Real-Time Tracking of Human Location and Motion using Cameras in a Ubiquitous Smart Home

Dongkyoo Shin, Dongil Shin, Quoc Cuong Nguyen, and Seyoung Park
Department of Computer Engineering, Sejong University,
98 Kunja-Dong, Kwangjin-Ku, Seoul 143-747, Korea
Phone: +82-2-3408-3242, Fax: +82-2-498-4273
[e-mail: {shindk, dshin}@sejong.ac.kr, {cuongnqc, sypark}@gce.sejong.ac.kr]
*Corresponding author: Dongkyoo Shin

Received January 5, 2009; revised February 4, 2009; accepted February 5, 2009; published February 23, 2009

Abstract

The ubiquitous smart home is the home of the future, which exploits context information from both the human and the home environment, providing an automatic home service for the human. Human location and motion are the most important contexts in the ubiquitous smart home. In this paper, we present a real-time human tracker that predicts human location and motion for the ubiquitous smart home. The system uses four network cameras for real-time human tracking. This paper explains the architecture of the real-time human tracker, and proposes an algorithm for predicting human location and motion. To detect human location, three kinds of images are used: $IMAGE_1$ - empty room image, $IMAGE_2$ - image of furniture and home appliances, $IMAGE_3$ - image of $IMAGE_2$ and the human. The real-time human tracker decides which specific furniture or home appliance the human is associated with, via analysis of three images, and predicts human motion using a support vector machine (SVM). The performance experiment of the human's location, which uses three images, lasted an average of 0.037 seconds. The SVM feature of human motion recognition is decided from the pixel number by the array line of the moving object. We evaluated each motion 1,000 times. The average accuracy of all types of motion was 86.5%.

Keywords: Real-time human tracker, smart home, ubiquitous computing, pattern recognition, support vector machine

This work was supported by the Korea Research Foundation Grant funded by the Korean government (MOEHRD) (KRF-2007-313-D00727).

DOI: 10.3837/tiis.2009.01.004

1. Introduction

The ubiquitous smart home provides an automatic home service via analysis of human and home contexts [1]. Furthermore, the smart home uses a unified context in the form of Who, What, Where, When, Why, and How for analyzing patterns of human behavior [2]. For example, when a human sits on sofa watching a TV program, the ubiquitous smart home analyzes the human's preferred channel and automatically provides a home service.

The ubiquitous smart home deals with multiple contexts from the home environment and the human, but the most important context is information about the human's location and motion [3]. Studies in location recognition are classified according to their sensor equipment; there are many methods, such as the use of a camera, infrared light, and pressure sensors without a camera. Almost all current location recognition methods use a camera but this involves many aspects. Firstly, images could be recognized from different forms of the image based on the light intensity. Secondly, object movement has to be distinguished from human movement. Finally, the information has to be updated when the furniture and home appliance are relocated and moved.

Human tracking for a smart space was studied by various methods. Pfinder is a real-time system for tracking a person, which uses a multi-class statistical model of color and shape to segment a person from a background scene. It finds and tracks peoples' heads and hands under a wide range of viewing conditions [4]. Tominaga and Hongo proposed a method for extracting human movement and hand gestures from multi-channel motion images captured in Percept Room [5]. Percept Room has 16 cameras on the wall. Eight cameras are installed at a height of 220 *cm* from the floor and eight more cameras are installed at a height of 90 *cm* from the floor level [5][6]. However, Percept Room did not provide useful service, because Percept Room did not consider the relationship between the furniture, home appliances, and human location. **Fig. 1** shows the configuration of Percept Room. We proposed a method of identifying the human location corresponding to the furniture for an extended home service. The other enhancement better suited to home service is motion recognition. Human motion is recognized in terms of various types including "lie down", "sit", "stand-up", and "walk".

