Deleted Interpolation using a Hierarchical Bayesian Grammar Network for Recognizing Human Activity

Kris KITANI†, Yoichi SATO†, and Akihiro SUGIMOTO††

† Institute of Industrial Science, University of Tokyo 4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, Japan
†† National Institute of Informatics 2-1-2 Hitotsubashi, Chiyoda, Tokyo 101-8430, Japan

Abstract From the viewpoint of an intelligent video surveillance system, the high-level recognition of human activity requires a priori hierarchical domain knowledge as well as a means of reasoning based on that knowledge. We approach the problem of human action recognition based on the understanding that actions are hierarchical, temporally constrained and temporally overlapped. While stochastic grammars and graphical models have been widely used for the recognition of human action, methods combining hierarchy and complex inference have been limited. We propose a new method of merging and implementing the advantages of both approaches to recognize actions in real-time. To address the hierarchical nature of human action recognition and to recognize temporally constrained and overlapped human behavior, we implement a Hierarchical Bayesian Network (HBN) based on a Stochastic Context-Free Grammar (SCFG) . The HBN is applied to different permutations of the new evidence and a limited set of past evidences via deleted interpolation to calculate the probability distribution of the current state of action. Results from the analysis of action sequences from a video surveillance camera show the validity of our approach.

Key words Human Activity Recognition, Hierarchical Bayesian Network, Stochastic Context-Free Grammar, Deleted Interpolation

1. Introduction

The automation of video surveillance is a topic of growing interest in recent times. This is because understanding human activity as it happens and differentiating between normal and abnormal trends of activity enable us to identify potential danger. By automatically analyzing human activities in real-time, we can recognize suspicious activity or even predict certain actions before they happen. Whereas current video surveillance systems are use to analyze human actions after they have occurred, real-time feed back from video images allow us to gather more useful information. A fully automated system such as this can then be used to alert security about hazardous behavior or even identify a shop-lifter.

To implement such a system, our task then becomes twofold. First, we need a framework to be able to characterize high-level human action and secondly, we need a method of interpreting those actions when they occur or even before they are completed.

The characteristics of human actions can be learned from perceptual psychology [1]. Actions are hierarchical. That is, actions are taxonomically organized, existing at various levels of abstraction. Walking and running are a *type of* moving. Actions are also partonomical and temporally constrained. The action of walking consists of a right foot forward action and a left foot forward action, i.e sub-actions are temporally constrained ordered parts. Actions can also be temporally overlapped. For example, sitting in a chair might be interpreted as a transition from a standing action to a sitting action, where the transitional action is an overlap of the two actions. There is an inherent ambiguity in differentiating human actions, especially at transition phases between two actions.

To address the latter half of the problem, namely the recognition of human actions from a sequence of video images, we need an efficient method of incorporating the characteristics of actions mentioned above. The high-level action recognition system must encode hierarchical information about actions, capture temporally constrained actions as well as temporally overlapped actions.

This paper addresses the issue of hierarchy by implementing a stochastic context-free grammar (SCFG). We also use Bayesian Networks (BN) to capture the temporally constrained nature of actions. We combine both the SCFG and the BN to create a Hierarchical Bayesian Network (HBN) and apply the HBN via deleted interpolation to the stream of observations to recognize overlapped actions in the stream. By implementing this system we are able pre-process substrings of low-level input data and identifying temporally overlapped actions to produce a richer set of intermediate-level actions (mid-level actions between low-level and high-level actions) to be used for higher levels of recognition. Furthermore, our approach lays the framework for predicting the next action based on the presupposition that certain actions can be temporally overlapped.

It is noted here that we are not directly addressing the issue of extracting symbols from a video sequence. Instead, we assume that a set of reliable low-level input observations (e.g. appearance and movement attributes) are available to us, allowing us to focus on building up a scheme for action recognition based on those observations. Also, the mechanics of grammar creation is not the focus of this paper because grammars are domain dependent and vary according to the application.

2. Related Research

The recognition of high-level human action from video can be seen as an intersection of two fields, namely (a) Computer Vision/Pattern Recognition and (b) Plan Recognition.

