Towards Automated Workflow and Activity Recognition at a Pediatric Intensive Care Unit

Colin Lea & Jim Fackler, Greg Hager, Russ Taylor

Abstract— **In this work we investigate techniques for developing a system for automated workflow and activity recognition in an Intensive Care Unit using a set of RGB-D cameras. The ultimate goals are to increase safety, optimize workflow, increase resource allocation efficiency, and distribute activities based nurse skill. For this project we learn to automatically recognize a set of 7 high-level activities. We take a three step approach to classifying actions from depth images. People are extracted from the images by background subtraction and then segmented into temporal action sequences based on their center of mass. A 23-dimensional feature vector is generated from each action sequence. We evaluate this with two types of discriminative classifiers and achieve around 60% accuracy on two datasets with 5 and 7 actions per set. Given that this is the first work doing automated recognition using 3D cameras in an ICU, we provide thorough discussion of potential future research directions.**

I. Introduction

An Intensive Care Unit (ICU) is a hectic place. Dozens of staff members come in and out on a regular basis and perform a large number of small but important tasks necessary for a patient's proper recovery. Estimates from doctors at the local Pediatric ICU indicate that there may be around 800 micro-tasks completed during the average patient's stay. Enumerating all of these tasks is a problem of it's own and evaluating when an individual task is completed provides additional complications. Some typical tasks include giving medicine, checking diagnostics, inserting IVs, performing arterial-line insertions, emptying chest tubes, and documenting vitals.

Our goal is to develop a system for automatic monitoring of personnel activities using a set of 3D cameras placed in an Intensive Care Unit. There are four key benefits that this type of system could have which benefit both patients and the hospital. (1) Safety can be increased by checking if certain activities are completed, such as giving medicine at the appropriate time. (2) Workflow can be optimized by determining which activities are the most time consuming and spread them out. For example, sometimes there are multiple nurses in a room doing different activities. It may be more efficient to have both actions be completed by the same person. (3) By monitoring the needs of each patient, the hospital can efficiently allocate the number of staff members and resources on site. This could prevent problems of over- or under-staffing. (4) Additionally, the quality of the nurses could be measured to determine how well they work with different types of patients. This could increase the quality of care by partnering nurses with specific patients that better suit their skills.

There are a number of studies in the medicine literature with regards to nurse workflow. If our work is expanded to identify individual people then it would be possible to evaluate the cognitive load of hospital staff and potentially increase the quality of working conditions. For further reference to cognitive load see [1], [2], [3] and [4].

Fig. 1. A frame from one of our experiments at the Pediatric ICU. Shown are two personnel on the right and back and the patient in the middle.

While automatically determining activities with a fine granularity would be very useful, it is unrealistic solely using camera data. For example, it is clear that a low resolution camera would be able to detect when a nurse gives a patient a specific medicine. However, the general idea of giving medicine may be possible. In this exploratory work, we pick a set of five to seven actions common in an ICU. These include actions like emptying urine tubes, minor procedures on the patient, and checking diagnostics.

It is important to note that with the aid of other sensors more refined detection my be possible. For this preliminary study we chose to use the XBox Kinect for its potential to detect scene-level activities. The Kinect captures color and depth imagery resulting in a colored 3D point cloud. Unfortunately, for this study access to the color data is limited due to confidentiality reasons. While we do not currently use the color data, we anticipate using it in the near future to extract color features to help discriminate personnel in the scene.

It is worth noting that there are multiple facets to automated activity monitoring. There are two key directions: retrospective and real-time analysis. The retrospective approach aims to detect personnel actions throughout the course of a patient's stay and is helpful for the aforementioned reasons. The other component, real-time analysis, is useful for problem prevention. This could be done to track the

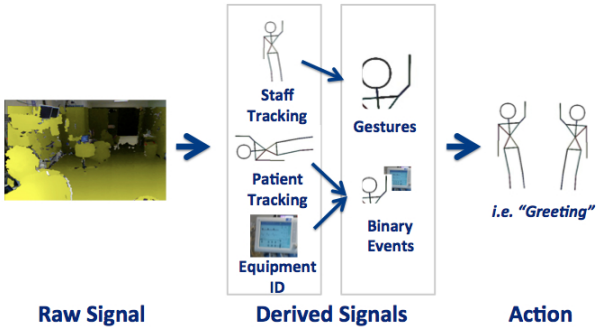


Fig. 2. The pipeline includes three key elements: (left) recording data from RGB-D cameras at the ICU, (middle) extracting features based on cues from our body tracking, and (right) classification of each segment into high-level actions.

