Modeling and Online Recognition of Surgical Phases Using Hidden Markov Models

Tobias Blum[1], Nicolas Padoy[1,2], Hubertus Feußner[3], and Nassir Navab[1]

[1] Computer Aided Medical Procedures (CAMP), Technische Universität München, Germany
[2] LORIA-INRIA Lorraine, Nancy, France
[3] Department of Surgery, Klinikum Rechts der Isar, Technische Universität München, Germany

Abstract. The amount of signals that can be recorded during a surgery, like tracking data or state of instruments, is constantly growing. These signals can be used to better understand surgical workflow and to build surgical assist systems that are aware of the current state of a surgery. This is a crucial issue for designing future systems that provide context-sensitive information and user interfaces.

In this paper, Hidden Markov Models (HMM) are used to model a laparoscopic cholecystectomy. Seventeen signals, representing tool usage, from twelve surgeries are used to train the model. The use of a model merging approach is proposed to build the HMM topology and compared to other methods of initializing a HMM. The merging method allows building a model at a very fine level of detail that also reveals the workflow of a surgery in a human-understandable way. Results for detecting the current phase of a surgery and for predicting the remaining time of the procedure are presented.

1 Introduction

Over the last decades an immense amount of information has become available in the OR. Novel imaging modalities, image processing methods and electronic patient recordings allow obtaining and accessing huge amounts of data. This development is to appreciate but also raises new challenges. During surgery, clinical personnel have to cope with information overload while experiencing high stress. Often, technological innovations are not well integrated into surgical workflow and there are deficiencies in effective information handling, which is also subject of recent critique [1,2]. Low acceptance concerning systems that are driven by emerging technologies and not by the need of the surgeons has been criticized [3]. To develop better solutions, that are well integrated into the surgical workflow and are adapted to the needs of the surgeon, there are several issues to deal with.

In this work statistical models are discussed that represent a surgery based on signals that are obtained from the OR. Such a model can be used in many ways. Offline for analyzing surgeries in order to better understand their workflow. Online for recognizing the current phase of a surgery, which can be used

for providing context-sensitive user interfaces, monitoring or predicting the remaining duration of a surgery. With signals, we refer to any kind of information that can be obtained automatically from the OR. This includes e.g. the use of surgical instruments, the state of devices, biomedical data or the position of the surgeon. In this work, Hidden Markov Models are used to model a surgery based on such signals. Instead of using simple HMM topologies we propose using a model merging approach to automatically derive the HMM topology from the data. An important advantage compared to simpler methods for initializing an HMM, is that with this approach we obtain a human-interpretable model of a surgery. In [4] it was already discussed how to use HMMs for visualizing workflow at different levels of detail. In this work we discuss how to recognize the current phase of a running surgery. The method is evaluated based on the capability to recognize the current phase of a laparoscopic cholecystectomy. Also results for predicting the remaining duration of a procedure are presented.

In the medical domain, HMMs have been used to model surgical activities for skill evaluation. In [5] 6-DOF data was recorded from a laparoscopic simulator and used to train four-state HMMs for classifying subjects according to their skill level. A Markov Model was build in [6] based on position, orientation, force and torque data of two endoscopic tools. Using the Markov Model, difference in the skill of subjects with different levels of training could be shown. Segmentation of surgical motions recorded from the da Vinci robot for skill evaluation has been done in [7]. In [8] video images were used to detect a patient entering or leaving the room, and the beginning and end of an operation. These four states were recognized using a combination of support vector machines and HMMs. Information obtained with an eye-tracker combined with visual features were used in [9] to recognize start and end of one surgical phase in a porcine laparoscopic cholecystectomy with an accuracy of 75%. In previous work, we already have segmented surgical phases of a laparoscopic cholecystectomy using a combination of Dynamic Time Warping (DTW) and AdaBoost [10]. While this method achieved an accuracy of 99.7%, it is limited to offline use, as the whole surgery must be recorded before DTW can be applied for segmentation.

2   Method

First we briefly describe the surgery and signals that are used in this work. Then the general method for recognizing the current phase of a running surgery using a HMM is explained. Next, simpler HMM topologies and the model merging approach are explained.

