Fall Alert Detection System (FADS)

Justin [redacted]
Missouri State University
Springfield, Missouri, United States
[redacted]

Connor [redacted]
Missouri State University
Springfield, Missouri, United States
[redacted]

Joe Enright
Missouri State University
Springfield, Missouri, United States
<enright321@live.missouristate.edu>

Josh [redacted]
Missouri State University
Springfield, Missouri, United States
[redacted]

***Abstract*— In this document, we propose a method to developing a fall detection system that uses several layers of image processing that identifies the differences between the first frame of the video feed (which is used as a baseline) and the current frame. The necessary components include a regular camera connected to our FADS software system to perform all of the necessary operations. Our solutions detect subjects moving withing the camera's view that meet or exceed a specified size criteria. This movement will trigger different log events such as when the user has entered the room and when the user has suffered from a fall. According to [1], "more than one out of four older people fall each year, but less than half tell their doctor". The death rate caused by people suffering from falls has continued to increase over the years. This death toll can be significantly reduced with a proper fall detection system, the issue is getting the majority of the population to actually want to use the product. The idea is to design and implement software that effectively tracks a user's movements using a Gaussian mixture model for background subtraction, which eliminates the need for extra accessories or devices that the end user would typically need for a system like this to operate effectively. OpenCV is a python-based library that our system utilizes to detect and extract pixel changes in the frame that allows the system to track the target in the foreground. The Imutils package will be used with OpenCV in order to resize the frame as well as grab the contours of the person in frame. Our method analyzes the x and y axis of the person being tracked, and determines the status of the individual based on their position. If the individual in frame is detected in a horizontal position on the floor, the system generates an alert that a fall has been detected. The simplicity of use and installation will appeal to a higher percentage of people in comparison to other products that require them to wear a monitoring device at all times, which will ultimately help reach a broader range of people and contribute to lowering the death toll of falling senior citizens.**

Keywords—Gaussian mixture model, OpenCV, Imutils package

I. Introduction

There are a variety of different software applications that exist in the security and protection space with the objective of tracking and detecting human activity. Many of the existing software program's serve a similar purpose as to what we have attempted to accomplish with our system, yet we have a slight variation at an attempt to enhance the end users experience. Many of the preexisting software systems have taken the route of using devices or mobile application's that will detect the status of the individual. Sensors for the user to keep on their body that can detect arm motion velocity, track irregularities in heart rate, or some other method containing a device that looks for inconsistencies with the user. For instance, [3] uses a sensor through a mobile application that detects the balance state of the individual using the product to preemptively create an alert prior to a fall event. Another system includes the use of a smart helmet integrated with a wearable camera, accelerometers, and gyroscope sensors [6]. While at times these may be effective approaches and it is necessary to perform studies relating to security in the fall detection area, however the question we want to address is how to implement a system that users are willing to actually use.

We have decided to focus on a simpler approach of using a camera system to track the user activity in their home. The system is able to lock on and follow the motions of the individual in the foreground by use of extracting each individual frame from the camera, and performing a Gaussian blur operation that implements a filter on the background projection of the image. Each new frame is subtracted from the calculated reference image and thresholds the result. Through manipulating each individual frame using image-processing techniques, the system is able to successfully distinguish between objects in the background from moving objects in the foreground. Once the Gaussian blur is put into effect, the system uses other built in OpenCV methods to hone in on the movements of the person in frame. Event-driven architecture is utilized so that events such as when the user enters the frame, when the user leaves the frame, when the user suffers a fall, etc. trigger functions in the implementation to activate so that logs are created and added to the log file. The system implements a function for monitoring the person in frame, containing a series of if statements that allows the system to decipher between whether the person is in an upright versus horizontal position. The status of the person is constantly being monitored while in the frame of the camera, and changes depending on whether the person is classified as standing or in a prone position. In the event that the person has fallen to a prone position and can't get back up, an alert is created and logged to the system's log file.