KidRoom is a tracking system based on "closed-world regions", which are regions of space and time in which the specific context of what is in the regions is assumed to be known [7]. Guohui Li and Jun Zhang presented an effective approach for detecting a moving object from a video stream based on a background template, integrating multiple techniques addressing illumination changes, shadows and noisy disturbances [8]. EasyLiving (Microsoft) uses two sets of color stereo cameras for tracking people in a living room. The stereo images are used for locating people, and the color images are used for tracking their identities [9]. However, EasyLiving did not consider object movement and had to use a pressure sensor for checking whether the human was sitting on the sofa or not. The House_np project at MIT proposed a method for extracting a number of humans in a small room using an installed ceiling camera [10]. However, House_np found it hard to distinguish between humans and objects in the large area, because the camera has to be installed on the ceiling. Unlike previous studies, the real-time human tracker system can decide which furniture or home appliance a human is associated with, and can predict a human's motion using four network cameras. In this paper, four types of motion are recognized: "lie down", "sit", "stand-up", and "walk".

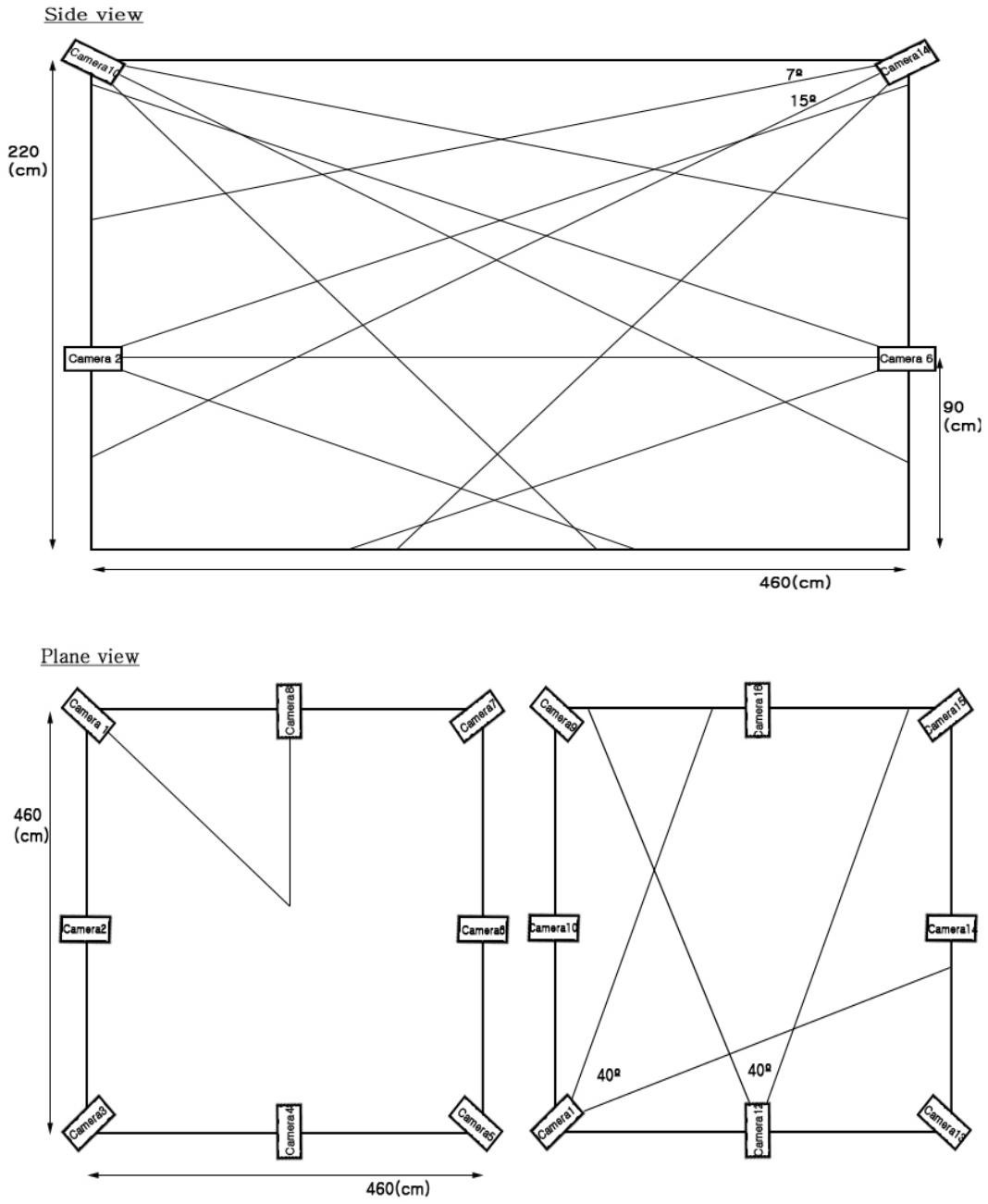

Fig. 1. Configuration of Percept Room

2. Architecture of Human Tracker

2.1 Experimentation room configuration

The real-time human tracker uses four digital network cameras to acquire a real-time image of the human. The camera system used in our experiment contains a 10 based-T LAN and consisted of one server camera with a 32-bit RISC CPU and three client cameras. **Fig. 2** shows

the camera arrangement in our ubiquitous smart home. The view scopes of the four cameras were found to be limited to an angle which is smaller than 90 degrees in each corner.

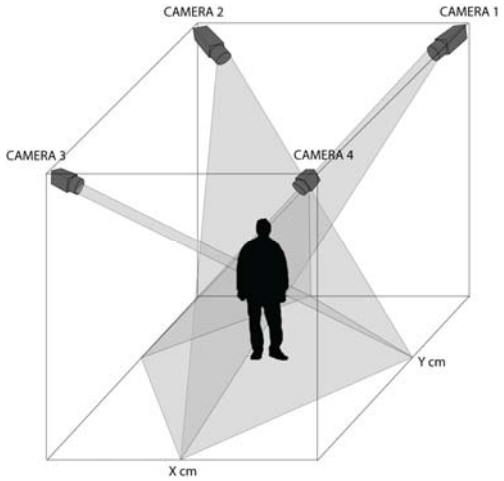

Fig. 2. Structure of the Camera system

2.2 Real-time Human Tracker

