Automatic Motion Recognition and Skill Evaluation for Dynamic Tasks

Todd E. Murphy[1], Chet M. Vignes[1], David D. Yuh[2], and Allison M. Okamura[1]

[1] The Johns Hopkins University, Dept. of Mechanical Engineering, 3400 N. Charles St, Baltimore, MD 21218 USA
{tmurphy, aokamura}@jhu.edu, cvignes@engineering.uiowa.edu
http://www.haptics.me.jhu.edu/
[2] Johns Hopkins Medical Institutions, Cardiac Surgery, 600 N. Wolfe Street, Baltimore, MD 21218 USA
dyuh@csurg.jhmi.jhu.edu

Abstract. Along with the development of computerized training systems has come increasing demand for objective methods of evaluating skill. For scenarios where skill is related to specific user motions, we have developed a system using hidden Markov models (HMMs) to recognize motions performed in a virtual environment with a haptic device. The output of this system is a list of motions used during completion of a task. We first explore which observations are most important for accurate recognition of user motions. Second, we use a sequence of motions to evaluate skill by analyzing the performance of individual users over multiple repetitions of a dynamic task. The results reveal that the system is able to achieve consistent recognition with a wide variety of observations. We also highlight numerous challenges to be solved before it can be used in practice. This approach could be used for validating the skill level of specific users as well as evaluating the efficacy of various training methods.

1 Introduction

Computerized and virtual reality training systems—many of which utilize haptic feedback—have gained increasing acceptance and sophistication in recent years. These tools, such as laparoscopic surgical simulators, open the door to a host of techniques not available with traditional training methods. For example, it is possible to create training scenarios that contain important complications that are rarely encountered in practice. Other benefits include the low cost of repetition, the opportunity to fail without consequences, and (potentially) increased realism in comparison to traditional training methods. An additional feature of these computerized systems is the wealth of data that may be collected during a training session. Presumably this data can be used to develop meaningful and objective metrics for skill, but in many applications the best way to do so remains unclear. A sampling of previous work in the medical field reveals systems that perform low-level analysis of the positions, forces, and times recorded during training on simulators and teleoperation systems [2, 10, 12, 14].

In this paper, we investigate the first steps in the development of an evaluation system utilizing hidden Markov models (HMMs) to recognize operator motions. HMMs have been applied extensively to recognition for speech [9] and handwriting [5]. They have also been used for recognition in tasks related to human motion [1], driving behavior [8], sign language [11], and human-computer interfaces. Rosen, *et al.* have done a great deal of work using HMMs to evaluate skill in laparoscopic surgery [10]. In one approach, a model was developed for each subject in the test. Skill was assessed by a comparison of the statistical distance between the models of suspected novices and recognized experts. Our works differs in that we train many models—one for each gesture—and seek to assess skill through an analysis of all the gestures used in the completion of a task. Previous work in our laboratory has used HMMs for gesture recognition in a cooperative (admittance control) human-robot system [6] to provide appropriate assistance in the form of virtual fixtures [7]. The observation vector included only forces applied by the user. Our previous work used tasks that are simpler and more constrained than what we present in this paper, and until now we have not evaluated skill using motion recognition.

This research is being done with an eye towards surgical applications. For many surgical tasks, intuition suggests that the skill of the surgeon is intrinsically tied to the motions used during the task. However, despite the demand for objective skill assessment [3], previous attempts [4, 13] have had difficulty in collecting truly objective data that correlated well with outcomes. Recent advances in robotic devices for minimally invasive surgery have created an unprecedented environment for collecting data during surgical procedures. In the existing systems, the surgeon sits removed from the patient at a console equipped with a display showing the operation site as seen through a laparoscope. The surgeon manipulates hand-held instruments that digitize his or her hand motions, and these inputs are then used to control the surgical tools held by the robotic device. Ultimately, our goal is to exploit the ability to collect information in this environment and implement a version of the system presented here on such a robotic device.

