A Vision-Based Emergency Response System with a Paramedic Mobile Robot

Il-Woong JEONG†[a], Student Member, Jin CHO†[b], Kyusung CHO†[c], Yong-Ho SEO†[d], and Hyun Seung YANG†[e], Nonmembers

SUMMARY Detecting emergency situation is very important to a surveillance system for people like elderly live alone. A vision-based emergency response system with a paramedic mobile robot is presented in this paper. The proposed system is consisted of a vision-based emergency detection system and a mobile robot as a paramedic. A vision-based emergency detection system detects emergency by tracking people and detecting their actions from image sequences acquired by single surveillance camera. In order to recognize human actions, interest regions are segmented from the background using blob extraction method and tracked continuously using generic model. Then a MHI (Motion History Image) for a tracked person is constructed by silhouette information of region blobs and model actions. Emergency situation is finally detected by applying these information to neural network. When an emergency is detected, a mobile robot can help to diagnose the status of the person in the situation. To send the mobile robot to the proper position, we implement mobile robot navigation algorithm based on the distance between the person and a mobile robot. We validate our system by showing emergency detection rate and emergency response demonstration using the mobile robot.

key words: emergency response, mobile robot, human action recognition, object tracking

1. Introduction

As the world became industrialized, nuclear families became common. As a result of growing the number of elderly people who lives alone causes lots of social problems. Especially it is difficult to provide the proper medical services at right time to the living alone elderly people. To solve the problem mentioned above, we propose a vision-based emergency response system with a paramedic mobile robot. The emergency detection system detects emergency and the paramedic mobile robot helps to diagnose the condition of a person who is possibly in danger.

The conventional video surveillance system needs some human resources to monitor the captured images and it is very hard work for anyone to monitor the static scene all day. In general, there could be too many surveillance cameras to monitor under the limited number of security guards in a building, so a surveillance system should be intelligent to provide appropriate services to people who trust the system by recognizing the context of the scene from the cameras.

An intelligent video surveillance system consists of three parts; detecting moving objects, tracking multiple moving objects, and detecting abnormal situations using information on tracked objects. There still exist a lot of problems to solve to apply intelligent video surveillance system into the real world. Input images can be affected by illumination change. And it is very difficult to make a model of moving object, especially in case of human due to the variety of pose that human can make. Occlusions, multiple fragment belongs to one object also make the problem more intractable. There are a lot of studies on developing a vision-based emergency detection system that can detect suspicious situation by analyzing image sequences from surveillance cameras automatically. As mentioned in Aggarwal and Cai’s work, these researches can be used for providing appropriate services according to users’ intention from their actions [2].

In the ideal case, a blob of an image represents an object, but in the real cases multiple objects may exist in a single blob, or an object is consisted of several blobs. Bose et al. proposed an object tracking method considering with fragmentation problem [15]. They defined a generic object model and proposed an object tracking algorithm using inference graph.

There are some researches to make a model for an action. Bobick and Davis proposed Motion History Image (MHI) [5]. Blank et al. proposed Spatio-Temporal Volume Descriptor [9]. Yilmaz and Shah proposed Actions Sketch [10]. These studies make models to describe human motion by using whole body appearances such as shape, contour, texture and motion. These models can be used to recognize human actions. Finite State Machine (FSM) [1], [4], Bayesian Network [3] and Hidden Markov Model (HMM) [6] are used to recognize users’ actions. Ogata et al. proposed a method to recognize human action based on MHI and an eigenspace [13].

A mobile robot that can provide various services in home environment have been studied by many researchers. Pineau et al. developed a mobile assistant robot Pearl to assist nurses and elderly people in their daily activities in assisted living facilities [8]. Michaud et al. proposed a telepresence system for home care assistance using a mobile
robot to decrease load on health care system, reduce hospitalization period and improve quality of life for patients who need medical care [16].

Although a vision-based emergency detection system and robotic systems have been studied for a long time, the trial to combine the surveillance system and robotic system is still not common in the research area. In the Jin et al.’s work, environment helps the robot by giving position on the moving objects to track them successfully [12]. In the Shiom et al.’s work, environment provides museum visitors’ prerecorded profile information to the robot for the interaction [17].

