An Effective Video Based System for Human Fall Detection

Hoang Le Uyen Thuc, Pham Van Tuan

Abstract— **Falling is known as a major health risk to cause injuries and even death among seniors. An early fall detection system is therefore extremely important to reduce the serious consequences of fall. In this study we aim to design and implement a practical three-stage video-based system for detecting fall events in elderly living alone at home. A common camera is used in the first stage to capture the video of a person doing his/her daily activities and then transfer to the following processing unit. In the second stage, from each frame of the video, we first separate the interested person from the background using adaptive background Gaussian Mixture Model; then the extracted object is converted into a five-dimensional feature vector using ellipse model; and finally we analyze those extracted features to recognize a fall using Hidden Markov Model trained by a challenging stimulated fall/non-fall database. The final stage is to immediately convey an SMS alert message to the assigned phone number to ask the timely medical assistance as soon as fall detected. Experimental result through the real-life video captured within one month in a crowded public area gives the average fall recognition rate of 97.47% with the delay time from 1 to 5 seconds, providing the high applicability of the system in real world.**

Index Terms — **Fall detection, Gaussian Mixture Model (GMM), ellipse model, Hidden Markov Model (HMM).**

I. Introduction

Fall is the most remarkable external cause of unintentional injury for the elderly. As reported by World Health Organization, approximately 28% - 35% of people aged of 65 and above fall every year and the rate increases to 32% - 42% for ones over 70 years of age. Fall is the major public health problem that often requires urgent medical attention. In addition, fall may result in a post-fall syndrome such as dependence, immobilization, depression, etc., which leads to a further restriction in daily activities. Fall accounts for 40% of all injury deaths. Consequently, the economic impact of falls is critical to family, community, and society [1].

Hence, early detection of fall in order to timely support the victims is the key strategy to mitigate fall-related injuries and their severe consequences. There have been a series of research projects all over the world on solutions for fall detection.

Manuscript received Aug, 2014.
Hoang Le Uyen Thuc*, Department of Electronic and Telecommunication Engineering, Danang University of Technology, Danang, Vietnam.*
Pham Van Tuan*, Department of Electronic and Telecommunication Engineering, Danang University of Technology, Danang, Vietnam.*

Fall detection approaches can be categorized into three main classes: wearable device based, ambient device based, and video based [2]. Wearable device based method relies on the sensor such as accelerometers [3], or fusion of accleleromerty and posture sensors [4] to detect the motion and location of the body of the person. Although this method is cheap, it causes the specific drawbacks such as the burden of time for sensor placement and the intrinsic intrusive nature, making it to be unfavorable choice for the elderly. Most ambient device based approaches use pressure sensors [5], based on the principle of sensing high pressure of the object due to its weight for detection and tracking. Pressure sensor is very cost effective and less intrusive; however, it is necessary to place pressure sensors to many things around the monitored object and thus, leading to generate false alarms in the case of detecting a fall. In recent years, video based methods have draw much attention in the field of fall detection, thanks to the easy installation, operation and maintenance of the video camera system, as well as the great achievements of the video analysis techniques. A numerous number of researches on video analysis have been proposed for detecting a fall. For example, Foroughi *et al.* conduct some methods to detect the fall [6 - 9] based on human shape variation. Extracted features, including combination of best-fit approximated ellipse around the human body, projection histograms of the segmented silhouette and temporal changes of head pose, are fed to a multi-class SVM [6] or an MLP ANN [7] for reliable classification of motions and determination of a fall event. Other features, also widely used for fall detection, are based on the combination of integrated time motion images and Eigen space technique [8, 9].

Most of video based fall detection systems consist of three stages: video acquisition, video analysis and notification communication [2], in which the video analysis stage plays a key role in the performance of the whole system.

In this paper, we design and implement a practical video based fall detection system applied for the elderly living alone at home, using the video analysis algorithm from our previous work [10], including adaptive background *Gaussian Mixture Model (GMM)* for human object segmentation, *ellipse model* for feature conversion and *Hidden Markov Model (HMM)* for fall detection.