patient and alert nurses if they are moving too much or getting out of their bed. We saw during our second data collection that this is a real problem. The night before we setup, the toddler ripped out a tube sown into him which caused complications and ultimately more work for doctors. This is also a problem in adult ICUs where patients will attempt to leave the room without permission. They may be heavily medicated and feel fine but are actually in a worse condition that they think. This may result in the patient falling and injuring themselves even more.

Our pipeline takes the following approach, as shown in figure 2. In the ICU, we record several hours worth of footage using a set of RGB-D cameras. From this data we then create a lower dimensional feature vector for activity recognition. This comes in the form of derived signals. Using the depth images we are able to extract people using a straight forward background subtraction technique. Features such as body position are gathered. Action sequences are created by tracking these segments over time. We feed features from these sequences into two different classifiers to output the label of each action.

In Section II, we present related activity recognition work. Sections III through V detail our segmentation algorithms, feature descriptors, and recognition methods respectively. Section VI shows results based on the two datasets that we gathered. Finally, Section VII closes with our thoughts and ideas for future work.

II. Related work

Video-based activity monitoring has become a prevalent research topic over the past couple of decades. Significant work has been done using different variations of Hidden Markov Models including [5], [6] and [7]. Spatio-temporal techniques [**?**] have proven moderately successful for video with a large number of classes. Context free grammars and hierarchical graphical models that encode sub-actions into "processes" have also been shown to provide good results [8].

Similar work to us has been done in a mock operating room (OR) at the Technical University of Munich, in the "Aware Home" at Georgia Tech, and in the Quality of Life Center "Kitchen" at Carnegie Mellon University. At TUM, Padoy *et. al* employs "Workflow HMMs" to encode a probabilistic flowchart-like design to determine things like the time remaining in a procedure [7]. At the Aware Home, Muhammad *et. al* uses unsupervised based methods with suffix trees to detect everyday activities. In the CMU Kitchen, researchers attempt to answer similar questions to us, however they use a greater variety of sensors including egocentric cameras and inertial measurement units (IMUs) [9].

The recent advent of inexpensive 3D imaging systems like the XBox Kinect has made it more cost effective to work with 3D data. Using 3D data helps some of the problems that algorithms using traditional 2D cameras face. Topics like segmentation become easier because of the additional depth information. There are a number of recent activity recognition projects using the Kinect such as [10] which uses it for in-home elderly fall assessment.

Recently there has been increased interest in gesture recognition. In Section VII, we discuss an approach we implemented that has similarities to [11]. In this work they develop a technique for learning arbitrary gestures for use in an Operating Room. There is a competition run by a group of prominent computer vision faculty at CVPR 2012 to develop the best One Shot Learning gesture algorithm. One Shot Learning is a technique where only one example is used to train a classifier. Preliminary results from the competition have been released [12]. Some of the leading competitors use techniques such as Hidden Markov Models, Dynamic Time Warping, and Histogram of Oriented Gradients (HOG)/Histogram of Oriented Flow(HOF) features.

Many of these gesture techniques require knowledge of the skeletal information, such as the positions of the hands and the head. The OpenNI Kinect drivers come with the NITE module which gives skeletal information as described in [13]. After collecting data in the ICU we found that this software did not work due to the high number of occlusions in the room and the variability in staff pose. Normaly when using the Kinect, a user initiates their skeleton by distinctly posing in front of the camera. It is not feasible to do this in our setting. The work in [14], which uses geodesic extrema on the body as seeds for the body pose estimation, has laid the groundwork for a number of other related works. Our progress in this area is described in section VII.

III. Segmentation

Our goal is to determine what action each person is performing at any time. As the first step in accomplishing this we employ two types of segmentation: image-based and temporal-based. To isolate each person and their corresponding actions we must setup a method of detecting people and tracking them over time.

