computation

MDPI

Article

# Model for the Detection of Falls with the Use of Artificial Intelligence as an Assistant for the Care of the Elderly

**William Villegas-Ch.** [1,2,*], **Santiago Barahona-Espinosa** [1], **Walter Gaibor-Naranjo** [3] **and Aracely Mera-Navarrete** [4]

[1] Escuela de Ingeniería en Tecnologías de la Información, FICA, Universidad de Las Américas, Quito 170125, Ecuador
[2] Facultad de Tecnologías de Información, Universidad Latina de Costa Rica, San José 70201, Costa Rica
[3] Carrera de Ciencias de la Computación, Universidad Politécnica Salesiana, Quito 170105, Ecuador
[4] Departamento de Sistemas, Universidad Internacional del Ecuador, Quito 170411, Ecuador
* Correspondence: william.villegas@udla.edu.ec; Tel.: +593-98-136-4068

**Abstract:** Currently, telemedicine has gained more strength and its use allows establishing areas that acceptably guarantee patient care, either at the level of control or event monitors. One of the systems that adapt to the objectives of telemedicine are fall detection systems, for which artificial vision or artificial intelligence algorithms are used. This work proposes the design and development of a fall detection model with the use of artificial intelligence, the model can classify various positions of people and identify when there is a fall. A Kinect 2.0 camera is used for monitoring, this device can sense an area and guarantees the quality of the images. The measurement of position values allows to generate the skeletonization of the person and the classification of the different types of movements and the activation of alarms allow us to consider this model as an ideal and reliable assistant for the integrity of the elderly. This approach analyzes images in real time and the results showed that our proposed position-based approach detects human falls reaching 80% accuracy with a simple architecture compared to other state-of-the-art methods.

**Keywords:** artificial intelligence; artificial vision; fall detection; machine learning

check for updates

**Citation:** Villegas-Ch., W.; Barahona-Espinosa, S.; Gaibor-Naranjo, W.; Mera-Navarrete, A. Model for the Detection of Falls with the Use of Artificial Intelligence as an Assistant for the Care of the Elderly. *Computation* **2022**, *10*, 195. https://doi.org/10.3390/computation10110195

Academic Editors: Joshua Thomas, Leo Mrsic, Román Rodríguez Aguilar, Pandian Vasant and José Antonio Marmolejo Saucedo

Received: 17 September 2022
Accepted: 25 October 2022
Published: 2 November 2022



## 1. Introduction

People over 65 years of age are considered older adults, at this stage, it is important to cover certain needs that are specific to age. One of these needs is physical, several studies establish that falls are the main causes of mortality in older adults, accidental falls as the age of a person advances, they have a higher percentage of complications and their occurrence. These works mention that approximately 30% of those over 65 years of age and 50% of those over 80 years of age fall at least once a year [1]. Falls can cause fractures, external injuries, internal bleeding, and loss of functionality in organs or limbs, reducing people's independence. When there is a fall, rapid and effective intervention is necessary to avoid further damage to the affected person [2].

The care that older adult needs are one of the main causes that affect the timely detection of events such as falls. Since, according to reviewed studies, the care of the elderly in the vast majority depends on family members and very few sectors entrust the care of the elderly to specialized places [3]. Of these cases, the older adults who are at greater risk are those whose care oversees relatives or who even spend alone in a home. Generally, homes are not suitable areas for the care of older adults, spaces and objects can be the cause of falls or blows that cause people to lose their balance.

This work aims to create a fall detection model that helps people who do not have continuous care or sectors that need to adequately monitor the well-being of older adults [4]. The model has been designed to monitor and identify a fall within a normal area of a home. To do so, artificial intelligence (AI) and machine learning tools, artificial vision, and devices such as cameras are used. The creation of the algorithm that is part of the detection model identifies postures and movement patterns. When detecting a fall, the model makes use of libraries that allow an alert to be sent to a previously registered user or group of users, who receive a message via Telegram with information and an image of the event that occurred [5]. For the evaluation of the model, work has been done with a group of people who acted out various scenarios, where positions such as sitting, bending over, falling, or simply the act of walking were simulated. The results obtained to guarantee the use of the system and its proper functioning in the classification and identification of falls.

## 2. Materials and Methods

For the design of the method, several concepts are used that are detailed below and guarantee the integration of technologies in event detection environments. The concepts and techniques considered are part of the analysis of similar works and the needs of the population that participates in this study.

### *2.1. Identification of the Environment and Its Needs*

This work aims at the older adult population, according to global data, every second two people turn 60 years old. In Ecuador, the country where this work is carried out, an elderly person is 65 years of age or older. According to the national institute of statistics and censuses of Ecuador [6,7], in the country by 2020 the number of older adults stood at 1.3 million people, out of a total of 17.5 million Ecuadorians. By 2030 it is estimated that the elderly will become 30% of the national population. With the increase in the older adult population, government institutions form national protection systems that seek to create strategies that allow efficient care for this population in matters of prevention, care, accompaniment, and reparation of their rights. In addition to the search to guarantee the rights of older adults, due to their biological and social conditions, they are considered vulnerable [8]. The factors that cause this vulnerability are caused living in situations of risk determined by personal, economic, family, community, and access to health resources.

In works reviewed, establish that the risk factors that make older adults more vulnerable are living in economically, socially, and geographically remote depressed regions. Another factor that is considered is not having the accompaniment of a person for their care, in addition, several studies present results indicating that 15.8% of older adults have the low functional capacity, 53.6% are overweight or obese, the 27.9% are in the very sick category, among others. These data allow us to establish that one of the most significant problems is the physical problems in this population and how they are exacerbated by not having adequate support [9]. One consequence of physical problems is that older adults are more likely to fall, the most common causes being muscle weakness, changes in their walking and balance, heart disease, decreased vision, and the use of canes and walkers. improper way.

### *2.2. State of the Matter*

Fall detection is a serious problem for the elderly population, regardless of the environment and supervision that they have. Of the works reviewed, there is a common factor and that is that they take ICTs as the ideal tools to create alert systems that focus on monitoring events and issuing alarms that can safeguard the lives of the vulnerable population [10,11]. Under this analysis it is possible to categorize the works into two groups, the first group focuses on the use of devices such as sensors that are part of an embedded system that the person must carry with them so that they can monitor events. The second group of works is part of the integration with integrated artificial intelligence techniques that use cameras or other devices to monitor a given environment.

