

# On-line Recognition of Surgical Activity for Monitoring in the Operating Room

N. Padoy<sup>1,2</sup> and T. Blum<sup>1</sup> and H. Feussner<sup>3</sup> and M-O. Berger<sup>2</sup> and N. Navab<sup>1</sup>

<sup>1</sup>Technische Universität München  
Fakultät für Informatik / I16  
Boltzmannstrasse 3  
85748 Garching bei München  
Germany  
{padoy,blum,navab}@cs.tum.edu

<sup>2</sup>LORIA-INRIA Lorraine  
BP 239  
54506 Vandoeuvre-lès-Nancy Cedex  
France  
{padoy,berger}@loria.fr

<sup>3</sup>Klinikum Rechts der Isar  
Chirurgische Klinik und Poliklinik  
Ismaninger Str. 22  
81675 München  
Germany  
feussner@nt1.chir.med.tu-muenchen.de

## Abstract

Surgery rooms are complex environments where many interactions take place between staff members and the electronic and mechanical systems. In spite of their inherent complexity, surgeries of the same kind bear numerous similarities and are usually performed with similar workflows. This gives the possibility to design support systems in the Operating Room (OR), whose applicability range from easy tasks such as the activation of OR lights and calling the next patient, to more complex ones such as context-sensitive user interfaces or automatic reporting. An essential feature when designing such systems, is the ability for on-line recognition of what is happening inside the OR, based on recorded signals.

In this paper, we present an approach using signals from the OR and Hidden Markov Models to recognize on-line the surgical steps performed by the surgeon during a laparoscopic surgery. We also explain how the system can be deployed in the OR. Experiments are presented using 11 real surgeries performed by different surgeons in several ORs, recorded at our partner hospital.

We believe that similar systems will quickly develop in the near future in order to efficiently support surgeons, trainees and the medical staff in general, as well as to improve administrative tasks like scheduling within hospitals.

## Introduction

The surgery room is a crucial unit within the hospital, where many changes have been predicted to occur in the next years (Cleary, Chung, & Mun 2005). While many different high-end technologies are made available to the surgeon, they are up to now neither connected to a common interface nor do they include any complete monitoring system. With the increase and improvements of minimally invasive surgeries, that rely on new tools and new imaging technologies, many electronic signals are however made easily available. They could be used for the design of a system that is aware of the surgical context. It could recognize from these signals what is occurring in the operating room (OR) and use this information to automatically write a report at the end of the surgery, present a context sensitive user interface and trigger simple events or reminders, like calling the next patient or switching on and off the lights. These two last actions are

often a source of hassle in busy ORs. Calling the next patient in time is also a huge healthcare issue, since, if done too soon, the next patient will stay longer anaesthetized as needed. If done too late, the OR will remain unused for some time. Furthermore, providing contextual information to the surgical staff, for example which tools need to be prepared next, would also be particularly valuable for trainees or new personnels. A context-aware system could therefore bring clear benefits to the OR. But additionally, it would also improve the overall workflow of the hospital as the information can be made available to the administration, for scheduling or computing overall statistics.

While surgeries are complex operations, surgeries of the same kind are often performed with a similar and reproducible workflow. A system can therefore be trained from surgeries of the same kind. This is especially relevant as the standardization of the workflow and the specialization of the ORs to certain kind of surgeries have been identified as an important source of improvement to the hospital efficiency (Herfarth 2003).

We present in this paper an approach for monitoring laparoscopic surgeries, which belong to the most common minimally invasive procedures. They are performed using tools inserted through small incisions in the patients' body, with one incision reserved for the laparoscope, a camera observing the operating field. Compared to open surgery, this results in reduced patient trauma, shortened hospitalization and decreased risk. On the other side, the field of view of the surgeon is reduced, the operation lasts longer and emergency cases are more complicated to handle. The huge advantage of laparoscopic surgeries for a monitoring system is the usage of many electronic systems, which can easily provide signals about the surgical actions without having to change the workflow or to install complex tracking systems.

We explain in the following how to process information from the used tool and from the endoscopic camera to detect on-line which phase of 14 surgical phases is taking place. The method uses a left-right Hidden Markov Model (HMM) and is evaluated on the example of the laparoscopic cholecystectomy. A cross-validation on 11 real surgeries shows an on-line detection rate of 93%. We further discuss its deployment in the OR and a few useful applications, like triggering easy events and predicting the remaining time of the surgery.

