A novel web-based depth video rewind approach toward fall preventive interventions in hospitals

Moein Enayati, Tanvi Banerjee, *IEEE Student Member*, Mihail Popescu, *IEEE Senior Member,*
Marjorie Skubic, *IEEE Senior Member* and Marilyn Rantz

Abstract— **The purpose of this study was to implement a web based application to provide the ability to rewind and review depth videos captured in hospital rooms to investigate the event chains that led to patient's fall at a specific time. In this research, Kinect depth images are being used to capture shadow-like images of the patient and their room to resolve concerns about patients' privacy. As a result of our previous research, a fall detection system has been developed and installed in hospital rooms, and fall alarms are generated if any falls are detected by the system. Then nurses will go through the stored depth videos to investigate for possible injury as well as the reasons and events that may have caused the patient's fall to prevent future occurrences. This paper proposes a novel web application to ease the process of search and reviewing the videos by means of new visualization techniques to highlight video frames that contain potential risk of fall based on our previous research.**

I. INTRODUCTION

Falls in the hospital rooms are considered a huge burden on healthcare costs. They can lead to injuries, extended length of stay, and increase in cost for both the patients and the hospital. It can also lead to emotional trauma for the patients and their families [1]. Having Microsoft Kinects installed in the hospital rooms to capture and process every movement in the room, we deployed our previously developed fall-detection system to detect naturally occurring falls, generate a real-time fall alarm and broadcast it to hospital nurses for immediate intervention. These systems also store a processed and reduced version of the 3D depth videos on a central file storage to provide information to the dedicated nursing team for post-fall quality improvement process. The compression technique that helps reducing video size by omitting non-movement frames from it also makes it almost impossible for the hospital staff to find the event that led to a fall alarm. There was a need to visualize fall events and the video contents accordingly. In this paper, we describe a web-application with a handy user interface to easily search among terabytes of depth videos to facilitate the finding and reviewing of the chain of events that lead to a patient fall. We will also discuss the improvements in the new version of the application which reduced the size of transferred videos by converting them to MP4 videos and makes the application platform free. This improvements in speed and compatibility on different browsers, caused more user satisfaction and more frequent use of the web-application.

II. Related Works

There is a wide variety of techniques implemented in the whole process of storing, reloading and reviewing patient monitoring videos taking special considerations to three aspects of personal privacy, video storage and meta-data visualization. In 2011 Guttmann et al. [2] developed a content aware video manipulation method (CAVM) to visually alter a video with respect to the relative importance of various regions. During this two phase procedure each frame is processed to assign a degree of importance to each pixel based on movements in the scene, while the second phase transforms and manipulates video content to a desired format more valuable for storage and later use.

Also in 2011, Wongsuphasawat et al. [3] proposed an event sequence analysis method to study the transfer patients within the hospital for quality control. They also had a previously developed system to create huge amounts of data from hospitals and were able to browse among them, but the review tasks were often overlooked. They introduced a visual overview on the event sequences called *LifeFlow* and used the same idea with sequences of time stamped events to create a meta-data of the original log files and present the visualization based on those sequences.

On the other end of the spectrum, some novel techniques have been implemented to present a visual overview of the entire video. *Panopticon* is a parallel video overview system designed and implemented by Jackson et al. [4] to create a more efficient video player having a storyboard of the most important frames in a grid format. Using this system, an overview of the context could be presented to the users with the expense of losing smooth motion from frame to frame.

We first present our approach for creating the reference meta-data of frames and then describe the techniques to visualize and present information to the end user in a fast and user friendly web application.

III. Motivation and Case Study

There is a major concern among physicians and engineers to develop techniques for fall detection and more

This work was supported in part by the Agency for Healthcare Research and Quality under grant R01-HS018477 (Rantz, PI). The content is solely the responsibility of the authors and does not necessarily represent the official views of the Agency for Healthcare Research and Quality.