2.1   Setup and Signals

Laparoscopic cholecystectomy is a common but complex surgery. With 95% of the cases performed laparoscopically it is one of the most established minimally invasive surgeries. The objective is to remove the gallbladder. Most of the signals
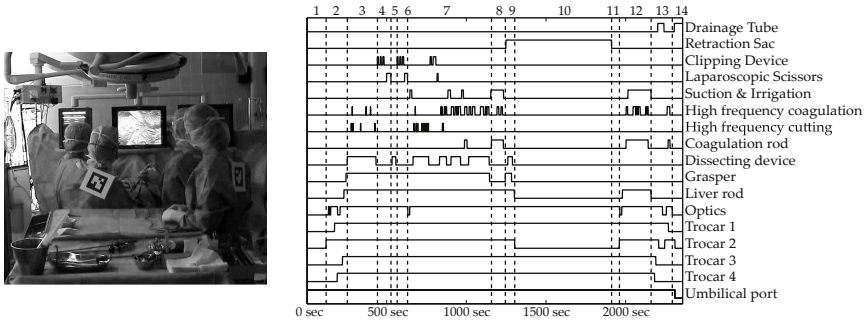
Fig. 1. View of the operating room (a) and the signals representing the instrument use during one surgery (b). The dotted lines indicate the 14 surgical phases.

used in this work are binary and represent the use of surgical instruments. Additional signals are whether one of the four trocars is inserted and high frequency (HF) coagulation and cutting, which are performed by applying a current to an instrument. This data was collected for 12 surgeries $\mathbb{O} = \{\mathbb{O}_1, \mathbb{O}_2 \ldots \mathbb{O}_{12}\}$, where $\mathbb{O}_i = O_{i1}O_{i2} \ldots O_{iT}$ and T is the length of a surgery i in seconds. Each O_{it} is a vector consisting of 17 binary values, representing instrument use. These instrument vectors will also be referred to as observations. An example of this can be seen in figure 1. Each surgery was segmented into 14 surgical phases. The surgeries were performed by three different surgeons in several operating rooms. All surgeries were videotaped and the use of the instruments was labeled manually afterwards. However, all of the signals can be obtained automatically. Information like the use of HF coagulation and cutting can be taken directly from the devices. The main challenge is to obtain the information about instrument use. This could be done using e.g. computer vision, barcodes or radiofrequency tags. A solution that is currently under development together with our medical partner is to use trocars equipped with a sensor that can detect instruments that are inserted and removed.

2.2 Hidden Markov Models

A HMM is a statistical model, describing a process over time. It is defined by the parameters $\lambda = (N, M, A, B, \pi)$, where N is the number of hidden states $S_1, S_2, \ldots, S_N$; M the number of observation symbols, A the transition probability distributions, B the observation probability distributions and π the initial state distribution. Given a HMM λ and a partial observation sequence $\mathbb{O}_{it} = O_{i1}O_{i2} \ldots O_{it}$, where $t \leq T$, the probability of the HMM having generated the sequence, $P(\mathbb{O}_{it}|\lambda)$, can be computed using the forward algorithm. While doing this, also the probability of being in one of the hidden states S_n at time t, $P(S_n|\mathbb{O}_{it}, \lambda)$ is computed. To estimate the current phase, we

introduce the phase probability distribution c_n. Given that we are in state S_n, the probability of being in phase i is given by $c_{ni} = P(phase = i|S_n)$. Using

$$\sum_{n=1}^{N} P(phase = i|S_n)P(S_n|\mathbb{O}_{it}, \lambda), \tag{1}$$

we compute the probability of being in phase i, after having observed the surgery up to time t.

The straightforward way of using this, would be to build a 14-state left-to-right HMM where each state represents one phase and it is only possible to stay in one state or move on the state representing the next phase. The phase probability distribution is directly set to $c_{ii} = 1$ and to 0 for all other phases. The transition and observation probabilities can be derived from the segmented training data by counting the length of the phases and the number of observations per phase. As can be seen later in the results section such a simple model already allows automatic phase recognition, but shows room for improvement.