1.1 Hidden Markov Models

The goal of modeling a system is to develop a set of rules for predicting that system's output. Many system models are created by attempting to understand and describe the basic principles that govern the system's behavior. Such deterministic models require a sophisticated understanding of the task to be accomplished by that system, and must explicitly encode the effects of noise, disturbances, etc. Hidden Markov models, on the other hand, are stochastic models that seek to predict the output of the system based on past observations.

HMMs operate under the premise that a system may be described as being in one of a set of distinct states. The observable output of the system is a probabilistic function of which state the system is in. As time progresses, the system may change state. When it does, there will be corresponding changes in the output. The states are arranged in a network that defines which states the system may change between. While

each HMM consists of a network of states, an entire system (or task) exists of a higher-level network of HMMs. This concept is illustrated in Figure 1.

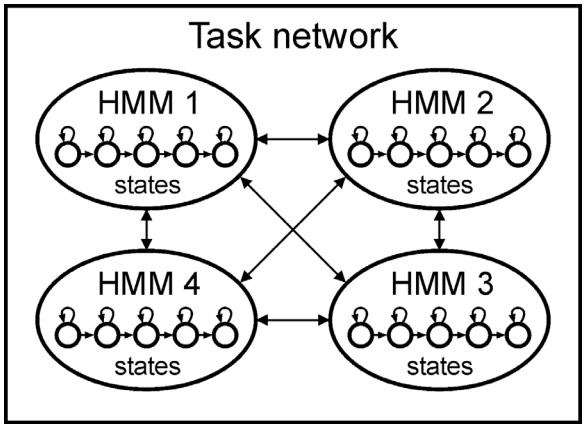


Fig. 1. A network of hidden Markov models (HMMs), where each HMM consists of a number of "hidden" states.

An HMM captures this type of system structure with three components that define the probability of transitioning from one state to another, what observations a state is likely to produce, and the initial conditions of the system. These three components are formally known as the state transition probability distribution matrix A, the observation symbol probability distribution matrix B, and the initial state probabilities π. A model λ can be succinctly defined by writing $\lambda = (A, B, \pi)$. Rabiner [9] explains the three basic questions that emerge when using HMMs to model real-world systems:

Problem 1: Given the observation sequence $O = o_1, o_2, ..., o_T$ and a model $\lambda = (A, B, \pi)$, how do we efficiently compute $P(O|\lambda)$, the probability of the observation sequence, given the model?

Problem 2: Given the observation sequence $O = o_1, o_2, ..., o_T$ and a model $\lambda = (A, B, \pi)$, how do we choose a corresponding state sequence $Q = q_1, q_2, ..., q_T$ that best explains the observations?

Problem 3: How do we adjust the model parameters $\lambda = (A, B, \pi)$ to maximize $P(O|\lambda)$?

1.2 Application of HMMs to Motion Recognition

The three basic questions identified by Rabiner have direct applications in motion recognition. Problem 1 is essentially the recognition problem. Given the model M for some motion and the series of observations O made from an unknown motion, solving Problem 1 reveals the probability these observations were produced by the model M. The forward-backward procedure is used to solve this problem.

Problem 2 has analogues in our application on two levels. We are interested in knowing the best state sequence for the purposes of recognizing each separate mo-

tion, but we are also concerned with knowing the most likely sequence of motions used during a task. Fortunately, the Viterbi algorithm, which is often used to solve this problem for state alignment, extends well to a network of models.

Finally, the answer to Problem 3 tells us how we can define appropriate models for each of our motions—models that we will need for the solutions to Problems 1 and 2. The approach to solving this problem is to use a series of observations known to be from a particular motion along with an iterative procedure such as the Baum-Welch method to estimate the model parameters.