Our first step towards identifying people was to perform an experiment to see how effective the built-in skeletal tracker from Microsoft works in the ICU. In good conditions this tracker will output a set of joints corresponding to positions

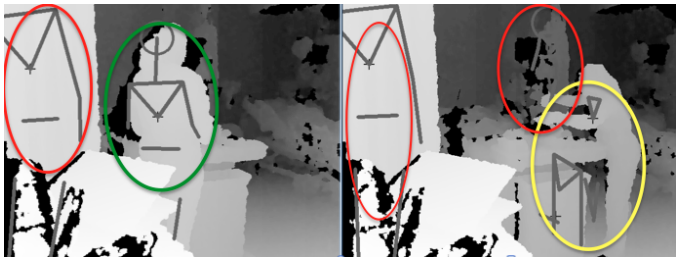


Fig. 3. Examples of Skeletons generated from the Microsoft tracker. Red means a bad skeleton, green means good, and yellow means, correct detection but bad pose estimation

of body parts including hands, the head, chest, and others. Our results show that performance in the ICU is lackluster, as depicted in Figure 3. The four biggest problems are as follows: (1) The skeletons commonly initialize a user incorrectly if they walk into the scene sideways. (2) The curtain on the left side of the image is depicted as a skeleton for a large part of the multi-hour recording. (3) When people walk next to other objects, like track cans, the skeleton sometimes merges together with the other object. There was even a case where a person left the scene and the skeleton was still detected on the other object. (4) Lastly, monitors in the background are sometimes miss-classified as people.

In orders to track people more robustly in the ICU we employ our own methods of segmentation and tracking.

A. Image Segmentation

Two methods of background subtraction were explored to detect people in our scene. The first used a spectral clustering method and the second used only a connected-components method with a gradient mask. The simple connected components method gave better results and was ultimately used. While this appears to be an easy problem, local variations in the image make it more difficult. Each person may have depth values ranging by about a quarter meter, so simple thresholding methods do not work.

Both techniques begin by differencing the current frame and a model of the background. The background is generated in three steps: (1) First, a set of five person-less images is averaged. (2) The Kinect has a large amount of pixel noise which is stored as NaN and Infinity in the image. To compensate we fill in error spots with their nearest neighbors. (3) Lastly, minimum and maximum thresholds are set to eliminate values that are realistically too close or far from the camera. This process is represented in Figure 4.

1) Cluster-based method: A technique using Density-based Spatial clustering of Applications with Noise (DBSCAN) was developed to attempt to segment people in the scene. The pixels of each human are clustered together in 3D-space thus we expected that this type of technique would work well. DBSCAN was chosen because, unlike K-Means and many other clustering algorithms, it does not rely on specifying a number of nodes. The implementation used is available in the SciKits-Learn machine learning library for Python. Our technique is as follows:

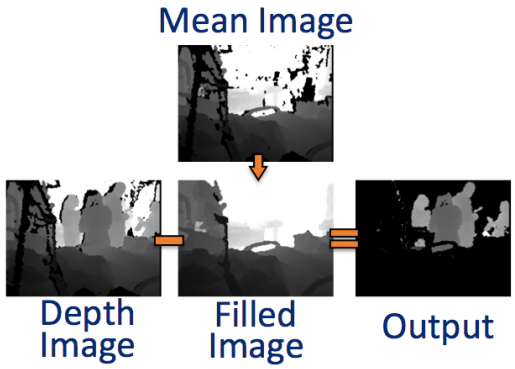


Fig. 4. The background subtraction technique starts by removing a background model of the image. This model is comprised of a mean image whose noise has been filled in using a nearest neighbor technique.

1) Morphological opening filter
2) Median Blur
3) Compute histogram
4) Fit clustering technique
5) Find connected components
6) Extract components above pixel count threshold

The output is a set of candidate people. While this technique works very well in many cases, it does not always find all of the people. Additionally, sometimes multiple people are segmented as one person when they are standing at a similar depth. This is a problem because there are often people standing at a similar distance.

2) Gradient-based: The second approach simply attempts to mask the outline of people based on local gradients. The idea is that if a gradient is large then it is likely a border between objects.