An intelligent video surveillance system can monitor the wide area but it can’t provide details on the specific situation at the various view. And a mobile robot can provide a closer look at the various view with its mobility, but it can’t monitor the whole area to protect. By combining the two systems, the performance of the whole system increases due to synergy that compensates disadvantages between the two systems.

We propose an emergency response system that can detect an emergency situation using vision based human action recognition and cooperate with a mobile robot to deal with the emergency. By combining these two systems, it is possible to examine details on specific situation by the mobile robot while the surveillance system monitors the whole area of concern. And we suggest an application of our system to the emergency response system.

After presenting overall process of our work in Sect. 2, we explain emergency detection algorithm using vision based object’ tracking and human action recognition in Sect. 3, and we introduce our mobile robot as a paramedic in Sect. 4. In Sect. 5, we show some experimental results of our work and we conclude our work in Sect. 6.

2. System Overview

An emergency situation can be detected by recognizing human action using image analysis. For the beginning, a human can be detected from image sequences captured by a fixed surveillance camera. The candidate area that seems to includes human image can be obtained by background subtraction process. After finding a region which includes human in the image, the person in the region is tracked while the background model is updating. A model for a target action can be generated based on the tracked image sequence which includes a person performing the target action, and we selected 5 actions to recognize and we created models for those actions. Using the proposed system, people in the view of the camera are tracked and managed by analyzing their actions to detect an emergency. If an abnormal situation is occurred during tracking the human actions, a mobile robot can be used to diagnose the situation. With its mobility and sensors, the robot can transfer some information on the situation such as sound and images at the close distance. According to the information from robot, it could be possible to determine the necessity of a rescue party for the situation at the emergency control center. The whole workflow mentioned above is shown in Fig. 1.

3. Emergency Detection System

The proposed human action recognition system can track multiple objects and recognize five simple human actions - waking, running, sitting, standing, and falling. The human action recognizer consists of three parts; blob extraction, object tracking, and human action recognition. Among the actions mentioned above, sitting and falling are used to decide emergency situation.

3.1 Blob Extraction

Given an input frame, we separate foreground from background. It is very important to segment correct foreground pixels efficiently, because tracking multiple objects and recognizing human actions are dependent on it. We, therefore, extract foreground pixels through two-level foreground detection: pixel-level and region-level foreground detection.

In pixel-level, we execute a color-based foreground detection using the running average approach (RA) that is a kind of background subtraction techniques. The RA models a pixel value (intensity or RGB values) as a Gaussian distribution and continuously updates the Gaussian distribution for adapting to background changes. It is comparatively simple and has low computation cost. And it has only one background model, that makes it possible to use the difference image, which is got from the input image and the background image, for additional process. For robust foreground detection, we use Jacques Jr. et al.’s detection method assuming that foreground pixels tend to appear in sets of connected points and occupy roughly the same portion in space in adjacent frames [14]. Let \( f(x, y) \), \( A(x, y) \), and \( \sigma(x, y) \) be the value, mean and standard deviation of pixel \( (x, y) \) at frame \( t \), respectively. We, then, get foreground mask \( \text{Mask}_{\text{color}} \) as follow:
Iting the shadow and highlight detection algorithm proposed by Jacques Jr.

\[ D'(x, y) = |I'(x, y) - \lambda(x, y)| \times \frac{1}{16} \left[ \begin{array}{cc} 121 & 242 \\ 121 & \end{array} \right] \]

\[ ND'(x, y) = \frac{1}{2}(D'(x, y) + D'^{-1}(x, y)) \]

\[ \text{Mask}_{\text{color}}(x, y) = \begin{cases} 1 & \text{if } ND'(x, y) \times k \alpha(x, y) \\ 0 & \text{otherwise} \end{cases} \]

However, in most cases, \( \text{Mask}_{\text{color}} \) includes shadow or highlight pixels induced by moving objects as shown in Fig. 2 (b). To remove shadow and highlight from \( \text{Mask}_{\text{color}} \), we assume that the intensity of a shadow pixel is scale-down of the intensity of the corresponding pixel in the background model, and we get \( \text{Mask}_{\text{sh}} \) as follows by adopting the shadow and highlight detection algorithm proposed by Jacques Jr. et al. [14].