II. Relevant Research

In a recent journal publication [2], Dr. Klaas A. Hartholt discusses the mortality rates from falls among adults in the U.S. aged 75 or older from the years 2000 to 2016. Fortunately, people are living longer lives now, in comparison to previous centuries, due to advancements in the science and medicine fields as well as other areas. However, people reaching the age of 65 and older can become fragile and more susceptible to fall related injuries. In the data from the U.S. National Vital Statistics mortality files, “The absolute number of deaths from falls among US adults aged 75 years or older increased from 8613 in 2000 to 25189 in 2016” [2]. In addition, in an article published by the CDC “Fall death rates among adults age 65 and older increased about 30% from 2009 to 2018” [1]. These tragic deaths have steadily inclined over the past 20 years, and can no longer be ignored. The article [2] also states, “Fall prevention strategies are typically recommended for adults older than 65 years”. Along with the discussion of fall prevention strategies, there needs to be a discussion for monitoring systems to help prevent these deaths from occurring. Providing incapacitated people with a way for quickly getting in touch with the authorities would be a great advancement in helping combat this growing issue. With all of the significant data discovered over the years demonstrating the seriousness of fall related incidents, it has become evident that there is a space for technology to enter into and contribute to helping lower the mortality from fall related incidents.

The company Apple has taken a stab at developing technology with fall detection capabilities in the latest series 4 and series 5 apple watches. David Phelan, senior contributor of consumer tech for Forbes magazine, discussed this potential breakthrough product in a recent Forbes article [3]. In this article, he discusses of this incredible story of a 34-year-old man cooking in his home when out of nowhere he started having back spasms and fell to the floor. Luckily, he was wearing the watch that contains motion sensors to detect the abnormal movement in his arm swing. The watch instantly prompted him with a button to press in the event that the authorities were needed. There is no question that this latest version of the apple watch is a great technological advancement that will hopefully save many lives. However, what if he hadn’t been wearing the watch because he took it off to take a shower and forgot to put it back on, or removed the watch before he started cooking for comfort. This is where our designed system can come into great use. With our visual-based motion detection camera systems set up around the user's home, the user doesn’t need to worry about putting on the device while walking around. Fall related incidents can happen when you least expect them, which is why we feel the best approach for attacking this issue is with a visual-based camera monitoring system.

In [4], authors emphasized the use of motion a sensor that users will wear around their waste. The hardware consists of a motion sensor, a Bluetooth sensor, and a WIFI sensor that provides the system with wireless capabilities so that alerts can be sent wherever the user is located as long as the system is connected to the internet. In our work, we eliminate the use of excessive hardware equipment to just a basic camera installation. The scope of our project is limited to the user’s home environment, which is in contrast to [4] attempting to monitor the individual’s behavior at locations outside their home. In addition, [4] does not utilize a camera system at all, but limits their detection technique to a sensor that calculates the speed and position of the user, which the system then obtains the data and uses to determine the person’s status. The system we have designed is a necessary and cost-effective alternative for people who want to have a sense of protection in the event of a fall, but also prefer to walk around their homes without having to wear a sensor belt. The maintenance costs ensuring that all of the equipment on the sensory belt is working as intended can be an added expense as well as a nightmare to determine which aspect of the system is not functioning properly. The durability of the sensor belt will be put to the test every time a fall event does occur, whereas the camera system will be the single source of maintenance and will not face the amount of physical toll that a sensory belt may have to endure.

Another method for detecting falls is discussed in [5], where the premise is based on the temperature distribution using an IR array sensor. People give off a distinctly higher temperature in comparison to inanimate objects in an indoor environment. This system is able to effectively target a user based on heat that the body gives off, which allows for implementing tracking ability to monitor the individual. This is an interesting approach and has great potential for success, however we feel that a visual based system tracking the user's posture and position is a more accurate approach to tracking. The heat seeking method, that many researches have attempted to implement, will have to deal with the trickiness of deciphering between a pet that may give off similar body heat that humans give off. We are confident in the approach of monitoring the area of the person in different positions to effectively capture when the user appears to be in danger.

