Modeling Bed Exit Likelihood In A Camera-Based Automated Video Monitoring Application

Paul Bauer
pbauer@ocuvera.com

Joshua Brown Kramer
jbrownkramer@ocuvera.com

Benjamin Rush
brush@ocuvera.com

Lucas Sabalka
lsabalka@ocuvera.com

Hospital inpatients often fall during an exit from the bed or in the ensuing seconds and minutes. Existing fall prevention technologies fail to provide adequate lead time to a patient bed exit or exhibit high rates of false alarms. To address these limitations and reduce risk of falls for patients and hospitals, we have developed a 3D camera-based system, named Ocuvera, for monitoring patients at risk of falling, without requiring a human monitor. The developed automated system looks for cues that predict a likely bed exit. If the system determines that the risk to patient safety is high, the system alerts nursing staff, often with enough lead time to prevent the exit. In this paper we discuss the algorithmic pipeline of the developed system, starting with the raw camera feed and ending with alarms. Emphasis will be placed on computer vision models of behaviors and objects, as well as a machine-learned bed exit risk model.

Introduction

Patient falls are a common, costly, and serious patient safety problem in hospitals, long-term care, and other care facilities. Approximately 2 to 3% of hospitalized patients fall each year [1, 2] resulting in nearly one million falls in U.S. hospitals; approximately one-fourth of these falls result in injury [1, 3]. Among patients who fall, the cost of care for the 2% of patients who sustain serious injury is nearly $14,000 greater than for patients who do not [4]. Consequently, falls are designated as one of eleven preventable Hospital-Acquired Conditions by the Centers for Medicare and Medicaid Services [5].

As many as 80% to 90% of falls in hospitals may be unobserved [6]. Unobserved and unassisted falls frequently follow unobserved and unassisted bed exits. Thus, one possible approach to reduce falls and fall-related injury in hospitals is to decrease the likelihood of unassisted bed exits.

Existing approaches to fall injury reduction have well-known shortcomings, as described in the Related Work section, including the inability to accurately and prospectively predict bed exits with enough lead time for healthcare professionals to respond and meet the patient's needs. This technological inadequacy presents a significant barrier to progress in reducing fall-related injury.

In this paper, we introduce a new method for monitoring hospital inpatients for fall risk. This method provides an ability to predict bed exits with adequate lead time for a response, and thereby potentially allowing nursing staff time to prevent an unattended bed exit and prospective subsequent fall. This prediction is done by detecting conditions or trends that indicate the patient may be attempting, about to attempt, to exit the bed.

The structure of this paper is as follows. We discuss Related Work of others to prevent falls among hospital inpatients. We introduce our new method for monitoring patients and predicting bed exits. Finally, the results are discussed and directions for future research are given.

Related Work

Interventions intended to decrease the risk of patient falls come in a variety of forms. As no single intervention has been proven to reliably decrease the risk of unattended bed exits and associated injury in patients at high risk for falls, these interventions must be implemented in conjunction, targeted at a patient's specific risk factors to be effective [7, 8]:

- Fall injury mitigation: Floor mats and low beds may reduce injuries from falls, but do not prevent unattended bed exits or alert hospital staff that a patient is attempting to exit the bed.
- Fall detection: Technologies designed to detect when a fall has occurred, including many wearable systems, but cannot prospectively notify personnel of a patient's intent to exit the bed.
- Patient restraints: Use of four bed rails (both upper and lower rails) and patient restraints are associated with an increased rate of injury without decreasing fall rates [9].

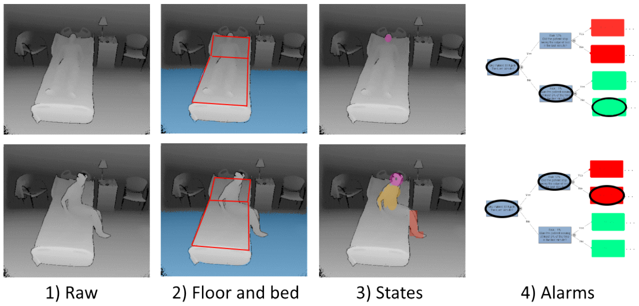

Figure 1. Our pipeline. We start with a 1) A raw depth frame, initially compute 2) A scene including bed and floor, then 3) Patient states, and 4) We fuse our state history with a decision tree to make an alarm decision

- Patient monitoring: Closed circuit or remote video monitoring can predict and detect unattended bed exits, but rely on additional, dedicated staff who are vulnerable to attention fatigue. In-room sitters–people who sit with the patient twenty-four hours a day–are frequently used for real-time monitoring, but there is little evidence that their use mitigates fall risk [8, 10].