\[ \text{Mask}_{\text{sh}} = \begin{cases} 1 & \text{if } \text{Mask}_{\text{color}} = 1 \\ \text{and } \sigma_r\left(\frac{f(x, y)}{\delta(x, y)}\right) < L_{\text{std}} \\ \text{and } L_{\text{low}} < \left(\frac{f(x, y)}{\delta(x, y)}\right) < L_{\text{high}} \\ 0 & \text{otherwise} \end{cases} \]  \hspace{1cm} (1)

where \( R(x, y) \) denote a \( 3 \times 3 \) region centered at pixel \( (x, y) \). \( L_{\text{std}}, L_{\text{low}}, \) and \( L_{\text{high}} \) are constants and determined experimentally. After removing shadow and highlight region, we get the result of foreground extraction as shown in Fig. 2 (c). Fragments sometimes occur, because true foreground pixels are removed as shadow or highlight if the texture and intensity of moving objects and background are similar. So we consider additional edge information from \( D' \). We multiply \( D' \) by \( \alpha \) for enhancing edge and use Sobel mask for the calculation of the edge gradient \( G = \sqrt{G_x^2 + G_y^2} \) where \( G_x \) and \( G_y \) are the horizontal and vertical differences in \( \alpha D' \). Then, we calculate an edge mask \( \text{Mask}_{\text{edge,sh}} \) through hysteresis thresholding. Lastly, we get the color-edge mask

\[ \text{Mask}_{\text{color,edge}} = (\text{Mask}_{\text{color}} - \text{Mask}_{\text{sh}}) \cup (\text{Mask}_{\text{edge,sh}} - \text{Mask}_{\text{color}}) \]

As shown in Fig. 2 (d), it describes a moving object better than Fig. 2 (c).

**Algorithm 1** Removing false positive regions from \( \text{Mask}_{\text{color,edge}} \)

1: procedure REGION-LEVEL(R) = R is the set of foreground regions
2: for all \( r_i \) do
3: if \( r_i \) is too small or shadow&highlight or invalid then
4: \( \text{Mask}_{\text{color,edge}}(x, y) \in r_i \leftarrow 0 \)
5: end if
6: end for
7: end procedure

In region-level, foreground pixels are clustered into regions via connected component analysis and we remove false positive regions from \( \text{Mask}_{\text{color,edge}} \), as shown in algorithm 1. First, small regions that are less than predefined size are removed as noise. Next, we check if \( r_i \) is shadow or highlight induced by added edges, by using equation 1 with \( r_i \) instead of \( R \). Last, we remove invalid regions whose boundary pixels do not have high gradient values, based on Javed et al.’s approach [7]. We define valid regions as follow:

\[ r_i \text{ is valid if } \sum_{(x, y) \in \partial r_i} \text{Mask}_{\text{edge,sh}}(x, y) \geq p_B \]

where \( \text{Mask}_{\text{edge,sh}}(x, y) \) is an edge mask of \( r_i \), \( \partial r_i \) is a set of boundary pixels of \( r_i \), and \( p_B \) is the percent of boundary pixels with high gradient value.

We, finally, represent regions that pass above region test as a set of blobs \( B' = \{ b'_i | i \in N, 0 \leq i \leq n \} \) where a blob \( b'_i \) is \( t \)-th set of connected foreground pixels at frame \( t \) and \( n \) is the number of blobs.

### 3.2 Object Tracking

In the ideal case, a blob represents an object, but in the real, multiple objects may appear as a single blob (grouping), or an object may be broken into several blobs (fragments). To handle these problems, we propose a realtime multiple objects tracking algorithm based on Bose et al.’s framework handling grouping and fragments[15]. As shown in Fig. 3, given \( B' \) which obtained from the blob extraction process mentioned above, we obtain a set of objects \( O^t-k \) at frame \( t-k \), where \( k \) is the size of a frame queue. Firstly, we determine blob association events by associating \( B' \) with \( B'_{t-k} \) in

**Fig. 3** An overview of our realtime tracking framework.
two consecutive frames and update the blob inference graph according to blob association events, and we label each vertex as one of a fragment, object, or group. Lastly, we statistically localize objects into the vertex using blob association event records and object events induced from the blob inference graph.

