

Novel and Robust Vision- and System-on-chip-based Sensor for Fall Detection

Kuo-Liang Chung,^{1*} Li-Ting Liu,¹ and Chi-Huang Liao²

¹Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, No. 43, Section 4, Keelung Rd., Taipei 10672, Taiwan, R.O.C.

²System Online Co., Ltd., 10F, No. 263, Section 3, Zhong-Siao East Rd., Taipei 10654, Taiwan, R.O.C.

(Received May 20, 2019; accepted July 22, 2019)

Keywords: accuracy, fall detection, foreground construction, foreground detection, system on chip (SoC), vision computing

In this paper, we propose a novel and robust vision- and system-on-chip (SoC)-based system as a sensor to effectively detect falls of the elderly. The proposed method consists of five steps: initial light stability confirmation, gradient-difference-based foreground detection, dilation- and multiframe-based foreground construction, false fall detection problem solving, and fall detection determination with a general-purpose input/output-based fall warning transmission. Real test videos have shown that our comprehensive experiments justify the low power, low hardware cost, and high detection accuracy merits of the proposed method when compared with related fall detection methods.

1. Introduction

According to Ref. 1, the percentage of people aged 65 or older in the world will be 16% in 2050. The main reason is that the number of elderly people is increasing owing to the improvement in health care provided by hospitals and medical advancements. In Taiwan, the percentage of elderly people aged 65 and above at the end of 2017 was 13.9%, and the percentage is expected to reach 20% in 2026.⁽²⁾ The statistical data indicated that, on average, 28–35% of the elderly fall at least once a year, and these falls often lead to serious injuries, especially if the elderly patient falls at home.⁽³⁾ Once a fall event occurs for an elderly person, if a fall detection signal can be sent out in time to call for first aid, the person can be rescued. The previous fall detection methods can be classified into four categories, namely, the wearable-device-based category, the ambiance-sensor-based category, the RGB-D [red-green-blue (RGB) color camera plus depth sensor]-based category, and the camera-based category. In the following brief survey of related works, we also point out their weaknesses.

*Corresponding author: e-mail: klchung01@gmail.com
<https://doi.org/10.18494/SAM.2019.2438>

1.1 Related work

In the wearable-device-based category, the user often carries a wearable device or a smartphone. Pierleoni *et al.*⁽⁴⁾ proposed a fall detection method in which the hardware covers a slim battery, a wireless receiver, a microprocessor, and a sensor used to gather information from a triaxial accelerometer, a gyroscope, and a magnetometer. Then, the collected data are analyzed by the proposed fall detection method. In Kau and Chen's fall detection method,⁽⁵⁾ the smart-phone used includes an e-compass and a triaxial accelerometer. From the gathered information, the discrete wavelet transform and support vector machine techniques are used to detect a fall event. Cheffena⁽⁶⁾ proposed a smartphone-audio-feature-based fall detection method. The main constraint of the above-mentioned wearable-device-based fall detection methods is that it is inconvenient for the elderly to carry a wearable device all day in order to detect falls.

In the ambiance-sensor-based category, Su *et al.*⁽⁷⁾ set up a Doppler sensor under the ceiling and proposed a coarse-to-fine-based discrete wavelet transform approach to detect falls. Shiba *et al.*⁽⁸⁾ proposed a hidden Markov model approach to measure the frequency distribution of vertical velocity trajectories in the foreground for determining the abrupt change occurring in a fall event. Erol *et al.*⁽⁹⁾ proposed a fall detection method using two Doppler sensors and the fusion-based approach to achieve better performance. The main constraint in the above Doppler-sensor-based approach is that the feasible distance is quite small from the sensor to the foreground.

In the RGB-D-camera-based category, the hardware mainly includes a Microsoft Kinect system and a personal computer (PC). Abobakr *et al.*⁽¹⁰⁾ proposed a skeleton-free fall detection method using the random decision forest model constructed from training depth maps. Finally, the fall event is determined by the support vector machine technique. Bian *et al.*⁽¹¹⁾ employed an enhanced randomized decision tree in fall detection such that human joint information can be extracted. Furthermore, joint trajectory information is provided to detect the fall event. Kong *et al.*⁽¹²⁾ proposed a multistep fall detection method. For fall detection, one merit of the RGB-D-camera-based approach is its independence from light changes, even in a dark room. However, the RGB-D camera, as well as the bundled PC, is expensive and takes up space.