## Related Work

Interest in workflow analysis inside the OR is recent. Most works either focus on the manual modeling of the interactions during the surgery or on the analysis of single movements.

(Jannin *et al.* 2001; Neumuth *et al.* 2006) address the understanding of a complete surgical workflow through manual modeling with the Unified Modeling Language (UML) or ontologies. Up to now the models are not related to real signals from the surgeries. It therefore allows off-line statistical analysis, but no monitoring. Surgical movements analysis is also of high interest, as it can be used for an objective evaluation of the surgical skills and for training. In (Rosen *et al.* 2006) and (Leong *et al.* 2006), force/torque or tracking information from endoscopic instruments is used to train Hidden Markov Models representing different surgical skill levels. They focus on a specific action performed on a simulator. In (Lin *et al.* 2005), a method to segment the steps of a knotting movement is presented. It uses data obtained from a robotic telemanipulator applied on a phantom and also serves for the evaluation of the performance.

Recognition of several events occurring in endoscopy has been addressed in (Lo, Darzi, & Yang 2003): visual cues from endoscopic images and a bayesian framework are used to classify four events including cauterisation and suturing. A further step is taken by (James *et al.* 2007), where information from an eye-gaze tracking system is combined with visual features in a neural network to detect one phase of a cholecystectomy performed on pigs, namely the clipping of the cystic duct.

With the objective to analyse the occupancy of the OR, (Bhatia *et al.* 2007) process external video information with Support Vector Machines and a 4-state HMM to classify the room occupancy into four elementary states. Eventhough the proposed method could be used on-line, only segmentation results using information from the complete data sequence are presented. Our approach use additional signals to be able to get more insight into the workflow and to monitor on-line the surgery.

Outside the hospital, many complex systems, often involving graphical models, have been proposed for the recognition of human activities, usually using continuous data from various sensors. (Oliver, Garg, & Horvitz 2004) propose for instance layered HMMs for real-time activity recognition in an office environment. (Shi *et al.* 2004) presented an approach to recognize concurrent activities, demonstrated on the monitoring of a blood glucose calibration system for elderly people at home.

In previous works of the authors (Padoy *et al.* 2007b; 2007a), a subset of the presented signals, where no automatic information from the endoscopic camera is involved, was used with a modified version of the Dynamic Time Warping algorithm or with 14-state HMMs to segment off-line the surgeries. To the best of our knowledge, there is no published work addressing the on-line recognition of surgical tasks during a complete surgery. A major difference to the related works cited above is the nature and the little amount of training data that can usually be obtained.



(a) Laparoscopic environment, with instrument table and video monitor.



(b) Dissection and suction-irrigation devices, as seen from the optics.

Figure 1: Instruments within external and endoscopic views from the surgical field.

## Medical application

While the method can be applied to any laparoscopic surgery and to other surgeries where a way to acquire signals is available, we focus for the experiments on laparoscopic cholecystectomy. This is a common but complex surgery performed laparoscopically in 95% of the cases, with a low conversion rate to open surgery. It is therefore convenient for the recording of the signals as well as for the demonstration of the method. The objective of cholecystectomy is to remove the gallbladder. It starts with the positioning of trocars on the patient, for insertion of the instruments inside the body, and finishes with their removal and the suturing of the induced holes. The most important intermediate steps are the dissection, clipping and cutting of the bile duct and of the cystic artery. The gallbladder is then separated from the liver and removed using a retraction sac. For it to pass through the endoscopic hole, the gallstones that caused the operation are removed one by one from it beforehand. This is followed by a final control phase of the abdominal area and the removal of all instruments. A view of the OR during such a surgery is shown in figure 1(a). All the worksteps from the insertion of the trocars till the suturing are displayed in table 1 and will be used for the evaluation. Two simple examples of events

1	CO2 Inflation	8	Liver Bed Coagulation 1
2	Trocar Insertion	9	Gallbladder Packaging
3	Dissection Phase 1	10	External Retraction
4	Clipping Cutting 1	11	External Cleaning
5	Dissection Phase 2	12	Liver Bed Coagulation 2
6	Clipping Cutting 2	13	Trocar Retraction
7	Gallbladder Detaching	14	Abdominal Suturing

Table 1: The fourteen phases used in the on-line detection.

to be triggered in this surgery are *calling the next patient* in phase 7 or *switching on the lights* in phase 10.