In order to evaluate the precision, stability, and real-time applicability of the proposed system in real world, we deploy a testing system including cameras for capturing the video, computer for analyzing the video, and communication module for transmitting the SMS alert message from computer to pre-defined cell phone number in case of a fall detected. Experimental result through the real-life videos recorded in long term shows the designed system can rapidly and appropriately notify the caregiver of the fall accidents. Besides, the comparative study gives the favorable precision of our system in various falling scenarios.

The rest of the paper is organized as follows: in Section II, we generally describe the basic architecture of the designed system. The video analysis step is then presented in Section III with more details. Experimental results, evaluation and comparative study are performed in Section IV, followed by conclusion in Section V.

II. Overview of Designed Video Based Fall Detection System

Our designed system has the basic architecture as shown in Fig. 1. The system follows three stages of operations: video acquisition, video analysis and notification communication.

A. Video acquisition

In this stage, we capture the video of one person doing his/her daily activities such as walking, doing exercises, sitting on chair, etc. then transmit to the computer for video analysis.

First, our system emphasizes to monitor the elderly in the indoor environment with almost static background and/or corridor with acceptable lighting condition. The type of camera we are interested is IP camera, thanks to a lot of benefits such as cable not necessary, remote accessibility, image quality, and video management. In fact, two IP cameras D-Link DCS-942L [11] are installed in the crowded areas with the hope that we can record fall accidents and activities similar to falls. The main reason of choosing D-Link DCS-942L for video capture, beyond the above benefits, is it can easily switch to night mode with infrared capabilities to ensure that the senior is surveyed uninterruptedly regardless of lighting conditions.

The recorded video data is then transmitted to the computer for video analysis, over a wireless router. In order to ensure the real-time processing requirement of the system, the selected router need the sufficient transfer rate (up to 300 Mbps). Besides, *language C++* [12] and *library OpenCV* [13] are used to design the interface between camera and computer. In addition, the selected data transfer protocol for IP camera is *Real Time Streaming Protocol (RTSP)*, which is suitable in end-to-end, real-time transferring the stream of video.

After being read from OpenCV, video data need to be analyzed for fall detection. In our system we employ Matlab in achieving this goal. Therefore, we use open source software package MexopenCV, which provides Matlab mex functions that interface Application Programming Interfaces (APIs) of OpenCV [14].

B. Video analysis

In this stage, video stream is analyzed frame-by-frame by our designed software to detect the occurrence of a fall, mainly based on our previous algorithm [10]. There are three main phases in this stage: first, the human object is segmented from the background of image by using the adaptive GMM; second, segmented object is converted into

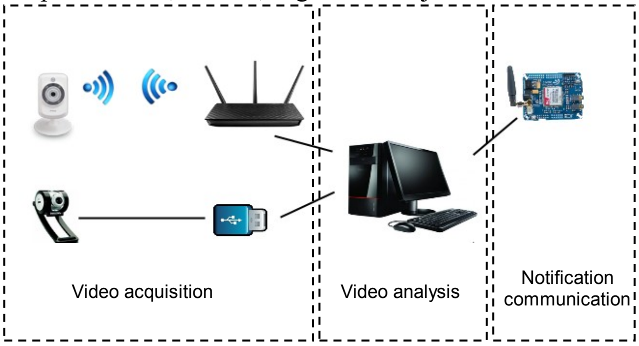


Figure 1. Basic architecture of the designed video based fall detection system.

five-dimensional feature based on the ellipse model; and finally, fall event is identified among other activities by HMM.

In our system, the above algorithms are programmed using available toolboxes of Matlab 2012a including Image Acquisition Toolbox, Computer Vision System Toolbox, HMM Toolbox, thanks to their great suitability in the sophisticated computing. The used computer requires the configuration as follows:

- Processor: Intel(R) Core(TM) Processor i3-2120 3.3GHz.
- Cache: 3MB L3.
- Ram: 2GB DDR3 1066Mhz 4 slot.
- Hard drive: 500GB SATA 7200RPM.
- Graphics: Intel HD Graphics.
- Data speed: 10/ 100/ 1000Mbps

The details of the three-step video analysis technique including human object segmentation, feature conversion, and fall recognition will be presented in Section III.