M. Enayati, T. Banerjee, M. Skubic are with the Electrical and Computer Engineering Department at the University of Missouri, Columbia, MO 65211 (e-mails: mektb@mizzou.edu, tsbycd@mizzou.edu, skubicm@missouri.edu).

M. Popescu is with Health Management and Informatics, University of Missouri Columbia, MO 65211 (email: popescum@missouri.edu)

M. Rantz is with School of Nursing, University of Missouri, Columbia, MO 65211 (email: rantzm@health.missouri.edu)

importantly, fall prevention. In 2008, some regulations and reimbursement policies were established by the Center for Medicare and Medicaid Services (CMS) to encourage the prevention of life-threatening conditions of patient falls in U.S hospitals. They address hospital falls as a principal indicator of a patient's safety which might result in other injuries, increased length of stay and complicated patient treatment [5]. For our work, we have been working with physicians and nurses from University of Missouri Hospital and TigerPlace, an independent living facility in Columbia, MO for years to develop methods to detect, report and prevent falls from occurring in hospital rooms as well as apartments using computer vision techniques.

Based on our latest research in fall detection techniques using the Microsoft Kinect [6] it is now possible to monitor a patient's movement and detect falls. Kinect cameras unobtrusively capture depth videos while taking into consideration a patient's privacy. [1] Videos are stored on local machines as 15 minutes avi files and then transferred to the central file storage over available Wi-Fi network considering the bandwidth limitations. A real-time fall alarm system generates alarms upon detecting a fall to the designated hospital nurses for immediate actions [7].

Our application is designed to go one step further to not only present the fall events in the patient's rooms but also provide an easy-to-use interface for hospital staff and nurses to review what incidents occurred before the fall event Fig. 1. It is imperative to detect falls inside hospitalized areas to provide fast and adequate medical treatments, but it is also vital to review the videos to understand the reason of the fall event and provide some preventive multifaceted or single interventions [8].

Since 2012, we installed pilot systems in 6 hospital rooms and for a duration of 8 months, depth videos were captured and transferred to the central file storage in separate locations for each hospital room. However, no falls were detected at that time. For this application, we installed 6 systems in University of Missouri hospital. In this new setup, after each fall alarm a message is sent to the hospital staff. They can then go back further to review the videos to see what happened and take subsequent necessary action if it was a real fall or they may want to review the typical behavior of that patient, e.g., compare yesterday's behavior with today's when the fall occurred (Fig. 2). In both cases it is not always easy to find the exact temporal location of that fall event especially if there was a lot of inactivity around that time and there were several black frames in the video sequence. Capturing videos 24 hours 7 days a week creates a huge amount of video data for each hospital room. During this continuous data collection, there are situations when the patient is asleep and there is no movement in the scene. In this case, the system automatically replaces actual video frames with blank frames which significantly decreases the size of videos. As a result, each video consists of a series of black and "actual" frames with a reference text file having timestamps and information about the motion content in the depth frame. This information is used to understand more about the data content in each frame. But going manually through videos containing several blank frames would make it difficult for nurses to find the underlying causes for the alarm. The staff member is also likely to miss one very short period of movement between several black frames of inactivity. For our prototype system, we received some complaints from the nurses requesting a better way to find frames having movement among the entire video frames.

Our proposed web-based application provides a handy user interface to easily search among terabytes of depth videos to find the video sequences of interest. We then introduce a novel technique to visualize high level information about these depth frames containing movement or falls. This can allow the hospital staff to keep track of fall events in order to facilitate the process of finding and reviewing the chain of events that led up to the patient fall for post-fall quality improvement process.

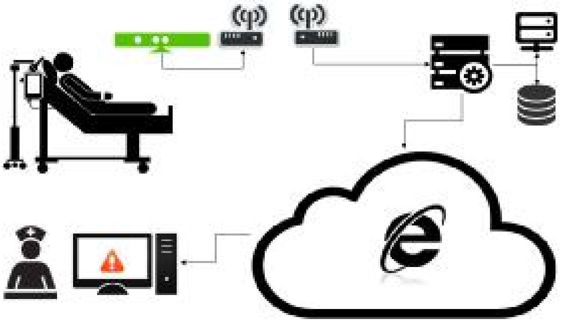


Figure 1. Fall Detection and depth video rewind data flow. Captured videos from Kinect cameras are transferred through Wi-Fi to the file server and a relevant record to the database. Nurses can review alarms and a whole relevant video series over the web application.