## Methods

### Signals

For monitoring, the signals recorded from the OR at a resolution of one second are represented by a multidimensional time-series  $\mathbb{O}$  where  $\mathbb{O}_t \in \{0, 1\}^K$ :

$$\mathbb{O}_{t,k} = 1 \text{ iff signal } k \text{ is active at time } t$$

These binary vectors contain two kinds of signals. In the  $K = 18$  signals available, 16 of them indicate what instruments are used in the surgery. The remaining two signals are derived automatically from the view of the endoscopic camera. Its images are however very challenging for vision algorithms as many perturbations occur, such as strong specularities, appearance of smoke, tissue deformations, occlusions and fast change of field of view. The first signal indicates whether the camera is present in the body and the second if clips were detected in the field of view. Signals for one surgery are displayed in figure 3.

**Instrument signals** These signals indicate which endoscopic tools are used at each time. They include the trocars and instruments such as the grasper, the cutting device or the high-frequency coagulation device, which are all displayed in figure 3. For this work, they were labeled manually because of administrative issues. It is however technically possible to obtain them automatically. This issue is further addressed in the discussion section.

**Endoscopic camera signal** As can be guessed from figures 1(a) and 1(b), colors are a main cue to obtain the state of the endoscopic camera. Since cameras with different settings are used, a color normalization (Paulus, Csink, & Niemann 1998) is performed on the images beforehand, which proved to improve the results. We used a small color histogram of size 20 as visual feature. The first 10 contain a hue histogram and the last 10 a saturation histogram of the image. Based on labelled images from training surgeries, two Gaussian Mixture Models (GMMs)  $G_{endo}$  and  $G_{out}$  are trained to model the color spaces of endoscopic images and outside images. The training consists in an initialization with principal component analysis followed by Expectation Maximization (EM) iterations. An image is classified as endoscopic when the probability of its histogram to belong to

$G_{endo}$  is higher than for  $G_{out}$ . Evaluation based on a few manually labelled complete surgeries provides a success rate of 92%.

Two kinds of images are difficult in the evaluation: the images where the camera is half inserted and as much of the metallic trocar can be seen as of the internal anatomy, and the images where the camera is entirely inside the abdomen but fully blinded by specularities due to its proximity to the tissue.

**Clip detection signal** Clips are small longitudinal metal objects used to close the blood vessels. In all the surgeries we recorded, dark blue clips are used, sometimes with additional stronger grey clips. To detect the blue metallic clips, we use color classification with a similar approach as before followed by few morphologic operations and a decision based on shape properties. Two GMMs  $G_{clips}$  and  $G_{bg}$  modeling the colors of the clips and of the endoscopic background are constructed based on a few training images segmented manually. They are used in a first step to classify the image pixels into clip and background. After morphological closing operations, the connected components are selected to be clips depending on two properties: sufficient size and longitudinal shape. When the endoscopic camera is detected to be in the body and such a connected component is found, the clip signal is set as active. Note here, that wrong positives in the detection are more an issue than wrong negatives. Indeed, detected clips before their introduction in the body give a wrong indication about the possible phase. It however often occurs that clips are small or hidden from the field of view after their introduction in the body. For this reason, the criteria are set such that clearly visible clips are detected, while far or partially hidden clips might not be detected. This reduces the number of wrong positives, which may occur as the dark blue clips have a close color to coagulated tissues and smoke and are also sensitive to specularities and reflections. The recognition model appeared however to cope with these wrong detections. The evaluation is not easy, as manual labeling of the clips is tedious and has to be done image per image, contrary to the instruments for which we only consider the presence in the body, not in the field of view. On a few manually labeled surgeries, where also far and hardly visible clips were noted as present, the detection rate is 76%.

### On-line segmentation model

As most signals get activated in several different phases, sometimes even differently by different surgeons, the phases cannot be recognized simply from the signals. This requires a learning algorithm including a temporal model. In order to detect on-line which surgical phase is taking place, we represent the surgery as a stochastic process using a left-right hidden markov model. The model is constructed from signals  $\mathbb{O}_{t,k}^l$ , where  $l \in 1 \dots L$  and  $L$  is the number of surgeries available in the training set. For each surgery  $l$  in the training set and for each time  $t$ , the phase  $\mathcal{T}(\mathbb{O}^l, t)$  that was taking place at that time is supposed to be known.