- Bed exit detection: Pressure pad alarm systems or "virtual bed rails" can detect bed exits, but have high false positive rates, which contributes to alarm fatigue and delayed response by hospital personnel [11].

- Bed exit prediction: Pressure pads that detect changes in posture may alert staff that a patient is planning to exit the bed; however, existing technologies have high false positive rates [11].

- Fall risk reduction training: Focusing on adaptive training by improving the standardization, planning, and real-time adjustment of the human response to fall risk can decrease fall rates [16]. However, without improved assistive technology, gains in fall risk reduction are limited by the technology used itself.

The prevalence of these limitations is consistent with the fact that little progress has been made in decreasing fall-related injury as a healthcare acquired condition [12]. To challenge the existing paradigm, we must address the limitations in existing fall prevention technologies. To prevent falls, technology should notify hospital personnel prospectively so that there is sufficient response time for a healthcare professional to assess and assist the patient.

Methods

In this section we will discuss our algorithmic pipeline, with an emphasis on the modeling aspects.

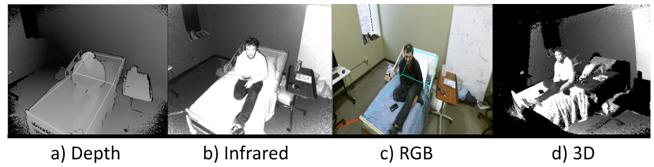

Figure 2. Image streams available from Kinect: a) depth, b) infrared and c) RGB. Image d) illustrates a 3D model of the scene that can be reconstructed from the depth information.

Raw Signal

Our algorithms are built on top of the raw data streams available from Microsoft's Kinect for Xbox One sensor [13]. The Kinect provides the following basic raw data streams: a 1080p RGB video feed, a 512×424 Infrared (IR) video stream (which can be used for night vision), a 512×424 depth stream, and a stereo audio stream. See Figure 2 for an illustration of the available image streams.

A depth image from the Kinect is a 512×424 image where the value at a given pixel represents the depth from a plane containing the camera perpendicular to the focal axis, in millimeters. For example, if the camera were directly facing a wall 1 meter away, every pixel on the wall would have a value of 1000, plus or minus some noise. This information can be used to reconstruct a 3D representation of the scene (See Figure 2d).

Microsoft provides the Microsoft Kinect for Windows SDK that computes a variety of potentially useful signals [14]. Most notably, it computes an estimate of a skeleton for people in view of the Kinect depth camera. While some research suggests this data can be useful [15], we have found this skeletal data to be of limited use for monitoring patients in inpatient settings. Skeletal data from the Microsoft Kinect for Windows SDK is most reliable when people are standing far from nearby obstructions and backgrounds. Inpatient subjects are typically supine or seated, are close to background objects such as beds and chairs, and are frequently occluded by staff or equipment.

Computation of States

Using the depth video, we estimate several states that are important for predicting bed exits. The states we use were chosen over a period of time in a highly iterative and hands-on modeling process. That is, we started with a simple predictive model built on a small number of states, tested our predictive ability, looked to determine what states might improve prediction, added them, and repeated.

Floor Finding. First, we find the floor. We do this by using a heuristic filtering step that selects many points in the frame as possible floor points. Then we use RANSAC, a well-known algorithm for finding shapes in points clouds, to find a large plane in that set.