As some phases consist of many actions, it would be appropriate to use multiple states to model one phase. In [11] we used a left-to-right HMM on similar data, where the number of states per phase is based on the average length of the phase. The training sequences for one phase are divided into subsequences of equal length. Transition and observation probabilities for each state are computed based on the length of the subsequences and the number of observations per subsequence. The phase probability is set to 1 for the corresponding phase. After building one HMM per phase, they are concatenated to obtain a model of the whole surgery that is capable of recognizing the current phase. This model shows good results, but has one important drawback. While the number of states per phase is related to the length of the phase, there is no direct relationship between HMM states and the surgical work steps a human would recognize.

2.3   Model Merging

Instead of using the method described before we propose to use a model merging approach, to generate a HMM that models the surgical workflow with much more detail and in a way that can be interpreted by humans. The method we are using is based on the model merging approach by Stolcke and Omohundro [12,13]. The idea is to build an initial model that represents every action from every training example by one state. Such a model exactly explains every step that occurred in the training data and is thus completely overfitted. By iteratively merging two states, a more compact model is generated that still explains the training data well, but does generalize better over unseen data.

The model is build phase-wise by first generating one left-to-right HMM for each training surgery. Each time one of the binary value changes, a new state is added to this HMM. Transition probabilities are set according to the duration of the actions. The initial model λ_0 is built by taking all these paths, each describing one surgery, in parallel. An illustrative example, using only a short action sequence from two surgeries is shown in the upper part of figure 2.
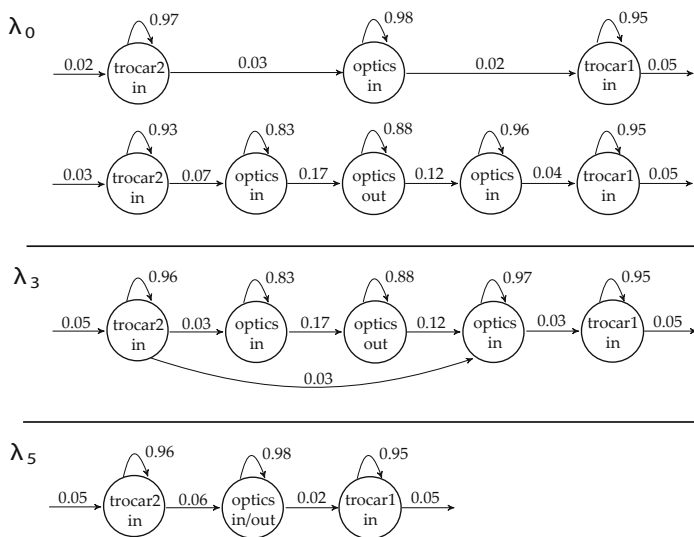
Fig. 2. This is an example of how the model merging would work using only two training surgeries. At the beginning of phase two, usually trocar 2 is placed, the optics is inserted and then trocar 1 in placed. In some cases the optics is taken out in between to clean it. For better comprehensibility, the nodes are not labeled with the binary instrument vectors, but with the last instrument that changed. The upper part of the image shows the initial model that is build from two surgeries and contains two parallel paths. In the middle part, the model is shown after three merging steps. The states *trocar2 in* , *trocar1 in* and *optics in* have been merged. In the lower part, the model after two additional merging iterations is shown.

Obviously, λ_0 is of no use for detecting the current phase, as it only explains the training data. To obtain a more compact HMM that generalizes better, states are iteratively merged based on the probability of explaining the training data. In each step i a new HMM λ_i is built, by merging two states from λ_{i-1}. using a best-first heuristic. For all pairs, $P(\mathbb{O}|\lambda_i)$ is computed, and the pair giving the highest probability is chosen. In figure 2, the results after three and after five merging steps are shown. In this example, $P(\mathbb{O}|\lambda_3)$ is only slightly lower than $P(\mathbb{O}|\lambda_0)$ and so λ_3 still explains the training data well. $P(\mathbb{O}|\lambda_5)$ is considerably lower as it has to represent two different observation with one state. Every additional merging step would drastically lower this probability. In order to obtain a compact HMM that still explains the training data well, we abort the merging as soon as the decrease in $P(\mathbb{O}|\lambda_i)$ reaches a threshold. As this value decreases very slowly at the beginning and very fast later, the exact value of this threshold has only little effect. To obtain a model that reflects the consecutive characteristics of the workflow, we have chosen to only allow merges that do not lead to loops in the HMM. This constraint did not change the results of the online phase recognition significantly, but leads to a model that can be interpreted more intuitively by humans.