Supposing now  $\mathbb{O}_{t,k}^{test}$  are the signals from a new surgery

that has to be monitored, the objective is to compute  $\mathcal{T}(\mathbb{O}^{test}, t)$  at each time step  $t$  while knowing only the partial signals  $\mathbb{O}_1^{test} \dots \mathbb{O}_t^{test}$  up to the actual time.

For this, a suitable Hidden Markov Model (Rabiner 1989) with parameters  $\lambda$  is constructed. A discrete Hidden Markov Model is a quintuplet  $(N, M, A, B, \pi)$  where  $N$  is the number of states  $(x_i)_{1 \leq i \leq N}$  in the model,  $M$  the size of the observation alphabet,  $A$  the transition probability matrix between the states,  $B$  the observation probability matrix and  $\pi$  a probability distribution over the initial states. Our model  $\lambda$  is constructed out of models that each represent one phase of the surgery. We first present how to use the model  $\lambda$  for monitoring. Its construction is explained in the next paragraph.

We are looking for

$$\begin{aligned} \mathcal{T}_{test}(t) &= \mathcal{T}(\mathbb{O}^{test}, t) \\ &= \underset{u}{\operatorname{argmax}} P(\text{phase} = u | \mathbb{O}_1^{test} \dots \mathbb{O}_t^{test}) \end{aligned}$$

Supposing we know from the construction of the model the probabilities  $P(\text{phase} = u | X = x)$  of being in phase  $u$  while being in the HMM state  $x$ , we have

$$\mathcal{T}_{test}(t) = \underset{u}{\operatorname{argmax}} \sum_{x_i} P(\text{phase} = u \mid X_t = x_i) \times P(X_t = x_i \mid \mathbb{O}_1^{test} \dots \mathbb{O}_t^{test})$$

where  $P(X_t = x_i | \mathbb{O}_1^{test} \dots \mathbb{O}_t^{test})$  can be computed from the so-called forward probabilities using dynamic programming.

## Model construction

In this paragraph, we explain the construction of  $\lambda$  and  $P(\text{phase} = u | X = x)$ . For each phase  $u$  a HMM with parameters  $\lambda_u$  is constructed out of the corresponding training data. The training data are made up of all subsequences of signals belonging to the phase  $u$  from each training surgery:  $\mathbb{O}_{t,k}^u$  where  $\mathcal{T}(\mathbb{O}^t, t) = u$ . These models  $\lambda_u$  are then appended to each other as in figure 2.

There exist numerous works on how to learn and train the parameters of an HMM. As very little training data is available, we take benefit from the discrete case and from the linear course of actions to model the phase as a left-right HMM that can this way be directly initialized from the data. As rule of the thumb, we choose the number of states to be the square root  $\sqrt{\frac{E}{2}}$  where  $E$  is the mean phase duration calculated from the training data. It performs better than choosing any constant, but other classical rules for model selection like  $(\frac{E}{\log(E)})^{\frac{1}{3}}$  perform equivalently well on this data.

The sub-models are trained by dividing each sequence of the training data into equal subsequences and by assigning them to the respective state of the submodel. The transition probabilities are then set such that the expected duration of staying in one state equals the mean duration of the subsequences assigned to the state. The observation probabilities are directly learned from these subsequences. The observation distribution is chosen to be the frequency of occurrences, while allowing a small probability for an unknown

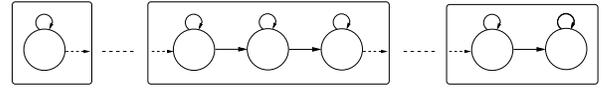


Figure 2: Example of appended HMMs.

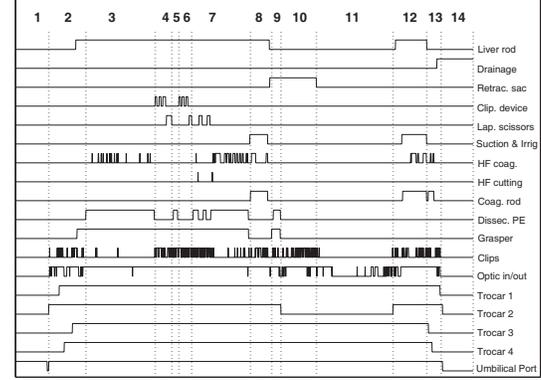


Figure 3: Temporal sequence of signal vectors for one surgery, with indication of the 14 phases.

observation. Assuming signals to be independent in the observation distribution has shown to yield equivalent results.