2 Experiment I: Selecting Appropriate Observations

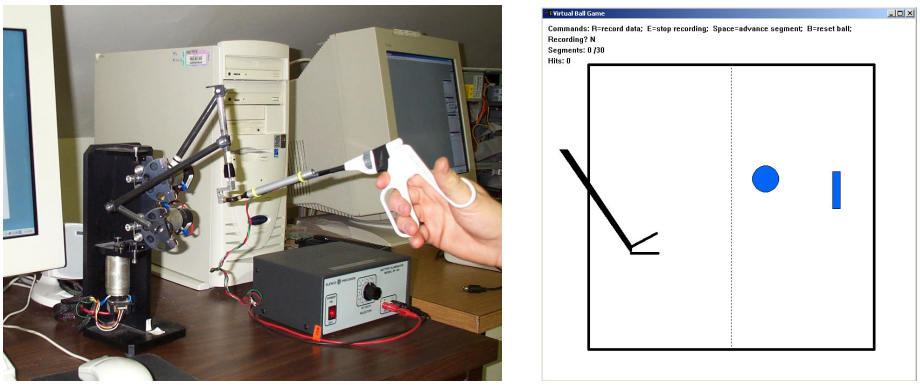
In the virtual environment developed for this work, it is possible to record all the information necessary to create the environment—in short, we have complete access to all states of the system, such as position, velocity, force, etc. (Note that these are different from the "hidden" states of the HMM, which do not necessarily have a physical interpretation.) This experiment sought to identify which of these variables contributed most beneficially to motion recognition with HMMs.

2.1 Experimental Setup

In this experiment, users interact with a two-dimensional virtual environment through use of a three-dimensional haptic device (3GM, Immersion Corporation) with a modified laparoscopic tool (Auto Suture Endo Shears) attached. Interfacing with the haptic device is accomplished through an Immersion Impulse PCI card. A Hall-effect sensor on the scissor-like handle of the laparoscopic tool is used to determine if the gripper is open or closed, and this data is obtained through a custom parallel port A/D card. Figure 2a shows the device in use.

A representation of the laparoscopic tool and an end-effector are drawn in the virtual environment (Figure 2b), where the user interacts with other objects. The virtual environment was created with Visual C++ and runs on a 800 MHz computer with the Windows 2000 operating system. The virtual environment is contained in a box. The limits of the environment are shown with dark lines, and forces from the haptic device prevent the tool tip from moving outside these boundaries. In addition to the tool, the environment contains a moving target (a thin rectangle) and a ball that can be picked up, carried, and thrown with the gripper. The target moves continuously up and down in a regular sinusoidal pattern. The ball behaves much like a ball in the real world: it is subject to a constant downward acceleration from gravity, viscous damping in air, and it will bounce off of the target and the floor. However, if it strikes either the left or right wall, it "sticks" to the wall and falls to the floor.

The goal of the task was to hit the moving target three times with the ball thrown from behind the dotted line drawn down the middle of the environment. If the ball misses the target, it strikes the right wall and falls to the floor, where it must be retrieved for another try. If the ball hits the target, it bounces back to the left wall, where it falls to the ground and is retrieved by the subject. Test subjects and system



(a) (b)

Fig. 2. Experimental setup: (a) The 3GM haptic device, and (b) the virtual environment.

trainers were instructed to refrain from catching the ball in mid-air but, rather, to wait until the ball had settled after each throw. This constraint was developed to simplify the number of potential motions to be recognized.

2.2 Recognition System

The HMM algorithms are implemented with the Hidden Markov Model Toolkit (HTK) from the Cambridge University Engineering Department. Use of a HMM system for motion recognition is preceded by a process of training models for each of the motions we desire to recognize. For the purpose of this experiment, any group of motions could have been selected. An analysis of the motions executed during the recorded sessions of an expert user and several test subjects resulted in the definition of ten basic gestures (chosen by the experimenter) that are used to classify all the observed motion. These ten gestures define the "vocabulary" of our recognition system and are described in Table 1. The training data was formed from 14 recordings of an experienced user executing these motions. Not every recording contained every motion; each motion had a minimum of seven examples in the training data. The data was used to train a plain, single mixture, single stream, five-state HMM for each of the basic motions.