3.2.1 Blob Tracking

At each frame, we determine blob tracks by associating current blob set $B^t$ extracted from the input frame with $\hat{B}^t$ predicted from previous blob set $B^{t-1}$ using the Kalman filter. We associate blobs by matching regions of $\hat{B}$ and $B$. Let $M$ be a binary $|B^{t-1}| \times |B^t|$ correspondence matrix whose entry is set as below:

$$M_{ij} = \begin{cases} 1 & \text{if } \frac{|\hat{b}_i \cap b_j|}{|\hat{b}_i|} > th_o \quad \text{or} \quad \frac{|\hat{b}_i \cap b_j|}{|b_j|} > th_o \\ 0 & \text{otherwise} \end{cases}$$

(2)

where $th_o$ is the threshold of overlap.

As shown in Fig. 4, we assume that blob association events are classified into five events as follows: continue, merge, split, appear, and disappear. After inferring a blob association, we update the velocity of blob centroid according to blob association events for later coherent motion test.

3.2.2 Blob Inference Graph Updating & Labeling

A blob inference graph $G$ is used to infer blob’s label and localize objects. A vertex is associated with more than one blob, and a directed edge represents spatial relation between two vertices. In the Bose et al.’s tracking algorithm[15], a new inference graph is created for every frame. On the other hand, only one inference graph is need for tracking during whole tracking in this work. An inference graph $G$ is updated according to the blob association events as below:

**Appear** ($b'_j \cap B^{t-1} = \emptyset$) : Add a new vertex $v(b'_j)$.

**Continue** ($b'_j = b^{t-1}_j$) : No change ($v(b^{t-1}_j) = v(b'_j)$).

**Merge** ($\cup_i b^{t-1}_i$ : Add a new vertex $v(b'_j)$ and directed edges whose tails are $v(b'_j)$ and whose heads are vertexes corresponding to $\{b^{t-1}_i | b^{t-1}_i \subset b'_j\}$.

**Split** ($\cup_i b^{t-1}_i$ : Add new vertexes corresponding to $\{b'_j | b'_j \subset b^{t-1}_i\}$ and directed edges whose tails are $v(b^{t-1}_i)$ and whose heads are vertexes corresponding to $\{b'_j | b'_j \subset b^{t-1}_i\}$.

After updating $G$, we need to refine $G$ for later labeling and localizing to satisfy two properties: if two vertexes are equivalent, their tracks should be associated with the same vertex; $v_j$ is the descendant of $v_i$ if $v_j$ is subset of $v_i$. For that reason, a vertex is represented by more than one Vertex Units Set (VUS), where a Vertex Unit (VU) is a leaf of $G$. VUS is also updated according to blob association events. We can compare two vertexes through VUSs.

We label each vertex as one of fragment (F), object (O), and group (G) using Bose et al.’s inference-graph labeling algorithm that stitches together blobs that belong to the same object. The loss of tracking information is, however, induced by the limit of the number of frames considered for tracking. To overcome this problem, we add a confidence flag to the vertex. If a vertex is reliable, its state is not initialized so that we can get more correct tracks.

3.2.3 Object Localizing

We get object tracks by placing objects in vertexes according to object events. The object events are classified into five events: object continue, object merge, object split, object appear/reappear, and object disappear, similar to the blob association event. Object events are estimated by the state change of a vertex’s label, as shown in Fig. 5.

A vertex whose label is fragment is unrelated to object events. If the vertex includes any false objects, we remove them from the vertex. A vertex is labeled as an object in three cases: an object appears or reappears (object appear/reappear), the previous object continues (object con-
3.3 Human Action Recognition

3.3.1 Action Modeling Using MHI

Unlike the posture recognition which uses single image frame, a sequence of images should be considered to recognize human action. Given a sequence of images, we adapt a representation of motion history image (MHI) for a model of human action. The MHI collapses an image sequence into a single image that captures spatial and temporal information about motion [5]. The MHI is known for its fast processing speed and ability to represent short-duration movement.

An MHI at frame $t$ is updated as

$$M^t_\delta(x,y) = \begin{cases} t/\delta & \text{if } \Psi(I'(x,y)) \neq 0 \\ M^{t-1}_\delta(x,y) & \text{otherwise} \end{cases}$$

where $\delta$ is the number of images used for the collapse, $I'(x,y)$ is the current blob mask image and $\Psi$ signals the presence of a blob at pixel $(x,y)$. The first four image of Fig. 6 show extracted silhouettes, and the image on the right-hand side is their corresponding MHI.