C. Notification communication

Whenever a fall is detected, the alarm sound and text notification are displayed in the monitoring screen. Besides, in our system, *notification communication* is designed to communicate with the caregiver by sending an SMS notification message to the pre-assigned cell phone number to ask for immediate assistance.

Module SIM900A is used in the system, providing benefits from small size and cost-effective solutions [15]. SIM900A delivers GSM/GPRS 900/1,800MHz performance for SMS, quite meet the cellular frequency band of mobile carriers in the country.

III. Video Analysis

The overview of the video analysis stage is shown in Fig. 2. The objective of this stage is to detect a fall incident of people in the incoming video stream. There are three main processing steps including human object segmentation, feature conversion, and fall recognition.

A. Human object segmentation

As the first process in video analysis module, human object

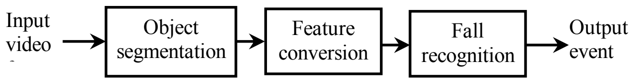


Figure 2. Functional diagram of video analysis module.

segmentation plays a key role in providing a proper silhouette for the following units to improve the fall detection accuracy under poor background conditions. In this study, we apply adaptive background GMM approach [16] to distinguish moving human from the rest of image which is called background.

At time t, the value of pixel at (x_0, y_0) in frame is X_t. The recent history of each pixel, $\{X_1, \ldots, X_t\}$, is constructed by a GMM with K components. The probability of observing the current pixel value can be calculated by,

$$p(X_t) = \sum_{i=1}^{K} \omega_{i,t} \eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \quad (1)$$

where $\omega_{i,t}$ is an estimated weight of the i^{th} Gaussian distribution at time t, η is the pdf of i^{th} Gaussian distribution, $\mu_{i,t}$ is the mean value of the i^{th} Gaussian at time t, and $\Sigma_{i,t}$ is the covariance matrix of the i^{th} Gaussian at time t.

Thus, a new pixel value will be represented by one of the major components of the mixture model and used to update the model. Every new pixel value is checked against the existing K Gaussian distributions, until a match (pixel value within 2.5 standard deviations of a distribution) is found [16].

When none Gaussian is matched, X_t is marked as a foreground pixel and the least probable component is replaced by a distribution with the current value as its mean, an initial high variance, and a low weight parameter.

As the parameters of the mixture model of each pixel change, in order to determine which of the Gaussians are most likely presented the background, the Gaussians are ordered by the value of ω/σ then the first BG distributions are chosen as the background model, where

$$BG = \arg\min_b \left(\sum_{k=1}^{b} \omega_k > T \right) \quad (2)$$

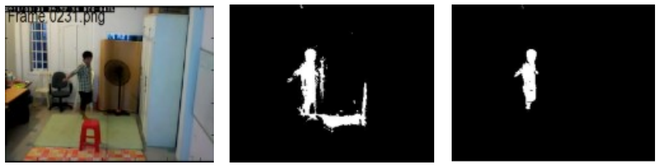
where T is the minimum fraction of the background model.

After extracting the human object, some morphological operations such as closing and opening are implemented to smooth the boundary and fill the small holes to create well-defined silhouette images. The results of human object segmentation process are shown in Fig. 3 [10].

B. Feature conversion

The silhouette image in each frame will be transformed into a reduced representation set of features. The chosen features should be informative and discriminative in describing an occurred fall event. In this study, we use five-dimensional features based on the ellipse model [10]. Five features are defined as follows:

1) *Current angle* – θ, is the angle between the major axis of ellipse and horizontal axis in the current frame. This feature is calculated by,



(a) (b) (c)

Figure 3. (a) Original image, (b) Segmented silhouette by background subtraction, (c) Segmented silhouette by adaptive GMM.

$$\theta = \frac{1}{2}.\arctan\left(\frac{2\sum i \sum j.x.y.I(i,j)}{\sum i \sum j.x^2.I(i,j) - \sum i \sum j.y^2.I(i,j)} \right) \quad (3)$$

where i, j: position of pixel, $x = i - O_x$ and $y = j - O_y$ (O_x, O_y: position of centroid, $I(i,j)$: value of pixel (i,j). Examples of current angle are shown in Fig. 4 [10].