In the analysis carried out, it is identified that the first group of works reviewed makes use of technologies such as embedded and takes as reference the concepts of the Internet of Things [12]. In general, it can be mentioned that the architectures that are based on these works require a strong electronic component, in addition to requiring people to transport the embedded system so that it can perform monitoring. In certain works, it is even mentioned that their systems are related to wearable technology, however, since they are prototypes, this usually presents discomfort to the person or generates false positives. These problems are generally due to the size of the prototypes, since, to detect a fall, these devices integrate sensors such as accelerometers, microcontrollers, communication modules, interface modules, etc. Other works analyze the work carried out by companies that design medical devices, this is the case of sense4care [13]. This company has developed a fall detector based on a triaxial accelerometer and algorithm approved by the Center for Technological Studies for Dependence and Autonomous Life. This device is presented as a small sensor that is connected to the person's waist and connects to mobile via Bluetooth [14]. This device called Angel4 registers a fall, makes an emergency call, and sends text messages indicating the situation and GPS location of the person. Another device with a similar function is the one proposed by [15], This device is governed by an expert system that analyzes the data conditions and determines if the event has been generated If the event is positive, the expert system warns of the fall using a telephone connection.

The second group of related works takes as its fundamental basis the creation of systems that do not depend on portability and costs. To meet this objective, the systems integrate AI techniques with machine learning that are responsible for monitoring an environment in real-time [16]. The methodology used in these works focuses on the prediction of falls according to the positioning of a person's body [17]. An example is the work developed by [18] where an application was developed that detects if there is a fall using the position of the person's skeleton [19–21]. The detection process uses a dataset that contains the position of the body, obtained from OpenPose. This is a library included in C++ and was part of the first system capable of real-time detection of body key points. Its operation depends on a multistage convolutional network with two branches, this allows a neural network to identify the position of the body and determine whether the person suffers a fall [22]. The works reviewed suggest that the use of AI algorithms allows falls to be detected with low computational cost and costs are reduced by having the adaptability to commonly used cameras [23]. However, other works analyze the hardware necessary for the implementation of this type of system and recommend the use of cameras with better technical characteristics to obtain high depth values [24]. An example is the Kinect cameras, their high quality simplifies the task of bottom extraction and ground detection [25]. In addition to the works that use AI as the main component for fall detection, there is a subgroup of work that uses artificial vision. These fall detection systems use computer vision and window frame image processing techniques to classify activities. With technological advancement, in monitoring devices such as cameras, it is possible to obtain high-quality in-depth images. This information is also analyzed for human tracking, user monitoring, and recognition [26,27], and monitoring and recognition of subjects' daily activities in indoor environments [28]. Classical vision-based fall detection and classification strategies consist of five phases [29], data acquisition from video sequences, feature extraction from images, feature selection, and learning and inference. In addition, other techniques base their operation on machine learning techniques used in the literature, such as SVM [30] or random forest [31]. Aziz et al. [32] proposed a fall detection system based on human silhouette shape variation in vision monitoring and SVM to identify postures. They then used HMM to classify the data into falling and non-falling events. In the work [33] followed the person's silhouette along with the video sequences. Shape deformation is quantified from the profiles based on shape analysis methods. Finally, dips in daily activities are detected using Gaussian Mixture Models (GMM).

To this subgroup of works are aligned those that use deep learning techniques. Through this technique, instead of teaching computers what an object is like to detect it, it is given enough information about the object so that the machine can solve it by itself [34]. For example, if you want to detect a mouse by artificial vision, instead of teaching it what the mouse looks like, you give it enough photos of mice and the machine will decide on its own if it is a mouse or not [35]. Deep learning is currently used in object detection, voice recognition, driving autonomous cars, or in the search for cancer cells. The Deep Learning model is based on the use of artificial neural networks [36].

### 2.3. Method

For the design of the prototype, an architecture based on the use of artificial vision techniques is established. In Figure 1, the components of the entire system are presented, it is necessary to emphasize that the proposed system focuses on the monitoring of a certain environment and can be coupled to any environment of a home. The first component is the area to be monitored. This work, being focused on the monitoring of older adults who present physical vulnerabilities and who do not have the assistance of a person, establishes its application within a home, considered a controlled environment [37]. Therefore, the monitoring areas can be a bathroom, a living room, a room, etc. This depends on the place where the older adult is usually found. The experimentation area has not been modified to apply the model in a totally real environment.

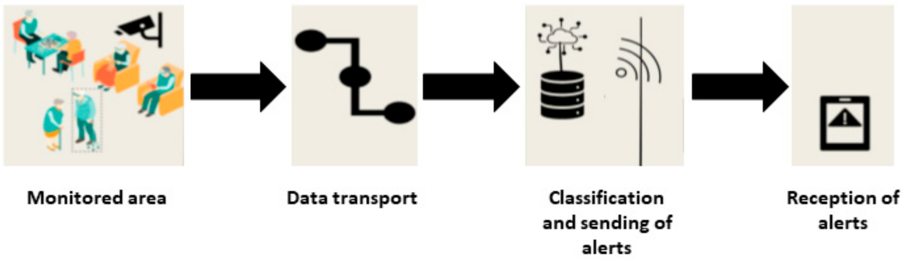


**Figure 1.** Components for a fall detection model.

The monitoring of the area uses a camera that must be physically located in a place with a line of sight without obstacles, in addition, the installation must be in the backlight. The device was located at a height of 1.45 m from the ground and the maximum monitoring distance is 4 m. The location of the camera with the indicated parameters guarantees an adequate field of vision for the detection of the area [38]. Focal errors are corrected with the use of the calibrated camera optics tool, which is part of the iPi Recorder software, the use of this tool is possible when using a Kinect camera. In addition to the correction of the focal error, it is necessary to verify and correct the aspect ratio of the pixels, the image resolution that is used is 1080 pixels, this is an adequate value for the recognition of objects in real-time.

The second component focuses on transporting data from the device to the micro-controller. For proper system operation, the transport must be executed in a fraction of a second. The next component integrates the computer system that, for image processing, considers four skeletonization points in the person's positioning.

