

國立臺灣大學電資學院網路與多媒體研究所

碩士論文

Graduate Institute of Networking and Multimedia

College of Electronic and Computer Science

National Taiwan University

Master Thesis



基於時序金字塔之第一人稱影像行為辨識
Activity Recognition in First-Person Camera View Based on
Temporal Pyramid

劉軒銘

Shane Ming Liu

指導教授：歐陽明博士

Advisor: Ming Ouhyang, Ph.D.

中華民國 102 年 6 月

June, 2013



致謝

碩士生活轉眼即逝，除了最後口試前緊張刺激的一段戰鬥期以外，其他時間可以說是我生命中最快樂、充實且成長最多的時光。首要感謝歐陽老師給予我們學習與研究上最大的自由度，並維持實驗室良好的運作及提供豐沛的資源，使我可以專心於學術上。也感謝所有學長姐們的幫助，不管是研究方向的提點，技術上的指導，或是任何和我們分享的經驗，都於我助益良多。特別是小鐵和 Winble，感謝他們最密切的支援。另外，同屆同學們的幫忙也一樣重要，我們一起來到研究所，從碩一開始一起修課、合作，直到最後口試互相支援。我相信少了他們，我沒有辦法如此順利地走過研究生活。最後是家人和妳的支持，那是在學校白天心靈受創後，晚上回到家永遠的給予支持的溫暖。感謝所有的人，沒有你們就沒有現在的我。



中文摘要

在本篇論文中,我們提出了針對於拍攝自第一人稱攝影機影片,進行主角執行中的行為的辨識方法.我們將此問題轉換為鏈狀條件隨機場 (Linear-chain Conditional Random Fields) 的序列標註問題.在本方法中使用高階視覺線索,也就是畫面中物件偵測的結果,來當做辨識特徵.另外也使用了時序金字塔 (Temporal Pyramid) 來實現在時間軸上的多重解析度,並證明其可以改善現行的物件偵測結果.另外也針對在日常生活中常會發生的事件交錯情況,提出在時序金字塔中找尋可能解的辦法.最後我們利用目前最新研究提供的資料 [1] 進行實驗,得出可匹敵的結果.再利用自行拍攝的影片資料,比較有無進行交錯事件搜尋的差別.

關鍵字: 日常生活行為辨識, 時序金字塔, 條件隨機場



Abstract

We present a simple but effective online recognition system for detecting interleaved activities of daily life (ADLs) in first-person-view videos. The two major difficulties in detecting ADLs are interleaving and variability in duration. We use temporal pyramid in our system to attack these difficulties, and this means we can use relatively simple models instead of time dependent probability ones such as Hidden semi-Markov model or nested models. The proposed solution includes the combination of conditional random fields (CRF) and an online inference algorithm, which explicitly considers multiple interleaved sequences by inferencing multi-stage activities on temporal pyramid. Although our system only uses linear chain-structured CRF model, which can be easily learned without a large amount of training data, it still recognizes complicated activity sequences. The system is evaluated on a data set provided by the work from state-of-the-art, and the result is comparable to their method. We also provide some experiment result using a customized dataset.

Keywords: activity of daily livings, temporal pyramid, conditional random files



Contents

| | |
|--|-------------|
| 致謝 | i |
| 中文摘要 | ii |
| Abstract | iii |
| Contents | iv |
| List of Figures | vi |
| List of Tables | viii |
| 1 Introduction | 1 |
| 1.1 Overview | 3 |
| 2 Related Work | 4 |
| 3 Method | 6 |
| 3.1 Visual Phrase Object Feature | 6 |
| 3.2 Activity Model | 7 |
| 3.3 Temporal Pyramid Feature Aggregation | 8 |
| 3.4 Online Inference Algorithm | 10 |
| 3.5 Algorithm | 15 |
| 4 Experiments and Results | 17 |
| 4.1 Experiment 1 | 17 |

| | | |
|----------|---------------------|-----------|
| 4.1.1 | Dataset | 17 |
| 4.1.2 | Evaluation | 19 |
| 4.2 | Experiment 2 | 22 |
| 4.2.1 | Dataset | 22 |
| 4.2.2 | Evaluation | 24 |
| 5 | Conclusion | 27 |
| | Bibliography | 28 |





List of Figures

| | | |
|-----|---|----|
| 3.1 | (a) Observation of the first two segments $(T_{0,0}, s_{0,0})$ and $(T_{0,1}, s_{0,1})$. (b) Stage tables for $a_{0,0}$ and $a_{0,1}$ | 10 |
| 3.2 | (a) A merged segment $(T_{1,0}, s_{1,0})$. (b) The stage table for $a_{1,0}$ | 11 |
| 3.3 | (a) A Possible stage orders for coming segment $(T_{0,2}, s_{0,2})$. (b) The stage table for $a_{0,2}$ | 12 |
| 3.4 | Conditions on new segment $(T_{0,3}, s_{0,3})$. (a) The grown temporal pyramid. (b) The stage table for $a_{0,3}$ | 14 |
| 3.5 | the system flow of our system | 16 |
| 4.1 | Comparison between with and without temporal pyramid. To the left is correctly detected "making tea" in sliding window with temporal pyramid ; while to the right(Bag) is a failed case "drinking water bottle" using sliding window alone. | 19 |
| 4.2 | Single-stage results detected by our system | 19 |
| 4.3 | The recall-precision curve of the ideal object detection. Dotted blue line is the result of sliding window without temporal pyramid. Solid red line is the result with temporal pyramid. Green dotted line is the result detected by [1] | 20 |
| 4.4 | The recall-precision curve of the real object detection. Blue dotted line is the result of sliding window without temporal pyramid. Red solid line is the result with temporal pyramid. | 20 |
| 4.5 | Sceenshots of our system working with the customized dataset. Note that the floating point are the detection probability. | 22 |



| | | |
|-----|--|----|
| 4.6 | Comparison of sliding window with or without temporal pyramid in our dataset. The red solid line is with pyramid; The blue dotted line is without pyramid | 24 |
| 4.7 | Comparison between single-stage and multi-stage activity inferencing, considering all activities. The blue dotted line is the result of single-stage activity inferencing ; The red solid line is the result of multi-stage activity inferencing. | 25 |
| 4.8 | Comparison between single-stage and multi-stage activity inferencing, considering multi-stage activities only. The blue dotted line is the result of single-stage activity inferencing ; The red solid line is the result of multi-stage activity inferencing. | 26 |



List of Tables

| | | |
|-----|--|----|
| 3.1 | The online activity recognition algorithm | 15 |
| 4.1 | The average precision of pre-segmented videos | 17 |
| 4.2 | The activities of daily life(ADLs) contained in the dataset. Multi-stage activities are those interleaved or co-occurrent. | 18 |
| 4.3 | The average precision of our system working with ideal active and passive object annotations, plus the comparison to previous methods. The column "bag" is the result detected by sliding window alone. The column "temporal pyramid" is the result detected by sliding window and temporal pyramid. Note that the precision of Ours is at the recall rate of 0.7. | 18 |
| 4.4 | The activities of lab routines contained in the dataset. Multi-stage activities are those interleaved or co-occurrent. | 23 |
| 4.5 | The average precision of objects detected by cascade classifier in our dataset. | 23 |



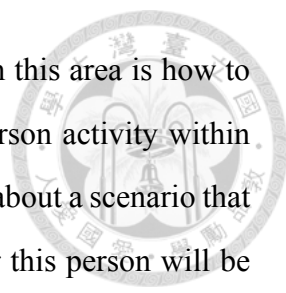
Chapter 1

Introduction

Activity recognition is one of the classic problems in computer vision. Defining and recognizing activities for applications in surveillance or sport analysis has been extensively studied in previous researches. However, there is usually less activity variation in these videos. We concentrate on recognizing activities of daily life in videos shot in first person view, which includes a wide variety of activities of daily life (ADLs). Nowadays, wearable devices such as GoPro camera and Google Glass can be easily equipped and generate high quality first-person-view videos. We believe that the combination of activity recognition and the video capture devices will promote appealing applications such as follows:

Life Logging First-person-view video is naturally suited for personal life-logging. As stated in [2]. People can wear convenient devices to record their daily life, which improves their memories better than diary or voice recording. By utilizing activity recognition system to parse the recorded video, users can reassemble memorable events and quickly filter out massive amount of meaningless events.