Like any other applications with user interfaces, a simple, fast and reliable application can only be created after a complete evaluation of the requirements and via continuous communication with the stakeholders. We first created a demo search and review desktop application and tested the application with nurses engaged in the program. We asked them to do a simple task of finding videos at certain time periods to play with the application and give feedback on what is missing in the application. In the earlier version of the application, the user could input a specific time (year, month, day, hour and minute) for a specific hospital room and the closest video to that time stamp was retrieved and presented. While the simple application was found very useful for reviewing the videos, it still was not found to be efficient in searching through the large amount of data. We struggled with search algorithms to extract a list of videos from the terabytes of video files stored in the file server related to a requested time period. During that time, we received suggestions about moving toward big-data or no-sql databases to resolve this issue. In the end, after many experiments, we ended up with using a MySQL database and a simple mapping table from video date/time to the actual physical path on the file server. The database is updated every time a new video uploaded to the server.

However, after a while we found that the larger the list of videos got, the harder it was to address a specific event. It was then imperative that we provide some contextual information on each video concurrently so that the user can get more information about its content. Also, the feedback from the nurses showed that they could not locate the fall event and they spent a lot of time trying to find the specific location of the fall event. In the next section, we resolved the issue by improving the interface to assist the clinicians to get the information they needed in a quicker and more efficient method.

IV. Method

For every new depth frame captured, the current data capture algorithm compares it with a buffer of previously captured frames within a ten second window. If there is any significant movement detected in the current frame, the frame gets stored in a compressed format in the video file. If not, it is replaced by a blank frame to reduce the video size. In the meantime a record of meta-data about that specific frame is generated and stored in the corresponding text file. This record contains a time stamp and information such as if the frame contains any movements or is blank. In our algorithm, we create a high level visualization of video content in two levels of Frame, if it is blank or contains movement or a fall, and File if contains any movements or falls event.

For the first level it was required to find a way to address frames inside each video, so that the user could easily understand which portion of video filled just by blank frames and which part contains movements or fall events. Having a meta-data file alongside each video file at the create time, eliminates any reprocessing on the video files which could be time consuming. As shown in Fig. 3 while processing the videos the fall detection algorithm [6] classifies each frame into three classes of “Blank”, “With Movement” and “With Fall Event” which we assigned colors “black”, “yellow” and “red”, respectively and the corresponding indicator for each of them is stored in the meta-data file. In the final visualization tool we will provide a value for each frame – timestamp – and present them in a specific color coding based on these class indicators so that the user is able to distinguish different frames at a glance.

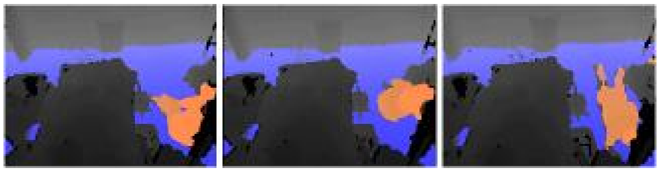


Figure 2. Hospital fall; blue area for ground and the orange object as foreground after using an adaptive background subtraction algorithm.

For the second level it is needed to read the related flags for all the frames and calculate the total number of non-zero flags which is the total number of frames having significant movement or fall events. Comparing this value with a lower bound threshold creates the decision boundary of video classification to “non-important” or “with significant movements” and highlighting video labels in the provided list of videos for ease of access could be done by similar color coding.

V. Design and Development

This application has a two layered design to fully implement separation of concerns. Each layer focuses on a specific portion of the problem and provides or uses services from the other layer.

