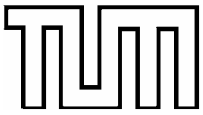

LEHRSTUHL FÜR REALZEIT-COMPUTERSYSTEME
TECHNISCHE UNIVERSITÄT MÜNCHEN
UNIV.-PROF. DR.-ING. G. FÄRBER

First Steps Towards Monitoring and Recovery of Surgical Workflow

Seyed-Ahmad Ahmadi

Bachelor Thesis

First Steps Towards Monitoring and Recovery of Surgical Workflow

Bachelor Thesis

Proposed and supervised by the Chair for Computer Aided Medical Procedures (CAMP)
Technische Universität München
Prof. Dr. Nassir Navab

Co-supervised by the Chair for Real-Time Computer Systems (RCS)
Technische Universität München
Prof. Dr.-Ing. Georg Färber

Advisors: Prof. Dr. med. Hubertus Feussner (Medical Partner)
Dipl. Inf. Martin Horn, CAMP
Dipl. Ing. Tobias Sielhorst, CAMP
Dr.-Ing. Dipl.-Kfm. Hans Oswald, RCS

Author: Seyed-Ahmad Ahmadi
Grasmeierstr. 25/121, 80805 Munich, Germany

Submitted in November 2005

Acknowledgment

I would like to thank...

- Prof. Dr. Nassir Navab for this very interesting project and for his support and advice during the conduction of my Bachelor's Thesis at the CAMP chair,
- Martin Horn and Tobias Sielhorst for being inspiring and helpful supervisors throughout my whole work,
- Prof. Dr. Hubertus Feussner for the discussions on the medical aspects of this thesis as well as for working with us on the recording of data during surgical procedures. Many thanks also to Armin Schneider from MITI who arranged the dates for data acquisition,
- Prof. Dr. Georg Färber and Hans Oswald for their co-supervision and for the management of all formalities regarding this thesis,
- Ralf Stauder for being a great and extremely capable project partner – without him, the outcome of this project would not be as satisfying, and
- Christoph Riedl for the mutual support, encouragement and invitations to coffee breaks during the conduction of our simultaneously running Bachelor theses.

Seyed-Ahmad Ahmadi
Munich, 11.11.2005

Abstract

In a previous paper by Sielhorst et al., a surgeon's hand movement was captured over time and in three dimensions. Several methods were compared of how to synchronize the resulting 3D trajectories. With the help of the most promising algorithm called Dynamic Time Warping (DTW), the operating procedures of different surgeons could be compared in an objective manner.
In this thesis, the existing approach using DTW has been expanded in order to automatically retrieve the workflow of a whole surgical procedure. The underlying surgery, a laparoscopic cholecystectomy, is a standard procedure with a distinct and systematic sequence of events. Nevertheless, time and workflow variations occur due to the surgeon's experience and anatomic conditions of the patient. In this thesis, a system is created which is able to synchronize key events in the data recorded from multiple operations despite these variations. The signal which is used for synchronization is a multi-dimensional trajectory reflecting which laparoscopic instrument is in use at every moment during the operation. After matching several procedures, an average surgery is created. Trigger events are used to identify 14 distinct workflow phases in the average curve. Using so-called time warp paths, which are obtained through the DTW synchronization, these phases are mapped onto each of the original surgeries. The results are then compared with the actual timestamps of workflow phases in each surgery. In six procedures with 13 trigger events each, the proposed system was able to identify 87% of the events with a tolerance of five seconds. Furthermore, it could be demonstrated that two events within the OR, switching on and off the OR lights and calling the next patient, can be successfully controlled by our system.
Using the obtained synchronization, various video and data streams are displayed in a comprehensive GUI for an objective comparison of the procedures.
Possible scenarios of a reliable workflow recovery system are better training methods for medical education, workflow optimization inside the OR and eventually the development of an "intelligent" operating theatre which is able to deliver support and context-specific information for the surgical team.

Table of contents

Index of Figures

Index of Tables

1 Introduction

In this thesis, a system will be introduced which is able to identify pre-defined workflow phases within surgeries of one kind. The structure of this thesis is as follows:

First, chapter 2 will lay the groundwork for the reader. Motivations for surgical workflow recovery will be given, the scope of this project is defined, and most importantly, the surgical framework will be explained. This contains a detailed description of the workflow steps that are performed during the surgeries which we are looking at in this thesis.

Chapter 3 will deal with the required materials and methods for workflow recovery. Test data is acquired and the Dynamic Time Warp algorithm is introduced as the basic tool for the proposed method of workflow recovery. The basic DTW algorithm is gradually enhanced in 3.3 to 3.6 in order to construct an average curve representing the average workflow of the analyzed procedure.

Chapter 4 is dedicated to the description of workflow recovery and to the results with the final dataset. Section 4.1 will describe how the workflow can be recovered once such an average surgery has been obtained. This is the core concept of the proposed system. Section 4.2 will eventually show the results of our system, evaluating the outcome of all algorithms that have been introduced before and deriving a success rate for workflow phase detection. Furthermore, a short demonstration will be given of how the obtained results are presented in the graphical user interface (GUI) of the system.

The results are then comprised in chapter 5. Strengths and weaknesses of the proposed method will be outlined. Furthermore, two other approaches for workflow detection will be discussed and a conclusion will be given with recommendations for further work.

Eventually, chapter 6 will give a short outlook on the possible development of the given approach and on workflow recovery in general.

2 Framework

2.1 Motivations for workflow recovery

In March 2004, a workshop was organized by the ISIS Center, Department of Radiology and the Georgetown University Medical Center which was dedicated to discuss ideas, requirements and objectives for the so-called OR 2020 – the Operating Room (OR) of the Future [1]. A group of over 100 industrial and academic experts gathered to discuss topics like “Surgical Informatics”, “Systems Integration and Technical Standards” or “Intraoperative Diagnosis and Imaging”. One major issue covered the area of “Operational Efficiency and Workflow”, where clinical needs, technical requirements and research priorities were identified for achieving greater efficiency in surgeries of various kinds and specialties.