3.3.2 Action Classifying

The bounding box of a MHI is normalized to 32x32 gray image (1024 features). And a PCA is conducted to the normalized image to reduce feature dimension. We then, add the ratio of the width to the height of the bounding box that is used for discriminating two actions - walking and running - which have similar shapes in their MHIs.

To classify actions, a multi-layer perceptron (MLP), a sort of robust neural network is used. We define five action classes: walking, running, sitting, standing, and falling. Figure 7 shows examples of MHIs that are used to train an MLP. The MLP is trained by using 320 actions from four subjects.

3.4 Emergency Region Detection

It is necessary to calculate the location of the emergency site to send the mobile robot to the appropriate position. If we use a projective camera model without a radial distortion, there exists the relation between a point $X = (X_w, Y_w, Z_w)^T$ in a world coordinate and the corresponding point $x = (u, v, 1)^T$ in a image coordinate shown in Eq. (4).

$$x = PX = K[R \ t] \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = K[r_1 \ r_2 \ r_3 \ 1] \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = H \begin{bmatrix} X_w \\ Y_w \\ 1 \end{bmatrix} = H\tilde{X}$$ (4)

where $P$ is a $3 \times 4$ projection matrix, $K$ is a $3 \times 3$ matrix representing intrinsic parameter, $R$ is a $3 \times 3$ rotation matrix, $t$ is a translation 3-vector, and $r_i$ is a i-th column of $R$. We assume that the fallen person lies on a floor and it is chosen such that the XY-plane corresponds to the floor, so the world point $X$ representing the person’s position on the floor can be rewritten as $X = (X_w, Y_w, 0, 1)^T$. Now, Eq. (5) can be obtained from Eq. (4).

$$x = K[r_1 \ r_2 \ r_3 \ 1] \begin{bmatrix} X_w \\ Y_w \\ 0 \\ 1 \end{bmatrix} = H \begin{bmatrix} X_w \\ Y_w \\ 1 \end{bmatrix} = H\tilde{X}$$ (5)

In (3), the relation between $\tilde{X}$ omitting a Z coordinate of $X$ and its corresponding image point $x$ can be represented by a $3 \times 3$ homography matrix $H$. Because the camera is fixed in real-time, we can calculate $H$ as well as $K$, $R$, and $t$ through a calibration stage in advance. Therefore, if the fallen person’s position is located in an image, his/her position $\tilde{X}$ on a world floor can be obtained from Eq. (6).

$$\tilde{X} = H^{-1}x$$ (6)

4. Emergency Response

When an emergent situation is occurred, the first thing to do for a paramedic is checking the patient’s consciousness. Although images from a surveillance camera are useful to
detect the emergent situation, it is hard to recognize the consciousness of a person in those images captured from the long distance. Furthermore, a surveillance camera can see the emergency site only at the fixed view and it has blind spots.

Instead of the surveillance camera, we used a mobile robot as a member of emergency response team. When an emergency is detected by the emergency detection system, a mobile robot moves to the emergency site and it sends image streams and sound from the emergency site to the control center. Then an agent in the control center can determine what emergency response is needed for the person according to the information sent from the mobile robot such images and sound on the person.

We used a mobile robot platform iRobiQ as a paramedic in our system. iRobiQ has been developed under URC (Ubiquitous Robotic Companion) project funded by Korean government that links the robotics with IT technologies to provide services to human in daily lives. It is a commercial robot that provides fun and convenience services such as teaching English in kindergarten and monitoring home at the remote site through internet for security use. It has ultrasonic sensors, infrared (IR) sensors, and bumper sensors to detect and avoid obstacles on its path. It also has camera, microphone, and speaker to communicate with a person. Figure 8 shows the appearance and specification of iRobiQ.

Although iRobiQ has several range sensors such as ultrasonic sensors and IR sensors, these sensors are not powerful enough to use for building a map or localizing its position. Instead of using built-in sensors of the robot, we used IR based localization sensor system called StarGazer for indoor environment to calculate the position of the robot in our environment. It analyzes infrared ray image reflected from a passive landmarkers to calculate the position and heading angle of a robot. The passive markers don’t need any power resources since the StarGazer uses IR LEDs mounted around the camera lens, along with the IR pass filters placed over the camera lens to track the markers that are designed to reflect IR rays. As the result, the landmarkers can be installed without any limitations except the one simple requirement; the ceiling must be plane. Figure 9 shows the StarGazer and its landmarkers.