Contributions from computer vision and pattern recognition started with Brand in [2] and [3], when he utilized a deterministic action grammar to interpret a video sequence a person opening a computer housing unit. Multiple parses over a stream of outputs from the low-level event detector were ranked and stored, taking the interpretation of the highest ranking parse. Ivanov *et al* [4] first used an SCFG for action recognition, using the Earley-Stolcke parser to analyze a video sequence of cars and pedestrians in a parking lot. Moore *et al* [5] also used an SCFG to recognize actions in a video sequence of people playing Blackjack. They extend the work of Ivanov and Bobick by adding error correction, recovery schema and role analysis. Minnen *et al* [6] build on the modifications made by Moore and Essa by adding event parameters, state checks and internal states. They apply the SCFG to recognize and make predictions about actions seen in a video sequence of a person performing the Towers of Hanoi task.

From a background in plan recognition, Bui [7] used a hierarchy of abstract policies using Abstract Hidden Markov Model (AHMM) implementing a probabilistic state-dependent grammar (PSDG) to recognize action. The system recognizes people going to the library and using the printer across multiple rooms. AHMMs closely resemble the Hierarchical Hidden Markov Models (HHMM) [8] but with an extra state node. Nguyen *et al* [9]used a AHMEM, which is a modified version of an AHMM for the same scenario as Bui.

The aforementioned works use domains with high-level actions delineated by clear starting points and clear ending points, where the incoming low-level input observations are assumed to describe a series of temporally constrained actions (with the exception of Ivanov [4]). However, in our research we focus on a subset of human actions that have the possibility of being temporally overlapped. We show that these types of actions can be recognized efficiently by observing only a small window of time.

3. Modeling Human Action

Human actions are ordered hierarchically much like words in a natural language. Thus an understanding of hierarchy can be leveraged to reason about actions, just like one can guess at the meaning of a word from context. The SCFG [10] and the BN [11] lay the groundwork for applying hierarchical information to human activity recognition.

Our justification in using an SCFG to model human action is based on the idea that it models hierarchical structure that closely resembles the inherent hierarchy in human action. Furthermore, just as in the case of a natural language, an SCFG is able to handle multiple interpretations of an action at any variable length.

The merit of using a Bayesian network is found in the wide range of queries that can be executed over the network. In addition, BNs can deal with negative evidence, partial observations (likelihood evidence) and even missing evidence.

4. Recognition System Overview

Our recognition system consists of three major parts (Figure 1). The first is the action grammar (an SCFG) that describes the hierarchical structure for all the actions to be recognized. Second is the HBN that is generated from the action grammar. Third is the analysis module that takes a stream of input symbols (low-level action symbols) and uses deleted interpolation to determine the current probability for each possible output symbol (level 2 action symbol).

We give the details of our system based on the use of the CAVIAR data set [12], which is a collection of video sequences of people in a lobby environment. The ground truth for each agent in each frame is labeled in XML with information about position, appearance, movement, situation, roles, and context. In the next section we explain how this ground truth data was used to create an input stream for our system.

4.1 Action grammar

The set of nonterminals is defined as **T**= {*en, ex, mp, wa,*

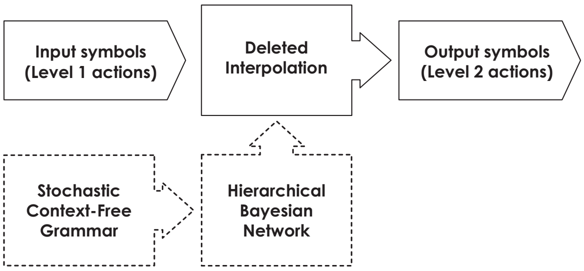

Figure 1 System Flow Chart. Dashed lines indicate off-line components and bold lines indicate online components. Level 1 actions symbols and the HBN are merged via the deleted interpolation step to produce level 2 actions.

in, br, pu, pd } (also called level 1 action symbols) and their meanings are given in Figure 2.

