Automated Workflow and Activity Recognition at the Pediatric ICU

Paper Review

Colin Lea
March 2012

Paper Selection:

A. Bigdelou, T. Benz, L. Schwarz, N. Navab. "**Simultaneous Categorical and Spatio-Temporal 3D Gestures Using Kinect."** *IEEE Symposium on 3D User Interfaces (3DUI), Orange County, CA, March 2012*

A. Bigdelou, L. Schwarz, N. Navab. "**An Adaptive Solution for Intra-operative Gesture-based Human-Machine Interaction.**" *International ACM Conference on Intelligent User Interfaces (IUI), Lissabon, Portugal, February 2012*

L. Schwarz, A. Bigdelou, N. Navab "**Learning Gestures for Customizable Human-Computer Interaction in the Operating Room**." *Medical Image Computing and Computer Assisted Intervention (MICCAI), Toronto, Canada, September 2011*

Overview:

In our project, in which we are developing methods for activity recognition in the ICU, there are several distinct algorithmic components to keep in mind. These include tracking people, identifying equipment, and deciphering actions carried out by nurses and staff. In our approach we establish a connection between these actions and literature on the topic of gesture recognition. In the following, a line of three papers by Ali Bigdelou at the Technical University of Munich is detailed which has a close link to our work. The papers feed off of each other and thus will be discussed jointly.

The overarching goal of this work is to develop a system capable of recognizing a set of human gestures. The key distinction over prior literature is in the ability to learn and recognize both categorical (discrete) and spatio-temporal (continuous) gestures. For example, a categorical state may be a waving gesture and the

spatio-temporal component may be a normalized hand position between the start and the end of the wave. Two dimension reduction-based approaches are implemented and compared with two sensing modalities – the Xbox Kinect and a set of Inertial Measurement Units (IMUs) placed on the users' arms. Each paper includes a user study using an image viewing application in an operating room. In general, this user interface is preferred over traditional mouse and keyboard setups.

Methods:

Both sensing modalities chosen are high dimensional. The Kinect skeleton tracker outputs 15 three-dimensional body positions resulting in a 45 dimension sampled up to 30 times per second. Only the orientation component is used in the IMUs, so each device contains 4 dimensions – representing a quaternion – resulting in a total of 16 dimensions between the four devices. In order to reduce this dimensionality, Principal Components Analysis (PCA) and Laplacian Eigenmaps are employed. Additionally, the authors demonstrate substantial noise in the categorical gesture label using the Laplacian Eigenmap technique, thus a Particle Filter is used to smooth the results. A maximum likelihood approach is used to establish the categorical state and a Kernel Regression Map is used to determine the normalized spatio-temporal value.

PCA – Intuitively PCA outputs a low dimensional representation of a dataset that includes a set of basis vectors in the directions of greatest variance. Figure 1 highlights the process used for using PCA on the Kinect data. The idea is to run PCA on each 45-dimension vector per class per time step. Thus, for each class of gestures, the number of basis vectors will be a function of the number of time steps. The per class set of bases are normalized to represent the continuum of spatio-temporal positions. This procedure is done for each class.

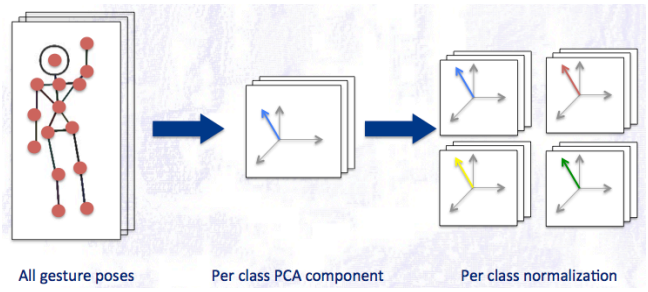

Figure 1 PCA-based dimensionality reduction

Laplacian Eigenmaps – Manifold learning techniques, such as this, are used to find a low dimensional linear embedding from non-linear high dimensional spaces. For example, in this application Laplacian Eigenmaps can embed the 16 dimensional IMU data into a 2D gesture representation. There are three key parts to this technique. First, find the nearest neighbors in the high dimensional space. This can be done efficiently using approximate nearest neighbor methods. The second step is to calculate a similarity function based on the local neighborhood using a heat kernel. Lastly, to get the low dimensional output, the eigenvalues of the similarity matrix are computed. Note that this output is a Euclidian space. In this paper they also define a gesture phase model which is simply based on the start, middle, and end of the manifold, as shown in figure 2.

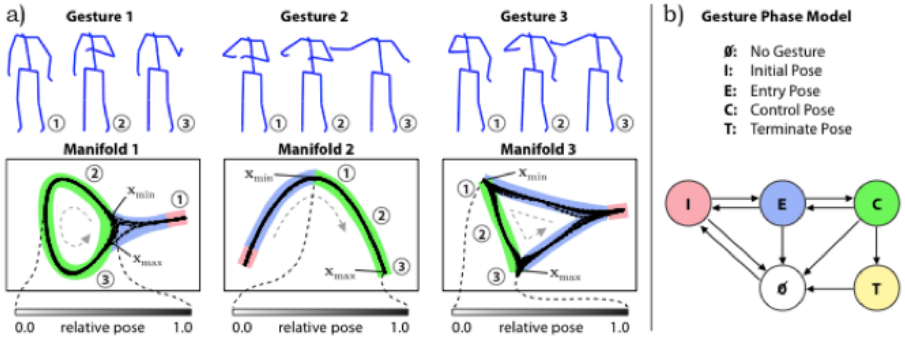

Figure 2 Laplacian Eigenmap technique applied to a sample gesture using the inertial measument units

A Particle Filter is applied when using the manifold technique to track the gestures over time. At each timestep a set of particles is added to manifolds based on the probability of a sensor reading given the class labels and the low dimensional representation. The maximum likelihood solution if determined to find the categorical state and the low dimensional outputs are averaged to

determine the spatio-temporal position.