The optimization of the workflow a patient has to undergo in a hospital, from his admission to his discharge, is the focus of many research communities. The motivations range from better cost efficiency [4] to an improved, measurable and consistent treatment quality for the patient [5]. Independent of the motivations for workflow optimization, as Herfarth [4] denotes, it is crucial that

> *“the surgical unit, as the area of greatest cost intensity, must cooperate completely and smoothly with the units responsible for outpatient care and preoperative assessment, and with the intensive care, intermediate care and short-stay wards”.*

In this respect, many of the approaches so far examine the overall workflow of patient care in the hospital whereas the time inside the OR more or less is a black box in the overall workflow. Information about the status and the progress of the surgery is scarce, whereas a proper information flow during surgery would be very valuable to the organization both inside and outside of the OR.

The resulting benefits have been extensively formulated by the experts in the OR2020 workshop [1]. Some of the described benefits, both for the administration and the medical side of a hospital, will be given in the following paragraphs.

Context-specific controlling and support Benefits for the medical staff include a context-specific support of the surgeon and the surgical team. A system which is reliably able to identify and follow the workflow of a surgery is in its ideal case able to serve as a backend support system in the OR.
One big potential, for example, lies within the intelligent management and

controlling of electronic devices in the OR. Routine tasks like switching on and off the OR ceiling lights, depending on the current work steps being performed, could easily be cared for by a context-aware system. A more sophisticated system could also care for a context specific presentation of critical information. It could take care that the right piece of information is presented at the right location and at the right time inside the OR. For surgeons, this would mean an improved workflow with fewer interruptions since they would not have to give instructions anymore in order to receive the information they require.

Anticipation of next steps Since such a system could anticipate next steps, it would be able to notify or even actively help the team surrounding the surgeon to prepare necessary tools and devices for the next surgery phase. In this thesis, a first step towards such phase recognition will be taken. The ultimate goal is the establishment of an intelligent agent with opinionated consultancy inside the OR. This long-term goal was formulated by the working group for workflow efficiency during the OR2020 Workshop. For such an intelligent surgical agent, however, the identification of workflow during surgery is crucial.

Administrative optimizations Apart from the benefits inside the OR, such a system could also be beneficial to the administration of a hospital. By synchronizing several surgeries of one kind and evaluating the spatial relations of objects inside the OR during same events, the layout of the operating theatre could be improved and the setup of devices could be optimized to match the requirements of the surgery setting.

Automated documentation Monitoring surgical workflow requires the recording of many signals and events inside the OR. Combining such signals with knowledge of the regular workflow allows this system to create surgery documentation in a very high quality. Until now, surgeons have to create documentation post-operatively by either writing the documentation themselves or dictating it to assistants. This is a tedious task which demands a lot of time and effort from the surgeon. A workflow sensitive system on the other hand could be designed to create surgical reports automatically. Such detailed documentation could be useful for other aspects of hospital management as well. For example, the legal administration of a hospital could rely on very precise data when dealing with accusations from patients instead of relying on documentation from the surgeon which is often subjective and inaccurate.

Simulation and education A further, very important application for a workflow retrieval system lies in the field of education. A workflow-aware system could be used to simulate surgeries in an arbitrary level of detail, depending on the overall workflow sensitivity of the system. Medical students could perceive simulated surgeries and compare their performance with the performances of their instructors. For medical teams, such a surgery simulator could furthermore serve as the perfect training instrument for pre-operative preparation.

2.2 Scope of this project

Medical workflow analysis is one of the research branches of the “Chair for Computer Aided Medical Procedures” (CAMP) at the Technical University of

Munich, where this thesis was conducted. This thesis marks a first step towards the monitoring and recovery of surgical workflow. The underlying surgical framework is confined to surgeries of one kind and of one specialty. The procedure and its typical workflow are described in 2.4.

In the course of this work, a method will be proposed with which the status of surgical workflow can be identified. The overall scope of the project is made up of the following objectives:

- **Recording of surgeries**
  In order to retrieve signals for synchronization and workflow detection, several video and audio signals are recorded in the OR and evaluated offline after surgery.
- **Creation of an "average surgery"**
  By non-linearly synchronizing an arbitrary number of surgeries in the time domain, an average surgery can be generated and unknown surgeries can be compared to this average surgery.
- **Automated weighting of signal relevance**
  Not all signals that are recorded have the same relevance for the task of workflow recovery. In order to receive better synchronization results and thus, a better average, a method for automated weighting of signals and their importance is implemented.
- **Workflow recovery**
  By assigning workflow phases to the average surgery, the workflow of any other surgery can be identified. Furthermore, events and devices inside the OR can be controlled in a context-specific manner.
- **Presentation in a GUI**
  Finally, the evaluated data is used for presentation in a graphical user interface. Several surgeries with synchronized timelines can be shown simultaneously and their respective workflow phases will be indicated. A first attempt of controlling the workflow is made by proposing points in time where certain events such as switching on and off the lights in the OR or calling the next patient should take place.

2.3 Previous and related work

This thesis and the proposed method of workflow retrieval are based on a paper by Sielhorst and Blum on synchronization of 3D movements for quantitative comparison and simultaneous visualization of actions [6]. As mentioned above, several methods were discovered and evaluated of how to synchronize the 3D hand movement of a surgeon. The methods evaluated included synchronization by

- Simple scaling,
- Longest Common Subsequence (LCSS), and
- Dynamic Time Warping (DTW).