In the color-camera-based category, the hardware used mainly includes a color digital camera and a PC. Liao *et al.*⁽¹³⁾ combined a Bayesian belief network model, the motion activity measure, and human silhouette shape variations to solve the fall detection problem. Lin *et al.*⁽¹⁴⁾ first applied the Gaussian mixture model and the ellipse approach to approximate the foreground. Furthermore, using the angular orientation and motion acceleration information, a fall event is detected. Nguyen *et al.*⁽¹⁵⁾ approximated the foreground using an ellipse and the smallest rectangle. According to the angle of the ellipse and the ratio of the height to the width of the rectangle, a rule-based classification scheme is proposed to detect a fall event. Abdelhedi *et al.*'s method⁽¹⁶⁾ was first applied to the Type-2 Fuzzy Gaussian mixture model to extract the foreground, and then the human fall behavior is analyzed by fuzzy logic to detect a fall event. Although the computational time is short, the camera and bundled PC are still not cheap for consumers.

Currently, on the basis of a low-powered and low-cost system-on-chip (SoC) framework, Chung *et al.*⁽¹⁷⁾ first applied the Gaussian mixture model to estimate the background model and then generated the foreground model by subtracting the estimated background from the current image frame. Furthermore, according to the variation of the centroid, a fall detection procedure is carried out. Experimental data indicated that there is room for improvement in its accuracy and execution time, especially in a complicated environment. Besides the method in Ref. 17, the three methods in Refs. 14–16 are also included in the comparative methods.

1.2 Motivation

The main motivation of this paper is twofold: (1) to develop an effective vision- and SoC-based fall detection method with low power, low hardware cost, and high detection accuracy to overcome the limitations existing in the above-mentioned four fall detection categories, and (2) to design a more robust foreground construction process to improve our preliminary result,⁽¹⁷⁾ and thereby improving the accuracy of the fall detection method.

1.3 Contributions

In this paper, we propose a new, effective vision- and SoC-based fall detection method that consists of five steps: initial light stability confirmation, gradient-difference-based foreground detection, dilation- and multiframe-based foreground construction, false fall detection problem solving, and fall detection determination associated with a general-purpose input/output-based fall warning transmission. In particular, the proposed gradient-difference-based foreground detection step, which is robust against light changes, and the dilation- and multiframe-based foreground construction step can robustly construct the foreground model. Because the extracted foreground is more complete, not only can the fall detection accuracy be increased, but the false fall detection problem caused by a squat posture can also be resolved. Real test videos have shown that our comprehensive experiments have justified the low power, low hardware cost, and high detection accuracy of the proposed fall detection method when compared with related methods.^(14–17)

The rest of this paper is organized as follows. In Sect. 2, the SoC-based fall detection system setting in a home room is presented. In Sect. 3, the proposed fall detection method is presented. In Sect. 4, the experimental results are reported. In Sect. 5, some conclusions are presented.

2. Vision- and SoC-based Fall Detection System Setting in a Room

The room environment simulated in the experiment is depicted in Fig. 1(a). As depicted in Fig. 1(b), the proposed vision- and SoC-based fall detection system contains two kinds of hardware, the SoC ‘Hisilicon-Hi3516CV300’ with a size of $37 \times 37 \text{ mm}^2$ and the bundled camera ‘M12-4IR(3MP)-C’. They are combined together and arranged at location A, as depicted in Fig. 1(c). With our fall detection method in the embedding system in SoC as a fall detection sensor, the power consumption of the proposed fall detection sensor is only 0.5 W.

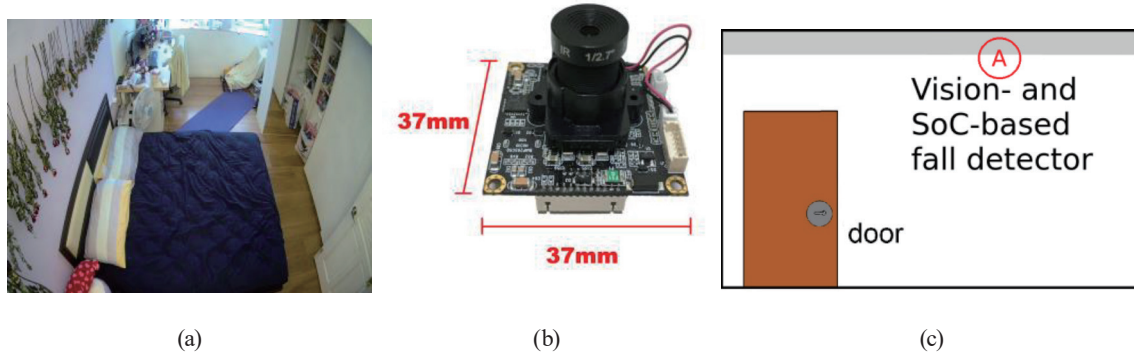


Fig. 1. (Color online) Our hardware setting in a home room. (a) Environment of the home room. (b) SoC 'Hisilicon-Hi3516CV300'. (c) Side view of the home room from the door.