We used Microsoft Robotics Developer Studio (MRDS) to provide a way of easy access to the robot. It is possible to create applications that enable user monitor or control a robot at the remote site using a web browser. Monitoring surrounding environment and controlling the movement of the robot were implemented as a service on a web browser in this system.

5. Experiments

5.1 Experimental Setup

The experimental environment consists of a surveillance camera and landmarkers to calculate the position of our robot. A landmarker can cover the 2.4 m radius area, so we installed 9 landmarkers under the ceiling of our environment to cover the 56 m² area. The experimental environment is shown in Fig. 10. To build a map for a mobile robot, map building process was conducted by StarGazer sensor before the experiment. The StarGazer can build a map for the environment by using the distance between landmarkers and the height of the ceiling.

5.2 Emergency Detection

We built 5 action models - walking, running, sitting, and falling - for this system, and each model was learned by 40 trials demonstrated by 4 different person. The average recognition result was 90.9%, and the recognition rate of sitting and falling that are used to detect emergency was 98.6% as shown in Table 1.
Table 1 Human action recognition result.

| Result | Walk | Run | Sit | Stand | Fall | Rate(%) |
|--------|------|-----|-----|-------|------|---------|
| P1     | 10   | 9   | 10  | 10    | 10   | 98.0    |
| P2     | 7    | 10  | 10  | 8     | 10   | 90.0    |
| P3     | 4    | 10  | 10  | 10    | 10   | 88.0    |
| P4     | 9    | 10  | 10  | 3     | 10   | 84.0    |
| Rate (%) | 82.9 | 98.6 | 97.1 | 75.7  | 100.0| 90.9    |
| Emergency | No   | No  | Yes | No    | Yes  | 98.6    |

Table 2 Performance comparison.

| View Range | The Proposed System | The Conventional Surveillance System | The Conventional Robotic System |
|-------------|---------------------|--------------------------------------|---------------------------------|
| Monitoring Wide Area | possible | possible | impossible |
| Examining Specific Spot | possible | impossible | possible |
| Providing Various View | possible | possible with limitation | possible |
| Multiple Object Tracking | possible | possible | impossible |

5.3 Emergency Response

Figure 11 shows a test case of this system. When a pedestrian is falling down (Fig. 11 (a)), the emergency detection system detects emergency at the scene (Fig. 11 (b)), and a mobile robot is sent to the site to diagnose the person’s state. The trajectory of the person and the robot was plotted in Fig. 11 (c). Then a monitoring agent can determine the status of the person based on information from the robot using a service on web browser as shown in Fig. 11 (d).

5.4 Performance Comparison

It is difficult to measure the performance of this system quantitatively since the proposed systems is a kind of new system and there are no common measures in this field. So we present a comparison table, Table 2 to estimate the whole performance the proposed system.

6. Conclusion & Future Work

We proposed an emergency response system that consists of a vision-based emergency detection system and a paramedic mobile robot. A vision-based emergency detection system extracts human from the image, recognizes human action, and detects emergency. After detecting human in image sequences, detected human regions are tracked using appearance features such as the size and color of the region. Human action models are created using motion history image, and human actions are recognized using a neural network. Among the recognized actions, sitting and falling actions are considered as emergency situations.

When an emergency is occurred, a mobile robot is sent to the emergency site to diagnose the situation at the close distance. A mobile robot can find proper path to the emergency site using IR-based landmarkers installed under the ceiling. After arriving the target location, it transfers information on the situation such as sound and images. This system is a new emergency response system that combines video surveillance cameras and a mobile robot to provide various services to users. This system can be used for monitoring elderly people who live alone or detecting intrusion for security system. This system is a new trial to combine a video surveillance system and a mobile robot, and as the result, the system can provide an emergency response at the specific spot by using a mobile robot while the surveillance system monitors the whole area of concern.

The proposed system has some limitations at this moment in time. The walking direction of the person in the view of the camera should be the lateral view to recognize his or her action. And the robot can’t move into the area that does not installed landmarkers under the ceiling. Finally, the robot can find the approximate location of a patient and it needs some help from human monitoring agent to examine the patient’s status.