When the HMMs  $\lambda_u$  are appended, the last state of  $\lambda_u$  is modified to have a transition to the first state of  $\lambda_{u+1}$ . The probability is chosen so that the expected time in the phase model equals the mean duration of the phase, computed from the training data. The function  $P(\text{phase} = u | X = x, \lambda)$  is then defined to be 1 for all states  $x$  from  $\lambda$  that were appended from the submodel of phase  $u$  and to be 0 for the others. The initial probability distribution is initialized for the HMM to start in its first state.

It is possible to refine the resulting model  $\lambda$  with the Expectation Maximization algorithm and to update the function  $P(\text{phase} = u | X = x, \lambda)$  accordingly at the same time. We however choose not to do it, as it sometimes worsen the results. The reason is that very short phases with observations also appearing in the neighbouring phases tend during the EM steps to be absorbed by the neighbouring ones, as they are modeled by a few states. This also comes from the fact that we are interested in the phase segmentation when we construct the model, which is not directly correlated to the log-likelihood of the training data that is optimized by the EM.

## Experiments and Results

### Recording Setup

We recorded video information from 11 real surgeries, performed by three different surgeons in several operating rooms of the same hospital. An external video including surgical staff, instrument table and situs, as well as the view of the endoscopic camera were recorded synchronously during

these surgeries.

The usage of laparoscopic instruments is, at the moment, derived manually from the video data using a labeling software. So far, they cannot be obtained automatically from the videos without human knowledge, since the laparoscopic tools look very similar and can only be distinguished by their tooltips which are very small. Moreover, many occlusions and strong illumination changes occur in the external video, due to personal movements and to the status of the OR lights. In the laparoscopic video, in addition to the aforementioned perturbations occurring in the images, not all the instruments are present in the field on view. Automatic tool identification is however technically possible, as addressed in the discussion section.

## Results

The method is evaluated using a full cross-validation on the data: the framework is trained on each available subset of 10 surgeries and evaluated on the remaining one. The objective being the *on-line* recognition of the surgical phases, we use three kinds of errors:

- **overall classification error** - percentage of wrong detections compared to ground truth information in the complete surgery
- **mean error per phase** - mean between the percentages of wrong detections inside each phase
- **number of skipped phases** - number of phases that were completely detected as another phase

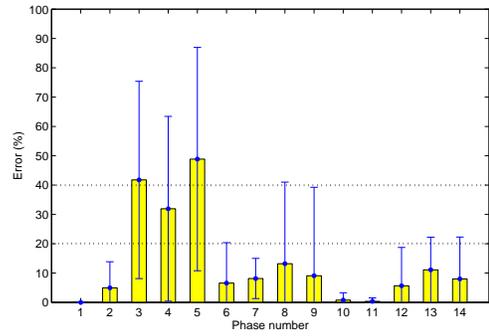
While computing the errors, a tolerance of 5 seconds before and after the ground truth definition of the phase is set. It is required as phase changes are defined at changes of instruments, but in practice instrument change (removing the past instrument from the trocar, grasping the next one from the scrub nurse and inserting it in the trocar) takes in average 10 seconds. Moreover, detection errors inside a phase are more important than at the loose border.

Results of the cross-validation are displayed in table 2. Each row shows the mean errors of the cross-validation tests for a different HMM. The model  $\lambda$  computed as explained in the method section has the lowest errors. For comparison, the errors computed with the model  $\lambda^{(1)}$  where only one state per phase is used, or  $\lambda^{(2)}$  with two states per phase, are also displayed.

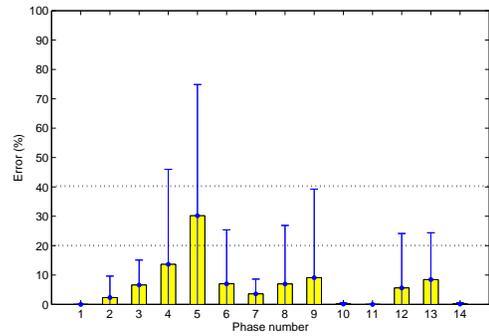
In table 3, we compare the results when no information from vision is considered. It shows as expected that using automatic information extracted from the endoscopic images does improve the results. For instance, since the camera is present in the body when the surgeon operates with the tools, but is also removed for cleaning as soon its view gets obscured, it can indicate whether the surgeon is working inside or outside the body or whether the phase contains blood.