2) *Coefficient of motion – Cmotion,* is to show the motion rate of the human in the current frame t. It depends on the *Motion History Image (MHI)* of 15 consecutive frames from the $(t\text{-}14)^{th}$ frame to the t^{th} frame. In an MHI, pixel value is a function of the recency of motion in a sequence at that pixel, i.e., more recently moving pixel is brighter than the past ones. Fig. 5 shows an example of MHI from walking sequence [10]. Cmotion is calculated by,

$$Cmotion = \left(\frac{\#Gray\ pixel}{\#Gray\ pixel\ +\ \#White\ pixel} \right) \quad (4)$$

When a fall occurs Cmotion is rather high [17].

3) *Deviation of the angle – Ctheta,* is standard deviation of 15 angles θs of 15 consecutive frames. Ctheta is usually higher when a fall happens [17].

4) *Eccentricity –e,* can be thought of as a measure of how much the surrounded ellipse deviates from being circular. The eccentricity at current frame is computed as,

$$e = \sqrt{1 - \frac{b^2}{a^2}} \quad (5)$$

where a and b are semi-major and semi-minor axes of ellipse, respectively. In case of falling human facing directly to camera, e will decrease rapidly [17].

5) *Deviation of the centroid – Ccentroid*, is defined as standard deviation of centroid coordinates from *15* successive frames. Ccentroid decreases rapidly when a fall occurs [17].

C. Fall recognition

In our system, fall recognition is performed using discrete-state HMM [18].

An HMM is completely characterized by a set $\lambda = \{A, B, \pi\}$, where A = transition matrix = $\{a_{ij}\}$, with a_{ij} being the transition probability from state q_i to q_j, $(i, j) \in [1: J]$; B = observation matrix = $\{b_j(k)\}$, with $b_j(k)$ being the probability of observed output (discrete) symbol v_k at state q_j, $k \in [1: K]$; $\pi = \{\pi_i\}$, with π_i being the initial state probability; J = number of states; K = number of observed symbols.

There are several types of HMMs depending on the transition matrix A [18]. For our application, 5-state left-to-right model as in Fig. 6 has been found to be most

Figure 4. Examples ellipse model and its current angle.

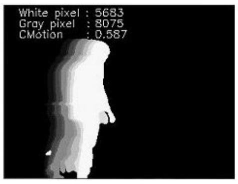


Figure 5. An example of MHI and coressponding Cmotion,

appropriate. In this model, initial state probability is $\pi = \{1, 0, 0, 0, 0\}$,

The use of HMM covers two phases which are training and testing. The data to feed HMM is from self-built stimulated fall/non-fall videos. The whole database is divided into two sets for training and testing.

1) The *training* process is carried out in the two following steps:

- *Vector quantization:*

In order to feed data to discrete-state HMM, a vector quantization process (e.g., K-means clustering) is necessary to map each training feature vector to a symbol among K symbols. On the other hand, quantization process helps to build a codebook with K codewords. Clearly, vector quantization causes quantization distortion and it is required to keep this distortion as small as possible. However, this implies a large value of K and that leads to slow processing speed. In this study, through experiments, we choose the value of K as *96* to balance the quantization distortion and processing speed. Thus, vector quantization converts a sequence of 5-dimensional feature vectors into a corresponding sequence of integer numbers (or codeword indexes) of the same-length.

- *Model building:*

In order to detect the fall action from the other non-fall actions, two HMMs (one for fall and one for non-fall action) are built to optimize the corresponding state transition matrices As and observation matrices Bs in order that the symbols generated by corresponding HMM can fit for the observed feature vectors. The observed feature vectors for fall/non-fall model are feature vectors which were quantized to become integer numbers (or codeword indexes) from training set of fall/non-fall videos. The matrices A and B are initially assigned with random values, then the Baum-Welch algorithm [18] is run until convergence condition is satisfied. If it runs over *10,000* times without convergence, it will be forced to be stopped.