To assess the performance of the recognition system, it is necessary to have a standard for comparison. As with speech recognition systems, our standard is a transcription detailing the motions used and times of transition between motions. This transcription was identified manually by using the capability of the virtual environment to replay recorded sessions. As the recording is replayed, the data is labeled and segmented manually by the experimenter. (In previous work [6, 7], we allowed the users to segment the task during execution by pressing a key on the control computer when they intended to change motions, but for the dynamic task presented here, this method generates significant errors in transcription due to increased mental load.) During each recording, data was collected at 100Hz. In all, twelve observations were recorded: position (x_{pos}, y_{pos}, z_{pos}) and velocity (x_{vel}, y_{vel}, z_{vel}) of the tool tip in three

Table 1. Motion Vocabulary Chosen for a Dynamic Task

Label	Description
A	Moving downward to retrieve ball, ends after ball is grasped.
B	Moving primarily upward with ball.
C	Throwing the ball. Ends at time the major components of motion in the direction of the throw cease.
D	Horizontal movement to the left without the ball.
E	Moving forward and down to retrieve ball. Ends after ball is grasped.
F	Moving left and up with ball in gripper.
G	Moving backward and down to retrieve ball. Ends after ball is grasped.
H	Moving forward and up with ball in gripper.
I	Wasted motion—low magnitude in any direction, does not result in major position change or end by retrieving or throwing the ball.
J	No motion; silence.

dimensions, position of the ball (x_{ball}, y_{ball}), the distance separating the ball and the tool tip (d_s), the status (open or closed) of the gripper (g), and the magnitude of forces (f_x, f_y) being exerted on the tool tip by objects in the environment.

2.3 Experimental Procedure

The experimental process began with nearly 60 test runs used to adjust several system parameters to baseline values that produced reasonable results. Among these parameters were the model transition penalty, the pruning threshold, and the number of states in each model. The transition penalty affects the Viterbi-like algorithm used for recognizing the most likely sequence of models, known as the Token Passing Model. The algorithm works by passing tokens through the network of possible models and discarding tokens that travel low probability paths. The transition penalty is a fixed value that is added to the log probability of each token as it jumps to a new model. The pruning threshold defines the width of search used during the forward-backward procedure for model estimation.

Both the transition penalty and the pruning threshold have a strong effect on performance of the system. In general, a lower transition penalty results in a greater number of insertions—situations where the system recognizes a motion that was not performed. Conversely, a higher transition penalty leads to more deletions, when the system does not recognize motions that were performed. The effect of the pruning threshold is less dramatic (the main benefit is decreased computation), but making it larger tends to increase the number of insertions and vice versa. Both parameters will require further fine-tuning for optimal performance and different data sets.

The first batch of tests also verified that the quantity of training data was sufficient for robust model estimation by using only half of the data and obtaining comparable results to tests using twice as much data. With the values of these parameters settled, we set out to determine which observations were most important to achieving good recognition.

Table 2. Word accuracy percentages for recogntion of training data.

Test	Observations	Accuracy %
1	x_{pos}, y_{pos}, z_{pos}, x_{vel}, y_{vel}, z_{vel}	73.83
2	x_{pos}, y_{pos}, x_{vel}, y_{vel}	73.83
3	x_{vel}, $y_{vel,}$ g	71.96
4	x_{vel}, y_{vel}	71.96
5	x_{vel}, y_{vel}, f_x, f_y	71.96
6	x_{pos}, y_{pos}, x_{ball}, y_{ball}, d_s, f_x, f_y	73.83
7	x_{ball}, y_{ball}, d_s, f_x, f_y	81.31
8	x_{ball}, y_{ball}, d_s	81.31
9	x_{ball}, y_{ball}	81.31

2.3 Results

Table 2 shows the results of nine different tests including various combinations of observations in the training data. The recognition was performed on the training data. For all tests, the transition penalty and pruning threshold parameters of the HTK system were set at –200 and 1000, respectively.