### 2.4. Development of the Fall Detection Model

For the development of the fall detection model, the Python programming language, and the libraries available in the application of artificial vision techniques such as Tensor-Flow, OpenCV, pyTelegramBotAPI, etc. are used. In addition to the GitHub interface for Kinect offers a set of routines and access to Python functions for data acquisition from the Kinect sensor [39,40]. In the first stage of the design, the algorithm initializes the interface, and the Kinect sensors are enabled for motion capture and position data collection. The captured images are in color and the person is monitored and skeletonized as shown in Figure 2. As a reference point, a diagonal line was drawn from the base of the hip joint to the base of the shoulder to track and monitor the person's skeleton [41]. The reference points and lines are superimposed on the body of a person, highlighting it from the rest of the common objects in the monitored area, which is taken as a visual reference object in the development for the improvement of the algorithm in the precision adjustments. For posture recognition, the Open Pose Estimation library is used, which allows the reference points of the human body, such as the face and extremities, to be detected in real-time. This library integrated into OpenCV allows reading the activity of a person in an environment, from a video [42].

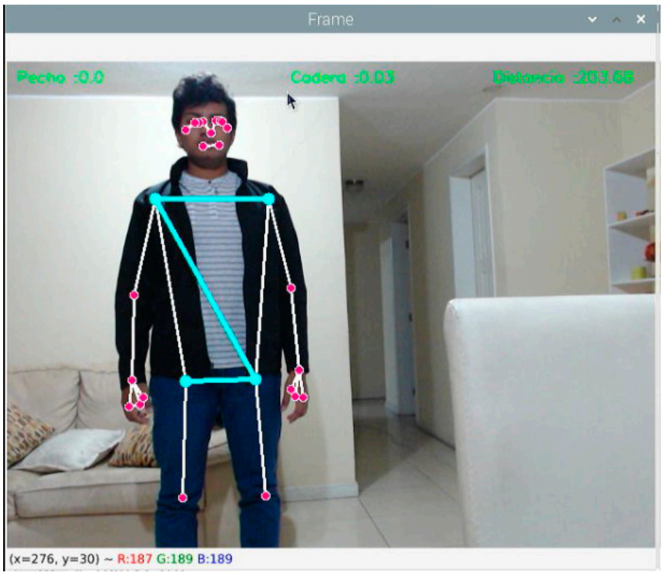


**Figure 2.** Skeletonization of a person for the detection of positions with reference points.

After the "x", "y" and "z" coordinates of the skeletonization reference point are determined, the "y" coordinate corresponding to the movement direction coordinate is selected. For this, an average of 75 data is calculated and subtracted from the current position to establish the reference point of the movements [43]. With the position data the speed is determined, calculating the derivative with the backward differentiation method of the first order, this speed data is sent to a neural network if it detects a fall, it generates an alarm, in case of detecting a movement other than one fall the system continues to show the information on the speed and the images with its skeletonization [43], as shown in Figure 3.

For the development of the classification system, it is necessary to carry out sampling for the generation of data, with which to measure the speed and distance. For data collection, samples are taken from three people of different heights, they made several repetitions of natural movements of people such as walking, sitting, bending over, and falling [44]. The speed data is stored for about two seconds, considering that the Kinect captures 30 images per second and the processing rate of the sampling algorithm is 8 milliseconds. The samples were randomly selected and with these, the speed ranges for each action are established, these values are presented below:

- Walking, −90–28 cm/s
- Sit, 25–130 cm/s
- Crouch, 150–300 cm/s
- Falling, 285–535 cm/s

The neural network operates in the following way, it takes the reference data, which is the diagonal between the reference points, and a threshold of 25 units is established on the Y axis. In addition, various parameters were established to identify a positive drop, when plotting the reference points and connecting lines between the points, as shown in Figure 4.