Out of these methods, the DTW algorithm performed best in the given application [6]. Using DTW, Sielhorst et al. could synchronize the process of giving birth to a virtual child in a delivery simulator, using a forceps to grasp the head of the child. The 3D movement of the hand holding the forceps is performed differently in every case, but the underlying process of grasping the baby's head is the same to all attempts. Due to this, all curves share certain similarities which have to be synchronized in order for the observer to be able to simultaneously display two movements and compare them quantitatively. The paper was created at the CAMP chair as well and presented at the ISMAR conference 2005, receiving one of the Best Student Paper Awards.

For this thesis, the approach is further widened in order to evaluate multi-dimensional curves, each of which representing one recorded surgery. Another extension is the implementation of the DTW in an iterative procedure which allows us to synchronize an arbitrary number of surgeries in contrast to synchronizing only two curves at a time.

2.4 Surgical framework

In order to understand the workflow analysis in this thesis, the reader should be familiar with the underlying surgical framework. Therefore, it will be described shortly.

General description
The surgery chosen for this work is a so-called laparoscopic cholecystectomy (LC). In non-medical terms, this is a minimal-invasive procedure during which the gallbladder of a patient is removed. The gallbladder is a pear-shaped organ that rests beneath the right side of the liver, helping to store and concentrate bile, a digestive secretion which is produced in the liver. Gallbladders are removed when gallstones block the normal flow of bile and cause an unnatural swelling of the gallbladder, resulting in sharp abdominal pain, digestive difficulties and other issues like vomiting.

History and statistics
The first laparoscopic cholecystectomies were performed in 1985 by Mühe and in 1987 by Mouret [7]. Since then, the minimal-invasive procedure was constantly improved, gradually replacing the traditional method of open cholecystectomy, where the abdomen of the patient is opened with an incision of 12-20cm length [8]. Today, the minimal-invasive procedure is preferred over open surgery due to several reasons: instead of one big incision, there are four small incisions of 5 to 10 mm. This is aesthetically more pleasing and it causes less post-operative pain which allows the patient to return to every-day activities more quickly. Meanwhile, 95% of all cholecystectomies are performed laparoscopically. The conversion rate is less than 5% [2]. This is the rate of surgeries that have to be changed to open surgery intra-operatively due to complications. The frequent performance of LCs and their stable outcome make it a practical type of surgery for this kind of analysis. Most important in this respect, however, is the fact that the workflow is systematic and recurrent.

Clinical partner
In order to provide a sound evaluation of the workflow of this surgery, more than two procedures had to be taken into respect. In the course of this work, a total number of six LCs were recorded by courtesy of Prof. Feussner, head of the surgical department at the "Klinikum Rechts der Isar" in Munich. Prof. Feussner was the surgeon of all six surgeries.

General workflow of laparoscopic cholecystectomies
In the following, a rough summary of a common workflow for a LC will be given. The understanding of the workflow is essential for relating to the materials and methods that have been used for workflow recovery in the course of this thesis. For a short explanation of the medical terms used in this part, please refer to the table "Medical terms and explanations" in the appendix.

After anaesthesia and further preparation of the patient, like disinfection of the abdominal region, a small incision is made at the umbilicus or navel of the patient. A needle is inserted through which the abdomen of the patient is inflated with CO_2 in order to create an abdominal cavity. This marks the beginning of the procedure to us. When the inflation is sufficient, a trocar of 5mm diameter (trocar number 3, see Figure 1) is inserted into the incision and the laparoscope is put into the body. Under visual control, three other trocars of sizes 5-10mm are punctuated into the abdomen in counter-clockwise order. The laparoscope is pulled out of trocar 3 and inserted into trocar 4. A metal rod is inserted into trocar 1, the liver is pulled up and the gallbladder attached beneath it is revealed.

A grasper and a dissecting device, inserted through trocar 2 and 3, are used to identify the positions of the gallbladder's two main vessels, the common bile duct and the bile artery. With the dissecting device, these two vessels are uncovered from adhesions and adjacent tissue. When sufficiently dissected, the vessels are clamped using a clipping device and subsequently cut using a pair of laparoscopic scissors. As a next step, the gallbladder is removed from the liver. In order to separate the gallbladder from the liver, a high-frequency current is applied in two modes, cutting and coagulation mode. The metal tips of most instruments serve as electrodes which are able to dissect the tissue electrically. During the removal of the gallbladder, it is likely that bleedings are caused in the area of the so-called liver bed, which is a large portion at the lower side of the liver where the gallbladder used to reside. After the gallbladder has been removed completely, these bleedings are stopped, again using high-frequency current in coagulation mode.

In order to retrieve the gallbladder, a retraction sac is inserted and the gallbladder is put into the plastic bag with two or three graspers. Trocar 3 is pulled out and the plastic bag is retracted through the umbilical port. After removal, trocar 3 is inserted again and in a last and final control phase, the liver bed and the surrounding tissues are controlled for bleedings and coagulated respectively. The remaining vessel stumps are controlled for correct and clean clipping. A drainage tube is inserted in order to lead away the saline irrigation fluid after surgery. All instruments and trocars are retrieved and the four incisions are closed with stitches. The beginning of the abdominal suturing marks the end of laparoscopic activity and thus the end of the workflow that we are considering for this work.

Figure 2 shows a rough sketch of the workflow during a LC that is relevant to us. In the course of this thesis, a method will be proposed with which it will be able to recover these phases in a recorded surgery.