Senior Citizen Caring There are benchmarks used to evaluate ADLs for clinical application [3]. If we could do these evaluations by mounting a camera for individuals and detecting these ADLs, at-home monitoring may become a reality and can save huge resources for senior citizen caring.



Context-Aware Service "The next chapter" in previous research in this area is how to acquire more powerful context-aware service by integrating first-person activity within other context such as location-based service. Imagine a pinpoint task about a scenario that a person is leaving home for work: A possible activity sequence for this person will be "taking briefcase", "wearing shoes", "opening door" and "without taking umbrella". With weather prediction from a location-based service, the system can raise a notification to the user to bring umbrella if the probability of precipitation is high.

It is challenging to analyze daily activities in a first-person-view video due to variability in duration, interleaved activities and various levels of semantic meanings. Previous works in activity recognition approaches can be divided into single-layered methods and hierarchical methods [4]. Single layer methods are usually constrained to low activity level. On the other hand, hierarchical methods give more flexible meaning by concatenating events in lower hierarchies of activity models. A temporal pyramid feature representation method proposed in [1] aggregates bag of features in temporal hierarchy. Using temporal pyramid encodes the event-duration variation and avoids complex probabilistic modeling such as hierarchical hidden semi-Markov Model (HHSMM).

In this paper, we purposed an online system, which aims to detect the activities in a simple but effective way by using a temporal pyramid to migrate observations in time domain to provide improved detection results. Unlike previous method only taking the pyramid as a multi-resolution feature, we further propose a Viterbi-like sequential inference strategy to find the possible interleaved activities. And since the system is working in an online manner, once embedded with real-time object detection it becomes a real-time context-aware service system. This is exactly "the next chapter" application in computer vision and can be worthwhile to further researches in the future.

1.1 Overview

The organization of this paper is as follows. We use active/passive objects shown in the video frames as high level visual clue as described in section 3.1. In section 3.2, we explicitly define the problem we are going to solve and model it into a linear-chain conditional random field sequence-labeling problem. The temporal pyramid is explained in sections 3.3, containing the reason why using temporal pyramid and the definition of it. In section 3.4, we use a laundry example to illustrate the inference process and followed by system flow and the algorithm. In Chapter 4, the results of our experiment are presented including using ideal object annotation and deformable part-based model object detection. And the comparison between sliding window with or without temporal pyramid is also provided in this chapter. Finally we conclude the result in Chapter 6.

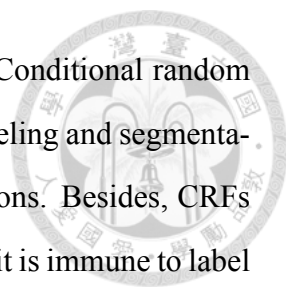


Chapter 2

Related Work

There are different activity recognition methods that aim at different types of input data such as video footage and sensor input. A recent research direction is to recognize ADLs in ego-vision[cite wearable camera] [1]. They proposed a method of using temporal pyramid to transfer the local features into temporal extension and classified the activities by support vector machine. There are also attempts to recognize activities by using non-visual cues. Some researchers used data from a simulating robot tag space to compare the performance between different statistical models [5]. An activity recognition system is embedded on smart phones based on 3D accelerometers data analysis [6], but it can only recognize a few types of primitive behavior such as walking. Another sensor-based approach is proposed to handle sequential, interleaved and concurrent activities sequence [7]. However, the systems works in a sophisticated smart home environment with RFID equipments.

The mainstream in previous methods is using state-based model for activity representation. Hidden Markov model (HMM) and its variants are widely used in those methods. Along with the increasing trend of newly-wed taking records, an automatic wedding video segmentation is proposed [8]. The system recognizes multiple wedding events by using HMM and segments a wedding video. A hierarchical hidden Markov model (HHMM) is used to describe events containing multiple semantic levels [6]. Switch Hidden Markov model (SHMM) is a two layer case of HMM [9]. It encodes both duration variation and semantic level combination for ADL recognition [9]. However, hierarchical statistical



models are complex so that the computational cost is usually high. Conditional random fields (CRFs) is a discriminative probabilistic model for sequence labeling and segmentation [10]. It relaxes the independence assumption between observations. Besides, CRFs provides single joint probability distribution of entire sequences, thus it is immune to label bias problem of maximum entropy Markov models (MEMMs), which only models joint probability in local states. CRF is shown to be more robust than HMM for activity recognition when the observation distribution violates the independent assumption of HMM [5]. A two-level probabilistic framework for concurrent and interleaving goal recognition [11] utilizes skip-chain CRF to achieve their goal. However, it is somewhat complex utilizing SCCRF and the inputs of their system are from non-visual sensors.

Various types of feature descriptors are used as observation for probabilistic models. Methods using low level features are often limited in the type of activity they can recognize [12]. Semantic features enhances activity recognition especially for composite events, such as wedding ceremony [8]. Their proposed Wedding feature descriptors of [8] are speech discriminator, flash light detector and bride indicator. The system performance reaches over 70% precision and recall rate among most of the event types. However, these feature descriptors are discriminative since they are supported by strong low level features in newly-wed video. ADL videos usually lack prominent characteristics and the event variation is high. So far, object detectors seem to be most stable feature descriptor in ADL recognition [1]. It is also useful to describe objects in an interactive-event space by visual phrases [13], e.g., "Man sitting in sofa" or "Person riding bike". Proposed temporal pyramid of [1] is a powerful feature representation. It provides a temporal hierarchy which aggregates object descriptors of short interval to those of large interval, which reduces effect of inaccurate object detection in short interval. The recognition is greatly enhanced by concatenating feature segments of all hierarchies into a single feature vector, and then classifies them by using support vector machine (SVM) classifier. However, the performance of object detection in such realistic dataset is unreliable and the framework cannot deal with interleaved activities that are not shown in the training data.



Chapter 3

Method

3.1 Visual Phrase Object Feature

Many previous research has shown that using RFID or some other mechanism to acquire the interacting objects is a strong evidence indicating the ongoing activity of the subject [7] [14]. However, it needs more effort for the user to be gear up with sensors and the target objects also have to be embedded with RFIDs.

Our target is to use pure computer vision to solve this problem. We use object-centric features to represent the high level information in the egocentric frames. The appearance of objects in interaction may be much different from its original one and reasoning objects relations includes much complexity. The significant gains in considering the objects as visual phrases is shown in [13]. This approach is appropriate in our case because most of the ADLs are interactions between human hands and the objects. We denote a specific object in separate types, active and passive. In other words, there are two different feature for a single object class. For example, active_cup and passive_cup are considered two different object/feature in our system.

Intuitively, objects may tend to lie in similar positions and have similar sizes presented in the field of view for the same activity. However, after [1] augmenting some spatial reasoning into their object features, there is no noticeable improvement. Therefore we do not take the spatial or size into account. On the other hand, we consider objects in the manner of bag-of-words.