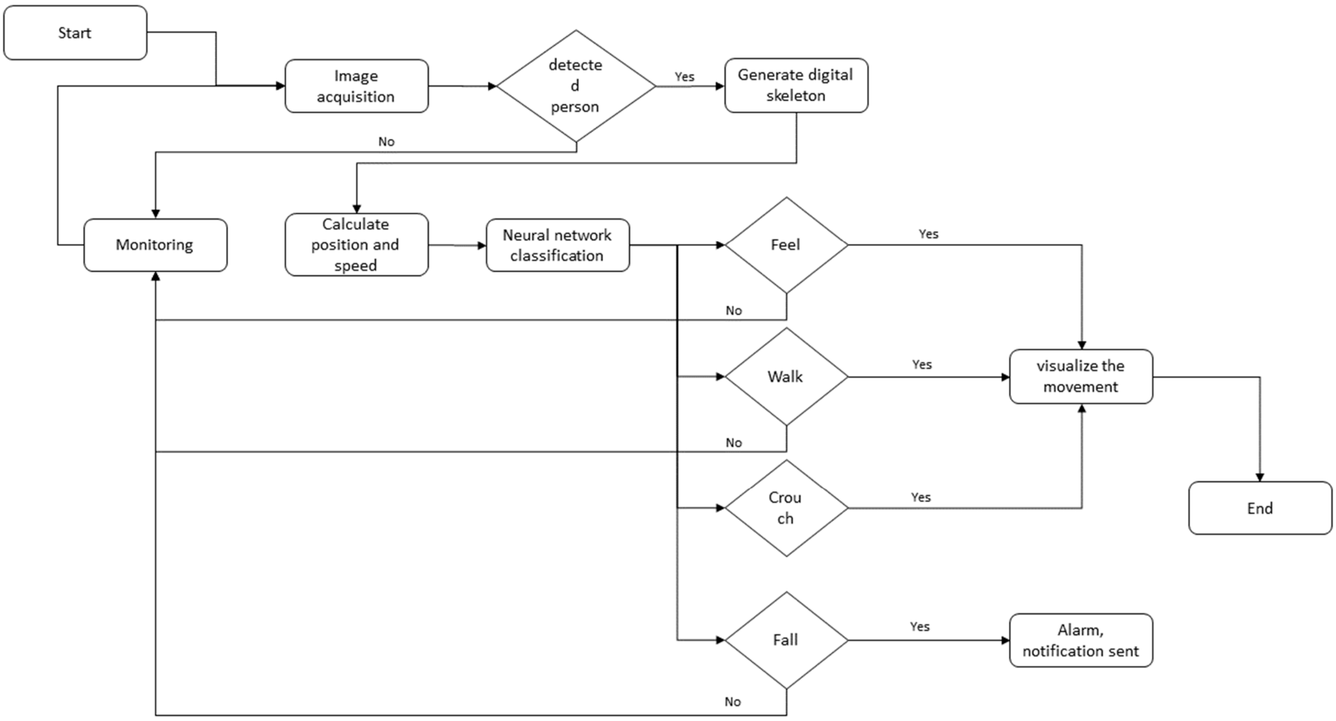


**Figure 3.** Flowchart for fall detection.

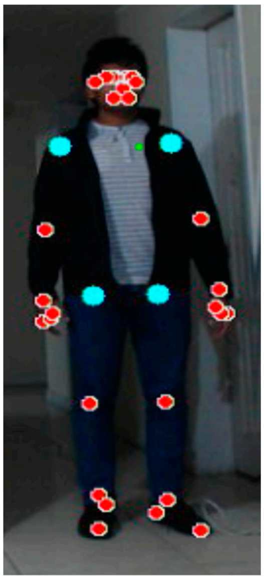

**Figure 4.** Reference points on a person for the identification of joints.

The lines are used to validate the position of the person, considering their characteristics, such as the length of the line, the speed of the movement, the inclination, etc. These parameters allow us to identify and classify the variations when the falls are lateral. The output dataset is assigned a constant value that corresponds to the action to be identified, that is:

- 1 walk action
- 2 actions sit
- 3 action crouches
- 4 fall action

By superimposing a digital skeleton on a person's body, it makes them stand out from the rest of the objects in a room. In addition, the skeleton serves as visual feedback for the improvement of the algorithm in the precision adjustments, as shown in Figure 5.

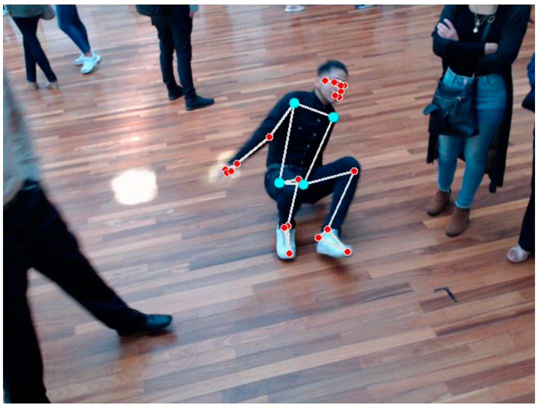

**Figure 5.** Skeletonization of a person by means of a fall detection algorithm.

By detecting changes in the size and inclination of the lines between the celestial points in Figure 6. The algorithm sends a message at the console level informing about the "fall detected" event, then captures the event and presents it in the console in the following format:

- print ('Fall detected')
- cv2.imwrite ('/home/local/Desktop/Program/ImagenGenerada.jpg', frame)
- cv2.putText (frame, 'Fall detected', (10, 100), font, fontScale, (255, 0, 0), thickness, cv2.LINEAA, False)

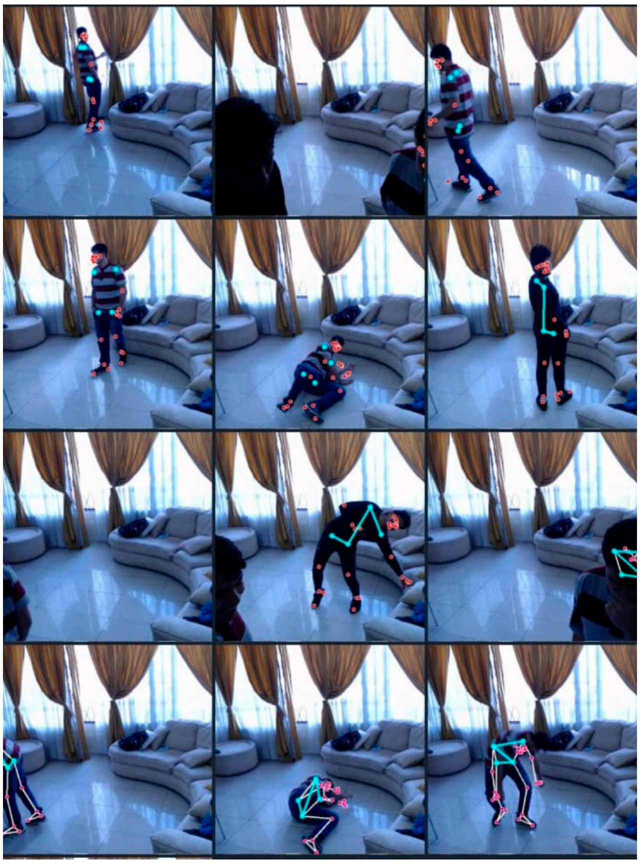

**Figure 6.** Fall detection sequence.

Sending notifications to contact groups or emergencies is done through Telegram. The process is executed by a bot, where the contact group is previously programmed [45]. In addition, the bot looks for the generated image and sends it with a brief description of the record with the date and time when the fall was detected. The message is configurable according to the room where the event occurred, the sending code is:

- bot.send message (chat, "Fall detected in the room")
- bot.send message (chat, now)
- bot.send photo (chat, img)

## 3. Results

The system implemented for three months, this being the time of tests to which it was subjected, presented a constant growth in its precision. In the first version and with a dataset, the accuracy of the model was 60% at the end of the implementation, adjustments, and tests, the accuracy reached approximately 80% about the environment, Figure 7 shows the samples made within the home.