2) The *testing* process is including two following steps:

- *Vector encoding:*

The Euclidean distance between each testing feature vector to each codeword in the codebook is calculated. Then the testing feature is encoded as the index of the nearest codeword.

- *Decoding and decision:*

First, a vector containing *15* integer numbers (i.e., *15* encoded feature vectors from *15* consecutive testing frames) is taken into decoding process. Then we calculate the probability of each model generating this sequence of feature vectors to measure the likelihood between the model and the testing frame sequence. After that, we compare these two probabilities to make the decision to label "*1*" (if fall model is more likelihood) or "*0*" (if non-fall model is more likelihood). Finally, we store the successive *20* labels in a *label- buffer*. If the total number of "*1*" in this buffer is greater than a predefined threshold – *Th* then the fall accident is detected. This procedure of label-buffer is for the purpose of decreasing

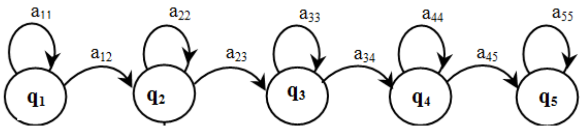


Figure 6. Five-state left-to-right HMM.

the rate of false detection.

IV. System Evaluation

The evaluation of the video analysis algorithm used in this system is performed through a self-built stimulated fall/non-fall database HBU-DUT [19]. In addition, the test of the entire developed system in real life conditions within a long term is also implemented to check the applicability of the system in real world.

A. Self-built database

Each of seven volunteers is instructed to perform a set of predefined actions including falling and non-falling. The volunteers are various genders, ages, weights, and heights. Each action is performed multiple times by multiple persons in various speeds. Some non-fall actions are extremely similar to fall actions such as bending, lying on floor, or creeping. Falling actions include different types caused by different reasons, such as fall by slip, fall by stumble, fall by roll, fall by faint, or fall by stroke. These help the database to be diversity and challenging. In total, we collect *134* stimulated videos including *65* fall and *69* non-fall videos. All videos are compressed in .avi format and recorded by camera D-Link DCS-942L [11] in a small room under good or quite good lighting conditions.

Based on the camera viewpoint we classify the action videos into three classes which are frontal view, side view, and arbitrary view. Thus, we have frontal-view fall, frontal-view walk, frontal-view creep, side-view fall, side-view walk, etc. Fig. 7 shows some examples of fall and non-fall actions in the database.

B. Experiments and evaluation

In this study, we divided the whole database into two sets: 31 videos including *15* fall videos and *16* non-fall videos for training and the rest for testing.

The testing set is grouped into three subsets which are named as Test 1, Test 2, and Test3 corresponding to three different testing scenarios as well-match (WM), medium-mismatched (MM) and highly-mismatched (HM) conditions, respectively. This grouping is designed to qualify

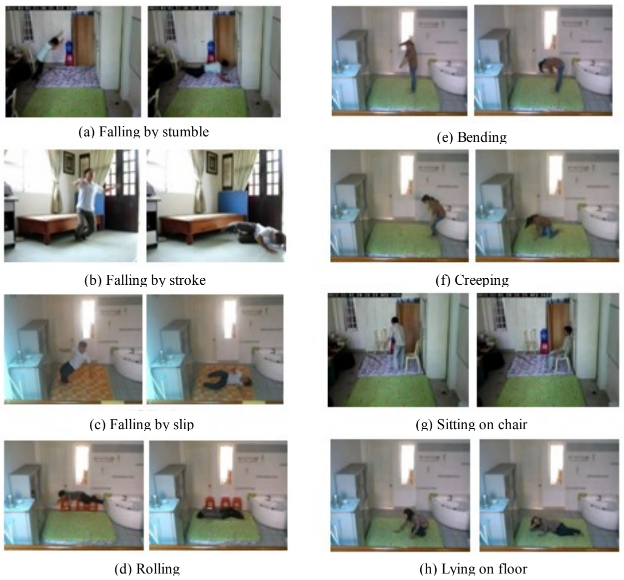


Figure 7. Image frames of fall and non-fall actions in the database.

robustness of the developed algorithms. WM set is very similar to training set. In each clip there is only one person doing actions with static background. MM set is similar to training set but the environment brightness and camera position are changed. HM set contains much changes in actions and recording conditions compared to training set such as: human is partly occluded, background is changed with added static objects, or there are more than one moving object in the sense.