3.2 Activity Model

Problem Definition A general description of our problem is defined as follows: Let $I = \{i|i = 1, 2, 3, \dots, n\}$ denote indexes of an input sequence with n frames, and $O = \{o|o = 1, \dots, m\}$ denote indexes of m objects in the scene. F is a set of feature vectors $\{\mathbf{f}_i|i \in I\}$ associated with the video frame, where \mathbf{f}_i contains m labels such that $f_i^o = 1$ if object o exists in frame i , otherwise $f_i^o = 0$. Let L denote a set of possible activities in the video frame, then the objective is to find activity labels $A = \{a_i|i \in I, a_i \in L\}$ such that

$$A^* = \arg \max_A P \{A|F\}, \quad (3.1)$$

where $P \{A|F\}$ denotes the probability of activity A given F . Since F is already known, the conditional probability $P \{A|F\}$ is proportional to joint probability $P \{A, F\}$. Therefore, eq (3.1) can be rewritten as

$$A^* = \arg \max_A P(A, F). \quad (3.2)$$

A linear chain-structured CRF can model the joint probability as

$$P(A, F) \propto \sum_{i=1}^n \sum_{j=1}^Q \lambda_j q_j(F, a_{i-1}, a_i) \quad (3.3)$$

Where Q is the number of feature functions and λ is the weighting factor learned by CRF training.

According to eq (3.3), we can derive

$$a_i^* = \arg \max_{a_i} \sum_{j=1}^Q \lambda_j q_j(F, a_{i-1}, a_i) \quad (3.4)$$

We use a sliding window to wrap up the video frames and aggregate them by temporal pyramid (will be describe in the following section). We call these aggregated frames a

segment $S = \{s_k | k \in \text{number of segments}\}$. Our system outputs activity label for each segment. The final problem definition is

$$s_k^* = \arg \max_{s_k} \sum_{j=1}^Q \lambda_j q_j(F, a_{k-1}, a_k) \quad (3.5)$$



Laundry Example Before going further, we use a simple example to explain the problem of using linear chain-structured CRF. The scenario is about a person doing laundry. A normal activity sequence is: "taking clothes", "using washing machine", and then "taking clothes" again. If we further consider multiple stages to describe an activity, the sequence will be: "laundry on stage 1", "laundry on stage 2", and then "laundry on stage 3". Since it is reasonable to take clothes just before and right after using washing machines, the transition possibility between states "laundry on stage 1" and "laundry on stage 2" should be strong after CRF model learning. However, it is also possible in a daily life such that the person does something not much related about laundry between these events. For example, he may receive a phone call before going to the laundry room. Therefore, the original event sequence becomes an interleaved activity sequence: "laundry on stage 1", "using cell phone", "laundry on stage 2", and then "laundry on stage 3". Since the transition possibility between "using cell phone" and "laundry on stage 2" is weak, CRF may misjudge "laundry on stage 2" as other activities after it observes "using cell phone". A possible solution is to consider earlier states rather than just consider the last state. However, this method will make the model learning become difficult since the CRF structure becomes more complex. And it is hard to determine how many neighboring events should be considered. Our idea is to use temporal pyramid encoding multiple temporal scales of an observation sequence and consider multiple interleaved sequences according to possible stages of an activity.

3.3 Temporal Pyramid Feature Aggregation

We adopt temporal pyramid [1] to model features of multiple temporal hierarchies. The reason we adopt this method are as follows:

Multi-resolution in Time Domain The ADLs are by nature having large variability in duration. Some previous methods try to solve this problem by training stage models depending on time. This makes state models complex, and still not a robust way for facing this problem. By observing the input video in different level of the pyramid is equivalent to seeing the data in multi-resolution of time domain, and this relaxes the problem of variation in duration of ADLs.

Alternative of Hierarchical Methods A main stream of activity detection is to model this problem into hierarchical layers. For example, atomic actions are detected in bottom layer and classifiers in higher layer of the hierarchy take over the atomic actions. High level result is then produced with higher symantic meanings. Building pyramid in feature space is also providing this hierarchical characteristic. We are able to use a single activity classifier, with hierachical feature pyramid, to achive the same goal.

Correction of Object Detection In the scenario of first-person-view videos, object detection becomes very difficult because of clutter, occlusion, different viewing angles and variation of illumination. Using temporal pyramid to merge segments by averaging the object detection score is in some sense correcting the detection result.

Temporal Pyramid Let T represent a temporal pyramid with p levels, and $T_{l,k}$ is k_{th} time segment on level l of the pyramid. We use $\mathbf{s}_{l,k} = \{s_{l,k}^o | o \in O\}$ to denote the corresponding feature vector for time segment $T_{l,k}$, where $s_{l,k}^o$ is the score of an object o in $T_{l,k}$. Now we define feature vectors on first level by

$$\mathbf{s}_{0,k} = \sum_{i \in T_{0,k}} \mathbf{f}_i. \quad (3.6)$$

The features on higher levels can be computed by aggregating features on last level:

$$\mathbf{s}_{l,k} = \frac{1}{2}(\mathbf{s}_{l-1,2k-1} + \mathbf{s}_{l-1,2k}) \quad (3.7)$$

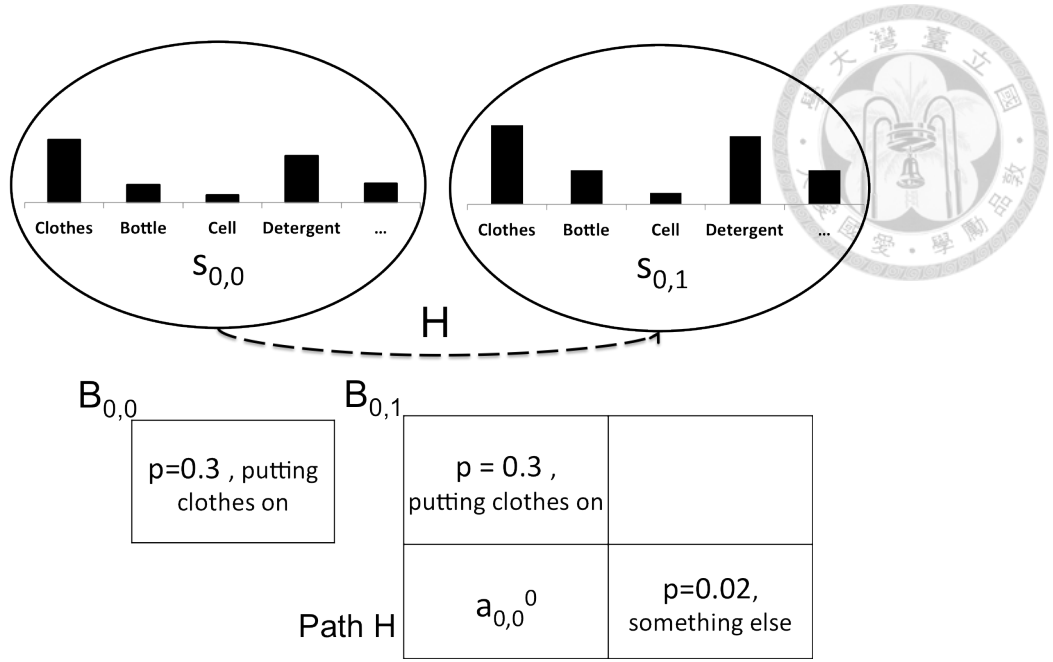


Figure 3.1: (a) Observation of the first two segments $(T_{0,0}, s_{0,0})$ and $(T_{0,1}, s_{0,1})$. (b) Stage tables for $a_{0,0}$ and $a_{0,1}$.

We translate $\mathbf{s}_{l,k}$ into a boolean vector $\mathbf{x}_{l,k}$ as input of CRF model, where $x_{l,k}^o = 1$ if $s_{l,k}^o > |T_{l,k}|/\alpha$, otherwise $x_{l,k}^o = 0$, where α is a constant.