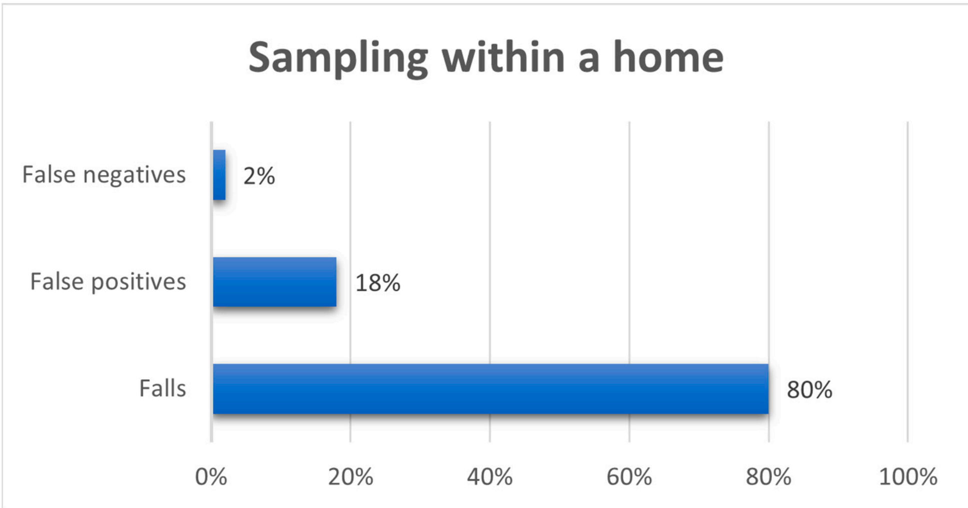


**Figure 7.** Sampling done in the living room of an average home.

The sampling was validated with three people with heights of 1.60 m, 1.70 m, and 1.75 m, respectively, the process consisted of simulating falls and common activities such as walking, bending over, and sitting down. Each person performed 15 times each action. From the testing stage, the system accurately detected the fall using the measured speed and the position of the person as parameters. Precision, sensitivity, and specificity were calculated with the following equations, where:

- A = Accuracy
- S = Sensitivity
- S1 = Specificity
- TP = True positive
- FP = False positive
- FN = False negative
- TN = True negative

$$A = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \quad (1)$$

$$S = \frac{TP}{TP + FP} \times 100 \quad (2)$$

$$S1 = \frac{TN}{TN + FP} \times 100 \quad (3)$$

The confusion matrices of the actions detected together with the total number of falls and non-falls for each person are shown in Table 1.

**Table 1.** Confusion matrix on the sampling of three people.

| | | | | | | |
|---|---|---|---|---|---|---|
| **Person 1** | | | | | | |
| | Samples | | | | Total Predictions | |
| | Fall | Crouch | Feel | Walk | Falls | No falls |
| Fall | 14 | 1 | 0 | 0 | TP 14 | FN 1 |
| Crouch | 1 | 14 | 0 | 0 | | |
| Feel | 0 | 1 | 14 | 0 | FP 3 | TN 39 |
| Walk | 0 | 0 | 0 | 15 | | |
| Accuracy = 92.9% | | Sensitivity = 93% | | | Specificity = 93% | |
| Person 2 | | | | | | |
| | Samples | | | | Total Predictions | |
| | Fall | Crouch | Feel | Walk | Falls | No falls |
| Fall | 15 | 0 | 0 | 0 | TP 15 | FN 0 |
| Crouch | 2 | 13 | 0 | 0 | | |
| Feel | 0 | 2 | 13 | 0 | FP 4 | TN 41 |
| Walk | 0 | 0 | 0 | 15 | | |
| Accuracy = 93.3% | | Sensitivity = 100% | | | Specificity = 91% | |
| Person 3 | | | | | | |
| | Samples | | | | Total Predictions | |
| | Fall | Crouch | Feel | Walk | Falls | No falls |
| Fall | 15 | 0 | 0 | 0 | TP 15 | FN 0 |
| Crouch | 0 | 15 | 0 | 0 | | |
| Feel | 0 | 1 | 14 | 0 | FP 1 | TN 44 |
| Walk | 0 | 0 | 0 | 15 | | |
| Accuracy = 98.3% | | Sensitivity = 100% | | | Specificity = 98% | |

From the results obtained in the table, the metrics of the proposed detection system can be obtained. The average precision is 94.8%, an average specificity of 94%, and an average sensitivity of 97.6%. The sensitivity obtained from the system is high and indicates that it is possible to classify several of the movements that were simulated as falls, the value is low due to the sample data obtained from the first person. The factor that affects the tests carried out is the size of the person and the location of the camera. In a second sampling, the proximity of the camera was changed, and the recognition improved significantly. However, this value is not presented in the section, since the change was made only for the indicated person and to establish the variations in their sample. In addition, it was shown that certain movements that people commonly perform, such as lifting an object or bending over, can be mistaken for a fall.

*System Evaluation and Adjustments*

The implemented system has shown constant growth in its accuracy, from the initial version to the current version, this pressure went from approximately 60% to 74%, varying after adjustments up to 80% about the environment. This increase in accuracy is mainly due to the optimization of the code. Currently, the code is without visible lines that join the skeletonization points, although they are maintained logically, the tracing is no longer performed. This change, which apparently should not have a greater influence, already in execution, triggers a lower use of the processing in each frame, meaning 69% percent of the processor destined to perform this function, or about a 57% load on its four cores. about. Consequently, it allows to raise in the rate of frames per second to 27FPS, the necessary ones to not lose any detail while a fall occurs. Table 2 presents the measurement results of the system and the existing variation when implementing improvements in the management of tracing and skeletonization.

**Table 2.** Comparison of the model, in measurement column 1 the results of the model with tracing are presented and measurement column 2 presents an adjustment by removing the skeletonization tracing.

| Criterion | Measurement 1 | Measurement 2 |
|---|---|---|
| Accuracy | 68.80% | 74% |
| Error range | 34.20% | 26% |
| Relative error | 3.14% | 11.24% |
| Processing | 110% | 69% |
| Memory | 63% | 15% |
| Storage | 0.038% | 0.006% |

Additionally, with the sampling of the data, its low margin of error of 20% is evidenced, mainly due to the false positives captured during the adjustments for the camera focus. In other tests carried out, the efficiency and precision of the system were affected when the system was changed from one environment to another, despite maintaining the system configuration, so that this problem does not exist, it is necessary to adjust the environment with parameters such as height, length, flat areas, etc. To visualize the effects of the adjustments, the results of the CHI square are used and implemented to affirm a relationship between the number of people and an increase in the generation of false positives. Table 3 presents the frequencies of the system. In this phase, a training phase is carried out with 507 records, of which 136 false positives and 11 false negatives were obtained, and 360 falls were detected. The relative frequency allows knowing about the portion that represents a given value in the set of available data on falls.