The level 1 action symbols were generated directly from the CAVIAR XML ground truth data using the *appearance* and *movement* information for each frame (Figure 3) . Again we note here that grammar creation is not the focus of our work but rather the method of processing the low-level terminal symbols.

Level 1 Actions	Meaning
en	enter
ex	exit
mp	move in place
in	inactive
wa	walk
ru	run
pu	pick up
pd	put down
br	browse

Figure 2 Description of the level 1 action (input) symbols.

		APPEARANCE	MOVEMENT
en	→	appear	
ex	→	disappear	
mp	→	visible	active
in	→	visible	inactive
wa	→	visible	walking
	$\mid$	occluded	walking

Figure 3 Level 1 grammar. The level 1 grammar is used to generate the level 1 action symbol (note: pd,pu,br are created using other information).

The set of action symbols (called level 2 actions) $\{BI, BR, TK, LB, PT, AR, DP\}$, were created manually to be a set of high-level actions to be recognized by the system. Respective definitions are given in Figure 4. Level 2 actions are a special subset of nonterminal symbols in the level 2 grammar. Level 2 actions are direct abstraction productions of S (start symbols), i.e. they are directly caused by S. The production rules $\mathbf{R}$ and their corresponding probabilities $\mathbf{p}$ are given in Figure 5.

Level 2 Actions	Meaning
BI	being idle
BR	browsing
TK	taking
LB	leaving behind
PT	passing through
AR	arriving
DP	departing
Intermediate Actions	**Meaning**
ST	stop in place
MV	moving
MT	move to
MF	move from

Figure 4 Set of all nonterminal symbols - Level 2 actions (directly produced by S) plus intermediate actions. Each nonterminal embodies an action in the action hierarchy.

S	→	**BI**			0.20	TK	→	pu			0.50
	$\mid$	**BR**			0.10		$\mid$	MV	pu		0.20
	$\mid$	**TK**			0.05		$\mid$	pu	MV		0.20
	$\mid$	**LB**			0.05		$\mid$	MV	pu	MV	0.10
	$\mid$	**PT**			0.30	LB	→	pd			0.50
	$\mid$	**AR**			0.15		$\mid$	MV	pd		0.20
	$\mid$	**DP**			0.15		$\mid$	pd	mp		0.20
BI	→	ST			0.10		$\mid$	mp	pd	mp	0.10
	$\mid$	MV	ST		0.10	PT	→	en	wa	ex	1.00
	$\mid$	ST	MV		0.10	AR	→	en			0.50
	$\mid$	MV	ST	MV	0.10		$\mid$	en	MV		0.50
	$\mid$	MV			0.20	DP	→	ex			0.50
	$\mid$	MF	mp		0.20		$\mid$	MV	ex		0.50
	$\mid$	MT	wa		0.20						
ST	→	in			0.50	MV	→	MF			0.20
	$\mid$	br			0.50		$\mid$	MT			0.20
BR	→	br			0.20		$\mid$	wa			0.30
	$\mid$	MV	br		0.20		$\mid$	mp			0.30
	$\mid$	br	mp		0.30	MF	→	mp	wa		1.00
	$\mid$	MV	br	mp	0.30	MT	→	wa	mp		1.00

Figure 5 Level 2 action grammar - The set of production rules $\mathbf{R}$ and their probabilities $\mathbf{p}$. The bold red nonterminal symbols are the set of level 2 action symbols.

4.2 Hierarchical Bayesian Network

We use the methods presented in [13] to transform the action grammar (level 2 grammar) into a Hierarchical Bayesian Network (HBN). We use the term HBN because information about hierarchy from the SCFG is embedded in the BN.

By converting the action grammar to an HBN, terminal symbols become *states* in the evidence nodes $\mathbf{E}$ of the HBN. Nonterminal symbols become *states* of the query nodes $\mathbf{Q}$. The set of production rules $\mathbf{R}$ become *states* of the hidden nodes $\mathbf{H}$ of the HBN.