Note that the three comparative methods^(14–16) were implemented in the PC-based system in which the CPU is Intel i7-8700@3.2GHz with RAM 24GB, and the power consumption is 70 W, indicating the low power merit of our vision- and SoC-based fall detection system. On the basis of a general-purpose input/output transmitter, our method sends out an alert signal to call for help when a fall event is detected by the proposed fall detection sensor.

The functionality of the proposed system is as follows. The CPU in SoC is ARM926@800MHz associated with 32 KB I-Cache, 32 KB D-Cache, 256 MB RAM, and 128 MB ROM. The operating system is Hisilicon Linux 32-bit and the HiLinux kernel is developed on the basis of the standard Linux kernel V3.18.y. The program development environment is the C standard library glibc-2.20. The power supply is via electrical wiring.

3. Proposed Fall Detection Method

Because the Gaussian mixture model for constructing the foreground model⁽¹⁷⁾ is time-consuming and the constructed foreground model is sometimes incomplete, we propose a new gradient-difference-based foreground detection approach and a dilation- and multiframe-based approach to construct a more complete foreground model. As shown in Fig. 2, the entire proposed fall detection method consists of the following five steps.

3.1 Step 1: Initial light stability confirmation

Let I_t and I_{t-1} denote the two images at time instances t and $t - 1$, respectively. To test whether the light condition has been stabilized before starting the formal fall detection task, we first compute the absolute difference between the average gray value of I_t and that of I_{t-1} , and then we compute the absolute average difference that can be calculated efficiently by the following histogram computation-based method:

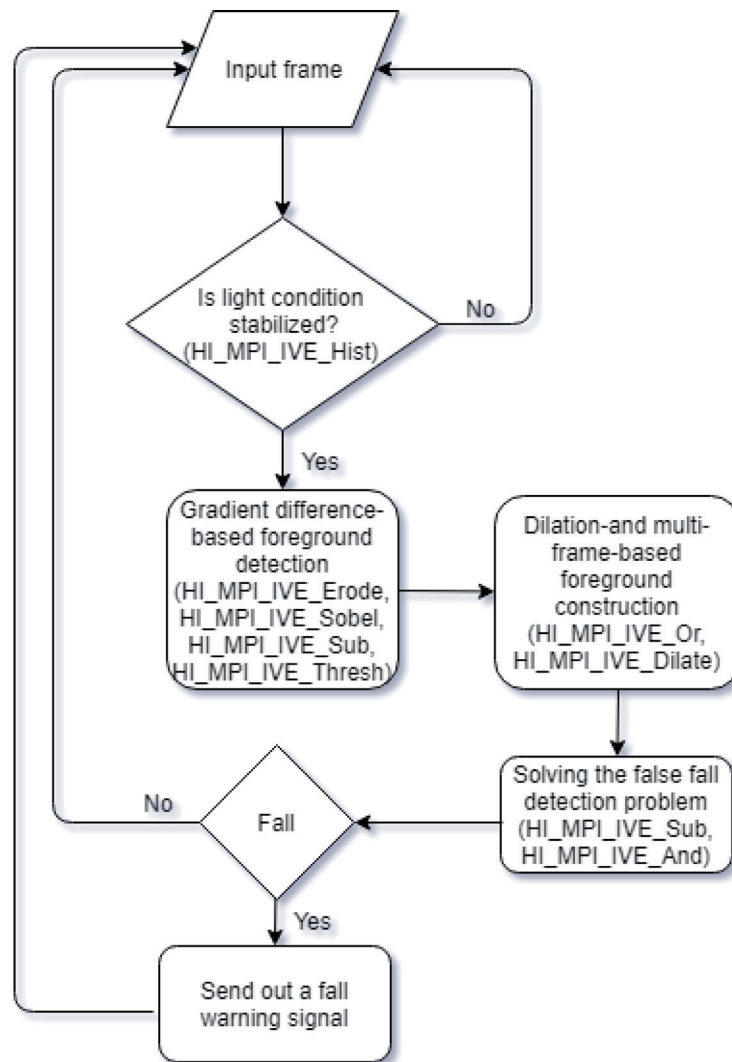


Fig. 2. Flow of the proposed vision- and SoC-based fall detection method.