Accuracy is computed using a common formula from the speech recognition literature: $(N - D - S - I)/N$, where N is the number of motions in the transcription, D is the number of deletions, S is the number of substitutions, and I is the number of insertions. This is an appropriate metric because it captures the number of each type of error during recognition. The results show there is considerable room for improvement before we achieve the success of other systems based on the same techniques. (Successful speech recognition systems typically have recognition accuracies > 95%.) However, they also highlight the flexibility of this method. Even when using nine different combinations of input observation vectors—some with more than three times as many components as others—the recognition rate remains relatively flat. The small variance in recognition rate prevents any sweeping conclusions, but some variables do appear to have advantages over others.

As shown in Table 2, test 1 represented a typical sampling of observations that would naturally be selected for a motion task. This combination also included the z-axis position and velocity. Despite the fact that the virtual environment is only two-dimensional, the haptic device is not constrained to this plane, and the possibility existed that movement along that axis could be of use in recognition. When compared to test 2, though, we see that the recognition is unaffected by the loss of the z-axis information, and we declined using it further. The observations in tests 3 and 4 were selected because these were most closely related to how the motions were defined (Table 2). The results suggest that despite this primary role, recognition can be improved with the inclusion of more information. Test 4 indicates knowing the gripper status contributes negligibly. Test 5 was the first to include the haptic forces the user experienced and shows that, although these forces improve the reality of the environment and may be beneficial to the user for completion of the task, they do not

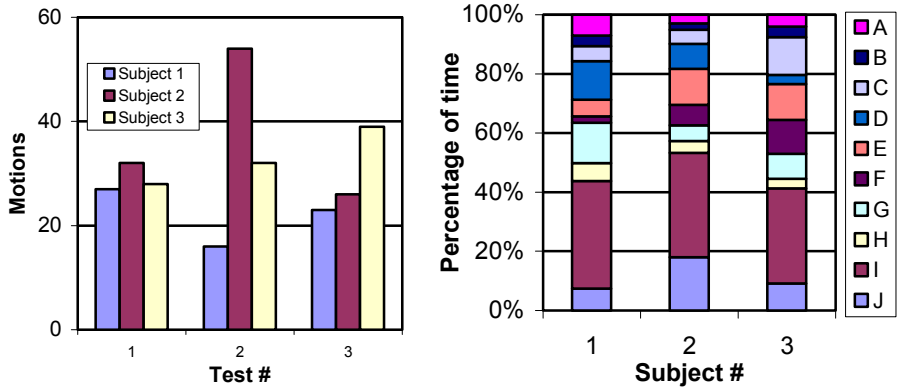
appear to have a useful effect on recognition. The results of tests 7 and 8 support this hypothesis. Test 6 used only observations that are not measured outside of the virtual environment. A small improvement in recognition encouraged tests 7-9, and these observations, particularly the position of the ball, produce the highest recognition rates. However, these results do not tell the complete story. First, the state of objects in a real environment may not be available for use in evaluation. Also, further analysis reveals that only one of the 13 examples of motion J (silence) in the data was correctly recognized in these tests. For that reason, this group of observations may not be the best choice in the context of our overall goal of skill evaluation.

3 Experiment II: Skill Evaluation

In this experiment we demonstrate how a recognition system can be used for skill evaluation purposes. Three different subjects, all with no prior experience using the system, completed a dynamic task in the virtual environment on three separate occasions and their performances were recorded. The task is the same as that describe din Experiment I.

Our approach is to record the performance of a user in the virtual environment, automatically recognize the sequence of motions executed by the user, and use this sequence to draw conclusions about the skill of each user. The final step of assessment could be done in several ways. Here we present a simple method comparing the total repetitions between users over multiple sessions to obtain a relative measure of skill.

One obvious concern about this type of system is that if the recognition system incorrectly identifies some of a user's motions, then any skill assessment based on this recognition will also be flawed. This is a real issue without a direct solution. One way



(a) (b)

Fig. 3. (a) Total number of motions used in completion of a dynamic task by three subjects over three tests. (b) Percentage of motion usage for each subject, averaged over three tests.

to justify the situation would be to view errors in the recognition system like noise in a more conventional measurement. With acceptably high recognition rates (perhaps >95%), the recognition system gives us a flawed picture of reality, but one that represents reality closely enough that it may still be used effectively to evaluate skill. Correlation of skill evaluation with external metrics, such as functional outcomes, could validate this technique.