In order to estimate the human's location, the camera handler acquires a color image with 720x486 pixels from a digital network camera every two seconds, and deals with status information from the network camera and detection of transmission errors and recovery. The moving object detector calculates the moving object's area using the difference value between the background image that is stored in the real-time human tracker and the acquired image from the network camera, in real-time. The object classifier extracts feature values from the moving object's area that are analyzed by the moving object detector.

The position recognizer estimates the human's location using information of the furniture and home appliances in the ubiquitous smart home. The real-time human tracker calculates the absolute coordinates of the human using the transmitted image from four network cameras. In addition, it calculates the region information using the furniture and home appliance location information of the fixed object manager. The fixed object manager provides a tool that is needed to establish the furniture and home appliance location for calculating the region information, and it updates and manages the furniture and home appliance location information. **Fig. 3** shows the architecture of the real-time human tracker.

The system requires an operator to be involved in fixed object management. This process arranges the furniture and home appliances in the home context that is acquired via identification of a background image and an image of furniture. The input image is received via the camera system. At first, the operator records an image of an empty room. Then, when furniture or home appliances are changed, the operator records the image again and specifies in the database the type of furniture along with the position information of the room.

When the system is employed and the human position and human motions are detected as described above, several schemes are raised suited to the appropriate situation. For example, when a human is detected as sitting on sofa, the TV automatically turns on the preferred channels. When a walk motion is detected, all the lights are turned on, whereas when the system detects a lie motion, all home appliances are turned off to ensure total silence.