Additionally, figure 4 shows the mean and standard deviation (std) per phase computed from the errors in the cross-validation tests. The model  $\lambda$  shows lower errors per phase, but also lower standard deviations compared to models with a fixed number of states per phase.

This also shows that the phases can be detected on-line. Higher errors occur for phases 5, 9 and 13. It comes from



(a)  $\lambda^{(1)}$



(b)  $\lambda$

Figure 4: Error per phase: mean over the cross-validation tests, for models  $\lambda$  and  $\lambda^{(1)}$ . The length of the vertical lines above the bars indicate the standard deviation.

the fact that these phases are performed very shortly in some surgeries, depending on the patient anatomy or on the surgeon. When it happens, a few seconds of wrong detection yields a high error, 100% in case the complete short phase is not detected. If all the phases are important for a context-aware supporting system, for instance to indicate the next instrument to hand in, phases 7 and 10 are crucial for a simple demonstration of the system to the surgeons. Indeed, calling the next patient is usually done at the beginning of phase 7, while switching on the light always occurs at the beginning of phase 10. Both of them are fortunately detected reliably.

Another direct application of the system is the prediction of the remaining time of the surgery, for instance to prepare the next patient and to gather the staff involved. It can be computed from the transition probabilities of the HMMs, using the computation of the most probable state at time  $t$ . Figure 5 shows the errors in the time prediction within each phase. They are computed for each second and averaged within each phase. The figure shows the mean and std of these values on all the cross-validation tests. As expected, the errors decrease when the surgery comes to its end. The mean prediction error is below 10mn from the phase 6 on, and it starts to be completely reliable in phase 10. More precise time prediction should take patient anatomy into account, as the obesity or the gallbladder size play an impor-

	overall(%)	mean per phase(%)	skipped
$\lambda$	7.6	6.7	0.3
$\lambda^{(1)}$	15.8	13.3	0.4
$\lambda^{(2)}$	12.6	10.0	0.4

Table 2: Comparison of errors for HMMs  $\lambda, \lambda^{(1)}$  and  $\lambda^{(2)}$ .

	overall(%)	mean per phase(%)	skipped
all signals	7.6	6.7	0.3
no vision	11.5	10.4	0.7

Table 3: Comparison of errors using model  $\lambda$  when including the visual features or not.

tant role. However, in its present form the system is reliable enough to warn automatically in phase 10 the staff that is supposed to be present for the next surgery. For comparison purposes, the mean of the durations of the 11 surgeries is 48mn, the standard deviation 14mn.

An illustrative video showing in accelerated speed a full surgery (see figure 6), including the endoscopic view, the external view, some signals, the phase detection as well as the triggered events, is available at <http://campar.in.tum.de/files/publications/MonitoringIAAI08.avi>

## Discussion

The previous section has demonstrated that on-line recovery of the surgical steps performed by the surgeon is possible based on the presented binary signals. It can be used in its present form for a demonstration in the OR. The last required step to install it in the real environment is of technical and administrative nature. All the binary signals which were labeled manually so far, can indeed be obtained automatically.

Signals like high-frequency coagulation and cutting or irrigation can be read directly from the system. If the usage of other instruments cannot be detected directly by vision because of the challenges mentioned before, their detection can be done using barcodes, RFIDs or other markers. The solution we tend to and which does not modify the workflow,

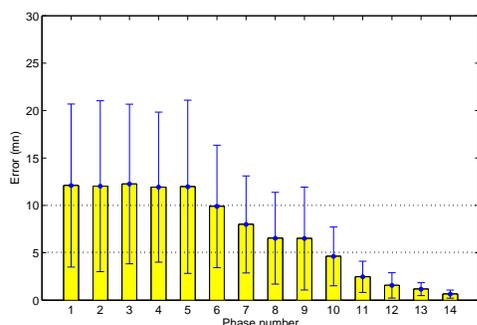


Figure 5: Errors in the prediction of the remaining time, from each phase. Mean and std on the cross-validation tests.



Figure 6: Screenshot of the illustrative video, displaying the external and internal videos, some signals and monitoring information.

is the usage of trocars equipped with a sensor that detects in real time which instrument is inserted or removed. A prototype solution is currently under development together with our medical partner.