In these experiments, we use three statistical measures which are Recall [%], Precision [%], and Accuracy [%] to evaluate the algorithm [20]. The performance of the employed fall detection algorithm obtained through three above experiments on three different testing sets is presented in Table I [10].

The overall performance of the algorithm is quite suitable to practice, achieving the maximal result (i.e, *100*%) for the least challenging scenario (i.e., WM test) and dramatically drop to *82.35*% for the most challenging scenario (i.e., HM test), but still provides the reasonably good performance.

To further evaluate the deployed fall detection algorithm, we also compare this algorithm with four previous algorithms using the same database under the similar experimental scenarios. Four algorithms used to compare are background subtraction– threshold matching (BGS-TM) [10], background subtraction – neural network (BGS-NN) [17], background subtraction – HMM (BGS-HMM) [10], and adaptive GMM - neural network (GMM-NN) [21]. The total accuracies are *69.90*%, *82.69*%, *81.35*%, and *86.38*%, respectively and are showed in Fig. 8 . Thus, with the total accuracy of *87.38*%, the fall detection algorithm used in this study is quite dominant.

TABLE I. STATISTICAL RESULTS OF THE FALL DETECTION ALGORITHM

	Test 1	Test 2	Test 3	Total
Recall [%]	100	86.67	84.00	88.64
Precision [%]	100	86.67	80.77	86.79
Accuracy [%]	100	86.21	82.35	87.38

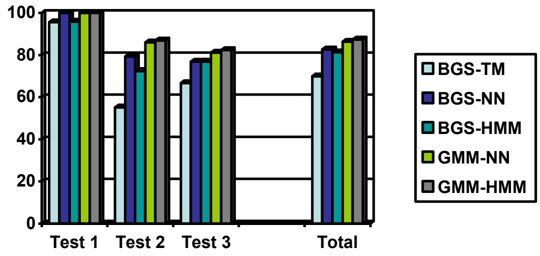


Figure 8. Fall detection rates in all testing circumstances for various video analysis methods.

C. Testing the developed system

The developed fall detection algorithm has been evaluated in many circumstances, providing promising results. In addition, to check the feasibility, stability, and real-time response of the whole system in real world, we build a testing system and perform experiments to continuously monitor and detect fall accidents within one month.

In fact, two IP cameras D-Link DCS-942L [11] are installed in the crowded areas - the lobby of building C and in front of Elect. & Telecom. Eng. Dep. office, Danang Uni. of Tech. (DUT), where a large number of students usually pass by with the hope that we can record fall accidents caused by students' carelessness, playing, or teasing, as well as the similar-fall actions.

Within one month, we collect *9* fall accidents with *4* falls by slip and *5* falls by playing. The number of non-fall actions are much larger. There are *665* non-fall actions, among which there are *202* one-object walks, *234* multiple-object walks, 153 down/upstairs-goes, 45 bends, 31 stairs-sits. Fig. 9 shows some fall and similar-fall images in real world.

The testing system monitors online and give alarms on the monitoring screen as well as sending SMS alert message to a pre-setup cell phone number. Fig. 10 is the designed monitoring screen in cases of detecting and non detecting a fall.