We denote the probability density function (PDF) for a level 2 action to be $\mathbf{P}(\mathbf{a}|\mathbf{e}^t)^{(1)}$ where $\mathbf{a} = \{a_0, a_1, \ldots, a_n\}$ is the set of all level 2 actions (states) and $\mathbf{e}^t = \{e^{t-l-1}, e^{t-l}, \ldots, e^t\}$ is a string of evidence at the evidence nodes of the HBN. The probability of a level 2 action is defined as the sum of the probabilities from each of those nodes,

(1) : $\mathbf{P}$ will be used when dealing with probabilities of multi-valued discrete variables. It denotes a set of equations with one equation for each value of the variable.

$$\mathbf{P}(\mathbf{a}|\mathbf{e}^t) = \mathbf{P}(Q_0 = \mathbf{a}|\mathbf{e}^t) + \cdots + \mathbf{P}(Q_m = \mathbf{a}|\mathbf{e}^t), \quad (1)$$

where $\{Q_0, Q_1, \ldots, Q_m\}$ is the set of all query variables $\mathbf{Q}$.

If there are $n+1$ different level 2 actions, $\mathbf{P}(\mathbf{a}|\mathbf{e}^t)$ represents a set of $n+1$ equations

$$\begin{aligned} P(a_0|\mathbf{e}^t) &= P(Q_0 = a_0|\mathbf{e}^t) + \cdots + P(Q_m = a_0|\mathbf{e}^t), \\ P(a_1|\mathbf{e}^t) &= P(Q_0 = a_1|\mathbf{e}^t) + \cdots + P(Q_m = a_1|\mathbf{e}^t), \\ &\cdots \\ P(a_n|\mathbf{e}^t) &= P(Q_0 = a_n|\mathbf{e}^t) + \cdots + P(Q_m = a_n|\mathbf{e}^t). \end{aligned}$$

Adding all the probabilities of the level 2 actions always sums to one because $\mathbf{a}$ is the set of all possible productions of S. Thus,

$$P(a_0|\mathbf{e}^t) + \cdots + P(a_n|\mathbf{e}^t) = 1. \quad (2)$$

4.3 Deleted Interpolation

Using deleted interpolation the HBN is applied to digressive subsets of the evidence within the analysis window when a new evidence symbol is encountered in the input stream. The current probability distribution is calculated as a weighted sum of select HBNs.

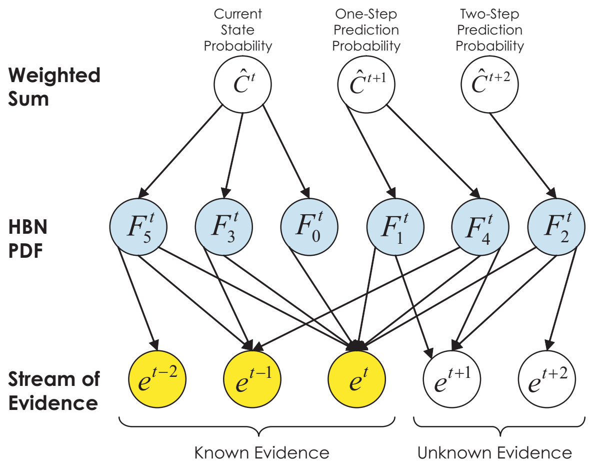

Figure 6 Deleted interpolation with a HBN ($l+1=3$) at time t (F_i^t is a node representing the output of the i-th HBN)

Figure 6 depicts the application of the HBNs to the stream of evidence at t when the maximum length of the HBN is three. The nodes e in the bottom layer represent the stream of evidence within the analysis window. The nodes F^t in the second layer are the PDFs of the level 2 actions at time t (i.e. the output of the HBN query nodes $\mathbf{P}(\mathbf{a}|\mathbf{e}^t)$) across digressive subsets of the evidence. The nodes in the third layer represent the weighted sums of the PDFs of the HBNs.

When $l+1$ is the maximum length of the HBN, all evidence before $t-l-1$ is ignored by the HBN because it has a limited number of evidence nodes.