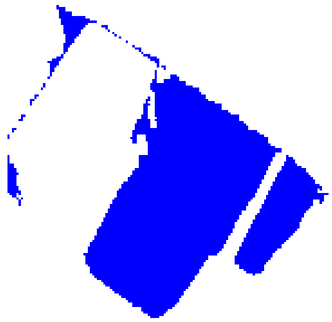

Figure 3. An overhead view of the scene used in determining the bed location. Blue points represent space that might contain an object of the right height and thickness to be part of the bed.

Bed Finding. Next we find the bed. In simple terms, there are two steps: bed location finding and bed modelling. In the bed location phase, we first find all the points on the floor that might be below the bed, in the sense that they could be under an object of the right height and thickness (see Figure 3). We then find all rectangles on the floor that are the size of a bed and contain nothing but those potential bed points.

Next, we pass each rectangle found into the bed modeler. The modeler closely examines the point cloud to find the foot of the bed, then the plane of the main segment of the bed, and finally the head of the bed (which can be at an angle, since most hospital beds bend for the convenience and comfort of the nurses and patients.

We essentially compute a fitness function for each bed model generated this way and take the best one. In practice we reduce computation time by first computing the fitness for a sparse set of bed locations, and then focus on a more fine-grained subset near high fitness models. The fitness function is meant to approximate the likelihood that the bed we have modeled would appear as it does in the depth frame. Essentially, the further the distance of points above the bed from the model, the lower the fitness, and if there are points that appear behind the supposed surface of the bed, this an obvious error that incurs a large fitness penalty.

After initially computing the floor and bed, we continually monitor to determine whether their positions should be re-computed.

Machine Learned States. In addition to the floor and bed, we also look in the scene for certain body parts, objects and poses. Several of these are estimated using machine-learned shape models developed by our team. For example, we have a classifier that detects heads. For each pixel in the image, the classifier determines whether or not it is part of a head. This classifier is a decision-tree whose splitting rules are basic geometric questions. Other examples of machine-learned shape models are classifiers that estimate the torsos of people facing the edge of the bed and the torsos of people leaning into the bed. Another example of body part estimation being used in applications is by Microsoft, which used body part estimation in developing the original Kinect SDK [16].

In addition to these per-pixel classifiers, we have learned convolutional neural networks for detecting more holistic states, such as whether or not a sequence of frames contains a bed exit.

People Tracking We monitor the location of people in the room. This is done by combining the foreground of the image (i.e., those points that are in front of a learned background) with machine-learned heads, and comparing the history of frames to determine which people in past frames correspond to the people in the current frame. Significant effort and computation go into keeping track of people over frames and into distinguishing between various people, especially as hospital personnel frequently physically interact with patients.

Other States For each person in the scene, we compute a variety of states relevant to the likelihood of bed exit. These include whether the person is sitting up, facing the edge of the bed, in the bed, or on the floor, whether their leg is hanging off the bed, whether they are near the edge of the bed, their level of activity, and so on. These states are primarily hand-written (rather than machine-learned) and largely rely on looking for certain sets of pixels or geometric shapes in certain locations with specific properties relative to the bed.

Alarms

Our system is designed to predict a patient bed exit and and trigger an alarm before one occurs. These predictive alarms rely on an explicit model of the risk of an impending bed exit. In addition to these predictive alarms, we throw an alarm if the patient exits the bed.

Each alarm relies solely on the resulting states discussed in the Subsection Computation of States above.

Predictive alarms. Our predictive alarms work by constantly assessing the current probability of a bed exit in the next 5 minutes and firing an alarm if that risk ever rises above a configurable threshold. The model we have chosen for this probability is a decision tree, where individual questions are about the recent history of a single state. See Figure 4 for an example of the type of tree we have learned. We chose decision trees over other model types (we considered Naive Bayes, Neural Networks, and Hidden Markov Models among others) because they achieve our performance goals and are easy to explain: our alarming policy can be read off in natural human language directly from the tree.

