg: A recurrent neural network-based model for semantic segmentation. In *CVPR Workshops*, pages 41–48, 2016.

[40] C. Vondrick, H. Pirsiavash, and A. Torralba. Anticipating visual representations from unlabeled video. In *CVPR*, pages 98–106, 2016.

[41] H. Wang, A. Kläser, C. Schmid, and C. Liu. Action recognition by dense trajectories. In *CVPR*, pages 3169–3176, 2011.

[42] L. Wang, X. Zhao, Y. Si, L. Cao, and Y. Liu. Context-associative hierarchical memory model for human activity recognition and prediction. *IEEE Transactions on Multimedia*, 2016.

[43] X. Wang and Q. Ji. Video event recognition with deep hierarchical context model. In *CVPR*, pages 4418–4427, 2015.

[44] X. Wei, P. Lucey, S. Vidas, S. Morgan, and S. Sridharan. Forecasting events using an augmented hidden conditional random field. In *ACCV*, pages 569–582, 2014.

[45] B. Yao and L. Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *CVPR*, pages 17–24, 2010.

[46] A. Zammit-Mangion, M. Dewar, V. Kadirkamanathan, and G. Sanguinetti. Point process modelling of the Afghan War Diary. *Proceedings of the National Academy of Sciences*, 109(31):12414–12419, 2012.

[47] Y. Zhu, N. M. Nayak, and A. Roy-Chowdhury. Context-aware modeling and recognition of activities in video. In *CVPR*, pages 2491–2498, 2013.