Gradient-based Technique:

1) Calculate gradient of image
2) Mask pixels that have too large of a gradient
3) Morphological filters
4) Find connected components
5) Extract components above pixel count threshold

In practice this approach works better than the cluster-based method. The downside is that to accurately disam-

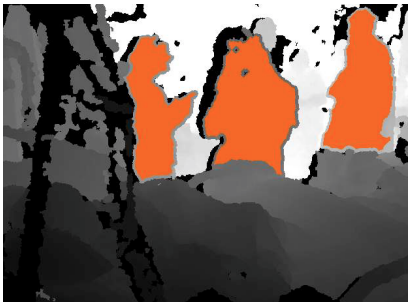


Fig. 5. The three people in the room are accurately segmented using a gradient-based approach.

Fig. 6. Example image and temporal segmentation from the second dataset. The orange part represents the image segmentation at this frame, the red square is the current center of mass, and the blue line is the center of mass at each time step in the sequence.

biguate people a large enough gradient mask must be used. Thus, the outline of a person is generally a few pixels inset from the actual boundary. An example segmentation is shown in Figure 5

B. Temporal Segmentation

To keep temporal consistency, each segment is grouped into an action set based on its center of mass. The problem here is that there are multiple segments at any one time and people are coming and leaving at different times. Additionally, people may move a substantial amount between frames which can try to throw off the tracker. To account for this we calculate a distance matrix between the segments in the current frame and people in the previous frames. The idea is that the distance between people in two frames is correlated and thus the closest new segment to a previous segment should be put in the same sequence.

In practice, there are problems with noise. Sometimes the center of mass is thrown off by a substantial amount, thus positions in new segments are compared with the moving average of the center of mass over the past three frames. Additionally, new segments are added to sequences if their distance to a previous person is less than 0.5 meters and the last person was seen within 5 frames.

The blue line in Figure 6 represents the nurse's position at all points of time in this particular sequence.

IV. Feature Extraction

Our approach aims to classify each sequence based on the prominent action taking place. The classification techniques in mind requires a static number of variables so there were several considerations to take into account. One trouble is that there are a variable number of people in a room at any point. Oftentimes the activities of these people are related. For example, there are times in our second dataset where two people were working together to perform the same procedure. In the first dataset it was common for nurses and parents to talk around the patient's bed. We want to incorporate these variables to improve the accuracy of our system.

In the end we developed four types of features: summary statistics, virtual touch sensors, orientation-based, and interaction coefficients. These amount to 23-dimensions as shown in Table I.

TABLE I

Features extracted from the depth images

Component	# Dimensions
Arc Length	1
Arc Velocity	1
Center of Mass	1
Touch Sensors	2
Orientation Histogram	12
Interaction Angles	4

A. Summary Statistics

Each time step of an action sequence includes two important pieces of information: center of has and current time. We use these to form three useful features that summaries activity. Path length is calculated as the integral of changes in center of mass over time. The arc velocity is simply the path length divided by the sequence duration. Lastly, the averaged center of mass is also included. Note that in practice we also tried using the frame count. However, we saw in the Decision Forest that this had a 0% importance weighting.

B. Virtual Touch Sensors

Looking through the data it is apparent that one of the distinguishing factors between whether a nurse is performing a procedure or checking vitals is how long these spend at either the head or foot of the patient. Drugs are generally inserted into tubes that are on the bottom half of the person's body. Using this knowledge we create vital touch sensors. These are activated if the bounding box of personnel comes within a certain radius of the sensor. The total number of activations for each sensor is used as a feature. The circles in Figure /refTouchSensors depict the locations of our two touch sensors in dataset 2.

C. Histogram of Orientations

Knowing the orientation of a person can be very useful because it gives an estimate of where they are looking and what they are doing. For instance, a nurse could be in the

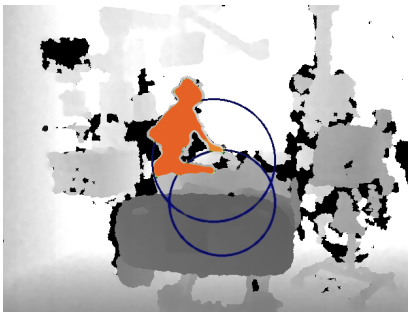
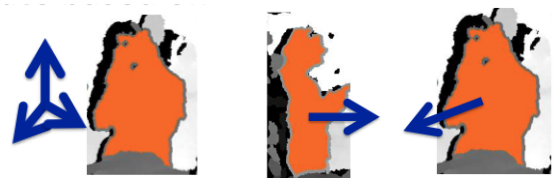


Fig. 7. Virtual touch sensors are used to see if a nurse is interacting with the upper or lower part of a patient's body.