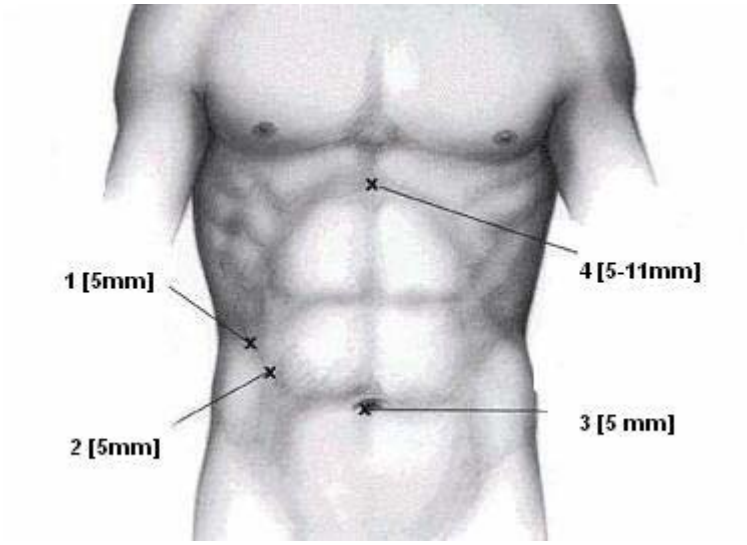

Figure 1 – Trocar placement in the abdomen.
Source: http://www.laparoscopy.net/chole/choletech2a.htm

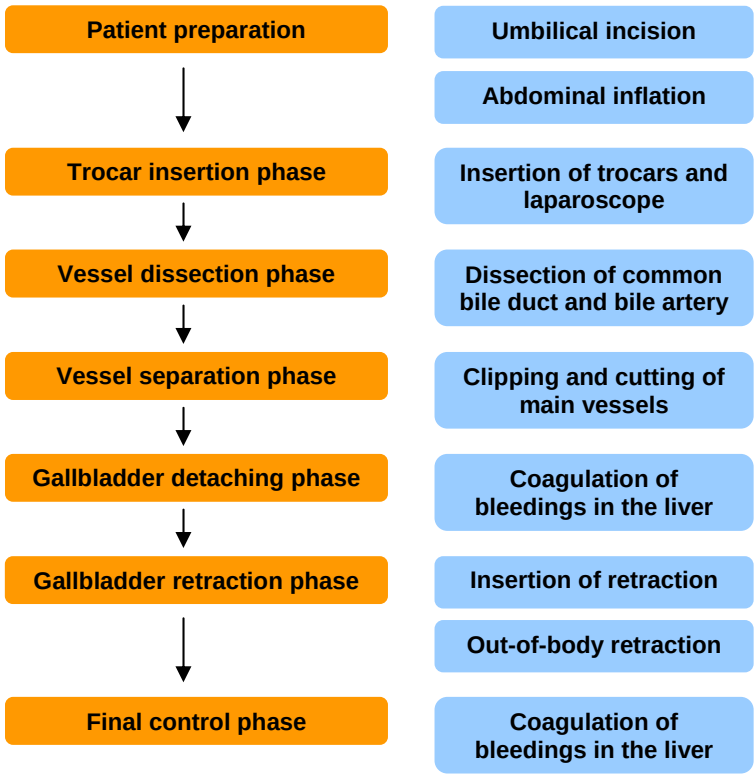

Figure 2 – Surgery phases during a laparoscopic cholecystectomy.

3 Materials and Methods

In the following sections, a method will be described with which same events in several surgeries can be automatically synchronized with each other. The typical workflow and events of a surgery have already been described. In order to represent it, however, an abstract and characteristic signal is required. In our case of a minimally-invasive surgery, the most representative signal is the information which laparoscopic instrument is being used during the surgery. For example, the clipping device directly reflects the workflow phase of clipping the artery.

Using the instrument signal, an algorithm called Dynamic Time Warping will be employed to synchronize the surgeries. In 3.2, the classic DTW algorithm is introduced. Sections 3.3 to 3.6 then describe how the classic algorithm is gradually enhanced in order to function as a tool for surgical workflow recovery. This will be further explained in chapter 4.

In order to be able to assess the quality of DTW synchronization during the implementation, we require test data. In our case, this test data is taken from real surgeries. Section 3.1 describes how we obtained this test data and how the data will further be illustrated throughout this thesis.

3.1 Acquiring of test data

3.1.1 Source of test data

The Dynamic Time Warp in its classic implementation enables us to synchronize two surgeries. The data with which this synchronization is tested was taken from real surgeries. This way, all test results directly reflect the capability and quality of workflow synchronization when using the DTW algorithm. The test data was obtained from three manual protocols which were created during three subsequent LCs at the Klinikum Rechts der Isar on July 15th 2005. Table 1 shows the beginning and end times of the surgeries, the duration and the operating surgeon.

Surgery	Start time	End time	Duration	Surgeon
$OP1_{test}$	08:46	09:28	42 minutes	Prof. Feussner
$OP2_{test}$	09:46	10:56	70 minutes	Prof. Feussner
$OP3_{test}$	11:40	13:19	99 minutes	Prof. Feussner

Table 1 – Details on the three test surgeries.

The three protocols can be found in the appendix 8.2. They took note of the time in a resolution of minutes, the laparoscopic instrument in use, the general action that was performed by the surgeon as well as the lighting condition in the OR.

3.1.2 Visualization of test data

The utilization of laparoscopic instruments was used as the signal with which the workflow of the surgeries is identified. As can be seen in the following paragraphs, the laparoscopic instruments give a strong clue about the actions that are being performed and about the progress of the surgery in general.

For a first visualization and for first tests using the classic DTW algorithm, a one-dimensional instrument signal was created from the protocols. In order to create the one-dimensional signal, numbers were assigned to the instruments. The numbers were chosen so that significant events in the workflow receive peaks in the curve. The most significant events after viewing three LCs appeared to be events such as the clipping and cutting of vessels, the retraction of the gallbladder with the plastic sac, and the insertion of the drainage tube into the body. Since these events correlate with the usage of specific instruments, the respective instruments received high numbers while the instruments that were used in between these events received low numbers. By doing so, the desired peaks were created.
Table 2 shows the laparoscopic instruments that were recorded in the protocols and the respective number with which they were modelled.

Figure 3 shows the three instrument streams over time.