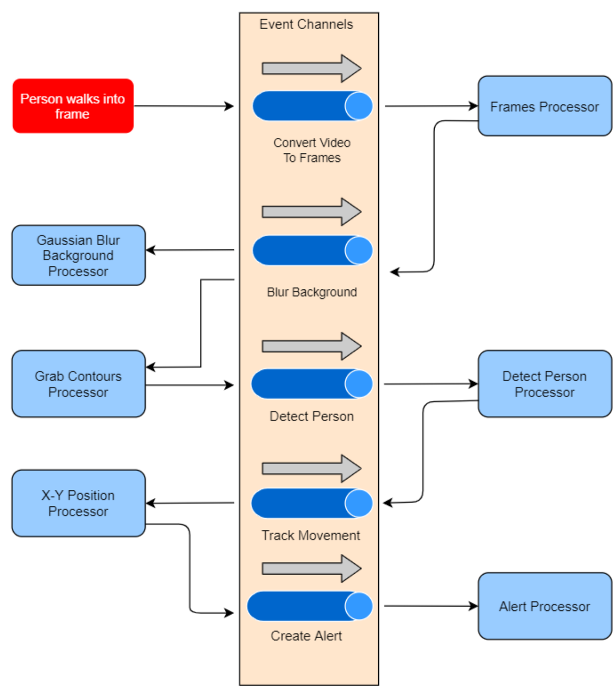

Figure 1. Event-Driven System Architecture

III. PROPOSED SOLUTION

In Fig. 1, we show the proposed high-level design of our FADS (Fall Alert Detection System) event driven system architecture utilizing the broker topology. The system contains a series of six different events chained together that trigger asynchronously when activated.

A. *Event Processing Components*

After the camera is turned on and connected to FADS, the system is activated when the user enters into the cameras line of vision triggering an event. The event travels through the event channel and reaches its destination at the frame's processor component. The frames processor component performs the business logic for converting the feed into individual frames. After the feed is successfully converted into frames, the event flows through the next channel and is received by the blur background processor. The system utilizes an OpenCV Gaussian blur method that implements a filter on the image, reducing the high frequency noise of the objects in the background so that the target in the foreground can eventually be separated out.

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{1}$$

With this two-dimensional formula, the Gaussian blur algorithm is able to perform in a linear time complexity. The Gaussian implementation convolutes the image, where each pixel is set to the average of the neighboring pixel. Upon the event processor's completion of processing the event, the event flows through to the grab contours processor. This process event effectively separates the target object in the foreground. After the grab contour processor event is completed, the event flows through the next channel to the detect person processor. This is where the system effectively renders a rectangle around the contour of the target object. The system is able to effectively determine the dimensions of the person that needs to be tracked, and a rectangle will be drawn to the screen creating a border around the target. Now that the target is successfully captured, the next processing component is for tracking the movement of the target. This component determines the threshold for the current state of the target based on the positioning in the x-y axis. The area of the subject's body position is obtained by the system, where they are classified as being in a horizontal or up-right position. After this event component is processed, the event goes to the final component of an alert being created. An alert is initialized upon the status of the person in frame. If the person is detected in the fallen, horizontal state, an alert is created. Otherwise, no alert is created and the system resets. This process is a loop that continues until an alert is created or the system is turned off.

Logging Events

Included in the implementation is a function for writing log events to the systems log file. The information in the log file includes the date and time for program initialization, when the camera feed is acquired, program initialization, subject entering the camera's view, suspected fall state, suspected safe state, subject leaving the camera's view, and the termination of the program. This information will be used by the system administrator to verify that the camera is connected with the FADS system, and the program is working as expected.