Kernel Regression Map – In order to determine the spatio-temporal position using PCA a generative kernel regression technique is used. Remember, the per class normalized basis vectors that were calculated are placed at discrete points along the continuum from 0 to 1. Essentially, during the testing phase each of the bases is weighted based on their similarity to the test basis. Weights for each per class basis is determined based on a Gaussian kernel using the distance from the test datum as its input using the following equation where s is a sample in high dimensional space and w is the weight.

$$w_i(s_t) = \exp\left(-\frac{1}{2}\left|\left|(s_t - s_i)/\sigma\right|\right|^2\right)$$

Additionally, the weights are smoothed based on the previous time step. The weights and their respective low-dimensional spatio-temporal value are evaluated to determine the final 1-dimensional normalized number using the following equation where x is the low dimensional value.

$$\hat{x}_t = f(\mathbf{s}_t) = \sum_{i=1}^{n} \frac{w_i(\mathbf{s}_t)}{\sum_{j=1}^{n} w_j(\mathbf{s}_t)} \cdot x_i$$

Results:

Given the relative simplicity to these models, the results are surprisingly good. For a low quantity of gestures they achieve 90%+ classification accuracy for both the Kinect and IMU datasets. Figure 3 shows the rates for the IMU dataset using PCA and manifold techniques for a set of 4 to 18 gestures. Note that MREG is the time-smoothed kernel regression that we previously mentioned and REG is the same method without using the previous classification data point. It's interesting to see that the PCA and manifold techniques achieve approximately the same accuracy on average – in one test PCA does slightly better and in another slightly worse. This is counter-intuitive if you think about the space of the data and the complexity of the algorithms. PCA should do worse at modeling non-linear data than Laplacian Eigenmaps. This shows, however, that this is not the case.

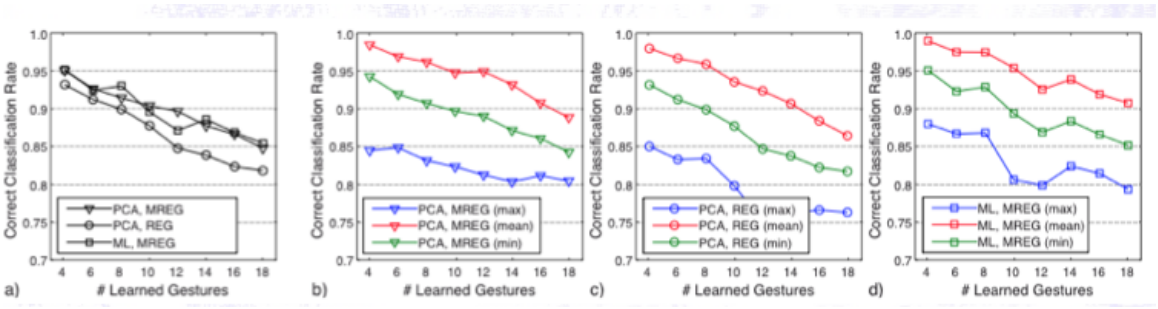

Figure 3 IMU results for PCA and manifold techniques. Plot A shows the average rates and B-D show the min, mean, and max rates for each algorithm.

Additional plots (not shown here) show that using a displacement-based representation of the Kinect data is superior to other distance measures. In this model, the position of each joint is normalized relative to the spine. An approach based on relative distances from the parent joint to a child joint also gets similar results. This fact is important to my work and has results that are similar with what I have seen.

User Study:

The application for this study is towards developing a natural, gesture-based user interface for viewing medical imagery in an operating room. Compared to the classical keyboard and mouse interface the participants slightly favored the Kinect interface. The IMU-based interface received a little bit less favorable results – a one-point difference on a scale from 0-5. As a whole they were satisfied with the new interface. Note, however, that users didn't like the voice activation method very much. In two of the studies the participant had to say "start" and "stop" to enable/disable the interface controls. Thus a more effective method of control is wanted.

Critique:

Between the three papers the authors didn't leave two many questions unanswered. A couple questions stem from the accuracy studies. They note that up to 18 gestures are used in the experiments. It would have been nice if the authors discussed the types of gestures more in detail. They mentioned actions such as moving your arm up and down to control a vertical sliding bar, but they didn't talk about how much the gestures varied. In my experience if two gestures are fairly similar then there is greater classification error. Per-joint and per-

gesture confusion matrices would have been useful to see if their miss-classifications are due to a single "bad" gesture or if the error is random.

Further analysis on the datasets would also be appreciated. In the papers they have a "common" dataset where all of the training data comes from a group of people and the testing data is from another person, as well as a "personalized" dataset where the same person does both the training and testing. In my current results using a fairly similar PCA-based approach I get a large discrepancy between "common" and "personalized" datasets. I get ~95% accuracy for the same person but only about 35% when training and testing on different people. I am interested to know more about their "common" dataset approach. They get only about a 10% difference between those data sets.

These papers have helped define my preliminary direction for the gesture recognition component of the ICU project. In the future I will be looking towards exploring other approaches such as time-series graphical models and multi-instance learning with support vector machines to compare how more structured models compare with the simplicity of PCA.