We currently ask questions about the states enumerated in the Computation of States section. We can also incorporate questions about the time of day; intuitively, a patient who moves a lot during the night is more likely to exit the bed than a patient moving around a lot during the day. We anticipate

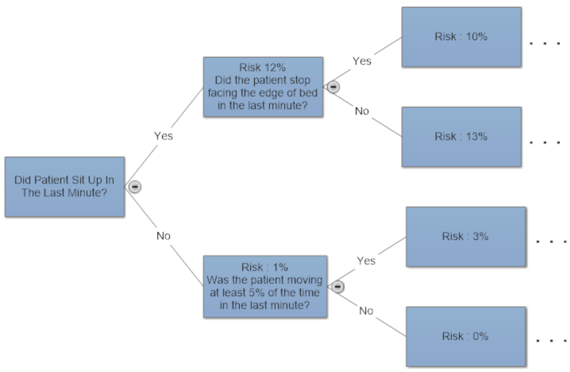

Figure 4. An example of the type of decision tree used to determine whether or not to predict a bed exit.

using diagnosis, medication, evaluation results, integration with other systems, and other such data in the future. The questions we ask can have the following structures.

- Is a state equal to a particular value? For example, is the patient sitting up?
- Did the state change from one particular value to another recently? For example, did the patient go from not facing the edge of the bed to facing the edge of the bed in the last minute?
- Has the current state been stable for some period of time? For example, has the patient been near the edge of the bed for at least 10 seconds?

There are other variants on these types of questions that are meant to be more general and to handle occasional state errors more robustly. In particular, we can ask about windows of time that don't include the current time ("did the patient hang their leg off the bed between 60 and 90 seconds ago?") and we can ask questions such as "has the patient been alone in the room for 90% of the time in the last two minutes?"

We learn our trees with a variant on a standard decision tree learning algorithm. Specifically, when choosing the first question to ask, we randomly generate a large pool of questions of the form described above, and choose the question from that pool that maximizes the information gain with respect to the question "will there be a bed exit in the next 5 minutes?" To get our left subtree, we repeat this procedure on just the data where the answer to our learned question is no. Similar for our right subtree.

There are some implementation details that might be of interest. We are able to try many questions quickly by 1) Computing states once and re-using them rather than running the entire computer vision pipeline for every possible question, and 2) Carefully implementing the questions to run in time proportional to the number of state changes, rather than the number of frames of video. Also of interest is that while we want to alarm on bed exits that occur while the patient is unattended, we found that our models generalized better if we trained on all bed exits, attended or not. We speculate that this improvement is due to 1) the resulting larger training set and 2) changing the responsibility of the learner from determining whether the patient would exit the bed ALONE to the easier question of whether or not the patient would simply exit the bed (alone or not).

Bed exit alarms. These alarms are designed to trigger when we detect a bed exit. The exact circumstances under which it fires can be configured on a per-patient basis and this configuration can affect the sensitivity and false positive rate of the alarm. For example, we can suppress bed exit alarms when the patient was not seen facing the edge of the bed. This allows the system to better distinguish, for instance, a patient exiting the bed from a nurse leaning into and then out of the bed to fix the sheets. On the other hand, this filter suppresses alarms in the rare, but real, circumstance that a patient exits the bed without sitting up at the edge of the bed.

Per-patient alarm configuration. We allow the alarms mentioned (Predictive alarms, Bed exit alarms, and other alarm types) to be individually enabled or disabled, as well as further configured, for instance by adjusting the risk level threshold, on a per-patient basis.

Experimental Environment

The described method was developed and tested using anonymous depth video data collected from multiple hospital sites. Said data provide the ability to retroactively and repeatedly train and test methods and algorithms using real patient behaviors.

Results

We present results of our algorithmic pipeline. The true efficacy and effectiveness of a fall prevention system is ideally determined by a clinical study determining its impact on fall and bed exit rates. For developing and testing our system, we indirectly estimate our impact on these rates by simulating alarm behavior on previously recorded anonymous depth video. We have randomly assigned this recorded data into two groups: video we observe to develop our algorithms and models, and video that we only use to measure our system. The results reported here are for the second group, which consists of roughly 13000 hours of video from 110 patients.

Our main metric is sensitivity: the percent of unattended bed exits that we alarm on. The main assumption we have made is that the number of falls after a bed exit is proportional to the number of unattended bed exits.