In the one-dimensional representation (cf. Figure 3), it becomes obvious that the three surgeries do indeed show a common workflow. The peaks denote the clipping and cutting of vessels, the usage of the retraction sac, and the insertion of the drainage tube. These peaks appear in all three surgeries, however at different times and at different distances in relation to each other. Despite these differences, a common workflow is visible and thus has to be recoverable.

Laparoscopic instrument	Code
No instrument in use	0
Trocar	1
Optics	2
Preparation forceps	3
Suction & irrigation device	4
Alligator forceps	5
Scissors	6
Clipping device	7
Retraction plastic sac	8
Drainage tube	9

Table 2 – Laparoscopic instruments and their code number for 2D signal representation

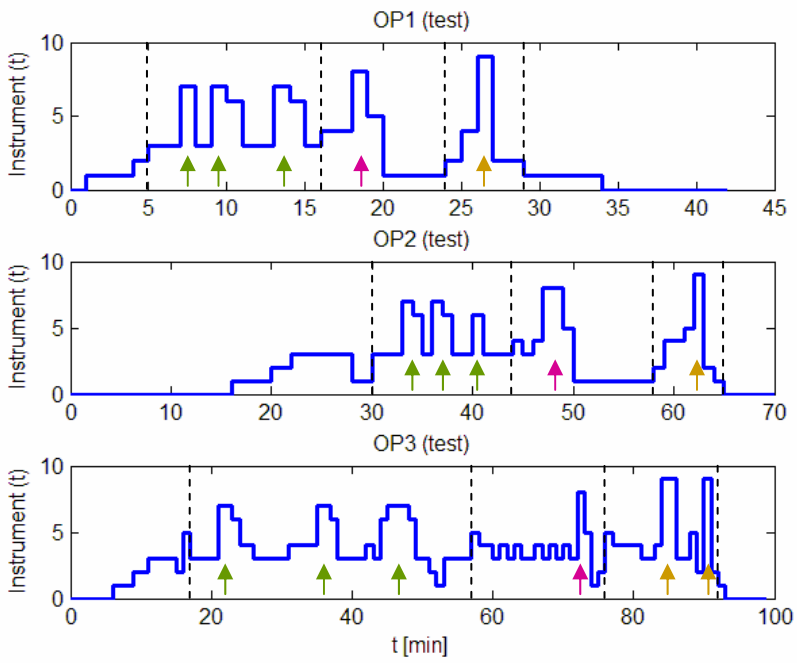

Figure 3 – Laparoscopic instruments as a 2D signal; arrows mark peaks representing common workflow events.

Table 2 is a generalization of the actual case. A one-dimensional representation has two major shortcomings. First, a one-dimensional model for the instrument signal allows only one instrument to be in use at time t. In an actual surgery,

however, the usage of four trocars allows the surgeon to use three instruments and the laparoscopic camera simultaneously inside the body. Second, the assignment of numbers to the instruments as shown in
Table 2 is biased due to a subjective view on the importance of events during the surgery.

Both shortcomings can be neglected for the moment, since a better model is used for actual testing of the algorithm. The one-dimensional representation was mainly used for first experiments with the DTW and for better illustration.

A more exact and more objective model is the multi-dimensional representation of a surgery with one dimension representing time and n_{Instr} additional dimensions representing the instrument usage tracks (cf. Figure 4). This model also allows a formal representation of the case that several instruments were used simultaneously. The usage of an instrument in this case was modelled as follows:

$$u(t) = \begin{cases} 1, \textit{if the instrument is used at time } t \\ 0, \textit{if the instrument is not used at time } t \end{cases}$$

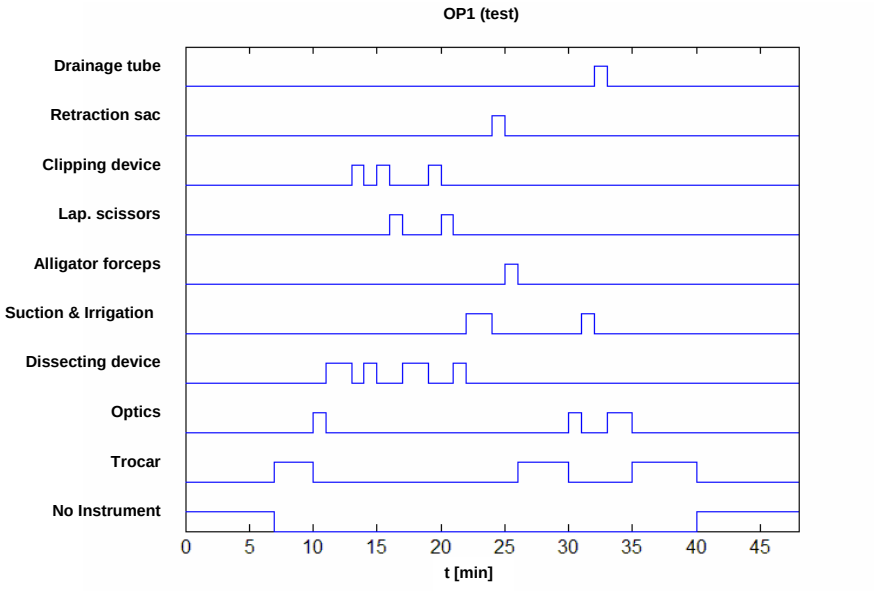

Figure 4 – Multi-dimensional representation of $OP1_{test}$.

3.2 The Classic DTW algorithm

3.2.1 Problem statement

As mentioned in 3.1, the curves of surgeries share common characteristics. However, these similarities are unevenly distributed over time. It would be easier

to compare such curves and study their similarities if they were synchronized so that common events occur at common points in time.

According to [9], a synchronization can be formally described as a mapping of two curves A and B with

$A = (a_0, a_1, \ldots, a_i, \ldots a_I)$ and $B = (b_0, b_1, \ldots, b_j, \ldots b_J)$, where a_i and b_j represent the i[th] and j[th] data point obtained during our measurement.