3.4 Online Inference Algorithm

Let us use the example "laundry" to illustrate the online inference algorithm. Suppose that the event sequence is: "laundry on stage 1", "using cell phone", "laundry on stage 2" and then "laundry on stage 3". Figure 3.1(a) shows the observation of first two segments $s_{0,0}$ and $(T_{0,1}, s_{0,1})$. In the beginning $((T_{0,0}, s_{0,0}))$ the most appeared object is "clothes", while in the second segment $((T_{0,1}, s_{0,1}))$ the score of "detergent" increases (the person is taking detergent for washing clothes). After merging $s_{0,0}$ and $s_{0,1}$, $\mathbf{s}_{1,0}$ shows that the most appeared object is "clothes" from beginning to current frame, as shown in Figure 3.2(a). Let $a_{l,k}$ denote the activity of segment $T_{l,k}$. Then the most possible activity $a_{0,0}^*$ in segment $(T_{0,0}, s_{0,0})$ is obviously:

$$a_{0,0}^* = \arg \max_a p(a_{0,0}, \mathbf{s}_{0,0}) \quad (3.8)$$

For the activity $a_{0,1}$ in segment $(T_{0,1}, s_{0,1})$, there are two possibilities of its stage:

- $a_{0,1}$ is on stage 1 of a new activity.

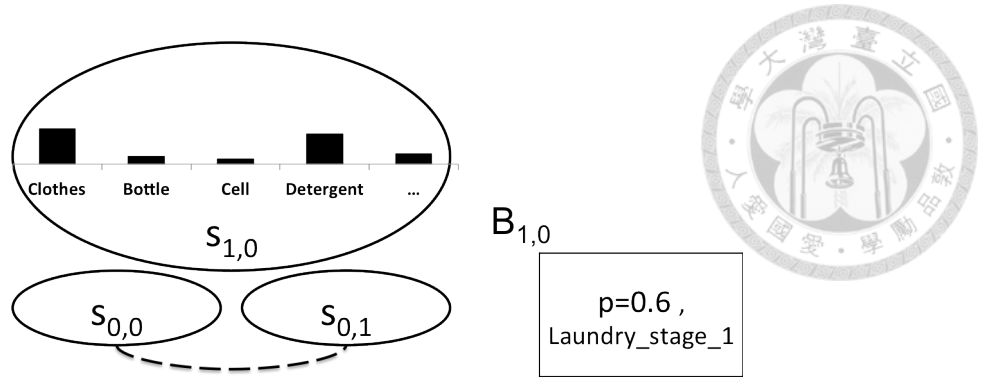


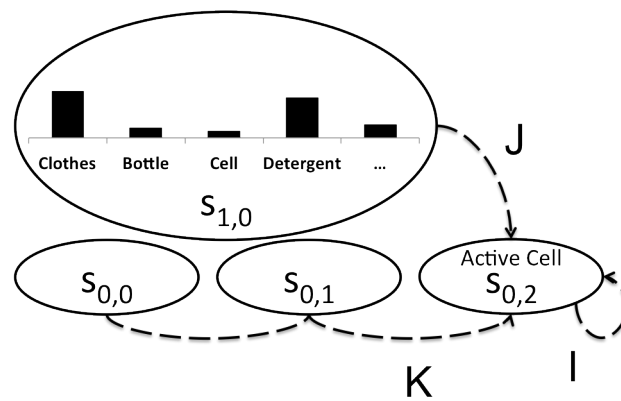
Figure 3.2: (a) A merged segment $(T_{1,0}, s_{1,0})$. (b) The stage table for $a_{1,0}$.

- $a_{0,1}$ is on stage 2 of the same activity of $a_{0,0}^*$. That is, $a_{0,0}^*$ is on stage 1.

Both cases are considered when solving $a_{0,1}^*$. We use $p_{0,1}^0$ to denote $p(a_{0,1}, \mathbf{s}_{0,1})$, and $p_{0,1}^1$ to denote $p(a_{0,0}^*)p(a_{0,0}^*, a_{0,1}, \mathbf{s}_{0,1})$, and $(a_{0,1}^0, a_{0,1}^1)$ to denote the most probable activities based on different stage reasoning, respectively. Then

- $a_{0,1}^0 = \arg \max_a p_{0,1}^0$.
- $a_{0,1}^1 = \arg \max_a p_{0,1}^1$.
- $a_{0,1}^* = \arg \max_a \max(p_{0,1}^0, p_{0,1}^1)$.

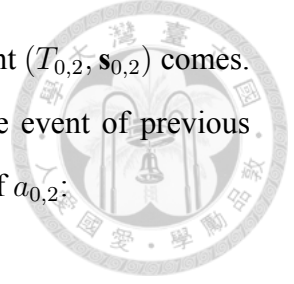
We use "stage table" to store the result of probability computation and activity labeling. Let $B_{l,k}$ denote a stage table of the activity $a_{l,k}$. Then $B_{0,0}$ is a 1×1 table, where $B_{0,0}(0,0)$ stores a 2-tuple $(a_{0,0}^*, p_{0,0}^*)$. Figure 3.1(b) shows that $a_{0,0}$ is "putting clothes on" but $p_{0,0}^*$ is low 0.3. It indicates that there exists noises (i.e., other existed objects in $T_{0,0}$). $sB_{0,1}$ is a 2×2 table, where the first column stores information of the activity on stage 1. $B_{0,1}(0,0)$ stores the result $(a_{0,1}^0, p_{0,1}^0)$ under the condition that $a_{0,1}$ is on stage 1. Otherwise, $B_{0,1}(1,0)$ stores $(a_{0,0}^*, p_{0,0}^*)$ and $B_{0,1}(1,1)$ stores $(a_{0,1}^1, p_{0,1}^1)$ under the condition that $(a_{0,0}, a_{0,1})$ are on stage 1 and 2 of a same activity, respectively. Now segment $(T_{0,0}, \mathbf{s}_{0,0})$ and new segment $(T_{0,1}, \mathbf{s}_{0,1})$ are merged to the segment $(T_{1,0}, \mathbf{s}_{1,0})$. Since there is no segment before it, $a_{1,0}^*$ is estimated in a way similar to $a_{0,0}^*$. Figure 3.2(b) shows that $a_{1,0}$ is "laundry on stage 1" and $p_{1,0}$ is higher, since the relative score of clothes is increasing after merging $\mathbf{s}_{0,0}$ and $\mathbf{s}_{0,1}$.



$B_{0,2}$

| | | | |
|--------|---------------------------------|------------------------------------|-------------------------------------|
| Path I | $p=0.8,$ $a=\text{use cell}$ | | |
| Path J | $a_{1,0}^0$ | $p=0.1,$ $a:\text{other event}$ | |
| Path K | $a_{0,0}^0$ | $a_{0,1}^1$ | $p=0.05,$ $a:\text{other event}$ |

Figure 3.3: (a) A Possible stage orders for coming segment $(T_{0,2}, s_{0,2})$. (b) The stage table for $a_{0,2}$.



We continue to explain the inference process when next segment $(T_{0,2}, \mathbf{s}_{0,2})$ comes. Similar to the case in $T_{0,1}$, activity $a_{0,2}$ may be related to the same event of previous segments. But its situation is a little bit more complicated than that of $a_{0,2}$:

- $a_{0,2}$ is on stage 1 of a new activity.
- $a_{0,2}$ is on stage 2 of the same activity of $a_{0,0}^0, a_{0,1}^0$ or $a_{1,0}^0$ (i.e., $(a_{0,0}^0, a_{0,1}^0, a_{1,0}^0)$ is on stage 1).
- $a_{0,2}$ is on stage 3 of the same activity of $a_{0,0}^0$ and $a_{0,1}^1$ (i.e., $(a_{0,0}, a_{0,1})$ are on stage 1 and 2, respectively).