$$D_{avg} = \left| \frac{\sum_{i=0}^{255} (i \times H_t(i))}{320 \times 240} - \frac{\sum_{j=0}^{255} (j \times H_{t-1}(j))}{320 \times 240} \right|, \quad (1)$$

where H_t and H_{t-1} denote the two histograms of I_t and I_{t-1} , respectively, in which each image has 320×240 pixels. The main reason why we adopt the histogram-based approach to calculate D_{avg} in Eq. (1) is that the histogram of each image can be quickly calculated by calling the building function ‘HI_MPI_IVE_Hist’ in SoC (Table 1).

Furthermore, we examine whether D_{avg} is less than the specific threshold, and empirically the threshold is set to 100. If not, it means that the light condition has not been stabilized and we go to Step 1; otherwise, we go to Step 2.

Table 1
Building functions in SoC.

| Subroutine name | Functionality | Purpose |
|-------------------|-----------------------|--|
| HI_MPI_IVE_And | Logical AND operation | Obtain the boundary change map |
| HI_MPI_IVE_Dilate | Dilate operation | Obtain a more complete foreground model |
| HI_MPI_IVE_Erode | Erosion operation | Remove noise in the image |
| HI_MPI_IVE_Thresh | Image binarization | Remove the pixel with weak motion information in the gradient difference map |
| HI_MPI_IVE_Hist | Histogram calculation | Calculate the gray value distribution of the image |
| HI_MPI_IVE_Or | Logical OR operation | Fuse five consecutive foreground models |
| HI_MPI_IVE_Sobel | Sobel edge detector | Obtain the gradient map of the image |
| HI_MPI_IVE_Sub | Subtraction operation | Calculate the gradient difference map |

3.2 Step 2: Gradient-difference-based foreground detection

To remove noise, we perform the erosion operation function ‘HI_MPI_IVE_Erode’ on the current image frame. Next, we perform the Sobel edge detection function ‘HI_MPI_IVE_Sobel’ on the current image frame I_t to obtain the gradient map G_t , and then we apply the building function ‘HI_MPI_IVE_Sub’ to calculate the gradient difference map between G_t and G_{t-1} . The calculated gradient difference map can be used to extract more foreground motion information against light changes. Furthermore, we apply the building function ‘HI_MPI_IVE_Thresh’ to remove the pixel with weak motion information in the gradient difference map, obtaining the coarse foreground F_t in I_t .

3.3 Step 3: Dilation- and multiframe-based foreground construction

To obtain a fine foreground model, each time, the coarse foreground model in Step 2 at the time instance t is fused by performing the logical OR operation ‘HI_MPI_IVE_Or’ on the five consecutive foreground models at the time instances $t, t - 1, \dots$, and $t - 4$, and the resultant foreground model is given by

$$F_t = F_t \cup F_{t-1} \cup F_{t-2} \cup F_{t-3} \cup F_{t-4}. \quad (2)$$

Because F_t still has several holes, the building function ‘HI_MPI_IVE_Dilate’ is performed on F_t to dilate each connected component in F_t , obtaining a more complete foreground model F_t' . For example, given an image frame in Fig. 3(a), after performing Step 2 and solving Eq. (2) for Fig. 3(a), the constructed foreground model F_t' is depicted by those white pixels in Fig. 3(b). Furthermore, we calculate the centroid of F_t' as

$$(x_c, y_c) = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right), \quad (3)$$