**Table 3.** Frequency table of the results obtained from the observed System.

| Cases | Absolute Frecuency | Relative Frequency |
|---|---|---|
| Falls | 360 | 0.7101 |
| False positives | 136 | 0.2682 |
| False negatives | 11 | 0.0217 |
| Total | 507 | 1 |

Table 4 presents the results of the system evaluated in two environments to establish the environment with the best characteristics for the implementation of the system. According to the results, in a room where the monitoring measures are less, the system is more effective, compared to an environment such as a room where there is a greater number of objects. However, these parameters must be configured according to the place where the elderly spend more frequently.

**Table 4.** Frequency table of the results obtained from the System observed in two environments.

| Entorno | Falls | False Positives | False Negatives | Total |
|---|---|---|---|---|
| Room | 100 | 25 | 2 | 127 |
| Living room | 260 | 112 | 9 | 382 |
| Total | 360 | 136 | 11 | 507 |

## 4. Discussion

The model has shown good performance in detecting simulated falls in a controlled environment. The system has an average accuracy of 94.8%, specificity of 94%, and sensitivity of 97.6%, with the ability to classify all simulated actions as falls. This result is relevant, compared to related works where values of precision, specificity, and sensitivity are lower than those obtained by this model. In contrast to the proposed system, some works present the use of a device that must be implemented directly in the patient, causing discomfort or rejection, even some prototypes require a power supply, which generates inconvenience to patients [24,46]. Based on the results, it is evident that the system works and is suitable for detecting falls in people.

Another important factor is that the best results were obtained in environments that had good lighting and few objects in the environment. In the evaluation of the model, it was also identified that an area with poor lighting makes it difficult for the estimation model to identify the skeletal structure of the person [47]. This directly affects the detection model, since, by losing track of the reference points of the person, the algorithm receives incorrect data, consequently, the number of false positives increases [48].

About other works and the results obtained, it can be mentioned that deep learning techniques are improving with the use of neural networks [49]. These all have the advantage of learning using previous training, in this way a feasible model is generated for the automatic extraction of the characteristics of an image. These models that have prior knowledge and that are applied to real environments, present higher reliability percentages, about systems that focus on image recognition [50].

The tests carried out on the system revealed several limitations that must be considered to improve the system in a later stage. Among these, lighting was detected as a factor to improve, since, having poor lighting, makes it difficult for the posture estimation model and the task of identifying the person's skeletal structure. This problem directly affects the method used to detect a fall since, by losing track of the person's body, the algorithm receives incorrect data, therefore, it detects falls when they do not occur in the sequence. Another limitation during system evaluation is environments that contain objects. Since, on certain occasions, these elements are identified as the position of a person. Due to this problem, the algorithm assumes that the person has changed his position drastically and will be fed with erroneous data that will raise false positives. In addition to these limitations, it was detected that certain positions such as bending over can be confusing for the system, especially if the person falls onto a chair or similar object; when the system detects the speed and inclination, it generates another false positive. The orientation and position of the camera are another crucial element when monitoring the environment. For this reason, it is important that the device is on a flat site with an excellent line of sight and that the height is regulated based on the average height of the people to be monitored.

## 5. Conclusions

The results obtained from the fall detection model have been evaluated under different scenarios. A prototype has even been created for its application in an embedded system, to improve portability. However, this process punished the processing capacity, therefore, the presentation of these tests was discarded until the hardware was optimized. Currently, the model relies on a laptop for processing and the use of a Kinect camera which can be seen as additional limitations. These parameters must be contrasted with the usefulness of the system and the affectivity of the algorithm, which can meet the needs of older adults.

For the detection of falls in the first phase, several libraries were used, such as Bounding Boxes. This library makes it possible to determine whether a fall has occurred based on the length of the sides. However, when applying the approximation, it was not possible to adequately filter the falls from the non-falls. The problem that was identified is that, when generating the required point cloud, it contained reference points that do not belong to the monitored person, which influences the length of the sides, leading to bad classifications.

When performing the tests and validation, cases were identified in which the precision was not a good indicator of the performance of the model. One of these cases is when the distribution of classes is unbalanced. Even the model in this case was able to predict all samples as the most frequent class, getting a high rate of accuracy, which doesn't make sense because your model isn't learning and it's just predicting everything as the top class, meaning that the model uses other metrics.