Figure 3.3 shows different interleaved cases. Let $a_{prev} = \{a|a_{0,0}^0, a_{0,1}^0, a_{1,0}^0\}$. The corresponding probabilities $(p_{0,2}^0, p_{0,2}^1, p_{0,2}^2)$ and most probable activities $(a_{0,2}^0, a_{0,2}^1, a_{0,2}^2)$ are given as follows.

- stage 1:

$$p_{0,2}^0 = p(a_{0,2}, \mathbf{s}_{0,2}), \quad a_{0,2}^0 = \arg \max_a p_{0,2}^0.$$

- stage 2:

$$p_{0,2}^1 = \max_{prev, a} p(a_{prev}^0) p(a_{prev}^0, a_{0,2}, \mathbf{s}_{0,2}),$$

$$a_{0,2}^1 = \arg \max_a p_{0,2}^1.$$

- stage 3:

$$p_{0,2}^2 = \max_{prev, a} p(a_{prev}^1) p(a_{prev}^1, a_{0,2}, \mathbf{s}_{0,2}),$$

$$a_{0,2}^2 = \arg \max_a p_{0,2}^2.$$

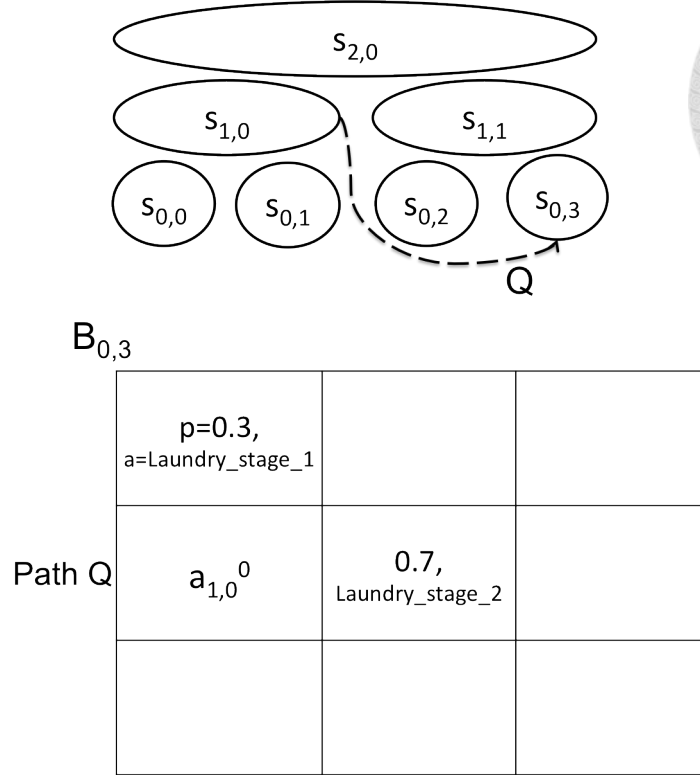


Figure 3.4: Conditions on new segment $(T_{0,3}, s_{0,3})$. (a) The grown temporal pyramid. (b) The stage table for $a_{0,3}$.

- Finally, $a_{0,2}^* = \arg \max_a (p_{0,2}^0, p_{0,2}^1, p_{0,2}^2)$.

Since $s_{0,2}$ indicates that the dominant object is "cell phone" (the person now is using a cell phone). $a_{0,2}^*$ is most likely "using a cell phone", while the possibility of other sequences are low, as shown in Figure 3.3(b). Now here comes the segment $(T_{0,3}, a_{0,3})$, we see that it indicates an active object as "washing machine". If $a_{0,3}$ is on stage 2, then there are four activity candidates on stage 1 of the same activity of $a_{0,3}$: $(a_{0,0}^0, a_{0,1}^0, a_{0,2}^0, a_{1,0}^0)$, the computation for $(a_{0,4}^1, p_{0,4}^1)$ is similarly to the case in $(a_{0,3}^1, p_{0,3}^1)$. Figure 3.4 shows that when $a_{1,0}^0$ is considered in stage 1, the probability of "laundry on stage 2" will be high. Because $a_{1,0}^0$ with a high probability of "laundry on stage 1"

Let a_w^* denote the most probable event at frame w , where w denotes the current position of the sliding window. Then as the sliding window moves through two segments, there are multiple activities candidates at same time. For example $a_{0,1}^*$ and $a_{1,0}^*$ since the two segments are overlapped. We simply choose the activity with higher probability for a_w^* . A complete description of the online inference algorithm is stated in Table 3.1.

3.5 Algorithm



Table 3.1: The online activity recognition algorithm

| |
|--|
| Input: |
| The feature vector of each frame: f , |
| Detection threshold: $Threshold$ |
| Output: |
| Activity label for each segment |
| for each segment |
| Incoming new segment s_{new} |
| If s_{new} is not similar to $s_{0,k-1}$ |
| $s_{0,k} = s_{new}$ |
| Phase 1: Merge and build the pyramid |
| Phase 2: Refresh the pyramid |
| Phase 3: Inference |
| If $p_w^* > Threshold$ |
| return a_w^* as the activity detected |
| ELSE |
| Discard s_{new} |

Phase 1: Merge and build the pyramid Here we omit the object detection in the procedure, which means that once a new segment s_{new} arrives, feature vector for each frame in the segment f_i is known. If the index of the new segment is multiple of 2, we go through each level of the pyramid and try to merge the existing segments to become new ones as eq 3.7 describes.

Phase 2: Refresh the pyramid Then we discard some segments in each level of the pyramid that are too far from the current segment because of two reasons: First, the high level segments merged by lots of low level segments may represent a long duration containing large amount of variation of observations, this is more likely that the subject is already changed the ongoing activity than remaining in the same one; Second, the connections between current segment and the segments happened before normally drops as time gone by.

Phase 3: Inference In the last phase, we will have new segments in each level come from the merging in phase 1 or the just arrived new observation. For each of the new seg-