Now let us be more concrete by assuming that the maximum length is three (i.e. $l+1=3$). The evidence available at time t is

$$\mathbf{e^t} = \{e^{t-2}, e^{t-l}, e^t\}.$$

Next we apply the HBN to the digressive subsets of the evidences starting with combinations of the current evidence e^t, then e^{t-1}, and finally e^{t-2}. Since $l+1=3$, there are six valid subsets given as [2]

$$\begin{aligned} \mathbf{e}_0^t &= \{e_1^t, e_2^{none}, e_3^{none}\}, \\ \mathbf{e}_1^t &= \{e_1^t, e_2^-, e_3^{none}\}, \\ \mathbf{e}_2^t &= \{e_1^t, e_2^-, e_3^-\}, \\ \mathbf{e}_3^t &= \{e_1^{t-1}, e_2^t, e_3^{none}\}, \\ \mathbf{e}_4^t &= \{e_1^{t-1}, e_2^t, e_3^-\} \text{ and} \\ \mathbf{e}_5^t &= \{e_1^{t-2}, e_2^{t-1}, e_3^t\}. \end{aligned}$$

A combination like $\{e_1^{t-2}, e_2^-, e_3^t\}$ is considered invalid because we know e_2^{t-1}. Likewise, $\{e_1^{t-2}, e_2^{none}, e_3^t\}$ is invalid because e_2 cannot be the end of the string since we know that a symbol at e_3^t exists.

The first combination $\mathbf{e}_0^t$ assumes that e^t is the only symbol in the input sequence. The second combination $\mathbf{e}_1^t$ assumes that e^t is known and that e^{t+1} is an unknown symbol expected to be the last symbol in the sequence.

Six combinations means that we have six corresponding PDFs

$$\mathbf{F}_i^t = \mathbf{P}(\mathbf{a}|\mathbf{e}_i^t) \text{ for } i = 0, 1, 2, 3, 4, 5.$$

The set of probabilities is

$$\begin{aligned} \mathbf{P}(\mathbf{a}|\mathbf{e}_0^t) &= \mathbf{P}(Q_0 = \mathbf{a}|\mathbf{e}_0^t) + \cdots + \mathbf{P}(Q_m = \mathbf{a}|\mathbf{e}_0^t) \\ \mathbf{P}(\mathbf{a}|\mathbf{e}_1^t) &= \mathbf{P}(Q_0 = \mathbf{a}|\mathbf{e}_1^t) + \cdots + \mathbf{P}(Q_m = \mathbf{a}|\mathbf{e}_1^t) \\ &\cdots \\ \mathbf{P}(\mathbf{a}|\mathbf{e}_5^t) &= \mathbf{P}(Q_0 = \mathbf{a}|\mathbf{e}_5^t) + \cdots + \mathbf{P}(Q_m = \mathbf{a}|\mathbf{e}_5^t). \end{aligned}$$

Observing the PDFs we notice that $\mathbf{P}(\mathbf{a}|\mathbf{e}_0^t)$, $\mathbf{P}(\mathbf{a}|\mathbf{e}_3^t)$ and $\mathbf{P}(\mathbf{a}|\mathbf{e}_5^t)$ all assume e^t to be the last terminal in the sequence. We also observe that $\mathbf{P}(\mathbf{a}|\mathbf{e}_1^t)$ and $\mathbf{P}(\mathbf{a}|\mathbf{e}_4^t)$ represent configurations in which one more symbol is expected before the end of the sequence. Lastly, we observe that $\mathbf{P}(\mathbf{a}|\mathbf{e}_2^t)$ represents a configuration in which two more symbols are expected before the end of the sequence.

For practical reasons we give weights to the different combinations of evidences and calculate their sums (these weights will eventually be replaced by a probability function). The current state probability is

$$\hat{\mathbf{C}}^t = w_0\mathbf{F}_0^t + w_3\mathbf{F}_3^t + w_5\mathbf{F}_5^t.$$

5. Experimental Results

The following experiment aims to show that our method of action recognition is well-suited for recognizing temporally constrained actions and temporally overlapped actions.