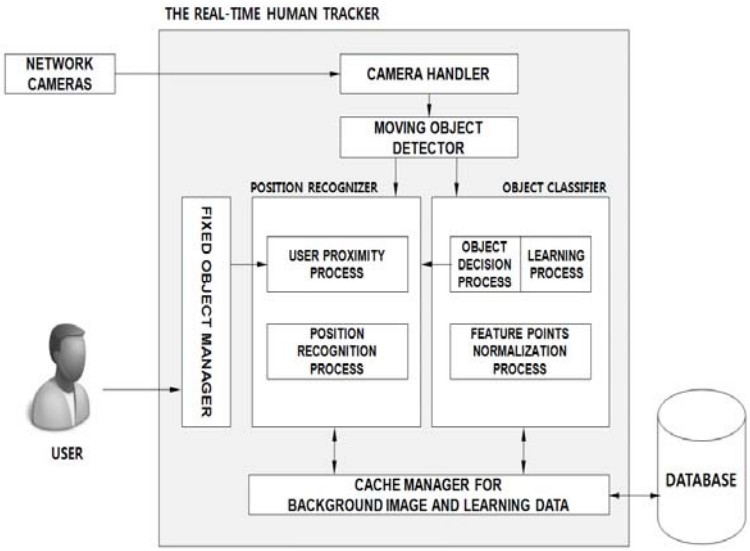

Fig. 3. Architecture of the real-time human tracker

3. Internal Algorithms of the Real-Time Human Tracker

3.1 Human classification

We use three images to decide which furniture or home appliance the human is associated with: $IMAGE_1$ - empty room image, $IMAGE_2$ - image of furniture and home appliances in the home, $IMAGE_3$ - image of $IMAGE_2$ and the human. **Fig. 4** shows the sample image for human classification. The real-time human tracker detects the image difference value between $IMAGE_1$ and $IMAGE_2$. And, it determines the location of the furniture in the ubiquitous smart home. The silhouette method is used for calculation of the location coordinates for the furniture, home appliances and human in the ubiquitous smart home [11]. However, the real-time human tracker needs additional methods, because the difference values between $IMAGE_1$, $IMAGE_2$, and $IMAGE_3$ might not be exact.

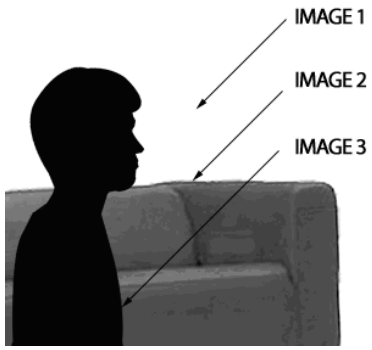

Fig. 4. Sample images for human classification

To process minute movements, we disregard a value that is smaller than a threshold value.

Human movement has to be distinguished from object movement. A change of $IMAGE_2$ means the movement of furniture and home appliances. A change of background due to object movement requires the update of new background information.

3.2. Human location recognition

Human location recognition is calculated by using the position recognizer, which calculates the absolute coordinates and region information of relative coordinates. Via the difference values between $IMAGE_1$ and $IMAGE_2$, the human tracker acquires the edges of the furniture and home appliances in the home, and calculates the absolute coordinates of the furniture using the pixel's array index (x, y) from the edge information. If the human enters the home, via the difference values between $IMAGE_2$ and $IMAGE_3$, the human tracker analyzes the pixel array index for the human image, and compares it with the pixel array index for the furniture or the home appliance. Region information means a closeness standard between the human and furniture or home appliance. It includes important information such as "in front of sofa" or "beside window." We use the moving object detector and object classifier to calculate the absolute coordinates of the human. The calculation of location coordinates use the silhouette method [11].

We define a region that can be covered by a camera as a *Region of Human*, and C_k (where k is the camera number) means *Region of Human*. We define the sum of *Region of Human* as F. F is represented as equation (1).

$$F = C_1 \cup C_2 \cup C_3 \cup C_4 \tag{1}$$

We define a region without humans as a *Deletable Region*, which is represented as D_k. *Projected Region* is calculated via the difference values between *Region of Human* and *Deletable Region*. T_k means *Projected Region*. T_k is represented as equation (2)

$$T_k = C_k - D_k \tag{2}$$

P is *Human Position*, which is calculated via the intersection of *Projected Region* T_k calculated from different cameras. P is represented as equation (3).

$$P = T_1 \cap T_2 \cap T_3 \cap T_4 \tag{3}$$

The information that is included in the neighborhood condition uses abstract neighborhood information instead of absolute coordinates such as "in front of sofa", "beside window" and "beside desk." The human location recognizer measures the width and height length per pixel, to calculate a closeness standard and select the major furniture and home appliance. The selected furniture and home appliance is defined as a fixed object. The fixed object manager manages the extracted edge of the fixed object, and saves and manages the object region. The human location recognizer selects the closeness of the fixed object by calculating the length between the human and the saved fixed object, and calculates the abstract neighborhood information.