Recognizing that our recognition system has not yet been refined to the point that it will provide accurate, reliable results, the results presented in this experiment refer to the manually identified sequence of motions utilized by each subject rather than results from the HMM system. Figure 3a shows the total number of motions used by each subject in each of three attempts to complete the task described. The left plot shows that test subject 1 used fewer motions than the other two subjects in all three trials. As shown in Figure3b, we were also able to compare the time of usage of motions I (wasted motion) and J (pause). Wasted motion accounted for 34% ±3% of the total time for all three subjects. More revealing was the usage of pauses, found to be 7.4% for subject 1, 18.0% for subject 2, and 9.1% for subject 3.

From these results we conclude that subject 1 was the most skilled of the group. For this to be a valid conclusion, we make the assumption that a skillful user will require the use of fewer motions to complete the task than a novice user and that this reduction will come, in part, from more efficient execution. Such "economy of motion" is often subjectively gauged for surgical skill evaluation.

This system enables numerous methods for assessing the skill. A comparative analysis like the one used here could be particularly useful if a recognized expert was included in the test group. Another possibility would be to track measures such as total motions and percentage of wasted motion, etc., over many repetitions of the same task by a single user in order to evaluate the user's learning curve. In addition, one could track the level of force applied during particular phases of a task. Over a large group of test subjects trained with different methods, such analyses could yield valuable insight regarding the efficacy of different teaching techniques.

4 Discussion

With such a large body of work in HMM-based speech recognition, it is useful to draw comparisons between those systems and our own. Speech systems typically have a much larger vocabulary than the simple 10-gesture vocabulary we have defined, often on the order of thousands of words. At first glance this may suggest that our objective is rather simple. However, speech recognition relies heavily on a well defined grammar that defines the probability one word may follow another. This grammar is derived from actual usage of the language and properly defining it has a tremendous effect on the success of the recognition system—perhaps as much as any other factor. Other HMM recognition systems have been for fairly deliberate (e.g., T'ai Chi Ch'uan [1]) or well-constrained (e.g., driving [8]) motions that lend themselves to a unambiguous vocabulary.

In our system, however, neither the vocabulary nor the grammar is pre-defined; the burden lies with the system designer to identify both of these things (first the vocabu-

lary and then the grammar). For now, the grammar has been defined by the transitions observed in the training and test data, with equal probability given to each transition. It is quite possible that a more appropriate vocabulary and a more sophisticated grammar—such as one that uses context-dependent transition probabilities—could yield significantly better results than we achieved here. However, it is not simply high-percentage recognition we seek, but we also desire to draw conclusions from the output. The results of Experiment I showed the best recognition was produced using a group of observations that almost completely failed to recognize one of the basic motions. Preliminary tests indicate that redefining the vocabulary, re-labeling the data, and performing the test again improves the recognition rate, but in doing so we discard the ability to identify pauses with the recognition system, which may be of use.

By comparing this work to our previous research on HMM recognition [6, 7], it is clear that the task domain is extremely important, both in determining the appropriate observation vectors and in the recognition accuracy that can be obtained. For complex tasks such as suturing, a detailed task analysis must be completed in consultations with surgeons in order to develop appropriate transcriptions for training the HMMs. While this process is cumbersome, it is important to remember that the trained HMMs will be available for recognition of motions executed by any user.

5 Conclusion

In this work we present the use of hidden Markov models to recognize the motions of a complex, dynamic task in a virtual environment and analyze the effect of different observation vectors. In contrast to previous work on HMMs in skill evaluation, we use each HMM to describe a particular phase of a dynamic task, and then use the resulting segmentation to analyze task execution. We found there are many factors affecting recognition performance, including the number of states in each HMM, the data used in the observation vector, the nature and amount of data used in the training process, and specific parameters used in the algorithms for training and recognition. Once an appropriate set of parameters is chosen, we can use the HMM recognition to evaluate the skill of users executing a dynamic task in a virtual environment.