We also recognize that nursing staff suffer from alarm fatigue, which can affect alarm response time, so we maintain a metric that captures the believability of our alarms. This metric is positive predictive value (PPV), i.e., the percent of alarms that are true positives.

Our third metric is lead time: the number of seconds before a bed exit that our system initially sends an alarm.

There are some subtleties in these definitions. What does it mean to "alarm on" on a bed exit if the alarm is predictive and occurs a minute or more before the actual exit? Should we count a predictive alarm as a true positive if the patient showed every sign of exiting the bed, but changed their mind, or was interrupted by a nurse? The choice we have made is to count alarms in the 5 minutes proceeding a bed exit and the five seconds afterwards as true positives, unless a human reviewer has determined that there were no signs of an intention to exit the bed at the time of the alarm. Furthermore, we count as a true positive any alarm thrown while the patient is doing behavior associated with an elevated risk of a bed exit, such as sitting at the edge of the bed while alone. In general, we take a conservative approach to grading such subtleties and therefore our results tend to represent a worst-possible grading. For example, imagine a patient exits the bed multiple times in short succession, and we alarm only on the first exit (perhaps because the system doesn't believe that the state has changed enough to warrant a second alarm). We count only the first exit as alarmed on in our sensitivity stats, even though the subsequent bed exits would likely have been prevented.

We report our sensitivity and PPV in the case where the patient is alone, anticipating that alarms will be disabled when nursing staff enter the room, either automatically by our system when detected, automatically by communication with a third-party nurse-location sensor, or manually by pressing a button to temporarily disarm the system.

Finally we report the stats with filters off (such as the filter described above, based on whether the person was recently seen facing the edge of bed). Turning filters on generally decreases sensitivity and increases PPV.

Sensitivity	PPV	Average Lead Time
95%	54%	45 seconds

Table 1
Conservative results from 110 patients.

Discussion

Here we discuss some limitations of our current metrics as well as directions for future research.

Subsampling. We do not run all of our video to compute our statistics; it would be prohibitively time consuming. Instead, we randomly sample 2% of our video and use that to estimate our PPV. The estimate given above is therefore unbiased, but likely differs somewhat from the PPV we would measure if we were to compute it exactly over all of our video. We run our algorithms against video of all bed exits, so the sensitivity is the exact sensitivity for the data we have collected.

Different population. The population of patients in our database may differ from the population that will use our system in practice. In particular, we have attempted to limit our data to patients who are at an elevated risk of fall, but the procedure for deciding who is at an elevated fall risk might be different at different facilities.

Alarms may change behavior. We passively recorded the video that our system is developed and tested on. It seems likely that activating our alarms will change patient and nurse behavior, which could have an affect on the metrics we are computing, including reducing the number of unattended bed exits.

Staff will have instant video feedback. One advantage of the technology presented is its ability to provide RGB, IR or depth images or video along with alarms. This provides nurses with additional context for the alarm, i.e., video of what the patient is currently doing. This will likely reduce the impact of a false positive–nurses will quickly ascertain that attending to the patient is unnecessary in a particular situation–and hasten response to a true positive–nurses will also be able to determine how important it is to respond quickly in a given situation. While this additional context is valuable, how it should be delivered to staff presents a challenge. Most existing alarm technologies do not provide nursing staff with devices that have the ability to display video, and the introduction of a new, separate device may not be optimal for staff.

Logistical issues. Furthermore, there are likely to be logistical issues that affect the efficacy of the system in practice. In particular, in discussions with hospitals and nursing staffs, several challenges have been anecdotally identified with delivery and response to alarms: staff is very busy, there are already many alarms from many different sources, existing alarm technology can be difficult to integrate with, technology and preferences are different on a per-hospital and even per-unit basis, and staff does not want to carry additional devices because of the inconvenience. These challenges must be overcome in order to deliver alarms to the right person, at the right time, and provide enough lead time to intervene and prevent an adverse outcome.

Future Directions. We have the automated video monitoring system active in several hospital rooms, and staff response has been quite positive. Based on this feedback, we believe Ocuvera is currently found to be useful by nursing staff. We continue to improve our predictive algorithms and the basic sensors that feed into them. Additional future use cases include monitoring of patients in chairs, as patient chair exits may also lead to falls.