To overcome the limitations of our current work, we are planning to conduct more researches to improve our system. We are planning to install multiple cameras and develop some algorithms for detecting accurate emergency locations, removing blind spots, and removing dependency on the walking direction. And by adopting some heterogeneous sensors such as StarGazer, vision sensor, and laser range finder, we are anticipating to expand the working area of our robot. And we are also planning to improve intelligence of the robot to recognize emergency situation without any assistance from human monitor such as finding the adequate location by finding the patient’s face to examine the
his or her status. Furthermore, we are planning to improve our action recognizer to recognize not only simple actions but also complex actions to deal with emergency situation well by adopting some techniques such as Support Vector Machine and Hidden Markov Model to enhance the recognition rate.

Acknowledgement

This research was supported by the Ubiquitous Computing and Network (UCN) Project, Knowledge and Economy Frontier R&D Program of the Ministry of Knowledge Economy (MKE) in Korea and a result of subproject UCN 09C1-J2-11T, and a grant (07High Tech A01) from High tech Urban Development Program funded by Ministry of Land, Transportation and Maritime Affairs of Korean government.

References

[1] A.F. Bobick and A.D. Wilson, “A state-based approach to the representation and recognition of gesture,” IEEE Trans. Pattern Anal. Mach. Intell., vol.19, no.12, pp.1325–1337, Dec. 1997.
[2] J.K. Aggarwal and Q. Cai, “Human motion analysis: A review,” Comput. Vis. Image Understand. CVIU, vol.73, no.3, pp.428–440, 1999.
[3] A. Madabhushi and J.K. Aggarwal, “A Bayesian approach to human activity recognition,” Proc. Second IEEE Workshop on Visual Surveillance, pp.25–32, June 1999.
[4] P. Hong, M. Turk, and T.S. Huang, “Gesture modeling and recognition using finite state machines,” Proc. Fourth IEEE International Conference and Gesture Recognition, pp.410–415, Grenoble, France, 2000.
[5] A.F. Bobick and J. Davis, “The recognition of human movement using temporal templates,” IEEE Trans. Pattern Anal. Mach. Intell., vol.23, no.3, pp.257–267, 2001.
[6] N. Oliver, E. Horvitz, and A. Garg, “Layered representations for human activity recognition,” Proc. IEEE International Conference on Multimodal Interfaces, pp.3–8, Oct. 2002.
[7] O. Javed, K. Shafique, and M. Shah, “A hierarchical approach to robust background subtraction using color and gradient information,” Proc. Workshop on Motion and Video Computing, pp.22–27, Dec. 2002.
[8] J. Pineau, M. Montemerlo, M. Pollack, N. Roy, and S. Thrun, “Towards robotic assistants in nursing homes: Challenges and results,” Robotics and Autonomous Systems, vol.42, pp.271–281, 2003.
[9] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, “Actions as space-time shapes,” IEEE International Conference on Computer Vision (ICCV), pp.1395–1402, 2005.
[10] A. Yilmaz and M. Shah, “Actions sketch: A novel action representation,” IEEE Conf. Computer Vision and Pattern Recognition (CVPR), pp.984–989, 2005.
[11] J. Choi, Y. Cho, K. Cho, S. Bae, and H.S. Yang, “A view-based multiple objects tracking and human action recognition for interactive virtual environments,” Int. J. Virtual Reality, vol.7, no.3, pp.71–76, 2008.
[12] T. Jin, K. Morioka, and H. Hashimoto, “Human-following robot using the particle filter in ISpace with distributed vision sensors,” J. Artificial Life and Robotics, vol.10, no.2, pp.96–101, 2006.
[13] T. Ogata, J.K. Tan, and S. Ishikawa, “High-speed human motion recognition based on a motion history image and an eigenspace,” IEICE Trans. Inf. & Syst., vol.E89-D, no.1, pp.281–289, Jan. 2006.
[14] J. Jacques Jr., C. Jung, and S. Musse, “A background subtraction model adapted to illumination changes,” IEEE International Conference on Image Processing, pp.1817–1820, 2006.