3.3. Human motion recognition

Human motion recognition classifies human motion using an object classifier. The object classifier recognizes the human's motion using the human's image pixels. The human motion recognition then is combined with the human location recognition to produce the human oriented services via association rules mining. The specific value of each image is inputted into the learned algorithms, to predict several types of human motion. The support vector machine (SVM) is used for the human motion recognition [12].

Fig. 5 shows a method for calculating the absolute position of the human: **Fig. 6** shows the structure of the object classifier. The features extracted from images are created from the moving object detector and are used in four linear-SVMs to recognize each respective motion. The linear SVM is learned by using black and white images, which is the difference value information between the changed image transmitted from the camera handler and background image, and it recognizes and predicts human motion using the edge information that is extracted from the black and white image.

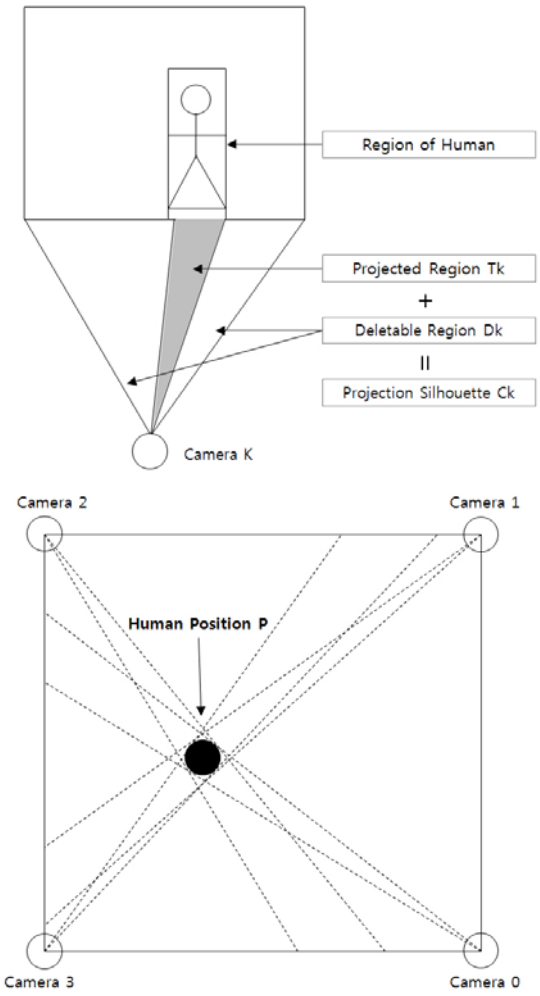

Fig. 5. Absolute position of human

Edge information pixels are in the range [0,255]. They are normalized to the range [0,1] and used as a normalized input value for SVM. Data obtained in the first learning period was saved

in the database and used for motion estimation and learning. The input value is refined by using the filter to narrow the data range, and the result of the learning and estimation period is saved in the database. The estimated result may be used in order to generate new actions with the association rules.

Four types of human motion are recognized: “lie down”, “sit”, “stand-up”, and “walk”. SVM is learned using a different image for each motion. In order to recognize each motion, the object classifier consists of four linear SVMs. The linear SVM selects human motion from the four types of human motion using the edge information. The object classifier recognizes whether the input image is human or not based on the estimated result of each linear SVM.