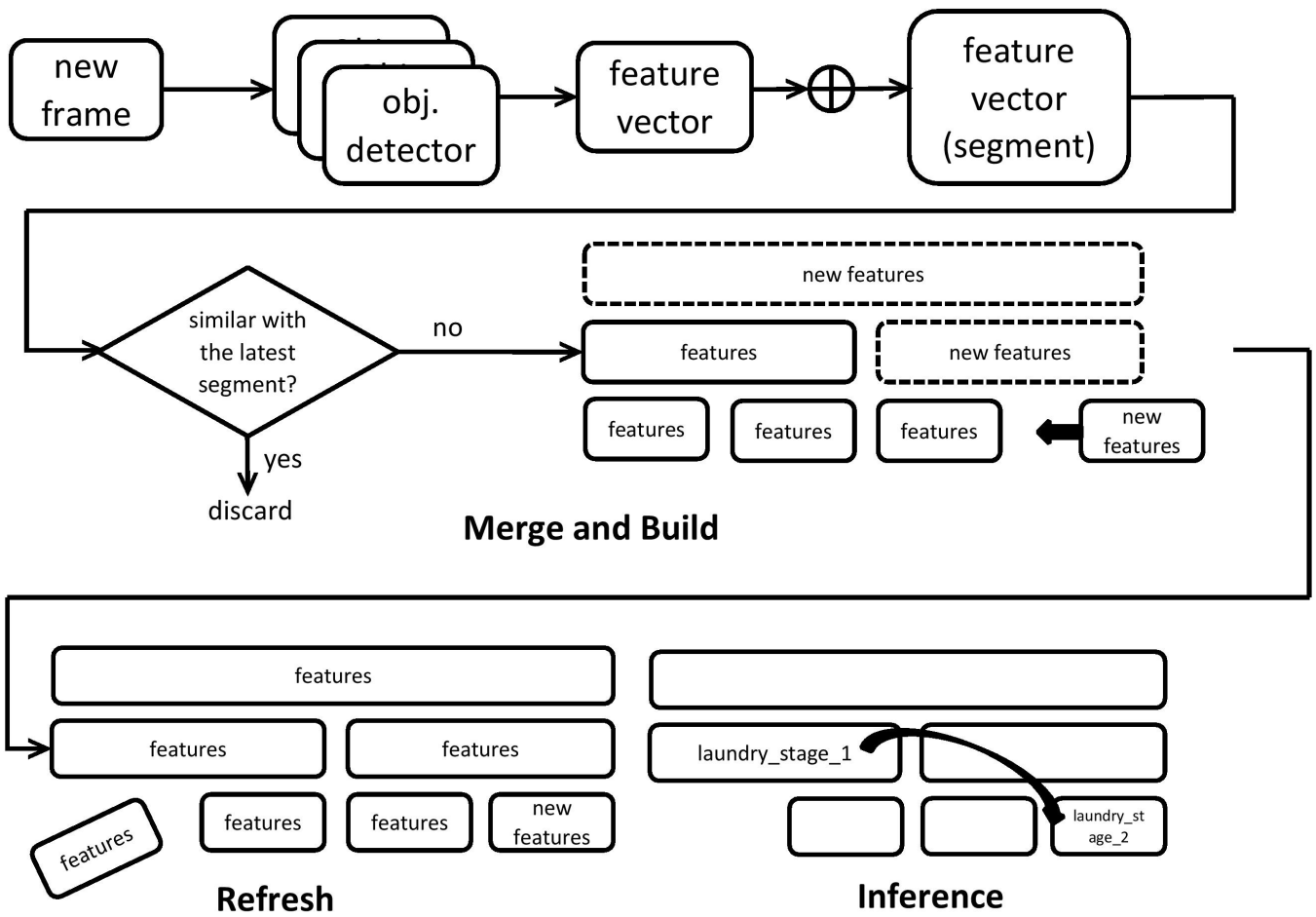


Figure 3.5: the system flow of our system

ments we use the process describe by above to find out the best activity label prediction of the current observation.



Chapter 4

Experiments and Results

4.1 Experiment 1

4.1.1 Dataset

We use the well-annotated, high-quality dataset introduced by [1]. It contains 10 hours long video clips composed of 20 people performing their activity of daily living (ADLs). The videos are shot by chest-mounted GoPro camera. The videos are in 1280x960 definition and 30 frames per second with 170 degrees viewing angle.

This dataset is challenging because of the unscripted activities, cluttered background and diverse environments. It contains 19 ADLs (table 4.2) suitable for experiments, four of them are multi-stage activities which means they are interleaved so the stages of one activity is not continuously shown in the video. Unlike [1] that considers this problem as a multi-class classification, our system utilizes conditional random fields, which transforms this problem into sequence labeling. We have embedded a general purpose CRF toolkit

| | Pre-segmented |
|--------|---------------|
| STIP | 0.165 |
| SVM+IO | 0.768 |
| CRF+IO | 0.785 |

Table 4.1: The average precision of pre-segmented videos



| activity | multi-stage | avg. duration(secs) | std(secs) |
|------------------------|-------------|---------------------|-----------|
| combing hair | no | 26.50 | 9.00 |
| make up | no | 108.00 | 85.44 |
| brushing teeth | no | 128.86 | 45.50 |
| dental floss | no | 92.00 | 23.58 |
| washing hands(face) | no | 76.00 | 23.33 |
| drying hands(face) | no | 26.67 | 13.06 |
| laundry | yes | 215.50 | 142.81 |
| washing dishes | no | 159.60 | 154.39 |
| moving dishes | no | 143.00 | 159.81 |
| making tea | yes | 143.00 | 71.81 |
| making coffee | yes | 85.33 | 54.45 |
| drinking water bottle | no | 70.50 | 30.74 |
| drinking water tap | no | 8.00 | 5.66 |
| making cold food/snack | no | 117.20 | 96.63 |
| making hot food | yes | 130.2 | 70.50 |
| vacuumning | no | 77.00 | 60.81 |
| watching TV | no | 189.60 | 98.74 |
| use computer | no | 105.60 | 32.94 |
| using cell | no | 18.67 | 9.45 |

Table 4.2: The activities of daily life(ADLs) contained in the dataset. Multi-stage activities are those interleaved or co-occurrent.

| | Temporal Pyramid | Bag |
|--------|------------------|-------|
| SVM+IO | 0.607 | 0.537 |
| Ours | 0.656 | 0.605 |

Table 4.3: The average precision of our system working with ideal active and passive object annotations, plus the comparison to previous methods. The column "bag" is the result detected by sliding window alone. The column "temporal pyramid" is the result detected by sliding window and temporal pyramid. Note that the precision of Ours is at the recall rate of 0.7.

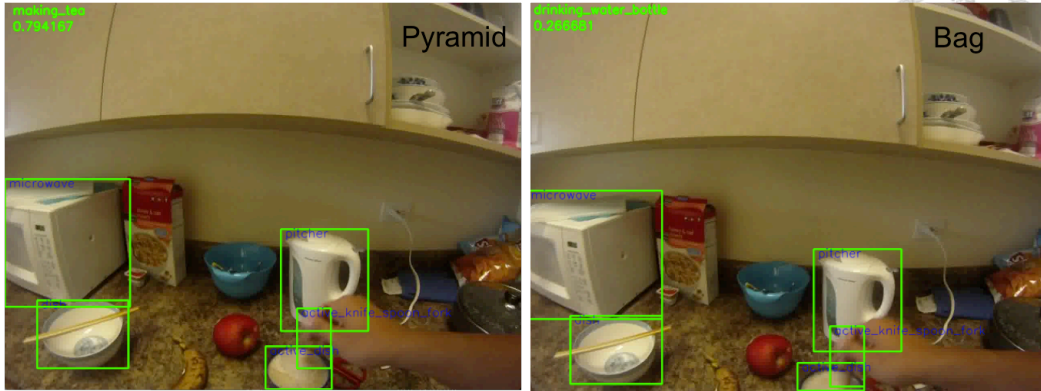


Figure 4.1: Comparison between with and without temporal pyramid. To the left is correctly detected "making tea" in sliding window with temporal pyramid ; while to the right(Bag) is a failed case "drinking water bottle" using sliding window alone.



Figure 4.2: Single-stage results detected by our system

[15] in our system for all the CRF utilities.