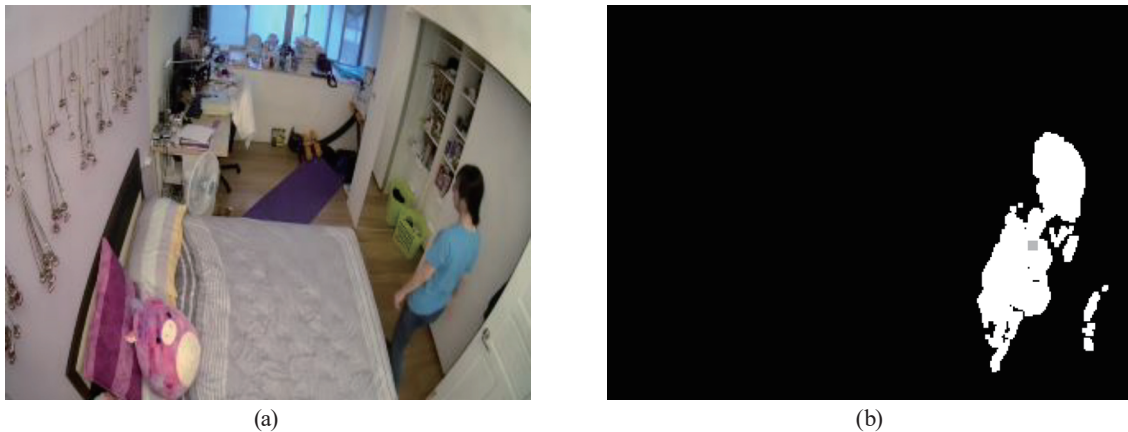


Fig. 3. (Color online) Image frame example and its final constructed foreground. (a) Input image frame. (b) Final foreground extracted by the proposed approach.

in which m_{10} , m_{01} , and m_{00} denote the traditional first-order, first-order, and zero-order moments of F_t' , respectively. The centroid calculated using Eq. (3) is used to identify the center of the constructed foreground model F_t' .⁽¹⁸⁾

3.4 Step 4: Solving the false fall detection problem

When one person is opening the door at time instance t to leave the current room, the light coming from the neighboring room and pouring into the floor of the current room may make the centroid of the foreground appear to return from the door area to the floor area of the room, which usually leads to a false fall detection. In what follows, we propose an effective approach to solving this false fall detection problem.

Considering the three consecutive gradient maps G_{t-2} , G_{t-1} , and G_t , we perform the difference operation 'HI_MPI_IVE_Sub' on two consecutive gradient maps to obtain the two boundary maps $B_{t-1} = G_{t-1} - G_{t-2}$ and $B_t = G_t - G_{t-1}$. Furthermore, we perform the logical AND operation 'HI_MPI_IVE_And' on the two consecutive boundary maps B_{t-1} and B_t to obtain the boundary change map $B_t' = B_{t-1} \cap B_t$, which is robust to light variations.

Let the number of white pixels in B_t' be $|B_t'|$. When $|B_t'|$ is small, those white pixels in B_t' may be treated as noise. To construct a stable boundary change map, we perform $B_t' = B_t' \cup B_{t-1}' \cup B_{t-2}' \cup B_{t-3}' \cup B_{t-4}'$. Then, we calculate the centroid of B_t' , (x_c', y_c') . Suppose the door region has been marked in advance. If the centroid of B_t' remains in the door region, we set the status of the door region to be $S_{door} = 1$; otherwise, $S_{door} = 0$. Because the constructed boundary change map is robust to light variations, the false fall detection can be avoided with the help of the door region status. Note that the status of the door region determined in this step could be considered together with the centroid calculated in Step 3 to determine whether a true fall event has occurred, as described in the next step.

3.5 Step 5: Fall detection determination

If the centroid obtained in Step 3 is on the floor area and the door region status obtained in Step 4 is $S_{door} = 0$, then the constructed foreground is in a candidate fall event on the floor region; otherwise, the constructed foreground is considered to be in a nonfall event. When the constructed foreground is in a candidate fall event on the floor region, if the centroid of B_t' determined by solving Eq. (3) remains in a fixed location for more than t seconds, e.g., $t = 3$, a true fall detection alarm signal is formally sent out via the general-purpose input/output-based transmitter to call for first aid help; otherwise, go to Step 1.

As shown in Figs. 4(a) and 4(b) in which each real image frame contains a different fall event, Figs. 4(c) and 4(d) demonstrate the corresponding detected fall events, respectively. Owing to the robustness of the proposed foreground detection process (see Step 2), the process for constructing a more complete foreground (see Step 3), and the boundary change map for providing the door region status, the detected fall event by our vision- and SoC-based method is much more reliable than our early work using the Gaussian mixture model approach.⁽¹⁷⁾