Figure 9. Practical fall/non-fall image frames

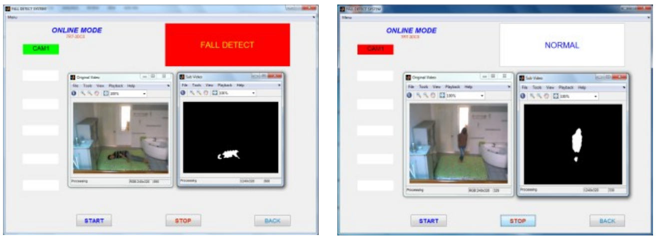


Figure 10. In-place alarm when detecting/non detecting a fall.

In the testing process, the developed system successfully detects 8 in 9 fall accidents, detects 16 non-fall actions as falls, accurately recognizes 649 non-fall actions. Therefore, the recognition rate achieves a remarkable result as below,

$$Acc = \frac{8+649}{674} = 97.47\%$$

Unfortunately, the delay time for giving an alarm is recorded as from 1 to 5 seconds. This delay can be tolerate; however it is bigger than we expected. The most reason is the time for transmitting video signals from camera to computer, for implementing video analysis algorithm by Matlab is rather long. We are thinking of using popular commercial processors instead of computer for the better processing speed as one of our future research directions to improve the developed system.

V. Conclusion

In this paper, we have designed and implemented a practical three-stage video-based fall detection system, by exploiting our previous work which is proven to be effective in stimulated fall/non-fall scenarios. Experiments in real life within a long term show that the performance of the developed system is quite good and robust even in poor conditions such as lighting changes, added background, varied camera viewpoint, occlusion, long-term sense. The ability for real-time processing can be acceptable. In summary, the system can provide the facility for rapid and precise alarm to caregivers at a distance in emergency situations via the commercial cellular phone network.

Acknowledgment

This study has been officially supported by Ministry Of Education and Training within the Research Project B2012 - 01 - 03. The authors would like to thank the TRT - 3DCS Teaching Research Team at DUT: Khue Tra, Phung T.Lai, Hieu V.Nguyen, Hoan V.Tran, Viet Q.Truong, Duy H. Le for their valuable contributions to the work.

References

[1] A. Kalache, D. Fu, and S. Yoshida, "WHO Global Report on Falls Prevention in Older Age," *World Health Organization Press*, Geneva, Switzerland, 2008.

[2] M. Mubashir, L. Shao, and L. Seed, "A Survey on Fall Detection: Principles and Approaches," *Neurocomputing*, pp. 144-152, 2013.

[3] M.A. Estudillo-Valderrama, L.M. Roa, J. Reina-Tosina, and D. Naranjo-Hernandez, "Design and implementation of a distributed fall detection system - personal server," *IEEE Trans. Inf. Technol. Biomed.*, vol. 13, pp. 874–881, 2009 .

[4] J.M. Kang, T. Yoo, and H.C. Kim, "A wrist-worn integrated health monitoring instrument with a tele-reporting device for telemedicine and telecare," *IEEE Trans. Instrum. Meas*., vol. 55, pp. 1655–1661, 2006.

[5] M. Alwan, P.J. Rajendran, S. Kell, D. Mack, S. Dalal, M. Wolfe, and R. Felder, "A Smart and Passive Floor-Vibration Based Fall Detector for Elderly," *IEEE International Conference on Information & Communication Technologies (ICITA)*, pp. 1003–1007, 2006.

[6] H. Foroughi, A. Rezvanian, and A. Paziraee, "Robust Fall Detection Using Human Shape and Multi-Class Support Vector Machine," *IEEE 6th Indian Conference on Computer Vision, Graphics & Image Processing (ICVGIP)*, pp. 413-420, 2008.

[7] H. Foroughi, B.S. Aski, and H. Pourreza, "Intelligent Video Surveillance for Monitoring Fall Detection of Elderly in Home Environments," *IEEE 11th International Conference on Computer and Information Technology (ICCIT)*, pp. 24-27, 2008.

[8] H. Foroughi, H.S. Yazdi, H. Pourreza, and M. Javidi, "An Eigenspace-based Approach for Human Fall Detection Using Integrated Time Motion Image and Multi-class Support Vector Machine," *IEEE 4th International Conference on Intelligent Computer Communication and Processing (ICCP)*, pp. 83-90, 2008.