References

[1] D. Oliver, F. Healey, and T. Haines, "Preventing falls and fall-related injuries in hospitals," *Clinics in Geriatric Medicine*, vol. 26, no. 4, pp. 645–692, 2010.

[2] J. Mahoney, "Immobility and falls," *Clinics in Geriatric Medicine*, vol. 14, no. 4, pp. 699–726, 1998.

[3] E. D. Bouldin, E. M. Andresen, N. E. Dunton, M. Simon, T. M. Waters, M. Liu, M. J. Daniels, L. C. Mion, and R. I. Shorr, "Falls among adult patients hospitalized in the united states: prevalence and trends," *Journal of Patient Safety*, vol. 9, no. 1, pp. 13–17, 2013.

[4] C. Wong, A. Recktenwald, M. Jones, B. Waterman, M. Bollini, and W. Dunagan, "The cost of serious fall-related injuries at three midwestern hospitals," *Joint Commission Journal on Quality and Patient Safety*, vol. 37, no. 2, pp. 81–87, 2011.

[5] C. for Medicare and Medicaid Services, "Hospital-acquired conditions (present on admission indicator)." http://www.cms.gov/HospitalAcqCond/, 2016. Accessed: 2016-08-30.

[6] J. M. Morse, *Preventing patient falls*. Sage Publications, 2000. 2nd edition.

[7] L. Currie, "Fall and injury prevention," in *Patient Safety and Quality: An Evidence-Based Handbook for Nurses* (R. Hughes, ed.), ch. 10, Agency for Healthcare Research and Quality, 2008.

[8] J. Degelau, M. Belz, L. Bungum, P. Flavin, C. Harper, K. Leys, L. Lundquist, and B. Webb, "Prevention of falls (acute care)." https://www.icsi.org/_asset/dcn15z/falls-interactive0412.pdf, 2012. Institute for Clinical Systems Improvement, Accessed: 2016-08-30.

[9] J. V. Agostinie, D. I. Baker, and S. T. Bogardus, "Prevention of falls in hospitalized and institutionalized older people," in *Making health care safer: a critical analysis of patient safety practices*, ch. 26, Agency for Healthcare Research and Quality, 2001.

[10] A. D. Harding, "The cost of serious fall-related injuries at three midwestern hospitals," *Observation assistants: sitter effectiveness and industry measures*, vol. 28, no. 5, pp. 330–336, 2010.

[11] R. I. Shorr, A. M. Chandler, L. C. Mion, T. M. Waters, M. Liu, M. J. Daniels, L. A. Kessler, and S. T. Miller, "Effects of an intervention to increase bed alarm use to prevent falls in hospitalized patients: a cluster randomized trial," *Annals of Internal Medicine*, vol. 157, no. 10, pp. 692–699, 2012.

[12] "Interim update on 2013 annual hospital-acquired condition rate and estimates of cost savings and deaths averted from 2010 to 2013." http://www.ahrq.gov/professionals/quality-patient-safety/pfp/interimhac2013-ap2.html, 2014. Agency for Healthcare Research and Quality, Accessed: 2016-08-30.

[13] "Kinect for xbox one." http://www.xbox.com/en-US/xbox-one/accessories/kinect, 2014. Microsoft, Accessed: 2016-08-30.

[14] "Meet kinect for windows." https://developer.microsoft.com/en-us/windows/kinect, 2014. Microsoft, Accessed: 2016-08-30.

[15] B. Hotrabhavananda, A. K. Mishra, M. Skubic, N. Hotrabhavananda, and C. Abbott, "Evaluation of the microsoft kinect skeletal versus depth data analysis for timed-up and go and figure of 8 walk tests," *Proceedings, IEEE Conference of the Engineering in Medicine and Biology Society*, 2016.

[16] J. Shotton, A. Fitzgibbon, A. Blake, A. Kipman, M. Finocchio, R. Moore, and T. Sharp, "Real-time human pose recognition in parts from a single depth image," in *CVPR*, IEEE, June 2011.