Flow Diagram/Algorithm

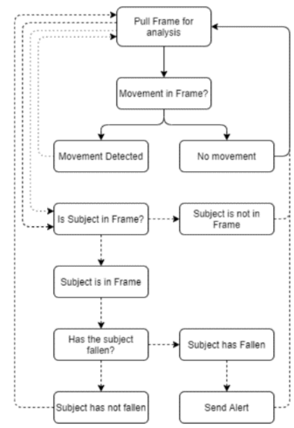

Figure 2. Flow Diagram

The program code is composed of a series of different functions that perform specific tasks in succession. The diagram in Figure 2 demonstrates the programs process once a subject has been detected. In the process video function we implemented, the program resizes the frame with an Imutils method then performs a series of OpenCV methods on the frame to grab the contour of the user. Once this information is processed, we implemented a for loop that will draw a rectangle around the contours of the user, and follow their movements until the target is out of sight of the camera. The next function monitors the subject's position, obtaining the height and width of the rectangle tracking the human. This is implemented with a series of if statements that will execute based on the x and y values of the user. Here is the algorithm used by the system to effectively track the user and create an alert when necessary:

1. The video feed is converted into frames for the system to work with.
2. The current frame is retrieved.
3. The current frame is processed through several filters.
4. The current frame is then compared to the first frame of the video feed for any differences.
5. If any differences are found, the system retrieves the contour of the new subject.
6. The horizontal and vertical position values are obtained by the system.
7. The status of the person is then updated when the horizontal values is larger than the vertical value.

IV. Experimental Results

Here are a series of images taken from a video feed that we recorded to demonstrate the different states of the system. This video was recorded with a regular 640x480 resolution camera to show that a high-resolution camera is not necessary for the system to operate effectively. We have performed a series of different tests, limited to the scope of a well-lit room.

A. Safe State

In this first image, we have a user entering into the frame in a safe, up-right position as depicted in the upper left corner of the window. The date and time are also displayed in the window at the bottom left corner. The system has successfully drawn a rectangle around the moving person that will track him until he is no longer in frame.

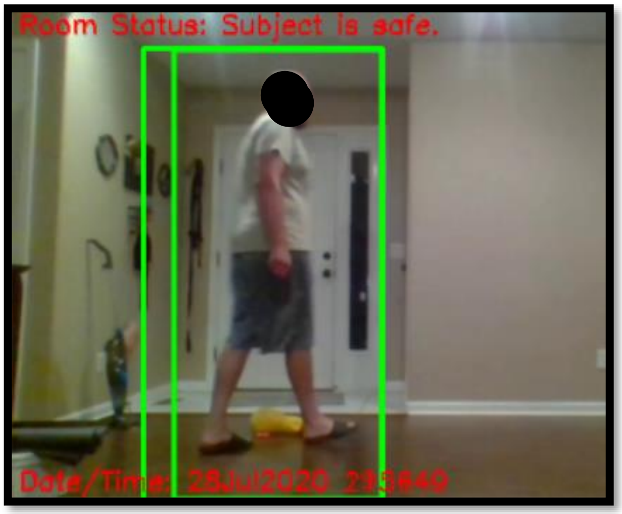

Figure 3. Subject is safe

B. Fall State

In Fig. 3, the status of the system changes when the person falls down. The tracker detects the point at which the x-value of the target is larger than the y-value. The room status is updated with a new message at the top of the frame.

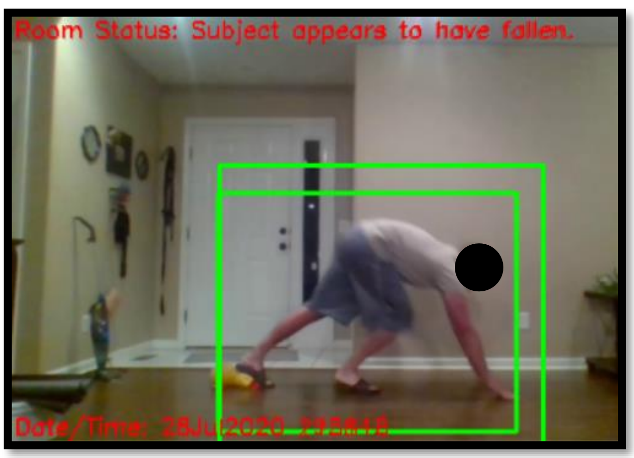

Figure 4. Subject has fallen

C. Alert Created

In this image, the object has remained in the fall state for over 10 seconds, triggering the alert event. We can see from viewing the timer that approximately 10 seconds have passed from Figure 4 to Figure 5.