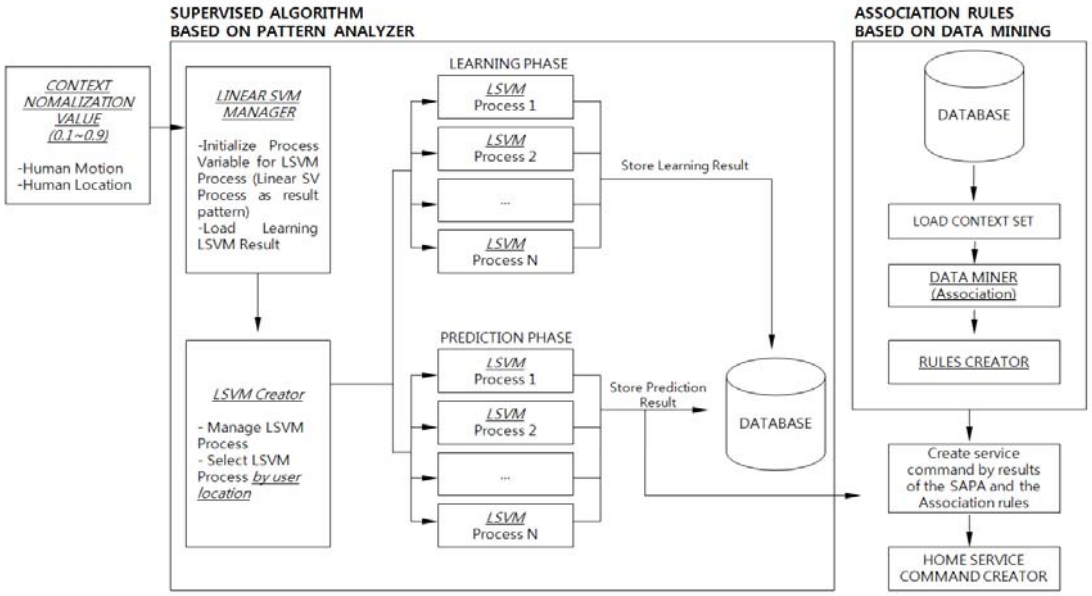

Fig. 6. Structure of the Object Classifier

The linear SVM we employed is a kind of *Linear Decision Boundary*, which is shown in equation (4)

$$w \cdot x + b = 0 \tag{4}$$

w and b are parameters of the model, and in this paper a two-dimensional training set was used. This means that equation (4) refers to x_a and x_b, which are different values of x and are two points located on the decision boundary. Main reason for choosing the Linear Decision Boundary SVM, instead of other linear techniques such as linear regression, the perceptron or Winnow, includes its flexible choices for margin of classifier. Those choices allow minimizing the error rate and estimating the parameter efficiently.

The final output of SVM is the decision result, which specifies what type of motion the input value belongs to. Combined with the human location, the rule is generated in the training process, or the expected services are raised based on the corresponding input value. If the context accuracy is smaller than a given pre-defined threshold, the system has to provide a substitute service list. Three methods are used to provide service for the system. First of all, the human could choose a high frequency service with the same context accuracy. Secondly, the human could choose a high frequency sequence with a certain period of time. Finally, a human

could choose a service with all kind of services.

4. Experiments and Evaluation

Typical motions handled by the real-time human tracker are shown in **Fig. 7**: “lie down”, “sit”, “stand-up”, and “walk”. The SVMs were trained to detect these motions separately. The white area in **Fig. 7** shows the feature values selected for human motion recognition, which represents the result of recognition analyzed by the object classifier.The edge information pixels are normalized to the range [0,1] and used as a normalized input value for SVM.

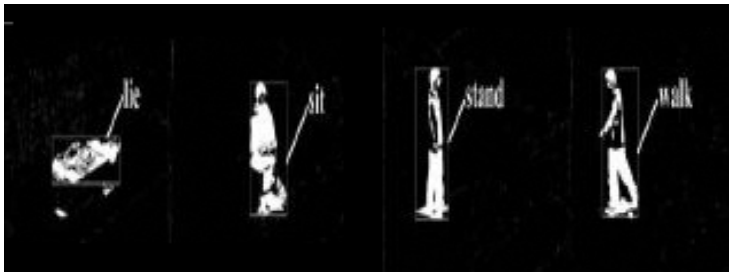

Fig. 7. Edge information for SVM

Table 1 shows the normalized input data for the edge information pixels for a certain motion. Each motion was evaluated 1,000 times; a dataset that included 1,000 records was used to train the SVM. Totally 4,000 records of the dataset were required to train all four types of motion. We separated the dataset used in training from the dataset used in testing. Each dataset contains 4, 000 records, corresponding to four types of motion.