4.1.2 Evaluation

We use 1-vs-all cross validation to train our models so that no subject will be include both in training set and test data. The training sentences are produced by action annotations including start frame, end frame, stage and activity label. Single-stage activities are single word sentences in the CRF; for those multi-stage activities being interleaved in the video, we take all the seperated stages for the particular activity to be sequential sentence as training data. The followings are:

1. Activity detection by conditional random field over pre-segmented videos

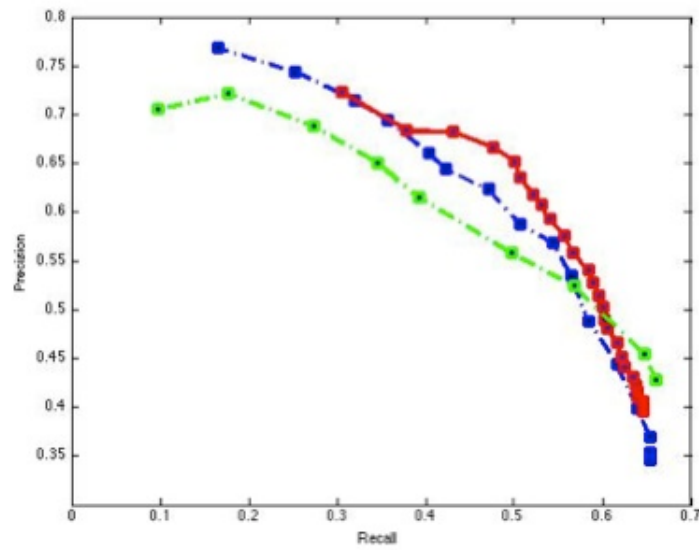
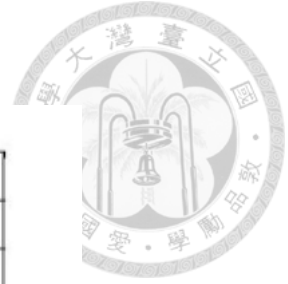


Figure 4.3: The recall-precision curve of the ideal object detection. Dotted blue line is the result of sliding window without temporal pyramid. Solid red line is the result with temporal pyramid. Green dotted line is the result detected by [1]

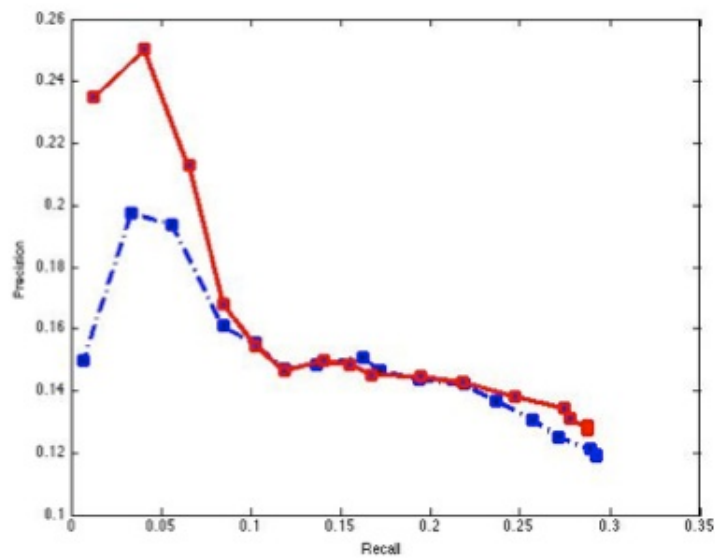


Figure 4.4: The recall-precision curve of the real object detection. Blue dotted line is the result of sliding window without temporal pyramid. Red solid line is the result with temporal pyramid.

2. Comparison of sliding window with or without temporal pyramid using ideal objects annotation
3. Comparison of sliding window with or without temporal pyramid using DPM objects detection



First we have tested the accuracy in pre-segmented video by the action annotation of the dataset. Each segment absolutely contains at least one activity performed in its interval. For the first experiment, result using spatial-temporal interesting points as feature [16] is also listed in table for reference. Note that the STIP method is unable to be applied with the ideal object annotation thus the accuracy is relatively low. The result shows that by using the same object feature, the performance of conditional random field is the same as SVM.

Then we have experimented the system with sliding window, namely a 300 frames sliding window as a segment in the first level of pyramid.

For Comparison of sliding window with or without temporal pyramid using ideal objects annotation, we use the ideal object annotation provided by [1] as object features. The result of (SVM+IO) [1] is listed in table 4.3. The result of our method is in the row (ours). The comparison of with and without pyramid recall and precision curve is figure 4.3. The result shows that with help of temporal pyramid, the precision is better than when recall rate is higher than 0.5.

For Comparison of sliding window with or without temporal pyramid using DPM objects detection, we use the deformable part base model (DPM) [17] detection result provide by [1]. The DPM detection result is the detection score of each object in each frame. The detection score higher than -0.75 for an object is considered "exist" in the frame. The comparison is in figure 4.4. It has clearly shown that by the help of temporal pyramid, the activity detection with imperfect object detection can be improved.



Figure 4.5: Screenshots of our system working with the customized dataset. Note that the floating point are the detection probability.

4.2 Experiment 2

4.2.1 Dataset

The dataset provided by [1] is in some sense a realistic one, however the activities presented in the dataset are not equally distributed and either not frequently presented. For example, there is only 2 out of 20 video containing interleaved activity of "making coffee"; there is only one video containing interleaved "making hot food". This means using this dataset to evaluate the characteristic of sequence labeling of conditional random field is not a good choice. For this reason, we have customized a dataset composed of 5 people performing 9 kinds of lab routines list in table 4.4. The activities are equally distributed and guaranteed to present in each video. We use this dataset to examine the effect of applying sequence labeling to finding interleaved activities.



| activity | multi-stage | avg. duration(secs) | std(secs) |
|-------------------|-------------|---------------------|------------|
| use computer | no | 17.8 | 2.77 |
| use cell | no | 17.4 | 5.37 |
| wash hand | no | 7.4 | 4.27 |
| drink water | no | 12 | 2.55 |
| talk to people | no | 23.4 | 12.05 |
| check the weather | no | 10.2 | 1.64 |
| reading | no | 20 | 5.7 |
| make coffee | yes | 10.8 , 18.4 | 7.56 , 5.6 |
| make photocopy | yes | 11.6 , 26.6 | 2.07 , 11 |

Table 4.4: The activities of lab routines contained in the dataset. Multi-stage activities are those interleaved or co-occurrent.

| object | average precision |
|-------------------|-------------------|
| passive cup | 0.055 |
| passive dispenser | 0.025 |
| active laptop | 0.34 |
| active cup | 0.073 |
| active book | 0.04 |
| active cell | 0.005 |
| active window | 0.03 |
| active papers | 0.08 |
| active human | 0.05 |
| active copier | 0.025 |

Table 4.5: The average precision of objects detected by cascade classifier in our dataset.

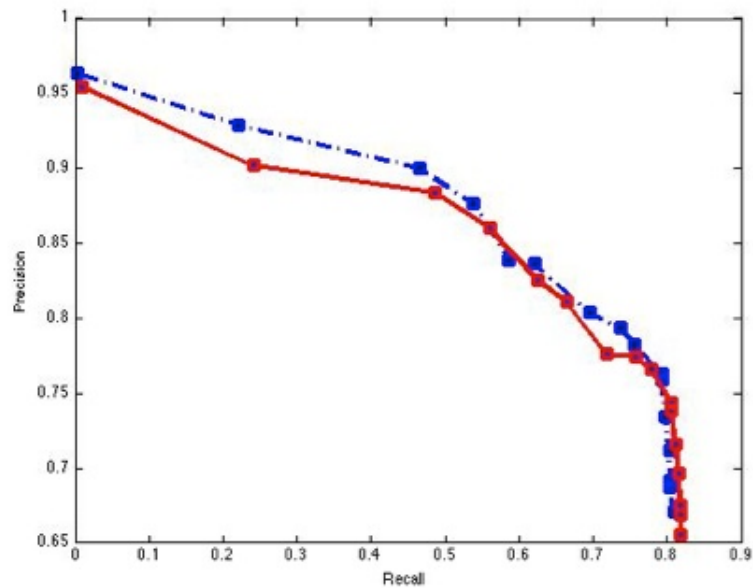


Figure 4.6: Comparison of sliding window with or without temporal pyramid in our dataset. The red solid line is with pyramid; The blue dotted line is without pyramid

4.2.2 Evaluation

We perform the following experiments:

1. Object detection using LBP feature object detection
2. Comparison of sliding window with or without temporal pyramid using ideal objects annotation
3. Comparison of single-stage inference and multi-stage inference using ideal objects annotation

First we use cascade classifier with LBP feature [18] to test the possibility of real-time application. There are 21 object types performed in this dataset but only 12 of them are suitable for cascade classifier training, in other words, the other 9 types of objects are rarely seen in the view, mostly shown with occlusion or the variation of view angle are too diverse. The detection result is in table 4.5. To tell the truth, it is not acceptable for further usage so we use ideal object annotations as input feature in the next experiment.