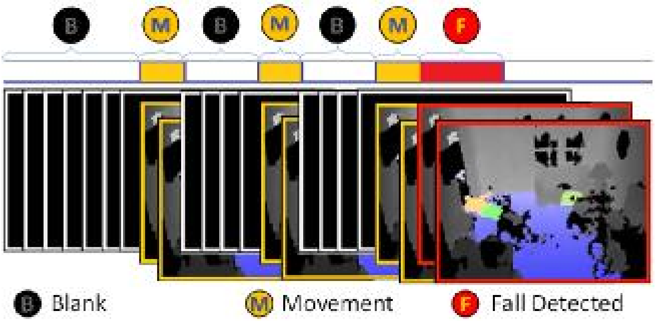


Figure 3. Visualizing frame content in form of colored area-chart using class indicators. Fall detection algorithm [6] is used to classify each frame into “B: Blank”, “M: With Movement” and “F: With Fall”.

A. Back-end

As previously mentioned, video files are transferred to the file server and simultaneously a record will be added for each file to the database. The backend has a two layered service oriented design and provides these services for the frontend: Search database, Retrieve video file and Retrieve meta-data file.

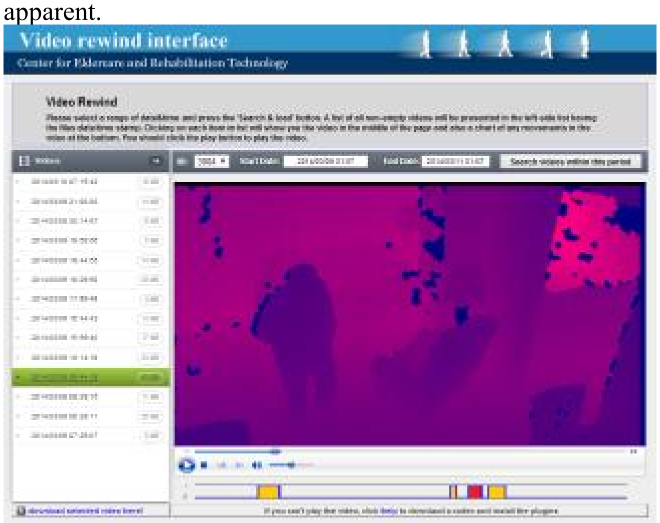
B. Front-end

In the design process, we implemented the latest HTML5 standard and notations to make the application reliable and portable for any device or browser. There were several HTML5 compatible video players that could be easily merged with customized html tags or java script events and we could design a video tracker having meta-data visualizations underneath. On the other hand because of the lossless compression provided by the avi format, all videos have been created by this format. But the only single standard component for playing avi videos on the web pages is the basic Windows Media Player Plugin. So we designed a separate visualization component to present the meta-data file’s content and bind it to the original video player to highlight blank, movement and fall frames and also update video tracker on mouse click.(Fig. 4)

C. Interaction between the Front-end and Back-end

The process of searching for specific videos begins with a user selection of a date-time period; an ajax request sends to a service on the server to extract the list of videos from the database and shows them to the user in the same page. After selecting each file name, a new request is sent to another service on the server to retrieve video file and its meta-data text file. The text file is read using a chart plugin to create a chart describing the content of the video frames.

All these steps are simply accessible via the user interface as shown in Fig. 4 at the top of the page there is a combination of search fields to specify a single room and a period of date & time which will be filled by the user to look for videos around that time. Search requests are sent to the server and the response, which is the left hand side list of available videos, is generated using Ajax. This list also contains some overall information about the videos such as video size and highlighted backgrounds representing the video content. Fig. 4 shows the interface of the system with the video names on the left, the ID information and time and date on the top, and the video in the center. Here the depth images are pseudo colored to make the contrast more apparent.



Video rewind interface
Center for Eldercare and Rehabilitation Technology
Video Rewind
Videos
Start Date:
End Date:
Search videos within this period

Figure 4. The Interface contains search area, result video list, video player and the visualization chart.

Then on selecting each video from the list, another Ajax request will be sent to the server and a comma separated list of meta-data will be returned along with the actual video files to be presented in the chart. As shown in Fig. 4, it is much easier to figure out which part of the video contains significant movements (colored yellow) and which frames contains the fall events (colored red).