Table 1. Normalized Input Data for SVM

Index \ Pixel number	1	2	3	4	5	6	7	8	9	10	…
Lie down 1	0.013	0.019	0.021	0.032	0.037	0.040	0.044	0.053	0.056	0.066	…
Lie down 2	0.006	0.009	0.010	0.012	0.013	0.014	0.016	0.017	0.020	0.019	…
…											
Lie down 1000	0.008	0.014	0.016	0.016	0.018	0.021	0.016	0.023	0.030	0.031	…
Sit 1	0.001	0.001	0.001	0.004	0.002	0.005	0.003	0.004	0.008	0.004	…
Sit 2	0.013	0.01	0.016	0.017	0.027	0.035	0.038	0.051	0.041	0.065	…
…											
Sit 1000	0.003	0.019	0.036	0.045	0.047	0.049	0.051	0.046	0.048	0.038	…
Stand-up 1	0.022	0.021	0.014	0.018	0.021	0.019	0.025	0.023	0.031	0.026	…
Stand-up 2	0.043	0.049	0.063	0.045	0.046	0.042	0.034	0.032	0.035	0.033	…
…											
Stand-up 1000	0.028	0.026	0.031	0.032	0.037	0.037	0.030	0.040	0.041	0.038	…
Walk 1	0.032	0.045	0.061	0.058	0.067	0.068	0.063	0.055	0.050	0.045	…
Walk 2	0.012	0.018	0.014	0.017	0.025	0.018	0.024	0.025	0.022	0.020	…
…											
Walk 1000	0.042	0.046	0.048	0.059	0.059	0.086	0.093	0.097	0.088	0.089	…

Table 2 shows the testing accuracy of the proposed real-time tracker. For each type of motion, the table shows the time (seconds) required to detect it, number of support vectors with estimated parameters (Number of SV), normalized value of the longest vector that occurs only once in the dataset (Norm of LV), and number of vectors in the range of the decision boundary (Number of KE). The performance experiment of the human's location lasted an average of 0.037 seconds. Each type of motion testing used 1,000 records of the dataset, and the average accuracy rate was 86.5%.

Table 2. Performance evaluation for the recognition of human location and motion (SV : support vector / LV : longest vector / KE : kernel evaluation)

Motion	Object location	Object motion			Total precision		
	Time (seconds)	Number of SV	Norm of LV	Number of KE	Total	Correct	Precision rate
Lie down	0.03699	171	1.75761	14,204	1,000	930	93.0%
Sit	0.03719	125	2.02598	13,085	1,000	911	91.9%.
Stand-up	0.03687	102	2.01333	13,377	1,000	821	82.1%
Walk	0.03695	98	2.14278	13,335	1,000	793	79.3%

6. Conclusion

We presented a real-time human tracker that predicts human location and motion for the ubiquitous smart home using four network cameras. The human tracker distinguishes the human by subtracting the background image from the input image and uses a silhouette method to calculate the absolute coordinates of the human. Human motion is predicted using linear SVMs. The SVM feature of human motion recognition is decided from the pixel number by an array line of the moving object. The performance experiment of the human's location, which used three images, lasted an average of 0.037 seconds. The average accuracy of all types of motion was 86.5%.

We are currently studying algorithms for eliminating the effect of shadows and lights in the real-time human tracker. Furthermore, we will study a new applicable algorithm for a smart home system and a ubiquitous health care system.

References

[1] S. K. Das and D. J. Cook, “Guest Editorial - Smart Homes,” *IEEE Wireless Communications*, vol. 9, no. 6, pp. 62-62, 2002.

[2] Seiie Jang and Woontack Woo, “A unified context-aware application model,” Springer, *Lecture Notes in Computer Science*, vol. 2680, pp. 178-189, Aug. 2003.

[3] Jonghwa Choi, Dongkyoo Shin, and Dongil Shin, “Research and implementation of the contextaware middleware for controlling home appliances,” *IEEE Transactions on Consumer Electronics*, vol. 51, no. 1, pp. 301-306, 2005.