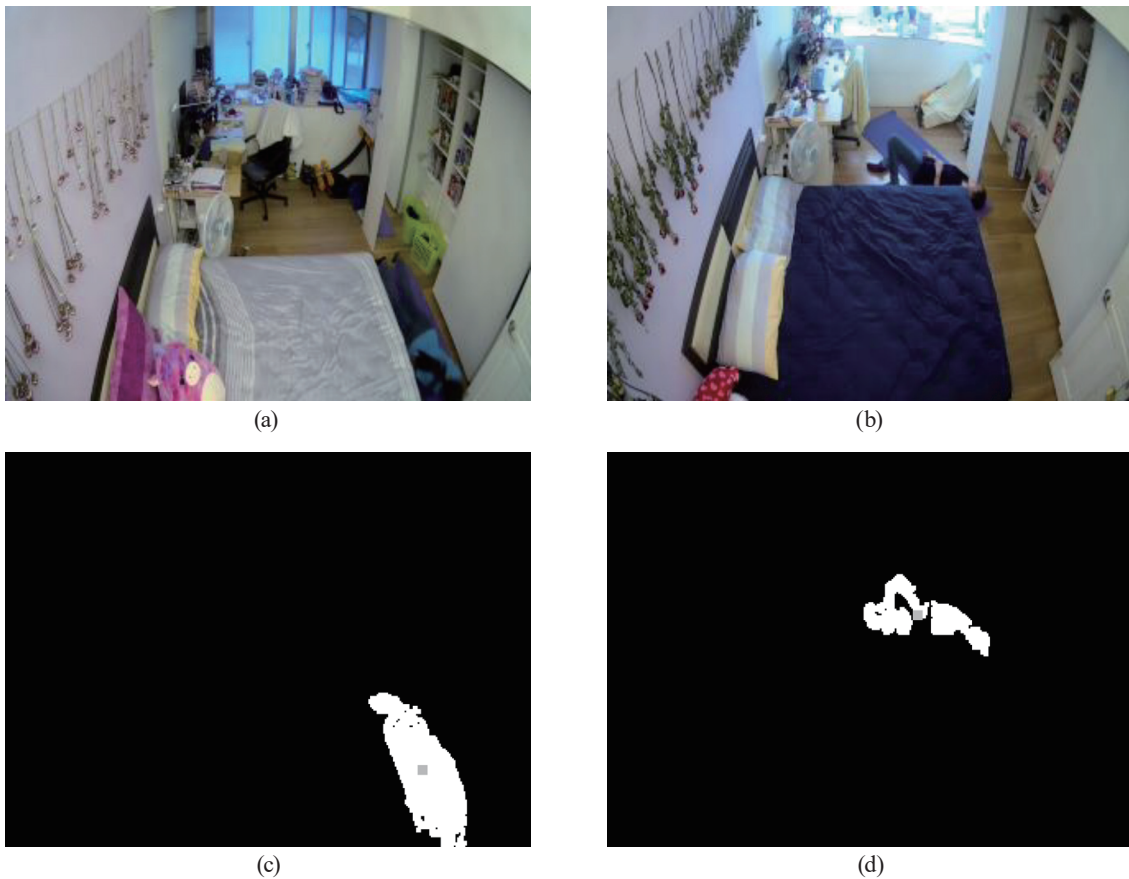


Fig. 4. (Color online) Two examples of fall detected by the proposed vision- and SoC-based fall detection method. (a) First sample image containing one fall event. (b) Second sample image containing one fall event. (c) Detected fall event for (a). (d) Detected fall event for (b).

4. Experimental Results

In this section, thorough experimentation has been carried out to demonstrate that in terms of accuracy, precision, and recall, the proposed vision- and SoC-based fall detection method outperforms our previous work⁽¹⁷⁾ and the three-image-processing- and PC-based fall detection methods.^(14–16)

4.1 Figures and tables

Before evaluating the fall detection accuracy performance of our method and the other related methods, four parameters, true positive (TP), true negative (TN), false positive (FP), and false negative (FN), are defined as

TP: number of fall events correctly detected by other related methods,

TN: number of nonfall events correctly detected by other related methods,

FP: number of nonfall events incorrectly detected as fall events by other related methods, and

FN: number of fall events incorrectly detected as nonfall events by other related methods.

The performance evaluation metrics, namely, accuracy, precision, and recall, are defined as

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}, \quad (4)$$

$$Precision = \frac{TP}{TP + FP}, \quad (5)$$

$$Recall = \frac{TP}{TP + FN}, \quad (6)$$

where ‘Accuracy’ denotes the total successful fall and nonfall detection rate of all events, ‘Precision’ denotes the successful fall detection rate of all fall events, and ‘Recall’ denotes the successful rate of all fall events.