[9] H. Foroughi, A. Naseri, A. Saberi, and H.S. Yazdi, "An Eigenspace-based Approach for Human Fall Detection Using Integrated Time Motion Image and Neural Network," *IEEE 9th International Conference on Signal Processing (ICSP)*, pp. 1499-1503, 2008.

[10] K. Tra and T. V. Pham, "Human Fall Detection based on Adaptive Background Mixture Model and HMM," *the 2013 International conference on Advanced Technologies for Communications (ATC)*, pp. 217-222, 2013.

[11] http://mydlink.dlink.com/products/DCS-942L

[12] W. Press, S. Teykolsky, W. Vetterling, and B. Flannery, "Numerical recipes in C++: the art of scientific computing," *Pearson Education*, 1992.

[13] http://opencv.org/

[14] http://vision.is.tohoku.ac.jp/~kyamagu/software/mexopencv/

[15] http://vntech24h.com/sim900-module.html

[16] C. Stauffer and W.E.L. Grimson, "Adaptive background mixture models for real-time tracking," *IEEE Conference of Computer Vision and Pattern Recognition*, vol. 2, pp. 246-252, 1999.

[17] Y.T. Ngo, H.V. Nguyen, T.V. Pham, "Study On Fall Detection Based On Intelligent Video Analysis", *The 2012 International Conference on Advanced Technologies for Communications (ATC)*, pp. 114-117, 2012.

[18] L.R. Rabiner : "A tutorial on hidden Markov models and selected applications in speech recognition", *Proc. IEEE*, vol. 77, no. 2, pp. 257-286, 1989.

[19] H.V. Tran, P.K. Lai, K. Tra, V.Q. Truong, D.H. Le, "DUT-HBU database", *Technical Report*, Danang University of Technology, 2012.

[20] T. Fawcett, "ROC Graphs: Notes and Practical Considerations for Researchers," 2004.

[21] P.K. Lai and T.V. Pham, "Gaussian Mixture Model Based Object Segmentation for Fall Detection," *2nd International Workshop on Industrial IT Convergence (WIIC)*, pp. 4-11, 2014.

Hoang Le Uyen Thuc received the B.E. degree on Electronics from Danang University of Technology (DUT), Vietnam, in 1994; the M.E. degree on Electronics-Communications from Hanoi University of Technology, Vietnam, in 1997. She is currently working toward the Ph.D. degree under the joint program between Electronic and Telecommunication Engineering (ETE) department, DUT, Vietnam and Information Processing Lab, Electrical Engineering department, University of Washington, U.S.A. She is now a lecturer of ETE department, DUT. Her research interests include signal processing, pattern recognition, and human behavior understanding.

Pham Van Tuan was born in Hoi An ancient town, Vietnam. He received Engineering and MSc. degrees in Electronics & Telecommunications and Dr. techn. degree in Electrical Engineering from Danang University of Technology (1999), Hanoi University of Science and Technology (2001) and Graz University of Technology (2007). Tuan was granted with a post-doctoral scholarship at the SPSC Lab of TUGraz from 2007 to 2008 and promoted to Associate Professor at ETE Department, DUT since 2012. He has been a lecturer at DUT since 1999, Vice-Chair of Electronic & Telecommunication Engineering Dept., DUT since 2010 and Vice-Director, Center of Excellence, DUT since 2011. He has published reference book and more than 50 conference, journal papers. His research interests are as follows: Detection, estimation, information theory; Speech/ Image/ Video processing and applications; Machine learning for signal processing; Real-time implementation and optimization of DSP algorithms. Assoc.Prof.Dr. Pham is IEEE member, IEEE SSCS Chapter member, member of Radio Electronics Association of Vietnam (REV). He functions in Editorial Board, REV Journal on Electronics and Communications, Vietnam. He joins TPC of many international conferences as IEEE-RIVF 2009, IEEE-ICCE 2010, IEEE-ATC 2011, GIIS 2011, UK-VN ASSW 2012, UK-VN ICBSBE 2012, IEEE-ATC 2014, REV-ECIT2014.