The merging process is computationally very demanding, as in every step the forward probability of the HMM must be computed for every pair of states. To speed up this algorithm, some approximations, that have already been proposed in [13] were used, namely the Viterbi path approximation and optimistic path updating. As we have a very sparse HMM, where most transition probabilities are 0, we also used a graph implementation to speed up computation. To build a model of the whole surgery, we run the model merging for each phase independently and set the phase probability distribution accordingly. After the merging process is finished, expectation maximization is used to refine the transition and observation probabilities. Next, the models of all phases are simply concatenated, to build a model that is capable of recognizing the current phase of a surgery.

3   Results

We have done a complete cross-validation, where the model was trained on all but one surgery and evaluated by remaining one. At each time t the phase with the highest probability was computed based on the partial sequence $\mathbb{O}_{it}$, as described in the last section. In table 1 the overall error and the mean error per phase are shown. The overall error is the percentage of time steps, where the phase with the highest probability differs from the ground truth. Some phases are extremely short, and even an error of some seconds would result in not recognizing most of this phase correctly. To account for this, the mean error per phase is computed based on the percentage of errors per phase.

We compared the results to both methods described in section 2.2. The model merging approach has clearly outperformed the 14-state model. Splitting the sequences into equal subsequences for initializing several states per phase has shown results that are comparable to the merging approach. However, the model merging has important advantages. The topology of the HMM allows humans to analyze the workflow. The constraint to build models without loops results in very comprehensible models, as can be seen in figure 3. This phase consists of one to four uses of the clipping device followed by the use of the scissors, which is also reflected in this model. As discussed in [4], this can be used for analyzing the workflow and statistical properties of a surgery. The labels in figure 3 have been created automatically by naming the state with the lowest number of instruments *idle*. All other states are labeled by the additional instruments. Such a visualization is not only important for analyzing surgical workflow but also for designing context-sensitive systems. It is possible to select one, or several states for triggering some action like displaying some information which is important for these steps. Using simpler HMM topologies, this would only be possible at the level of phases and not of single actions.

One application where automatic phase recognition could be used, is to predict the remaining time of a surgery. This is important for managing perioperative activities like anesthesia and scheduling of ORs and personnel. A HMM is a generative model, so we can use it to simulate the process it describes. To predict the remaining time, the most probable current state is computed and the remainder

Table 1. Errors obtained during a cross-validation on all 12 surgeries

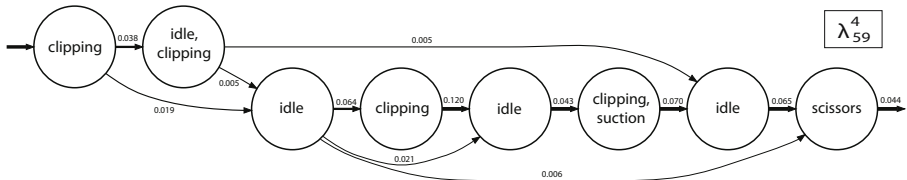|	overall error	mean error per phase
14 state HMM	14.01%	17.36%
subsequence initialization	7.49%	8.41%
model merging	6.73%	8.90%

Fig. 3. A visualization of the HMM of phase four that has been obtained using the merging method. For a clearer visualization, the probability of staying in one state has been omitted. This model was obtained after performing 59 merging steps.
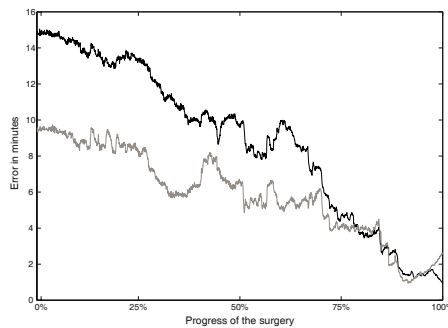

Fig. 4. Prediction of the remaining duration of a surgery. The darker line shows the error in minutes after a certain percentage of the surgery has been carried out. The lighter line, shows the standard deviation.

of the surgery is simulated several times. The remaining time is predicted as the average length of these simulations. If we would simply compute the average length of the surgeries, and assume that a new surgery would have the same length, the average error would be 14min 40sec. As can be seen in figure 4, the error at the beginning of a surgery coincides with this value. This is natural, as we do not have any information and can only assume the average duration. However, as the surgery goes on, our prediction gets better. Especially in the last quarter of a surgery, the prediction is very reliable. These results are based on a complete cross-validation. Note that the error is made of two components. The error in estimating the current state, and the error of predicting the remaining time from this state.