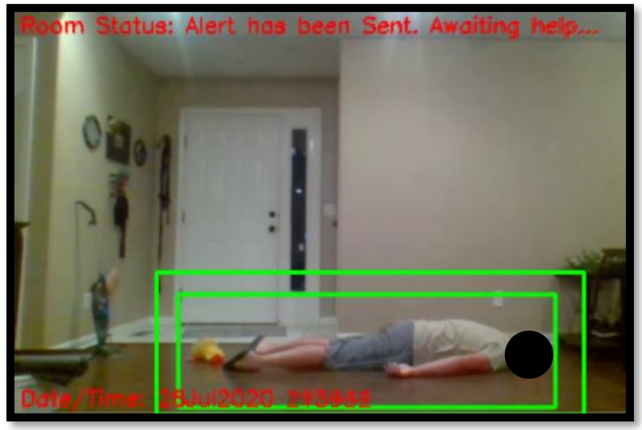

Figure 5. Alert is created

D. Back to Safety

This image was taken in a different video recording where the subject was able to stand up after a fall within the time frame of 10 seconds, effectively stopping the alert event from executing. The subject is now back to the safe state, and the system is reset.

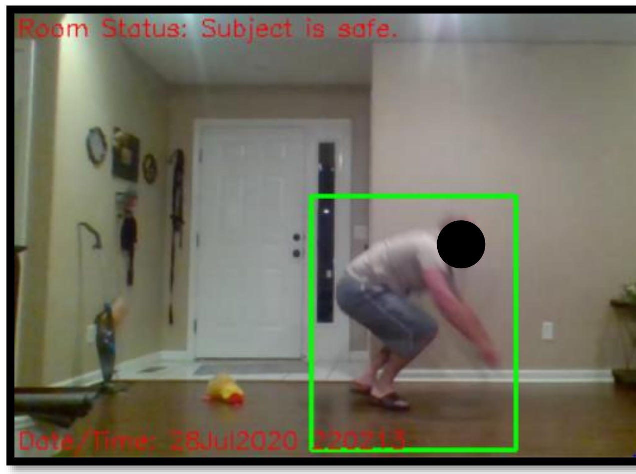

Figure 6. Subject is safe, again

The demonstration provided in Figures 2-5 indicate the different states of the system that will update based on the actions of the user. Figure 4 and figure 6 demonstrate two fairly similar positions by the user, yet their status according to the system indicates that they are not. When looking closely at Figure 3, we see that the x-value of the person is slightly bigger than the y-value. As soon as the posture of the user is longer horizontally than vertically, the status is updated to “subject has appeared to have fallen”. In Figure 5, the posture is slightly more vertical than horizontal, therefore the status is in the safe state. After performing a study of testing the effectiveness of the system in different locations around an indoor environment, the results are as follows:

Room:	Light condition:	Accurately track target user:	User State Accurately Detected:
Kitchen	Well-lit	5/5	4/5
Living Room	Slightly dim	4/5	4/5
Basement	Low light	2/5	1/5