4.2 Test dataset

As shown in Table 2, the types of simulated fall and nonfall events are nine and five, respectively, and the test simulation videos are captured in a real room. In the simulation, there are three persons, two females and one male, who simulate the above-mentioned fourteen fall and nonfall events. In Table 2, four types of events for the bed and chair are included to increase the practical application of the proposed vision- and SoC-based fall detection method. In total, we have 43 videos with 51 different segments covering the fourteen types of fall and nonfall events. The readers can access these test videos from the website.⁽¹⁹⁾

Table 2
Types of simulated fall and nonfall events.

| | |
|---------------|-------------------------|
| Fall event | Forward |
| | Backward |
| | Left front |
| | Left back |
| | Right front |
| | Right back |
| | Fall from the bed |
| | Fall from the chair |
| | Fall when standing up |
| | Walking |
| Nonfall event | Lying or sitting on bed |
| | Sitting on chair |
| | Squatting |
| | Fake fall |

Table 3
Accuracy and frame rate comparison among related methods.

| | Ac (%) | Pr (%) | Re (%) | Average fps |
|---|--------|--------|--------|-------------|
| Nguyen <i>et al.</i> ⁽¹⁵⁾ | 49.0 | 55.2 | 55.2 | 361.016 |
| Abdelhedi <i>et al.</i> ⁽¹⁶⁾ | 78.4 | 78.1 | 86.2 | 378.298 |
| Lin <i>et al.</i> ⁽¹⁴⁾ | 72.5 | 72.2 | 86.7 | 152.213 |
| Chung <i>et al.</i> ⁽¹⁷⁾ | 76.5 | 90.5 | 65.5 | 5.097 |
| Proposed | 98.0 | 100.0 | 96.6 | 15.981 |

4.3 Performance evaluation and comparison

From the above-mentioned test videos, in terms of the three performance metrics in Eqs. (4)–(6), Table 3 shows the performance comparison among our vision- and SoC-based fall detection method and the four other related methods. In Table 3, Ac, Pr, and Re are the abbreviations for accuracy, precision, and recall, respectively. From Table 3, we observe that the accuracy of our method is 98.0%, the detection precision is 100%, and the recall is 96.6%. The average number of image frames processed per second (fps) by our method is less than that using the PC-based approaches;^(14–16) however, the average number of fps obtained by our method is much higher than that in our previous work.⁽¹⁷⁾ When compared with the related fall detection methods,^(14–17) our method has the highest fall detection accuracy.

5. Conclusion

We have presented our vision- and SoC-based fall detection method for the elderly in a home environment. Our method consists of five steps, and in particular, the proposed Steps 2 and 3 can construct a more complete and reliable foreground; the false fall event detection problem can be resolved with the help of the door region status in Step 4. The data from thoroughly conducted experiments have demonstrated the low hardware cost, low power, and high detection accuracy of our fall detection method relative to the previous vision- and SoC-based method⁽¹⁷⁾ and the three previous methods^(14–16) in the color camera-based category. Our future work is to apply the model compression technique⁽²⁰⁾ to prune the number of parameters used in the recent deep-learning-based fall detection method⁽²¹⁾ such that the pruned deep-learning-based detection method can be run in a SoC-based environment.

Acknowledgments

We are grateful to Ms. C. Harrington for the proofreading that helped improve the manuscript.