Orientation Estimation Interaction Coefficient

Fig. 8. (Left) A segment and it's corresponding orientation vectors (Right) The forward vectors for two segments in a scene

same position when doing a procedure or looking at the video monitors. The key difference in the direction in which they are facing. We have found that we can calculate a rough estimate of orientation by using Principal Components Analysis. The input to PCA is the 3D information for a given segment. The output is a set of 3 orthogonal directions corresponding to the orientation. The vector associated with the largest eigenvalue is generally directed upwards. This is because there is the most variation in the vertical axis. The second eigenvalue is generally the direction the person is facing. Note that sometimes the second and third vectors get switched depending on which way the person is facing. Characterization of this estimation would be interesting future work.

In order to include the histogram estimate as a set of finite feature variables we calculate the histogram of orientations over the activity sequence. In practice we use use 12 bins in our histogram. Figure 8 depicts this feature.

D. Interaction Coefficient

There are two questions concerning how we want to include information about other people in the room. The first is in regards to how to include features for a variable number of people. The other question is conceiving what correlations we want to use between people. We answer this by using our previous orientation estimate. The projection of the orientation vector between each person is projected and used as a set of features. By calculating the maximum number of people in the room at any time in our datasets we decided that we should always use 4 variables for this feature. The values used are the average of the projections for the sequence. The rationale for using this feature is that you can tell if people are interacting based on if they are looking at each other. Figure 8 depicts this interaction.

V. Recognition

Our goal is to classifier between 5 and 10 activities in the ICU. There are two distinct ways of managing our data that have a large effect on the type of classifier we use. The first is to perceive all people and events in one giant soup. An activity label could be applied at every time step based on all of the people in the room. The second approach is to assign a label for every temporal segment. One of the key benefits is that it allows for multiple actions to happen simultaneously. From our data, we see that there are times when one person is emptying urine and the other person is doing documentation.

Ultimately, we decided on experimenting with Support Vector Machine and Decision Forest supervised learning methods to perform recognition. This in large part has to do with the large number of feature dimensions and the variable number of people and durations. While it is possible to employ techniques such as Hidden Markov Models, as done in [**?**] and other related work, learning the model parameters when the dimensionality of the feature-space is changing is very difficult. While other graphical models, such as Conditional Random Fields, can make this aspect more manageable, it is not obvious how to fully implement it in our situation.

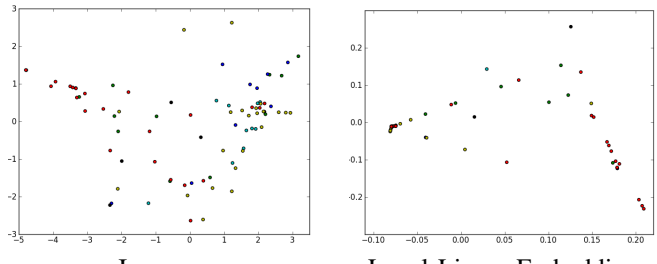
It is also not apparent whether this type of time-series model is even useful for our situation given that action sets are relatively independent of each other. We do not see a noticeable correlation between actions like emptying a urine tube and any other subsequent action. Also, in our case we see that the same actions can take very different amounts of time. For example, sometimes a nurse will come into a room, give medicine, and leave. Other times they will come in, observe for a few minutes, then give medicine.

Our model does have one fundamental limitation. Classification is done on whole activity segments. This means that if a person comes into the room, talks with others, inserts an arterial-line, talk with others, and leaves, it may be classified as only talking with others, because that was the task that happened for the longest period of time. The idea is that our classification algorithm will detect the prominent action in each segment.

A. Data Exploration

Before performing classification we explored the use of dimensionality reduction in hopes to better visualize the data and ideally to see how separable the classes were. Principal Components Analysis, Isomaps, and Local Linear Embedding (LLE) were used. Figure 9 shows the Isomaps and LLE techniques on our first dataset. The different colors denote different classes. These have been hand labeled and refer to the classes listed in Table II. It is apparent that Isomaps clusters the classes much better than LLE does.