Figure 7: Testing system under Different Conditions

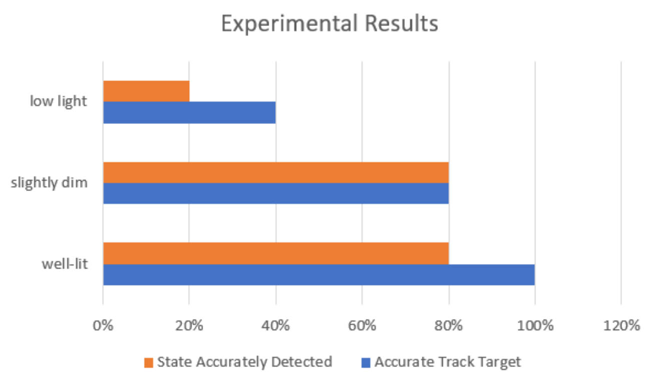

Figure 8: Graph of data from Figure 7

This chart displays the system's effectiveness under different lighting conditions in an indoor environment. Five separate tests were conducted in each room listed in Figure 7 in order to accurately simulate a practical demonstration of the software. The test produced accurate results when setup in a well-lit indoor environment, effectively tracking the moving person in the frame and determining the persons safety status. After a series of 5 different tests performed in a low light room, the system's ability to track the user in frame significantly decreased. On several occasions in the basement test, that system was unsuccessfully able to track the user and would the status would incorrectly depict the actual user's status. This will be an area for the team to improve upon when further implementing new functionality. In addition, our code will be publicly available so that anyone can access and use as a base to build their own fall detection system. The BSD License is included in the readme file along with the necessary python-based libraries that need to be installed.

V. Conclusion and Future Work

In this document, we proposed a system with the capability to track a person, determine if they may or may not have fallen, and send an alert to emergency personnel and/or caregivers all within the field of view of a regular surveillance camera. We have focused our research and development into creating a cost-effective and convenient system that is easy for any user to integrate into their homes without the worry of it intruding upon their independent lifestyle. With all the well published data concerning the ever-increasing mortality rate resulting from falls and fall related incidents, this technology can be a great contribution into both improving the lives and ensuring the safety of both senior citizens and persons with disabilities.

No project is ever fully "complete" as improvements can always be made to increase reliability and either add or improve upon features. As it stands, FADS could be improved by implementing a more concise object tracking ability for a wider range of lighting conditions. In addition, improving upon the object detection features will allow for a more accurate identification of what is and is not a human being as our focus is to determine when a person has fallen, not when a housecat decided to nap in the middle of the room. These two improvements alone would drastically improve upon the system's reliability.

In the future, we plan to add different functionality to the system, such as: sending an automated SMS alert to the administrator each time a fall event is detected, differentiating between varying severity of fall events, and identifying different motions that could preemptively detect a fall before the fall happens. Also, we would like to implement a sleep mode for the system so that energy is conserved until someone enters the frame, triggering the system to fully activate. Detecting motions that could signify fall events before they occur would cut down on the time it takes for the proper authorities to arrive and help the falling victim, which would be a great step in improving the system if done correctly.

References

[1] *"Important Facts about Falls." Centers for Disease Control and Prevention, Centers for Disease Control and Prevention, 10 Feb. 2017,* *www.cdc.gov/HomeandRecreationalSafety/Falls/adultfalls.html*.

[2] A. Hartholt, Klaas. "Mortality From Falls Among US Adults Aged 75 Years or Older, 2000-2016." *JAMA*, American Medical Association, 4 June 2019, jamanetwork.com/journals/jama/fullarticle/2735063.

[3] Phelan, David. "Has Apple Watch Series 4 Fall Detection Saved Its First Life?" Forbes, Forbes Magazine, 26 Oct. 2018, www.forbes.com/sites/davidphelan/2018/10/26/has-apple-watch-series-4-fall-detection-saved-its-first-life/#7d3525f47b45.

[4] Rathi, Neeraj, et al. "Wearable Low Power Pre-Fall Detection System with IoT and Bluetooth Capabilities ." Wearable Low Power Pre-Fall Detection System with IoT and Bluetooth Capabilities - IEEE Conference Publication, IEEE, 2017, ieeexplore.ieee.org/document/8268778.

[5] Ogawa, Yuya, and Katsuhiro Naito. Fall Detection Scheme Based on Temperature Distribution with IR Array Sensor. IEEE, 2020, ieeexplore.ieee.org/document/9043000.

[6] Shahiduzzaman, Kazi, et al. Enhancing Fall Detection for Elderly with Smart Helmet in a Cloud-Network-Edge Architecture - IEEE Conference Publication, IEEE, 2019, ieeexplore.ieee.org/document/8991972.