**Author Contributions:** W.V.-C. contributed to the following: the conception and design of the study, acquisition of data, analysis, and interpretation of data, drafting the article, and approval of the submitted version. The author S.B.-E. contributed to the study by design, conception, interpretation of data, and critical revision. W.G.-N. and A.M.-N. made the following contributions to the study: analysis and interpretation of data, approval of the submitted version. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Abobakr, A.; Hossny, M.; Nahavandi, S. A Skeleton-Free Fall Detection System from Depth Images Using Random Decision Forest. *IEEE Syst. J.* **2017**, *12*, 2994–3005. [CrossRef]
2. Behera, S.; Mohanty, M.N.; Patnaik, S. A Comparative Analysis on Edge Detection of Colloid Cyst: A Medical Imaging Approach. *Stud. Comput. Intell.* **2012**, *395*, 63–85. [CrossRef]
3. Pierleoni, P.; Belli, A.; Maurizi, L.; Palma, L.; Pernini, L.; Paniccia, M.; Valenti, S. A Wearable Fall Detector for Elderly People Based on AHRS and Barometric Sensor. *IEEE Sens. J.* **2016**, *16*, 6733–6744. [CrossRef]
4. Saleh, M.; Georgi, N.; Abbas, M.; le Bouquin Jeannès, R. A Highly Reliable Wrist-Worn Acceleration-Based Fall Detector. In Proceedings of the European Signal Processing Conference, A Coruña, Spain, 2–6 September 2019; Volume 2019.
5. Jung, S.; Hong, S.; Kim, J.; Lee, S.; Hyeon, T.; Lee, M.; Kim, D.H. Wearable Fall Detector Using Integrated Sensors and Energy Devices. *Sci. Rep.* **2015**, *5*, 17081. [CrossRef]
6. Instituto Nacional de Estadística y Censos. La Población Adulta Mayor Se Triplicaría En Los Próximos 40 Años. Available online: https://inec.cr/noticias/la-poblacion-adulta-mayor-se-triplicaria-los-proximos-40-anos (accessed on 8 August 2022).
7. Carmona Valdés, S.E. *El Bienestar Personal en el Envejecimiento*; Ciencias Sociales de la Universidad Iberoamericana: Distrito Federal, México, 2009; Volume 3, pp. 48–65.
8. Meeradevi, T.; Vikash Kumar, V.; Subhiksa, S.; Rajhan, V. *Wearable Fall Detector for Elderly People*; Innovations in Information and Communication Technology Series; Kongu Engineering College: Bengaluru, India, 2020. [CrossRef]
9. Silva de Lima, A.L.; Smits, T.; Darweesh, S.K.L.; Valenti, G.; Milosevic, M.; Pijl, M.; Baldus, H.; de Vries, N.M.; Meinders, M.J.; Bloem, B.R. Home-Based Monitoring of Falls Using Wearable Sensors in Parkinson's Disease. *Mov. Disord.* **2020**, *35*, 109–115. [CrossRef]
10. Medrano, C.; Igual, R.; García-Magariño, I.; Plaza, I.; Azuara, G. Combining Novelty Detectors to Improve Accelerometer-Based Fall Detection. *Med. Biol. Eng. Comput.* **2017**, *55*, 1849–1858. [CrossRef]
11. Silva, J.; Gomes, D.; Sousa, I.; Cardoso, J.S. Automated Development of Custom Fall Detectors: Position, Model and Rate Impact in Performance. *IEEE Sens. J.* **2020**, *20*, 5465–5472. [CrossRef]
12. Chaccour, K.; Darazi, R.; el Hassani, A.H.; Andres, E. From Fall Detection to Fall Prevention: A Generic Classification of Fall-Related Systems. *IEEE Sens. J.* **2017**, *17*, 812–822. [CrossRef]
13. Santos García, D.; López Ariztegui, N.; Cubo, E.; Vinagre Aragón, A.; García-Ramos, R.; Borrué, C.; Fernández-Pajarín, G.; Caballol, N.; Cabo, I.; Barrios-López, J.M.; et al. Clinical Utility of a Personalized and Long-Term Monitoring Device for Parkinson's Disease in a Real Clinical Practice Setting: An Expert Opinion Survey on STAT-ONTM. *Neurologia* **2020**, *1*, 8. [CrossRef]
14. Odunmbaku, A.; Rahmani, A.M.; Liljeberg, P.; Tenhunen, H. Elderly Monitoring System with Sleep and Fall Detector. In Proceedings of the Internet of Things. IoT Infrastructures Second International Summit, IoT 360° 2015, Rome, Italy, 27–29 October 2015; Volume 169.
15. de Ramón-Fernández, A.; Ruiz-Fernández, D.; Marcos-Jorquera, D.; Gilart-Iglesias, V.; Vives-Boix, V. Monitoring-Based Model for Personalizing the Clinical Process of Crohn's Disease. *Sensors* **2017**, *17*, 1570. [CrossRef]
16. Goeuriot, L.; Pasi, G.; Viviani, M.; Villegas-Ch, W.; Molina, S.; de Janón, V.; Montalvo, E.; Mera-Navarrete, A. Proposal of a Method for the Analysis of Sentiments in Social Networks with the Use of R. *Informatics* **2022**, *9*, 63. [CrossRef]
17. Villegas-Ch, W.; García-Ortiz, J.; Sánchez-Viteri, S. Identification of the Factors That Influence University Learning with Low-Code/No-Code Artificial Intelligence Techniques. *Electronics* **2021**, *10*, 1192. [CrossRef]
18. Shen, L.; Zhang, Q.; Cao, G.; Xu, H. Fall Detection System Based on Deep Learning and Image Processing in Cloud Environment. In Proceedings of the 12th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2018), Matsue, Japan, 4–66 July 2018; Volume 772.
19. Plechawska-Wójcik, M.; Rybka, J. Assessment and Comparison of Functionalities of Telemedical Applications. *Int. J. Comput. Appl.* **2014**, *107*, 8887. [CrossRef]
20. Nho, Y.H.; Lim, J.G.; Kwon, D.S. Cluster-Analysis-Based User-Adaptive Fall Detection Using Fusion of Heart Rate Sensor and Accelerometer in a Wearable Device. *IEEE Access* **2020**, *8*, 40389–40401. [CrossRef]
21. Alwan, M.; Rajendran, P.J.; Kell, S.; Mack, D.; Dalal, S.; Wolfe, M.; Felder, R. *A Smart and Passive Floor-Vibration Based Fall Detector for Elderly*; IEEE: Piscataway, NJ, USA, 2006.