One of the advantages of the Decision Forest, which we talk about later, is it's ability to show how important our input



Isomaps Local Linear Embedding

Fig. 9. (Left) Isomaps and (Right) Local Linear Embedding manifold learning techniques have been used to explore our data. Different colors denote different class labels Note that in Isomaps the classes are clustered together much better.

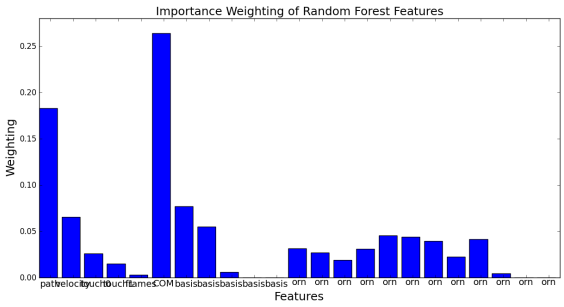


Fig. 10. Pipeline

features are. Figure 10 shows a 26-dimensional chart with the importance of each feature. We see that the average center of mass feature has the highest weight and the path length has the second highest. Originally, we used 5 variables to denote the interaction coefficients. In this chart we see that the last two variables in this group have no weight. As previously mentioned, we later decreased the number of variables based on the knowledge from this graph. Ten of the twelve bins from the orientation histogram are heavily weighted, which shows that despite only having a rough estimate of where the person is looking, it is still useful. It is interesting to see that the frame count has very little importance in our model. This is somewhat obvious after going through the data – the same actions can take very different amounts of time. For example, sometimes a person will observe for 20 seconds and other times they will be around for several minutes.

B. Classification

From the data exploration, it is apparent that the data is not linearly separable in a lower-dimensional space. Thus, we chose to experiment with Support Vector Machines and Decision Forests for classification. A Support Vector Machine (SVM) uses the idea that the data may be separable in a high dimensional state. A larger feature vector is generated which includes our inputs with addition variables that are functions of multiple inputs. The SVM finds the optimal hyperplane that separates classes. Different kernels can be used to efficiently transform the data into other forms which may have better delineation. For example a polynomial or radial-basis function can be used.

In the multi-class case there are two ways of using an SVM. The first is called the one-versus-all method which uses one classifier per class and trains on all of the data. It has two labels: class and not-class. This means there are generally many more not-class points than class points. The second method uses pair-wise classifiers. There are a total of N*(N-1) classifiers where N is the number of classes. An SVM is fit between every every class and every other class. The final classification is generally done by using the class that wins the most tests.

A Decision Forest generates a large number of simple decision trees where the nodes in the tree are randomly picked features. While the technique is relatively new, it has garnered a lot of attention due to it's success in many areas, including in computer vision for tasks like object classification. An extension of this method, Extremely Random Trees, was used based on it's superior performance.

Multi-Instance learning, a semi-supervised classification technique, was also explored on the first dataset. This technique was accurately able to split the datasets into multiple classes. In this method, there are two bags: positive and negative. The positive set contains at least one correct class and the negative bag only has non-examples. These are run through a classifier, in our case an SVM, and reclassified. Examples in the positive bag can then be reclassified as negative examples. In the first dataset this was able to accurately give a high-level split between tasks such as observing+rounds and procedures+checkups, but not lower level splits between procedures and checkups. The nice thing about this techniques is that not all of the data has to be labeled. This is especially helpful for larger datasets. Because we only had a few hours of data per camera we decided it wasn't necessary yet.

The SVM and Decision Forests that we used are available in the SciKit-Learn library for Python.

VI. Results

We recorded two sets of data at the Pediatric ICU. The first one was done as an improvement study without IRB approval. This means that the data is not publishable. After getting IRB approval we collected data once. Two Kinects were used in both efforts. After collection, the activities were hand annotated. First we talk about the recording software that we developed.