4   Discussion

In this work we have discussed different HMM topologies to model surgical workflow and detect the current phase. We propose the use of a model merging approach that is capable of detecting phases and delivers a very detailed model. Such a model allows detecting work steps at a finer level and is also interpretable by humans. This method has delivered good results, although only a small set of training data was available. The method is not limited to signals representing instrument use. When using vector quantization to build the initial model λ_0, it can also be applied to continuous data like tracking data or biomedical signals.

As the number of signals that can be obtained during a surgery is steadily increasing, methods for handling them are of growing interest. We believe that the use of such statistical models will be of great importance for developing future surgical assist system. They allow to better understand the workflow and to build systems that adapt to the current state of a surgery.

References

1. Plsek, P.: Complexity and the adoption of innovation in health care. In: Accelerating Quality Improvement in Health Care (2003)
2. Cleary, K., Chung, H.Y., Mun, S.K.: Or 2020: The operating room of the future. Laparoendoscopic and Advanced Surgical Techniques 15(5), 495–500 (2005)
3. Neumuth, T., Strauß, G., Meixensberger, J., Lemke, H., Burgert, O.: Acquisition of process descriptions from surgical interventions. In: Database and Expert Systems Applications, pp. 602–611 (2006)
4. Blum, T., Padoy, N., Feußner, H., Navab, N.: Workflow mining for visualization and analysis of surgeries. In: Computer Assisted Radiology and Surgery (2008)
5. Leong, J., Nicolaou, M., Atallah, L., Mylonas, G., Darzi, A., Yang, G.-Z.: Hmm assessment of quality of movement trajectory in laparoscopic surgery. In: Larsen, R., Nielsen, M., Sporring, J. (eds.) MICCAI 2006. LNCS, vol. 4190, pp. 752–759. Springer, Heidelberg (2006)
6. Rosen, J., Brown, J., Chang, L., Sinanan, M., Hannaford, B.: Generalized Approach for Modeling Minimally Invasive Surgery as a Stochastic Process Using a Discrete Markov Model. IEEE Trans. on Biomed. Eng. 53(3), 399–413 (2006)
7. Lin, H., Shafran, I., Yuh, D., Hager, G.: Towards automatic skill evaluation: Detection and segmentation of robot-assisted surgical motions. Computer Aided Surgery 11(5), 220–230 (2006)
8. Bhatia, B., Oates, T., Xiao, Y., Hu, P.: Real-Time Identification of Operating Room State from Video. Innovative Applications of Artificial Intelligence, 1761–1766 (2007)
9. James, A., Vieira, D., Lo, B., Darzi, A., Yang, G.Z.: Eye-Gaze Driven Surgical Workflow Segmentation. In: Ayache, N., Ourselin, S., Maeder, A. (eds.) MICCAI 2007, Part II. LNCS, vol. 4792, pp. 110–117. Springer, Heidelberg (2007)

10. Padoy, N., Blum, T., Essa, I., Feußner, H., Berger, M.O., Navab, N.: A boosted segmentation method for surgical workflow analysis. In: Ayache, N., Ourselin, S., Maeder, A. (eds.) MICCAI 2007, Part I. LNCS, vol. 4791, pp. 102–109. Springer, Heidelberg (2007)
11. Padoy, N., Blum, T., Feußner, H., Berger, M.-O., Navab, N.: On-line recognition of surgical activity for monitoring in the operating room. In: Innovative Applications of Artificial Intelligence (2008)
12. Stolcke, A., Omohundro, S.: Inducing probabilistic grammars by Bayesian model merging. In: Grammatical Inference and Applications, pp. 106–118 (1994)
13. Stolcke, A., Omohundro, S.: Best-first model merging for hidden markov model induction. Technical Report TR-94-003, ICSI, Berkeley, CA (1994)