22. Wu, G.; Xue, S. Portable Preimpact Fall Detector with Inertial Sensors. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2008**, *16*, 178–183. [CrossRef]
23. Sheikh, S.Y.; Jilani, M.T. A Ubiquitous Wheelchair Fall Detection System Using Low-Cost Embedded Inertial Sensors and Unsupervised One-Class SVM. *J. Ambient. Intell. Humaniz. Comput.* **2021**. [CrossRef]
24. Oh, J.Y.; Choi, H.S.; Jung, S.H.; Kim, H.S.; Shin, H.Y. Development of Pallet Recognition System Using Kinect Camera. *Int. J. Multimed. Ubiquitous Eng.* **2014**, *9*, 227–232. [CrossRef]
25. Roy, G.; Bhuiya, A.; Mukherjee, A.; Bhaumik, S. Kinect Camera Based Gait Data Recording and Analysis for Assistive Robotics-An Alternative to Goniometer Based Measurement Technique. *Procedia Comput. Sci.* **2018**, *133*, 763–771. [CrossRef]
26. Abudarham, N.; Shkiller, L.; Yovel, G. Face Recognition in Humans and Machines. *J. Vis.* **2018**, *18*, 156. [CrossRef]
27. de Oliveira, E.M.; Leme, D.S.; Barbosa, B.H.G.; Rodarte, M.P.; Alvarenga Pereira, R.G.F. A Computer Vision System for Coffee Beans Classification Based on Computational Intelligence Techniques. *J. Food Eng.* **2016**, *171*, 22–27. [CrossRef]
28. Tarlak, F.; Ozdemir, M.; Melikdglu, M. Computer Vision System Approach in Colour Measurements of Foods: Part I. Development of Methodology. *Food Sci. Technol.* **2016**, *36*, 382–388. [CrossRef]
29. Chmiel, M.; Słowiński, M. The Use of Computer Vision System to Detect Pork Defects. *LWT* **2016**, *73*, 473–480. [CrossRef]
30. Utami, N.A.; Maharani, W.; Atastina, I. Personality Classification of Facebook Users According to Big Five Personality Using SVM (Support Vector Machine) Method. *Procedia Comput. Sci.* **2021**, *179*, 177–184. [CrossRef]
31. Leong, W.C.; Bahadori, A.; Zhang, J.; Ahmad, Z. Prediction of Water Quality Index (WQI) Using Support Vector Machine (SVM) and Least Square-Support Vector Machine (LS-SVM). *Int. J. River Basin Manag.* **2021**, *19*, 149–156. [CrossRef]
32. Aziz, O.; Klenk, J.; Schwickert, L.; Chiari, L.; Becker, C.; Park, E.J.; Mori, G.; Robinovitch, S.N. Validation of Accuracy of SVM-Based Fall Detection System Using Real-World Fall and Non-Fall Datasets. *PLoS ONE* **2017**, *12*, e0180318. [CrossRef]
33. Liu, K.C.; Hsieh, C.Y.; Hsu, S.J.P.; Chan, C.T. Impact of Sampling Rate on Wearable-Based Fall Detection Systems Based on Machine Learning Models. *IEEE Sens. J.* **2018**, *18*, 9882–9890. [CrossRef]
34. Chhetri, S.; Alsadoon, A.; Al-Dala'in, T.; Prasad, P.W.C.; Rashid, T.A.; Maag, A. Deep Learning for Vision-Based Fall Detection System: Enhanced Optical Dynamic Flow. *Comput. Intell.* **2021**, *37*, 578–595. [CrossRef]
35. Islam, M.; Tayan, O.; Islam, R.; Islam, S.; Nooruddin, S.; Kabir, M.N.; Islam, R. Deep Learning Based Systems Developed for Fall Detection: A Review. *IEEE Access* **2020**, *8*, 166117–166137. [CrossRef]
36. Sarabia-Jácome, D.; Usach, R.; Palau, C.E.; Esteve, M. Highly-Efficient Fog-Based Deep Learning AAL Fall Detection System. *Internet Things* **2020**, *11*, 100185. [CrossRef]
37. Castro-Luna, G.; Jiménez-Rodríguez, D. Relative and Absolute Reliability of a Motor Assessment System Using Kinect® Camera. *Int. J. Environ. Res. Public Health* **2020**, *17*, 5807. [CrossRef]
38. Cevallos Salazar, G.F. Análisis Del Desempeño Del Sitio Web Del Instituto Ecuatoriano de Seguridad Social (IESS) Para Evaluar Su Accesibilidad y Usabilidad En Los Adultos Mayores de La Asociación de Jubilados de La "Hermandad de Ferroviarios" de La Ciudad de Quito. *Propues. De. ComHumanit. Rev. Científica De Comun.* **2020**, *11*, 149–178. [CrossRef]
39. Ghifari, H.G.; Darlis, D.; Hartaman, A. Pendeteksi Golongan Darah Manusia Berbasis Tensorflow Menggunakan ESP32-CAM. *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.* **2021**, *9*, 359. [CrossRef]
40. Brahin, N.M.A.; Nasir, H.M.; Jidin, A.Z.; Zulkifli, M.F.; Sutikno, T. Development of Vocabulary Learning Application by Using Machine Learning Technique. *Bull. Electr. Eng. Inform.* **2020**, *9*, 362–369. [CrossRef]
41. Huang, J. *Accelerated Training and Inference with the Tensorflow Object Detection API*; Huang, J., Ed.; Google AI Blog: Mountain View, CA, USA, 2017.
42. Hsieh, C.H.; Lin, D.C.; Wang, C.J.; Chen, Z.T.; Liaw, J.J. Real-Time Car Detection and Driving Safety Alarm System with Google Tensorflow Object Detection API. In Proceedings of the International Conference on Machine Learning and Cybernetics, Kobe, Japan, 7–10 July 2019; Volume 2019.
43. Aningtiyas, P.R.; Sumin, A.; Wirawan, S. Pembuatan Aplikasi Deteksi Objek Menggunakan TensorFlow Object Detection API Dengan Memanfaatkan SSD MobileNet V2 Sebagai Model Pra-Terlatih. *J. Ilm. Komputasi* **2020**, *19*, 421–430. [CrossRef]
44. Manajang, D.J.P.; Sompie, S.R.U.A.; Jacobus, A. Implementasi Framework Tensorflow Object Detection API Dalam Mengklasifikasi Jenis Kendaraan Bermotor. *J. Tek. Inform.* **2020**, *15*, 171–178.
45. Villegas-Ch, W.; García-Ortiz, J.; Mullo-Ca, K.; Sánchez-Viteri, S.; Roman-Cañizares, M. Implementation of a Virtual Assistant for the Academic Management of a University with the Use of Artificial Intelligence. *Future Internet* **2021**, *13*, 97. [CrossRef]
46. Al-Azzo, F.; Taqi, A.M.; Milanova, M. Human Related-Health Actions Detection Using Android Camera Based on TensorFlow Object Detection API. *Int. J. Adv. Comput. Sci. Appl.* **2018**, *9*, 9–23. [CrossRef]
47. Elgendi, M.; Picon, F.; Magnenat-Thalmann, N.; Abbott, D. Arm Movement Speed Assessment via a Kinect Camera: A Preliminary Study in Healthy Subjects. *Biomed. Eng. Online* **2014**, *13*, 88. [CrossRef]
48. Park, C.; Kim, J.; Sohn, J.C.; Choi, H.J. A Wrist-Type Fall Detector with Statistical Classifier for the Elderly Care. *KSII Trans. Internet Inf. Syst.* **2011**, *5*, 1751–1768. [CrossRef]
49. de Miguel, K.; Brunete, A.; Hernando, M.; Gambao, E. Home Camera-Based Fall Detection System for the Elderly. *Sensors* **2017**, *17*, 2864. [CrossRef]
50. Brownsell, S.; Hawley, M. Fall Detectors: Do They Work or Reduce the Fear of Falling? *Hous. Care Support* **2004**, *7*, 18–24. [CrossRef]