Additionally, many other signals could also be used to improve the results, as long as their acquisition does not change the workflow. For instance the CO2 pressure in the abdominal cavity, the data from the anaesthesia, the amount of liquid used to clean the abdominal area, or even the data from the administrative information system, containing patient information and used materials, could be useful and incorporated into the system. As the nature of the signals would change, this would bring new challenges, but also allow the detection of finer events within the surgical workflow.

## Conclusion

Contemporary operating rooms are usually equipped with high-end systems able to provide various signals. All these signals could principally be used for contextual support during the surgery. We have presented a working set of signals and an approach to monitor laparoscopic surgeries, which can also be extended to non-minimally invasive surgeries. The results obtained using 11 real surgeries yield a detection accuracy of 93% and allow the reliable triggering of a few useful events. We have also presented our future steps towards the deployment in a real environment, as well as some relevant applications for the surgeon. Similar systems, using machine learning techniques to analyse the signals from the OR, will probably be increasingly developed and applied in the near future as the ORs tend to become more and more high-tech. It will allow a stronger context-aware support to the surgical staff, the detection of anomalies, and a better administration of the hospital.

**Acknowledgments.** This research is partially funded by Siemens Medical Solutions.

## References

- Bhatia, B.; Oates, T.; Xiao, Y.; and Hu, P. 2007. Real-time identification of operating room state from video. In *Proc. of Innovative Applications of Artificial Intelligence (IAAI)*, 1761–1766.

- Cleary, K.; Chung, H. Y.; and Mun, S. K. 2005. Or 2020: The operating room of the future. *Laparoscopic and Advanced Surgical Techniques* 15(5):495–500.
- Herfarth, C. 2003. 'lean' surgery through changes in surgical workflow. *British Journal of Surgery* 90(5):513–514.
- James, A.; Vieira, D.; Lo, B. P. L.; Darzi, A.; and Yang, G.-Z. 2007. Eye-gaze driven surgical workflow segmentation. In *Proc. of Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 110–117.
- Jannin, P.; Raimbault, M.; Morandi, X.; and Gibaud, B. 2001. Modeling surgical procedures for multimodal image-guided neurosurgery. In *Proc. of Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 565–572.
- Leong, J.; Nicolaou, M.; Atallah, L.; Mylonas, G.; Darzi, A.; and Yang, G.-Z. 2006. HMM Assessment of Quality of Movement Trajectory in Laparoscopic Surgery. In *Proc. of Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 752–759.
- Lin, H. C.; Shafran, I.; Murphy, T. E.; Okamura, A. M.; Yuh, D. D.; and Hager, G. D. 2005. Automatic detection and segmentation of robot-assisted surgical motions. In *Proc. of Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 802–810.
- Lo, B. P. L.; Darzi, A.; and Yang, G.-Z. 2003. Episode classification for the analysis of tissue/instrument interaction with multiple visual cues. In *Proc. of Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 230–237.
- Neumuth, T.; Strauß, G.; Meixensberger, J.; Lemke, H. U.; and Burgert, O. 2006. Acquisition of process descriptions from surgical interventions. In *Proc. of International Workshop on Database and Expert Systems Applications (DEXA)*, 602–611.
- Oliver, N.; Garg, A.; and Horvitz, E. 2004. Layered representations for learning and inferring office activity from multiple sensory channels. *Computer Vision and Image Understanding* 96(2):163–180.
- Padoy, N.; Blum, T.; Essa, I.; Feussner, H.; Berger, M.-O.; and Navab, N. 2007a. A boosted segmentation method for surgical workflow analysis. In *Proc. of Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 102–109.
- Padoy, N.; Feussner, H.; Berger, M.-O.; and Navab, N. 2007b. Recovery of surgical workflow: a model-based approach. *International Journal of Computer Assisted Radiology and Surgery (CARS), Supplement 1 2*:481–482.
- Paulus, D.; Csink, L.; and Niemann, H. 1998. Color cluster rotation. In *Proc. of IEEE International Conference on Image Processing (ICIP)*, 161–165.
- Rabiner, L. R. 1989. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE* 77(2):257–286.
- Rosen, J.; Brown, J.; Chang, L.; Sinanan, M.; and Hannaford, B. 2006. Generalized approach for modeling minimally invasive surgery as a stochastic process using a discrete markov model. *IEEE Trans. on Biomedical Engineering* 53(3):399–413.
- Shi, Y.; Huang, Y.; Minnen, D.; Bobick, A.; and Essa, I. 2004. Propagation networks for recognition of partially ordered sequential action. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 862–869.