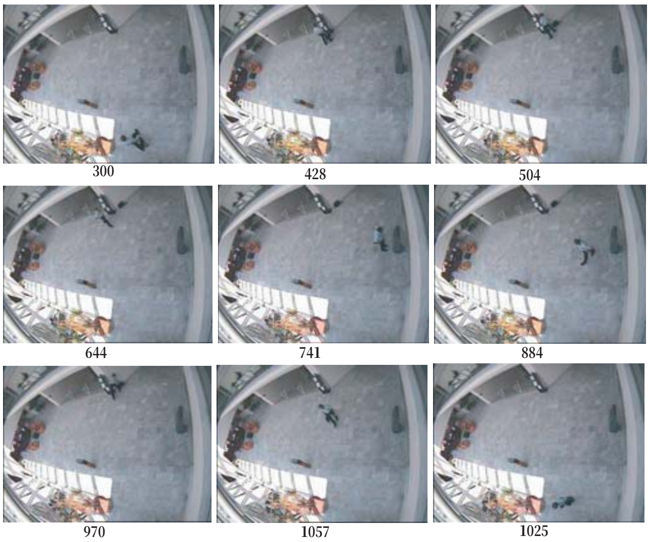

Figure 7 Key frames for the "Leave Behind and Pick Up" (Leave1) sequence.

Arriving	A period of time where the agent has just entered the scene. It must occur near a known exit or entrance.
Departing	A period of time where is seems that the agent is about to leave the scene. Ceases once the agent leaves the scene.
Passing Through	The agent appears to be simply walking through the lobby. Pattern should look like: Enter + passing through + exit. Agent is not looking around.
Browsing	A periof of time where the agent is near the counter, the magazine rack or the information booth. The agent appears to be looking at the landmark.
Being Idle	The agent appears to be walking around aimlessly. Usually characterized by walking slowly and stoping in place. Sometimes includes browsing.
Taking Away	The agent appears to be picking something up or preparing to pick something up. Includes movement just before and after picking up the object.
Leaving behind	The agent appears to be leaving something behind or preparing to leave something behind. Includes movement just before and after leaving the object.

Figure 8 Definitions for ground truth labeling.

The ground truth was compiled as a normalized sum of the interpretations of multiple people. Each labeler was given a definition for each level 2 action and asked to label each action separately. They were also given the option of labeling each frame with either a *yes*, *maybe* or *no*. No restrictions were placed on the number of times they could re-label the video sequences (i.e. they were given access to knowledge about future actions).

Analysis was run on four video sequences (Walk1, Walk2, Browse1, Leave1) to test the detection rate of the system. The system mistakenly labeled *Being Idle* as *Passing Through* for the Browse1 sequence (Figure 13) which adversely influenced the average recall and precision rate. This is expected however because the agent appears to be simply "passing through" the lobby before approaching the information booth. After frame 615 the system adjusts its interpretation of the scene once the agent is found to be browsing.

(2) : e^{-} represents missing evidence, e^{none} is a terminal symbol that represents the end of the sequence and the subscripts denote the start index i corresponding to the evidence nodes of the HBN.

		WALK1	WALK2	BROWSE1	LEAVE1	Average
Arriving	Recall	95%	62%	67%	22%	61%
	Precision	87%	99%	100%	100%	97%
Passing Through	Recall	87%	90%	0%	-	59%
	Precision	100%	96%	0%	-	65%
Being Idle	Recall	-	-	63%	80%	72%
	Precision	-	-	100%	94%	97%
Browsing	Recall	-	-	74%	57%	66%
	Precision	-	-	97%	83%	90%
Taking Away	Recall	-	-	-	18%	18%
	Precision	-	-	-	100%	100%
Leaving Behind	Recall	-	-	-	8%	8%
	Precision	-	-	-	100%	100%
Departing	Recall	62%	32%	39%	63%	49%
	Precision	100%	100%	56%	100%	89%

Figure 9 Recall and precision measurements.