References

- 1 United Nations: World Population Prospects: The 2008 Revision-Comprehensive Tables (UN, 2010).
- 2 National Development Council, Taiwan, R.O.C (2017). <https://www.ndc.gov.tw/en/cp.aspx?n=2E5DCB04C64512CC>
- 3 Good health adds life to years: Global brief for World Health Day 2012. https://www.who.int/ageing/publications/whd2012_global_brief/en/
- 4 P. Pierleoni, A. Belli, L. Palma, M. Pellegrini, L. Pernini, and S. Valenti: IEEE Sens. J. **15** (2015) 4544. <https://doi.org/10.1109/JSEN.2015.2423562>
- 5 L. J. Kau and C. S. Chen: IEEE J. Biomed. Health. Inf. **19** (2015) 44. <https://doi.org/10.1109/JBHI.2014.2328593>
- 6 M. Cheffena: IEEE J. Biomed. Health. Inf. **20** (2016) 1073. <https://doi.org/10.1109/JBHI.2015.2425932>
- 7 B. Y. Su, K. C. Ho, M. J. Rantz, and M. Skubic: IEEE Trans. Biomed. Eng. **62** (2015) 865. <https://doi.org/10.1109/TBME.2014.2367038>
- 8 K. Shiba, T. Kaburagi, and Y. Kurihara: IEEE Sens. J. **17** (2017) 22. 7561. <https://doi.org/10.1109/JSEN.2017.2760911>
- 9 B. Erol, M. G. Amin, and B. Boashash: 2017 IEEE Radar Conf. (IEEE, 2017) 0819. <https://doi.org/10.1109/RADAR.2017.7944316>
- 10 A. Abobakr, M. Hossny, and S. Nahavandi: IEEE Syst. J. **12** (2018) 3. 2994. <https://doi.org/10.1109/JSYST.2017.2780260>
- 11 Z. P. Bian, J. Hou, L. P. Chau, and N. M. Thalmann: IEEE J. Biomed. Health. Inf. **19** (2015) 2. 430. <https://doi.org/10.1109/JBHI.2014.2319372>
- 12 X. Kong, L. Meng, and H. Tomiyama: 2017 Int. Conf. Advanced Mechatronic Systems (ICAMechS, 2017) 269. <https://doi.org/10.1109/ICAMechS.2017.8316483>
- 13 Y. T. Liao, C. L. Huang, and S. C. Hsu: Pattern Recognit. **45** (2012) 24. <https://doi.org/10.1016/j.patcog.2011.04.017>
- 14 C. Y. Lin, S. M. Wang, J. W. Hong, L. W. Kang, and C. L. Huang: 2016 IEEE Second Int. Conf. Multimedia Big Data (IEEE, 2016) 237. <https://doi.org/10.1109/BigMM.2016.22>
- 15 V. D. Nguyen, M. T. Le, A. D. Do, H. H. Duong, T. D. Thai, and D. H. Tran: 2014 9th IEEE Conf. Industrial Electronics and Applications (IEEE, 2014) 994. <https://doi.org/10.1109/ICIEA.2014.6931308>
- 16 S. Abdelhedi, A. Wali, and A. M. Alimi: Afro-European Conf. Industrial Advancement (AECIA) **427** (2016) 227. https://link.springer.com/chapter/10.1007/978-3-319-29504-6_23
- 17 K. L. Chung, C. H. Liao, L. T. Liu, D. H. Liu, Y. S. Chan, and J. S. Cheng: Int. Workshop on Pattern Recognition (IWPR, 2019) (in press).
- 18 R. C. Gonzalez and R. E. Woods: Digital Image Processing (Pearson, NJ, USA, 2017) 4th ed.
- 19 <ftp://140.118.175.164/fall/videos>
- 20 S. Han, H. Mao, and W. J. Dally: Int. Conf. Learning Representations (2016). <https://arxiv.org/abs/1510.00149>
- 21 M. Yu, L. Gong, and S. Kollias: 19th ACM Int. Conf. Multimodal. Interaction (ACM, 2017) 416. <https://dl.acm.org/citation.cfm?id=3136802>

About the Authors

Kuo-Liang Chung received his B.S., M.S., and Ph.D. degrees in computer science and information engineering from National Taiwan University, Taipei, Taiwan, R.O.C. He has been a chair professor of the Department of Computer Science and Information Engineering at National Taiwan University of Science and Technology, Taipei, Taiwan since 2009. He was the recipient of the Distinguished Research Award (2004–2007; 2018–2021) and Distinguished Research Project Award (2009–2012) from the Ministry of Science and Technology of Taiwan, R.O.C. He has been the associate editor of the Journal of Visual Communication and Image Representation since 2011. His research interests include deep learning, image processing, and video compression. (klchung01@gmail.com)

Li-Ting Liu received her B.S. degree in Computer Science and Engineering from Tatung University, Taipei, Taiwan, R.O.C., in 2017. She is currently working towards her M.S. degree in Computer Science and Information Engineering at the National Taiwan University of Science and Technology, Taipei, Taiwan, R.O.C. Her research interest is in image processing. (m10515103@mail.ntust.edu.tw)

Chi-Huang Liao received his M.S. degree in Information Management from Fu-Jen Catholic University Institute, Taipei, Taiwan, R.O.C., in 2002. He is currently the general manager of System Online Information Co., Ltd., Taipei, Taiwan, R.O.C. His research interests include image processing and video-based health care applications. (james@sysonline.com.tw)