The synchronization should map the time axes i and j of both curves to a common time axis k with $c(k) = (i(k), j(k))$ and $k = 1 \ldots K$.

The mapping function for the two time scales of A and B onto each other will be referred to as a warping path or warp path w from now on:

$$w = ((i(1), j(1)), (i(2), j(2)), \ldots, (i(K), j(K))).$$

3.2.2 Shortcomings of other methods

In order to achieve such synchronization, several methods exist as described in [6]. The three methods described there are Simple Scaling, Longest Common Subsequence (LCSS) and Dynamic Time Warping (DTW). The DTW has certain advantages over the other two approaches. In the following, their shortcomings will be described shortly.

Simple Scaling When dealing with curves of unequal length, a simple synchronization method would be a plain scaling of curves in order to establish a common start and ending point. In our case, this would synchronize the beginning and the end of surgery workflow without fail. On the other hand, it is very likely that events within the surgery cannot be matched with this method. A simple example would be a pair of surgeries which have an equal duration and therefore do not have to be scaled at all. Nevertheless, it is very likely that due to anatomic dissimilarities of the patients, or due to different styles and experience levels of the surgeon, the different phases of the surgery are not equal in length. In the case of a LC, the dissection of the gallbladder vessels might require more time in one case while it is performed rather quickly in the other. The retraction of the gallbladder, on the other hand, might cause problems in one case while it is carried out rather quickly in the other case, since the surgeon, for example, decides to apply a greater incision.

This example shows that the duration of each surgery phase is difficult to predict. Therefore, our synchronization method has to take care that each part is adaptively adjusted in length so that the time length of the respective phase in the other surgery is met. This is a problem which requires a non-linear synchronization “that is able to locally translate, compress, and expand” [9] the surgery segments so that common events within the surgeries are mapped onto each other (cf. Figure 5).

The other two methods described in [6], LCSS and DTW, are both capable of solving such a problem non-linearly.

Longest Common Subsequence (LCSS) LCSS has several drawbacks in contrast to DTW. A detailed analysis is given in [6]. These shortcomings also hold true for our type of application. Thus, the description of this algorithm will be kept short. What LCSS does is to calculate the longest common subsequence within two strings, as the name already suggests. If two characters in two arbitrary strings are identical and if they appear in the same order within both strings, they are defined as a matched pair. The same principal can be used in order to find points that are common to two curves and appear in the same temporal order. The matched pairs are used as supporting points for a complete synchronization of the curves onto each other. The resulting synchronization is able to dynamically adjust and match the length of different surgery phases. But as evaluated in [6], the LCSS has difficulties when it comes to the synchronization of curves with different update rates. The same problem is present in our case due to different numbers of samples for different surgery curves. The problems that arise are described in [6] and will not be evaluated further. Moreover, the LCSS algorithm has difficulties with the synchronization of multi-dimensional curves and with the weighting of different variables in multi-dimensional problem settings. Unfortunately, as it has been described in 3.1, our signal is multi-dimensional. Furthermore, section 3.6 will show that a weighting of the variables can help to improve the results of synchronization.

Because of these two reasons, the DTW is more suitable for our task. The following paragraphs will describe the theory behind the DTW algorithm and how it can be implemented.

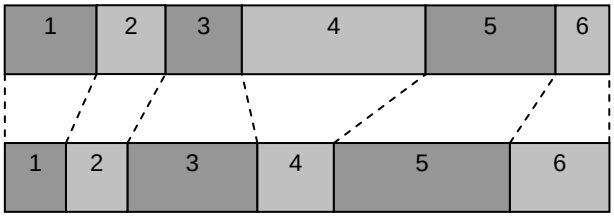

Figure 5 – Two surgeries with six phases of different lengths. Simple scaling would not be sufficient for synchronization in such a case.

3.2.3 Posing of the minimization problem

The minimization problem is posed and described in detail in [16]. A short summary shall be given here.

As mentioned in 3.2.1, the task aims at synchronizing two curves A and B. This is equivalent to a search for corresponding feature points a_i and b_j which are as similar to each other as possible for each matched time pair i and j. A measure for the similarity between two feature points is obtained by the distance function

$$d(c(k)) = d(i(k), j(k)) = \left\| a_i - b_j \right\|.$$

In this respect, k denotes a common time axis onto which both time axes i and j are transformed temporarily. Summing up the distances $d(c(k))$ for each corresponding point pair $(i(k), j(k))$ yields

$$D(w) = \sum_{k=1}^{K} d(c(k)).$$

The ideal mapping is achieved when this accumulated sum is minimized. The resulting time mapping is referred to as the DTW warp path.

Since physical processes follow timelines with respective restrictions, it is necessary to impose similar restrictions onto the warp path :

- Boundary conditions
  $i(1) = 0$, $j(1) = 0$, and $i(K) = I$, $j(K) = J$
- Monotonic conditions
  $i(k-1) \leq i(k)$ and $j(k-1) \leq j(k)$
- Continuity conditions
  $i(k) - i(k-1) \leq 1$ and $j(k) - j(k-1) \leq 1$

The first condition ensures the mapping of start and end points onto each other. The second and third condition ensure one crucial characteristic of the DTW. Every point of one curve is mapped onto at least one point of the other curve. The slope in the warping path therefore is always zero, one or infinite (cf. Figure 6).

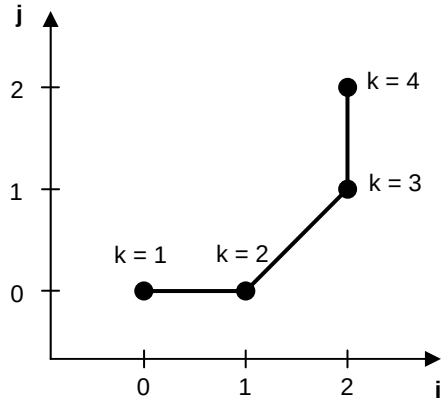

Figure 6 – Slopes on a DTW warp path can be zero, one or infinite.