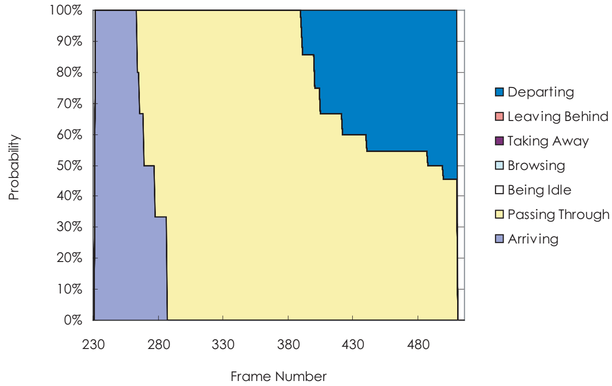

Figure 10 Ground truth for Walk1.

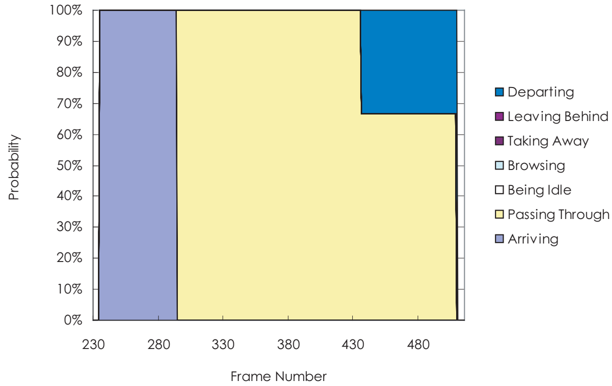

Figure 11 Output data for Walk1.

100% of the seven level 2 actions contained in the four sequences were detected using the proposed system. The overall recall rate was 91% while the average precision rate was 49%. The low precision rate can be attributed to the fact that the current system only changes states when there is a significant change in the movement and appearance of the agent (i.e. when a new terminal symbol is encountered). The gradual transitions between level 2 actions can be better characterized by utilizing likelihood evidence (probabilities associated with each terminal symbol) at the inputs of the system, which will be addressed in future work.

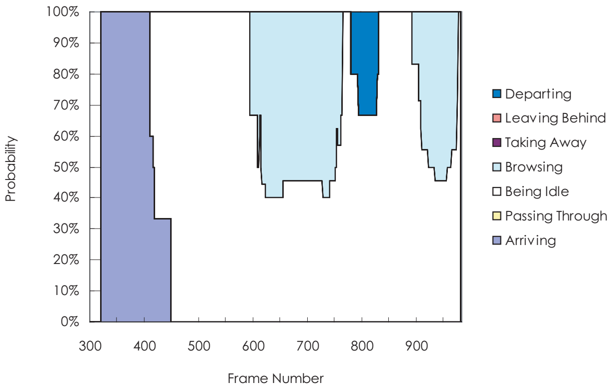

Figure 12 Ground truth for Browse1.

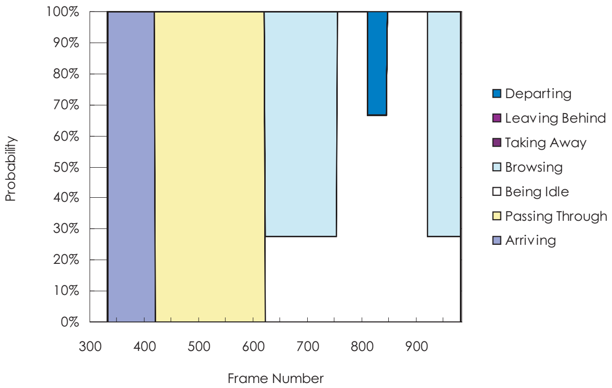

Figure 13 Output data for Browse1.

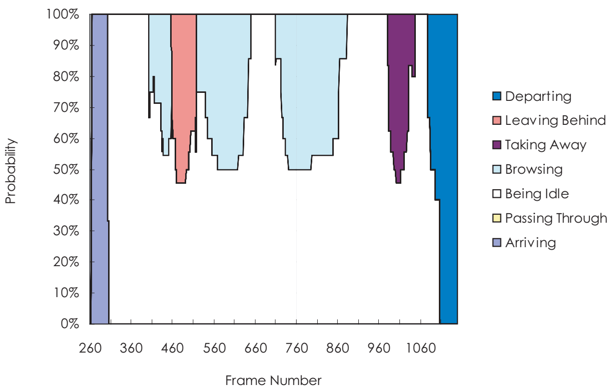

Figure 14 Ground truth for "Leave Behind and Pick Up" (Leave1).