A. Recorder

A dynamic frame rate Kinect recorder was written using C++ with the OpenNI device drivers to capture depth and color footage. A dynamic framerate is necessary because of the amount of footage we need. A compressed set of data requires 216 Gb of space per hour per Kinect when running at 30 frames per second. Our recorder only captures when there is motion in the room to decrease the necessary amount

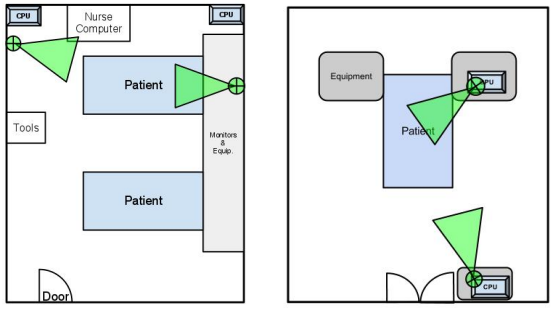


Fig. 11. Experimental setups. The green triangles represent the field of view of the Kinect. (Left) Dataset 1 which was done in a multi-patient room in the old JHMI PICU (Right) Dataset 2 which was done in a single-patient room at the new JHMI PICU.

of storage. Ours records at a rate of about 10 FPS if there is motion in the room and otherwise 1 frame every 3 seconds. This equates to between 20-135 Gb per hour.

In order to get the project passed through the IRB we said that all color data would be de-anonymized. In efforts to do this I experimented using two face detection algorithms. Unfortunately neither of them worked well enough, thus we opted to keep the color data locked up at the hospital.

B. Experiments

We recorded two datasets at the Johns Hopkins Medical Institute's (JHMI) Pediatric Intensive Care Unit (PICU). In each experiment we used two Xbox Kinects placed in different parts of the room, as shown in figure 11. The first set was collected at the old JHMI PICU which had multi-patient rooms. This was problematic because nurses and parents interacting with other patients would sometimes get in the shot of our cameras resulting in noisy data. The second set was collected at the new JHMI PICU which has single-patient rooms. This made our data much more clean. Unfortunately, we have not had time to sufficiently analyze the second dataset. Images from both datasets can be seen throughout this paper.

Using out classification techniques we were able to classify 5 actions in the first dataset and 7 in the second. See table II for the individual actions. In the first dataset our average classification rates were 48% using SVMs and 58% using Decision Forests. For reference, guessing the class by chance in this case results in an accuracy of 20%. Figure 12 shows the per-class recognition rates for the Decision Forest using this dataset. We see that some actions, like rounds, are classified very well but others, like observing, are much lower. While overall these are too low to be satisfactory, we believe that we will be able to achieve better results by using better quality data and possible using other types of features or classifiers.

TABLE II

ACTION LABELS FOR EACH DATA COLLECTION

Set I Actions	Set II Actions
Rounds	Documentation
Talking	Talking
Observing	Observing
Checking Diagnostics	Checking Diagnostics
Procedure	Procedure
Other	Urine Tube Removal
	Ventilator Use
	Other

While we recorded with two cameras in the first setup, we found that the camera near the patient's feet was not very helpful. One of the major advantages of using multiple cameras is that people can be tracked throughout the room. The cameras should have enough overlap so they can be registered in one frame such that that the same people can be detected in both cameras. In the first setup there was not enough overlap to properly do this. We accounted for this in the second dataset but have not had time to fully implement it.

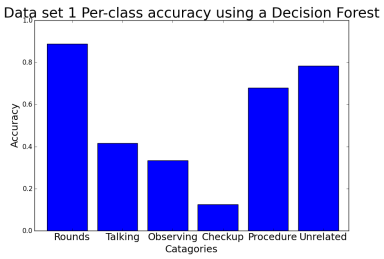


Fig. 12. Per-class accuracy for the first dataset using a Decision Forest.

The camera near the patient's head in the first dataset was very useful for tracking personnel as they worked with the patient on one side of the bed. However, whenever nurses came on the other side of the bed they would come in and out of the camera shot a lot. These would still be labeled as classes like procedure and checkup, however their characteristics would be very different than events on the other side of the bed. We believe that this is on the other reasons why we get low classification accuracy.

Preliminary results on the new dataset show high classification for certain classes, but the work isn't done enough to speculate too much. Given the decrease in noisy actions and the new camera setup we think the results will in general be higher.

VII. CONCLUSION

While at this point our results aren't complete, we have shown that automated activity analysis using 3D vision techniques may have the potential to be used in an everyday setting to increase safety, help efficiency, and optimize workflow. By performing this study we have opened a number of interesting research questions and directions.