[4] Christopher Wren, Ali Azarbayejani, Trevor Darrell, and Alex Pentland, “Pfinder : Real-time Tracking of the human Body,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 780 – 785, 1997.

[5] Masafumi Tominaga, Hitoshi Hongo, Hiroyasu Koshimizu, Yoshinori Niwa, and Kazuhiko Yamamoto, “Estimation of human motion from multiple cameras for gesture recognition,” Pattern Recognition, pp. 401-404, 2002.
[6] Hitoshi Hongo, Hiroki Watanabe, Mamoru Yasumoto, Yoshinori Niwa, and Kazuhiko Yamamoto, “Eye Regions Extraction for Omni-directional Gaze Detection Using Multiple Cameras,” In Proc. *IASTED Conference on Signal Processing, Pattern Recognition and Applications* (*SPPRA2001*), pp. 241-246, Jul. 2001.
[7] Aaron Bobick and James Davis, “Real-time recognition of activity using Temporal Templates,” In Proc. *Third IEEE Workshop on Application of Computer Vision*, pp. 1233-1251, 1996.
[8] Guohui Li, Jun Zhang, Hongwen Lin, Tu D, and Maojun Zhang, “A moving object detection approach using integrated background template for smart video sensor,” In Proc. *Instrumentation and Measurement Technology Conference*, vol. 1, pp. 462-466, 2004.
[9] John Krumm, Steve Harris, Brian Meyers, Barry Brumitt, Michael Hale, and Steve Shafer, “Multi-camera multi-person tracking for EasyLiving,” In Proc. *Third IEEE International Workshop on Visual Surveillance*, pp. 3-10, 2003.
[10] Rania Y. Khalaf and Stephen S. Intille, “Improving multiple people tracking using temporal consistency,” Dept, of Architecture House_n Project Technical Report, Massachusetts Institute of Technology, Cambridge, MA, 2001.
[11] M. Tominaga, H. Hongo, H. Koshimizu, Y. Niwa, and K. Yamamoto, “Estimation of human motion from multiple cameras for gesture recognition,” In Proc. 16th International Conference on Pattern Recognition, vol. 1, pp. 401-404, Aug. 2002.
[12] C. J.C. Burges, “A tutorial on support vector machines for pattern recognition,” *Data Mining Knowledge*, pp. 1-47, Dec. 1998.

Dongkyoo Shin received a B.S. in Computer Science from Seoul National University, Korea, in 1986, an M.S. in Computer Science from Illinois Institute of Technology, Chicago, Illinois, in 1992, and a Ph.D. in Computer Science from Texas A&M University, College Station, Texas, in 1997. He is currently a Professor in the Department of Computer Engineering at Sejong University in Korea. From 1986 to 1991, he worked at Korea Institute of Defense Analyses, where he developed database application software. From 1997 to 1998, he worked at Multimedia Research Institute of Hyundai Electronics Co., Korea as a Principal Researcher. His research interests include XML-based Middleware, Ubiquitous Computing and Digital-Rights Management for Multimedia.

Dongil Shin received a B.S. in Computer Science from Yonsei University, Seoul, Korea, in 1988. He received an M.S. in Computer Science from Washington State University, Pullman, Washington, U.S. A., in 1993, and a Ph.D. from University of North Texas, Denton Texas, U.S.A., in 1997. He was a senior researcher at System Engineering Research Institute, Daejun, Korea, in 1997. Since 1998, he has been with the Department of Computer Engineering at Sejong University in Korea where he is currently an Associate Professor. His research interests include Mobile Internet, Computer Supported Cooperative Work, Distributed Databases, Data Mining and Machine Learning.

Quoc Cuong Nguyen received a B.S in Computer Science from Thang Long University, Hanoi, Vietnam, in 2001, and an M.S in Computer Science from Andong National University, Korea, in 2005. He is currently pursuing a PhD degree at Sejong University. His research interests include Smart Homes, Image Processing, Data Mining and Machine Learning.

Seyoung Park received a B.S in Computer Engineering from Sejong University, Korea, in 2009. His research interests include Digital-Rights Management for Multimedia, Mobile Internet, Computer Networks and Ubiquitous Computing.