6. Summary and Conclusions

We have addressed the issue of hierarchical recognition of human action by basing our system on a SCFG. We then converted the SCFG to a HBN to recognize temporally constrained actions. We then applied a method of deleted interpolation using the HBN to the stream of low-level input symbols to recognize overlapped actions. We showed that preprocessing substrings of low-level input data and identifying temporally overlapped actions can produce a variety of high-level actions. Our results showed that our framework is a valid method for recognizing human activity.

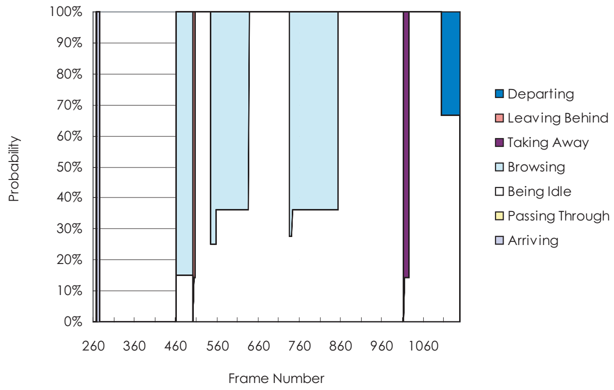

Figure 15 Output data for "Leave Behind and Pick Up" (Leave1). All major events are detected.

7. Acknowledgments

A part of this work was supported by Grants-in-Aid for Scientific Research from the Ministry of Education, Culture, Sports, Science and Technology of Japan (No. 13224051).

References

[1] J. Zacks and B.Tversky. Event Structure in Perception and Conception. *Psychological Bulletin*, 127, 2001.

[2] M. Brand. Understanding manipulation in video. In *Proceedings of 2nd International Conference on Automatic Face and Gesture Recognition*, pages 94–99, 1996.

[3] M. Brand. The Inverse Hollywood Problem: From video to scripts and storyboards via causal analysis. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence*, pages 132–137, 1997.

[4] Y.A. Ivanov and A.F. Bobick. Recognition of Visual Activities and Interactions by Stochastic Parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(08):852–872, 2000.

[5] D. Moore and I. Essa. Recognizing Multitasked Activities using Stochastic Context-Free Grammar. In *Proceedings of the Computer Vision and Pattern Recognition*, 2001.

[6] D.Minnen, I.Essa, and T.Starner. Expectation Grammars: Leveraging High-Level Expectations for Activity Recognition. In *Proceedings of the Computer Vision and Pattern Recognition*, 2003.

[7] H.H. Bui, S. Venkatsh, and G. West. Tracking and Surveillance in Wide-area Spatial Environments Using the Abstract Hidden Markov Model. *International Journal of Pattern Recognition and Artificial Intelligence*, 15(01):177–795, 2001.

[8] S. Fine, Y. Singer, and N. Tishby. The Hierarchical Hidden Markov Model: Analysis and Applications. *Machine Learning*, 32(1):41–62, 1998.

[9] N.T.Nguyen, H.H. Bui, S. Venkatsh, and G. West. Recognizing and Monitoring High-level Behaviors in Complex Spatial Environments. In *Proceedings of the Computer Vision and Pattern Recognition*, pages 620–625, 2003.

[10] C.D. Manning and H. Schutze. *Foundations of Statistical Natural Language Processing.* The MIT Press, 2003. ISBN: 0-262-13360-1.

[11] S. Russell and P. Norvig. *Artificial Intelligence: A modern approach.* Prentice Hall, 2003. ISBN: 0-13-790395-2.

[12] EC Funded CAVIAR project/IST 2001 37540. Found at http://homepages.inf.ed.ac.uk/rbf/CAVIAR/.

[13] D.V. Pynadath and M.P. Wellman. Generalized Queries on Probabilistic Context-Free Grammars. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(08):65–77, 1998.