The fact that the warp path slope can only take these three values raises another important implication: the number of points on the warp path is at least as large as $\max(I, J)$. Although this is known, the exact number of points on the warp path cannot be determined prior to computation [9].

3.2.4 Implementation with dynamic programming

A recursive algorithm helps to solve the non-linear problem of finding the optimal warping path.

$$DTW(A,B) = d(a_i, b_j) + \min\begin{pmatrix} DTW(A_{i-1}, B_{j-1}) \\ DTW(A_{i-1}, B) \\ DTW(A, B_{j-1}) \end{pmatrix}$$

The principles of dynamic programming are used to implement the recursion and thus obtain the optimal solution. The distances for all possible combinations of curve points a_i and b_j are calculated and stored in a matrix L, regarding the following rules:

- Start in the upper left field of the matrix (with indices i, j = 0) and calculate the distance $d(k=1) = d(a_0, b_0)$
- Increment i and j and thus fill up the matrix, whereby for each new pair a_i and b_j, the minimum of the left, the upper-left and the upper value is added to the newly calculated distance $d(a_i, b_j)$.

Once the matrix L is filled up, it has to be traced back from the lower right end. The lower right field $L(I,J)$ now contains the summed up distance $D(w)$ which is the minimal total distance that can be obtained due to optimal matching of the two curves. At the same time, $L(I,J)$ denotes the matching of the end points for the two curves. From there, we have to follow a path of minimal values in direction to the left, the upper left and the upper field. Mathematically, this is described as:

$$c(k-1) = \min\begin{pmatrix} (i(k), j(k)-1) \\ (i(k)-1, j(k)-1) \\ (i(k-1), j(k)) \end{pmatrix}.$$

This yields the desired warping path $w(k)$.

3.3 Adaptation to multi-dimensional surgeries

3.3.1 DTW synchronization with 2D surgery representation

Now that the DTW has been introduced, it has to be applied to our surgical data. First, the simpler, one-dimensional signal as explained in 3.1 was used in order to demonstrate the general feasibility of surgery synchronization with the DTW algorithm. The peaks representing important events in the workflow were expected to be matched. An exemplary result of DTW synchronization for $OP2_{test}$

onto $OP1_{test}$ can be seen in Figure 7. It shows that the matching of important events is satisfactory for one-dimensional surgery representations.

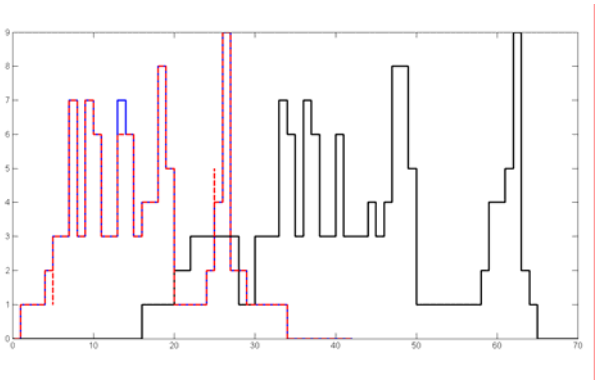

Figure 7 – DTW synchronization (red dashed) of $OP2_{test}$ (black) onto $OP1_{test}$ (blue)

3.3.2 Multi-Dimensional distance functions for the DTW

The one-dimensional representation of a surgery is not ideal (cf. 3.1). In the multi-dimensional representation of a surgery, we used feature vectors of the dimension $(1 \, x \, n_{Instr})$ instead of scalar feature points (cf. Figure 8). The DTW algorithm thus had to be adapted in order to be able to synchronize instrument vectors as modelled in 3.1. Fortunately, the DTW algorithm is flexible enough to be used without any major modifications. Only the distance function $d(c(k))$ had to be adjusted accordingly in order to calculate distances for vectors of arbitrary size.

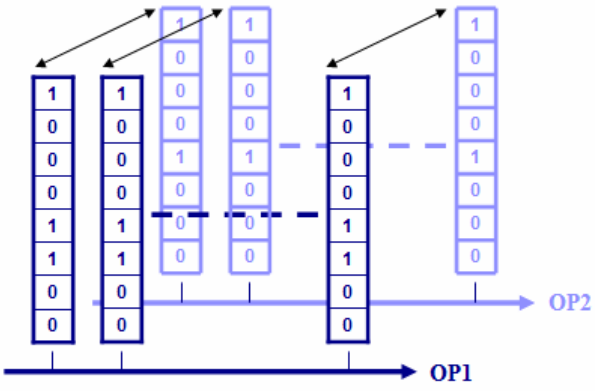

Figure 8 – Multi-dimensional instrument vectors for OP1 and OP2

Two appropriate types of distance functions for our binary instrument vectors were the XOR function and the Euclidian Distance Function (EDF):

XOR: $d = sum\left(XOR\left(a_i, b_j\right) \cdot W\right)$

EDF: $d = \left(\left(b_j - a_i\right) \cdot W \cdot \left(b_j - a_i\right)^T\right)^{0.5}$

The matrix W is a diagonal matrix with weighting coefficients for the single instrument signal tracks. The importance of weighting will be explained in 3.6. Although the DTW yielded the same quality for the XOR and the Euclidian distance function, the EDF could later be applied on continuous vector values as well. This made it more flexible than the XOR function and thus, it was used as the standard distance function in the further course of this work. Nevertheless, the DTW was implemented in a way that allows calling arbitrary distance functions while filling up the DTW matrix L. In this way, the multi-dimensional implementation of the DTW algorithm can be used for other projects as well, which might require other, more exotic distance functions.