This system has great potential for use in both training simulators and evaluation of robot-assisted surgery. Our work essentially takes advantage of the automatic data collection capability of such systems to provide an objective assessment of performance. The long-term objective of this research is to assess the feasibility and validity of objectively defining and assessing surgical technical competence in the performance of robot-assisted, minimally invasive cardiac surgery. In future work, we intend to recognize surgical gestures on position and haptic data acquired through the da Vinci computer-enhanced surgical robotic system (Intuitive Surgical, Inc., Sunnyvale, CA). Technical performance indices will be developed from the recognition results and these indices will, in turn, be correlated to objective functional outcome measurements to determine when a surgeon possesses the technical competence to safely perform robot-assisted cardiac surgery on actual patients.

6 Acknowledgements

This research is supported by the Johns Hopkins University Division of Cardiac Surgery and the National Science Foundation's Research Experience for Undergraduates program through grant #EEC-9731478. The authors would like to thank Dr. William Byrne for his expert advice on HMM theory and application.

References

1. Brand, M., Oliver, N., Pentland, A.: Coupled hidden Markov models for complex action recognition. Proc. IEEE Conf. on Comp. Vision and Pattern Recognition. (1997) 994 –999
2. Cotin, S., et al.: Metrics for Laparoscopic Skills Trainers: The Weakest Link! MICCAI, Lecture Notes in Computer Science, Vol 2488. (2002) 35-43
3. Darzi, A., Smith, S., Taffinder, N.: Assessing Operative Skills. British Medical Journal, Vol. 318. (1999) 887-888
4. Datta, V., et al.: The Use of Electromagnetic Motion Tracking Analysis to Objectively Measure Open Surgical Skill in the Laboratory-Based Model. J Am Coll Surg, Vol. 193. (2001) 479-485
5. Hu, J., Brown, M. K., Turin, W.: Handwriting Recognition with Hidden Markov Models and Grammatical Constraints. In Proc. 4th IWFHR, Taipei, Taiwan. (1994) 195–205
6. Hundtofte, C. S., Hager, G. D., Okamura, A.M.: Building a Task Language for Segmentation and Recognition of User Input to Cooperative Manipulation Systems. 10th Int. Symp. on Haptic Interfaces for Virtual Environment and Teleoperator Systems. (2002) 225-230
7. Li, M., Okamura, A.M.: Recognition of Operator Motions for Real-Time Assistance using Virtual Fixtures. 11th International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. (2003) (in press)
8. Liu, A., Salvucci, D. D., Modeling and prediction of human driver behavior. In Proceedings of the Ninth International Conference on Human-Computer Interaction. Mahwah, NJ: Lawrence Erlbaum Associates. (2001) 1479-14839.
9. Rabiner, L.: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proc. Of the IEEE, Vol. 77 (2). (1989) 257-86
10. Rosen, J., et al.: Task Decomposition of Laparoscopic Surgery for Objective Evaluation of Surgical Residents' Learning Curve Using Hidden Markov Model. Computer Aided Surgery, Vol. 7 (1). (2002) 49-61
11. Starner, T., Pentland, A.: Visual Recognition of American Sign Language Using Hidden Markov Models. Int. Workshop Automatic Face Gesture Recognition. (1995) 189–194
12. Verner, L., Oleynikov, D., Holtman, S., Haider, H., Zhukov, L.: Measurements of the Level of Expertise Using Flight Path Analysis from da Vinci Robotic Surgical System. Medicine Meets Virtual Reality (2003)
13. Wilhelm, D., et al.: Assessment of Basic Endoscopic Performance Using a Virtual Reality Simulator. J Am Coll Surg, Vol. 195. (2002) 675-681
14. Yamauchi, Y., et al.: Surgical Skill Evaluation by Force Data for Endoscopic Sinus Surgery Training System. MICCAI, Lecture Notes in Computer Science, Vol. 2488. (2002) 35-43