D. Improving user experience:

One of the biggest difficulties most of the users had with the first version of the application, were basically related to the video format and compression of the videos as of avi files. Although avi is a very famous and widely used video format mostly used on windows machines, there is just a single old web-based version of Windows Media Player which can play those videos. It's not compatible with all browsers and devices and makes it impossible to use our application from a Mac or Linux machine. To solve this problem and also decrease the size of transferred videos, we compressed converted all videos to MP4 format and used standard HTML5 video players.

All these changes provides a better and faster experience of Depth Video Rewind and makes it possible for a larger group of users to access this web-application using their own devices and browsers.

VI. Conclusion And Future Works

In this paper we presented a visualization technique and a web-based user interface to facilitate the process of reviewing depth videos to make fall preventive strategies based on the meta-data files generated by the fall detection algorithm. Compared to similar works described in the paper, our approach does not require any video reprocessing and all the required meta-data about the video frames cam be created slightly after the video is captured, using the same process for detecting falls. Thus, all of the required data is available whenever a user searches for a video and will be transferred to the client's machine in a fast and reliable method. We believe this is a high performance, low risk method to enable hospital staff to easily review depth videos and provide more reliable conclusions on the reasons of the fall events, to prevent similar cases in the future.

This web-application is designed to work as a global web framework and is compatible to present all kinds of visual information extracted from patient rooms. It is also possible to have a portal page with all the latest fall videos listed in a time oriented grid view so that a nurse could have an overview of all patients in a single page.

More precise researches could run on creating a thumbnail view of important scenes for a specific user. This feature could help the physicians review hours of videos just in a glance.

Acknowledgment

We would like to thank Bradford Harris for all his help and patience in installing pilot systems in hospital rooms. We also would like to thank Tatiana Alexander and Jie Chen for development of the demo versions to receive end users initial requirements.

References

[1] M. Rantz, T. Banerjee, E. Cattoor, S. D. Scott, M. Skubic, and M. Popescu, "Automated Fall Detection With Quality Improvement "Rewind" to Reduce Falls in Hospital Rooms", *Journal of Gerontological Nursing Technology Innovations*, 2014.

[2] M. Guttmann, L. Wolf, D. Cohen-Or,"Content aware video manipulation",*Computer Vision Image Understanding,* v.115 n.12, p.1662-1678, 2011.

[3] K. Wongsuphasawat , J. Guerra Gómez , C. Plaisant , T. David Wang , M. Taieb-Maimon , B. Shneiderman, "LifeFlow: visualizing an overview of event sequences" , *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2011.

[4] D. Jackson, J. Nicholson, G. Stoeckigt, R. Wrobel, A. Thieme, and P. Olivier, "A Parallel Video Overview System". *In proceedings of the 26th annual ACM symposium on User interface software and technology* (UIST 2013), 123-130, 2013.

[5] Sharon K. Inouye, M.D., M.P.H., Cynthia J. Brown, M.D., and Mary E. Tinetti, M.D.,"Medicare Nonpayment, Hospital Falls, and Unintended Consequences", *N Engl J Med*; 360:2390-2393, 2009.

[6] E. Stone, and M. Skubic, "Fall Detection in Homes of Older Adults using the Microsoft Kinect", in press, *IEEE Journal of Biomedical and Health Informatics*, 2014.

[7] E. Stone, and M. Skubic, "Testing Real-Time In-Home Fall Alerts Using Kinect with Embedded Depth Video Hyperlink", in press, *International Conference on Smart Homes and Health Telematics*, 2014.

[8] D. Oliver, J. B Connelly, C. R Victor, F. E Shaw, A. Whitehead, Y. Genc, A. Vanoli, F. C Martin, M. A Gosney,"Strategies to prevent falls and fractures in hospitals and care homes and effect of cognitive impairment: systematic review and meta-analyses", *BMJ, 334:82*, 2007.