We also evaluate the computation time of the object detection in our system. With a 30

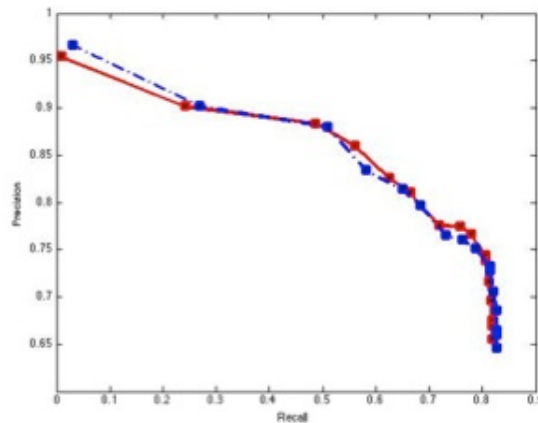


Figure 4.7: Comparison between single-stage and multi-stage activity inferecing, considering all activities. The blue dotted line is the result of single-stage activity inferecing ; The red solid line is the result of maulti-stage activity inferecing.

frames sliding window (the video is in frame rate of 30), it takes 11 seconds in average for object detection and 0.0292 seconds in average for the rest of the computation, which means the bottleneck is the object detection for a real-time application.

The comparison of sliding window with or without temporal pyramid using ideal objects annotation shows that the performance is almost the same. We believe that this is because of the variation in duration is relatively small than the dataset [1].

In this experiment, we evaluate the effect of using a strategy of label sequence to find the interleaved activities. The comparison is made by using two different conditional random field models, one is trained with sequence data, the other is trained with single state data only. In other word, the second model is similar to Naive Bayes model which does not consider neighboring state but only observations of the current state. The result is shown in figure 4.7 4.8. The result reveals that the multi-stage inferecing produce small but noticeable improvement.

We conclude that in such simple scenario where the interleaved activities are not similar with each other (for example, the objects feature have hardly any connection between make coffee and make photo copy), the benefit of multi-stage inference is limited. How-

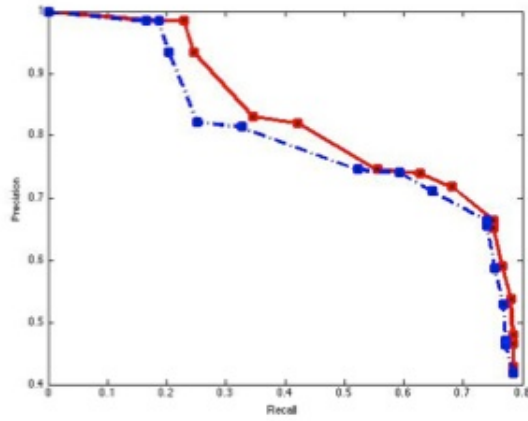


Figure 4.8: Comparison between single-stage and multi-stage activity inferencing, considering multi-stage activities only. The blue dotted line is the result of single-stage activity inferencing ; The red solid line is the result of multi-stage activity inferencing.

ever, we still have expectation that the multi-stage inference will come into effect when facing complex scenarios.



Chapter 5

Conclusion


In this paper, we propose a method to solve the problem of activity recognition in first-person-view videos by using high-level object appearance as visual clue, feature aggregation of temporal pyramid and conditional random fields.


- Conditional Random Fields
 - We are known to be first using CRF for activity recognition in first-person-view
 - We have proved that CRF is capable to handle this problem
- Temporal Pyramid
 - Improves imperfect object detection
 - Improves activity detection in complex environment
- Overall
 - Our method is better than state-of-the-art when using ideal object detection
 - Multi-stage sequence finding produce small but noticeable improvement



Bibliography

- [1] Hamed Pirsiavash and Deva Ramanan. Detecting activities of daily living in first-person camera views. *In CVPR, 2012*.
- [2] Steve Hodges, Lyndsay Williams, Emma Berry, Shahram Izadi, James Srinivasan, Alex Butler, Gavin Smyth, Narinder Kapur, and Ken Wood. Sensecam: A retrospective memory aid. *In International Conference on Ubicomp, 2006*.
- [3] B. Kopp, A. Kunkel, H. Flor, T. Platz, U. Rose, K.H. Mauritz, K. Gresser, K.L. McCulloch, and E. Taub. The arm motor ability test: reliability, validity, and sensitivity to change of an instrument for assessing disabilities in activities of daily living. *Arch Phys Med Rehabil*, 78(6):615--20, 1997.
- [4] J. K. Aggarwal, Michael S. Ryoo, and Kris M. Kitani. Frontiers of human activity analysis, 2011, Apr. [Online; CVPR2011 tutorial].
- [5] Douglas L. Vail, Manuela M. Veloso, and John D. Lafferty. Conditional random fields for activity recognition. *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*.
- [6] Young-Seol Lee and Sung-Bae Cho. Activity recognition using hierarchical hidden markov models on a smartphone with 3d accelerometer. *In HAIS, 2011*.
- [7] Tao Gu, Zhanqing Wu, Xianping Tao, Hung Keng Pung, and Jian Lu. epsicar: An emerging patterns based approach to sequential, interleaved and concurrent activity recognition. *In PERCOM, 2009*.

- 
- [8] Wen-Huang Cheng, Yung-Yu Chuang, Bing-Yu Chen, Ja-Ling Wu, Shao-Yen Fang, Yin-Tzu Lin, Chi-Chang Hsieh, Chen-Ming Pan, Wei-Ta Chu, and Min-Chun Tien. Semantic-event based analysis and segmentation of wedding ceremony videos. *Proceedings of the international workshop on Workshop on multimedia information retrieval*, pp. 95-104, 2007.
- [9] Thi V. Duong, Hung H. Bui, Dinh Q. Phung, and Svetha Venkatesh. Activity recognition and abnormality detection with the switching hidden semi-markov model. *In CVPR, 2005*.
- [10] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *In ICML, 2001*, pp. 282-289, 2001.
- [11] Derek Hao Hu and Qiang Yang. Cigar: concurrent and interleaving goal and activity recognition. *In AAAI, 2008*.
- [12] Ivan Laptev, Marcin Marszalek, Cordelia Schmid, and Benjamin Rozenfeld. Learning realistic human actions from movies. *In CVPR 2008*.
- [13] Mohammad Amin Sadeghi and Ali Farhadi. Recognition using visual phrases. *In CVPR, 2011*.
- [14] Emmanuel Munguia Tapia, Stephen S. Intille, and Kent Larson. Activity recognition in the home using simple and ubiquitous sensors. *In In Pervasive*, pages 158--175, 2004.
- [15] T. Kudo. Crf++: Yet another crf toolkit, 2007, Aug.
- [16] Heng Wang, Muhammad Muneeb Ullah, Alexander Kläser, Ivan Laptev, and Cordelia Schmid. Evaluation of local spatio-temporal features for action recognition. *British Machine Vision Conference, 2009*.
- [17] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *In PAMI, 2010*.

- 
- [18] Shengcai Liao, Xiangxin Zhu, Zhen Lei, Lun Zhang, and Stan Z. Li. Learning multi-scale block local binary patterns for face recognition. *In ICB 2007*.
- [19] Alireza Fathi, Xiaofeng Ren, and James M. Rehg. Learning to recognize objects in egocentric activities. *In CVPR, 2011*.