[15] B. Bose, X. Wang, and E. Grimson, “Multi-class object tracking algorithm that handles fragmentation and grouping,” CVPR 2007, pp.1–8, June 2007.
[16] F. Michaud, P. Boissy, D. Labonte, H. Corriveau, A. Grant, M. Lauria, R. Cloutier, M.A. Roux, D. Iannuzzi, and M.P. Royer, “Telepresence robot for home care assistance,” AAAI Spring Symposium: Multidisciplinary Collaboration for Socially Assistive Robotics, pp.50–55, 2007.
[17] M. Shiomi, T. Kanda, H. Ishiguro, and N. Hagita, “Interactive humanoid robots for a science museum,” IEEE Intelligent Systems, vol.22, no.2, pp.25–32, 2007.

II-Woong Jeong received his B.S. degree from the Department of Information and Computer Science, Ajou University, in 2002, and his M.S. degree from the Department of Electrical Engineering and Computer Science, KAIST, in 2004. He is currently pursuing a Ph.D. at the Artificial Intelligence and Media Laboratory, KAIST. His research interests include computer vision, humanoid robots and human-robot interaction.

Jin Choi received his B.S. and M.S. degrees from the Department of Electrical Engineering and Computer Science, KAIST, in 2002 and 2004, respectively. He is currently pursuing a Ph.D. at the Artificial Intelligence and Media Laboratory, KAIST. His research interests include computer vision, machine learning and ubiquitous computing.

Kyusung Cho received his B.S. and M.S. degrees from the Department of Electrical Engineering and Computer Science, KAIST, in 2003 and 2005, respectively. He is currently pursuing a Ph.D. at the Artificial Intelligence and Media Laboratory, KAIST. His research interests include computer vision, machine learning and augmented reality.

Yong-Ho Seo received his B.S. and M.S. degrees from the Department of Electrical Engineering and Computer Science, KAIST, in 1999 and 2001, respectively. He also received a PhD degree at the Artificial Intelligence and Media Laboratory, KAIST, in 2007. He was an Intern Researcher at the Robotics Group, Microsoft Research, Redmond, WA. He was a consultant at Qualcomm CDMA Technologies, San Diego, CA. He is currently a professor of the Department of Intelligent Robot Engineering, Mokwon University. His research interests include humanoid robots, human-robot interaction, robot vision and wearable computing.
Hyun Seung Yang is currently a Dean of Research and a Professor of Dept. of Computer Science of KAIST. He is also a Founding Director of KMCC (KAIST Microsoft Research Collaboration Center) at KAIST. He got his B.S. degree from Dept. of Electronics Engineering, Seoul National University, Korea and both M.S. and Ph.D. Degrees from School of Electrical Engineering at Purdue University, West Lafayette Indiana USA. He worked with Dept. of Electrical and Computer Engineering, University of Iowa as an assistant professor. He then joined Dept. of Computer Science, KAIST in 1988 and now a full and tenured professor since 1997. He is a Director of VSMM (International Virtual Systems and Multimedia) Society and Steering Committee Members of both ICAT (Intl Conf. on Artificial Reality and Telexistence) and Intl Conf. on Edutainment. He was a Governing Board Member of IAPR (International Association of Pattern Recognition), Vice President of KIISE (Korea Institute of Information Science and Engineering), Vice President of KMMS (Korea Multimedia Society), Vice President of KRS (Korea Robotics Society) and a Board Member of KBS (Korea Brain Society). He is now a Fellow of the National Academy of Engineering of Korea (NAEK). He was General Chair of 21CTIS 2009 (21C TransMedia Innovation Symposium), ISAC2009 (Intl Symposium for Arts and Contents), Program Chairs of ICEC 2008 (Intl Conf. for Entertainment Computing) and ASIAGRAPH Tokyo 2008, General Chair of ISAT2008 (International Symposium for Arts and Technology), General Chair of ICAT 2004 (14th International Conf. on Artificial Reality and Telexistence), Program Chair of VSMM 2002 (8th International Conference on Virtual Systems and Multimedia). He served Program Chairs/Program Committee Members of many international conferences on Artificial Intelligence, Virtual Reality, Computer Vision, Robotics, Computer Graphics and Multimedia. His research activities include Artificial Intelligence/Vision, Human Robotics, VR/MR, Ubiquitous/Wearable Computing, Media Art and Technology.