To continue in the same direction, future work may include looking at additional types of features. By implementing a skeletal tracker it will be possible to extract information about what personnel are doing during procedures. This should make it easier to differentiate what kind of procedure they are doing. Gesture recognition may be useful in this case and could be combined into a hierarchical framework to use in conjunction with the methods that we have employed. Other potential information includes traditional space-time features and using contextual pixel-level information in a bounding box around the person. For example, in the first dataset one action is a diaper change. It may be possible to classifier this type of event based on SIFT or other descriptors run on the window around the nurse.

It is apparent that knowing information about he room can be useful. For example, knowing that a nurse is standing next to a heart monitor may make it easier to classifier that they are checking vitals than if you know they were standing next to a computer. To this end, full scene analysis may turn out to be an important part of activity recognition.

Part of the importance of the recognition work as a whole is to detect when things go wrong. Anomaly detection is thus

an interesting area. The problem is that currently there isn't a list of tasks that must be generated (that in itself is another potential research area) which makes this type of detection very hard.

Differentiating between people may be an important task. This could be used to summaries how much time nurse spend in the room versus doctors or other personnel. Additionally, individual people could be detected and potentially even measured to determine how well they do at particular tasks. This could be used to allocate better staff to more problematic patients. One way to do this is to look at a color profile of the people. While we are not allowed to keep the color information remotely, due to the IRB, we could extract histograms or features from the people as features.

Looking at patient movement is also a useful problem. In the adult ICU it is common for people to try to get out of bed and slip and fall. In the pediatric ICU, patients will sometimes move around a lot and accidentally remove IVs and other tubes. We had first hand experience with this during our second data collection where on the previous night the child slid halfway down the bed and pulled out a tube in their back without anybody seeing. By tracking this, a nurse could be notified if there is too much movement.

In conclusion, we believe that the Intensive Care Unit is ripe for automation and that using 3D sensing can have a large impact on current workflow and safety concerns. While our current results are not outstanding, we think that with further improvements a similar system could be used to make the ICU a better place.

REFERENCES

[1] P. Carayon and M. Wall, "Impact of Performance Obstacles on Intensive Care Nurses Workload , Perceived Quality and Safety of Care , and Quality of Working Life," pp. 422–443, 2008.

[2] I. Care and P. Central, "Problems associated with nursing staff shortage : An analysis of the first 3600 reports submitted ..." 1998.

[3] M. Ma, M. Rnb, and C. L. Jd, "Add to My Projects A look into the nature and causes of human errors in the intensive care unit," pp. 1–9, 1995.

[4] A. Detsky, S. Stricker, A. Mulley, and G. Thibault, "Prognosis, Survival, and the Expenditure of Hospital Resources for Patients in an Intensive-Care Unit," *The New England Journal of Medicine*, 2010.

[5] Z. Tu, X. Chen, A. L. Yuille, and S.-C. Zhu, "Image Parsing: Unifying Segmentation, Detection, and Recognition," *International Journal of Computer Vision*, vol. 63, no. 2, pp. 113–140, Feb. 2005. [Online]. Available: http://www.springerlink.com/index/10.1007/s11263-005-6642-x

[6] K. P. Murphy, "Hidden semi-Markov models (HSMMs), Tech. Rep. November, 2002.

[7] N. Padoy, "Workflow Monitoring based on 3D Motion Features," in *International Conference on Computer Vision (ICCV)*, 2009, pp. 585–592.

[8] M. S. Ryoo and W. Yu, "One Video is Sufficient? Human Activity Recognition Using Active Video Composition," in *Workshop on Motion and Video Computing (WMVC)*, no. January, 2011.

[9] E. Spriggs, F. De La Torre, and M. Hebert, "Temporal segmentation and activity classification from first-person sensing," *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, June 2009. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5204354

[10] E. E. Stone and M. Skubic, "Evaluation of an Inexpensive Depth Camera for Passive In-Home Fall Risk Assessment."

[11] A. Bigdelou, T. Benz, and N. Navab, "Simultaneous Categorical and Spatio-Temporal 3D Gestures Using Kinect," in *3DUI*, 2012.

[12] I. Guyon, V. Athitsos, P. Jangyodsuk, B. Hamner, and H. J. Escalante, "ChaLearn Gesture Challenge : Design and First Results," *CVPR*, 2012.

[13] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-Time Human Pose Recognition in Parts from Single Depth Images," 2010.

[14] C. Plagemann, "Real Time Motion Capture Using a Single Time-Of-Flight Camera," in *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.