3.4 Synchronizing discrete curves

Discrete curves versus continuous curves When synchronizing discrete curves, the DTW might encounter passages of the curves which are exactly identical. This is a situation that does not occur when dealing with continuous curves since the probability is very low that exactly the same action is performed twice. The nature of continuous processes, uncertainties in the measurement results, and the presence of noise do not allow longer curve portions to be exactly identical. However, when dealing with curves that have discrete function values and which do not have any noise, as it is in our case, there might be time intervals with identical events.

How the DTW algorithm creates valleys for identical events During implementation, this resulted in a valley of identical values for the accumulated distance function in the DTW matrix. An example shall help to understand why the DTW algorithm creates such a valley. In the test surgeries $OP1_{test}$ and $OP2_{test}$, the external retraction of the gallbladder took the surgeon different amounts of time. For both surgeries, this phase is represented by a function value of "1" which tells us that only the trocar is in use. For $OP1_{test}$ or A, this segment is 3 minutes long (minutes 20-23) and for $OP2_{test}$ or B, this segment is 7 minutes long (minutes 50-57). When the DTW algorithm compares the first values of each segment (a_{20}, b_{50}), it calculates a distance in the matrix L depending on the predecessors to the left, to the upper and to the upper-left. For the following values $a_{21} \dots a_{23}$ and $b_{51} \dots a_{57}$, the new distance $d(a_i, b_j)$ is zero because all values are identical. Thus, the lowest value of the predecessors is adopted. In other words, two segments with identical values do not add any new distance to the DTW matrix. Because of that, a valley is created until one of the samples a_i or b_j changes its value.

Classic DTW handling of valleys The second step of dynamic programming requires backtracking of the DTW matrix L. A classic DTW implementation does not take valleys into respect during this process. The algorithm starts at the lower right end of L and follows the lowest of the three values to the left, to the upper and to the upper-left. As already described, the resulting series of corresponding time points results in the warp path w. However, the DTW cannot handle the case that all three values are identical. In its usual implementation, the DTW crosses

the valley diagonally until it hits a valley border. Then it follows the border until the upper left valley point is matched. After that, it continues with correct backtracking of L (cf. Figure 9).

Suggestion for improved handling of valleys If the valley is not square in shape, crossing it creates a horizontal or vertical warp path segment. In our case, this implies that several measurement points of B are mapped onto only one point of A. Ideally, however, the DTW should map identical events onto each other completely. As in our example, it makes sense to associate the longer trocar phase in B homogeneously with the short trocar phase in A. In the implementation, this means that whenever a valley occurs, the DTW should try to match the start and end points of identical events and cross the valley in between diagonally (cf. Figure 9).

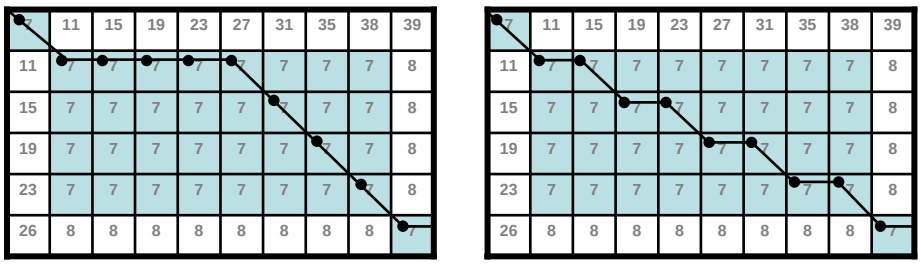

Figure 9 – Classic handling of valleys in the DTW matrix (left) and optimized, diagonal backtracking (right) when dealing with discrete functions

3.5 Synchronising m multi-dimensional curves

As it has been described before, the general concept of workflow recovery as proposed in this thesis is to synchronize several surgeries and create an average surgery. In section 4.1, it will be described how the workflow can be recovered from such an average. For now, the focus lies on creating an average surgery.

Sections 3.2 and 3.3 already showed how the DTW can be used to synchronize two surgeries at a time. For the creation of a characteristic average, however, more than two surgeries are required. Unfortunately, the classic DTW algorithm is not designed to synchronize more than two curves. Nevertheless, there are methods that manage to synchronize several curves that are based on the non-linear synchronization ability of the DTW. One method is proposed by [10] and is further improved in [11]. Another approach, similar to the one of [10], is proposed by [9].

3.5.1 Algorithm

In the following, an algorithm will be described with which an arbitrary number of curves can be synchronized using the DTW. The algorithm is adopted from Wang and Gasser [10].

The computation of an average curve according to [10] is performed in four steps:

(1) From a set of n curves $f_i, (i = 1 \ldots n)$, n-1 curves are warped onto one reference curve from the set. This yields n-1 warp paths $h_{i,raw}(t)$.

(2) The warp paths are normalized to the unit square and denoted as $h_i(t)$. Then the average of all warping paths is calculated.
This average timing can be marked as the first step towards the desired average curve.

$$h(t) = \frac{1}{m} \sum_{i=1}^{m} h_i(t)$$

(3) Once the common timing is obtained, the curves have to be warped onto this average time scale. In this respect, it is important to note that the average warp path is a mapping of an average timing onto the reference curve. The desired warp path, however, maps the time scales of all other (n-1) curves from the set onto the average timing. Therefore, the inversion of the average warp path can be used to map every curve onto the average timing. Figure 10 shall further help to understand this principle. Mathematically, these shift functions, as they are called in [10], can be described as

$$u_i(t) \equiv h_i\left(h^{-1}(t)\right)$$

(4) Now, each curve can be mapped onto the average timing. Once this is achieved, we will create the so-called "structural average" $f_0(t)$, which in the following will be referred to as the "average curve" or "average surgery".

$$f_0(t) = \frac{1}{m} \sum_{i=1}^{m} f_i(u_i(t))$$

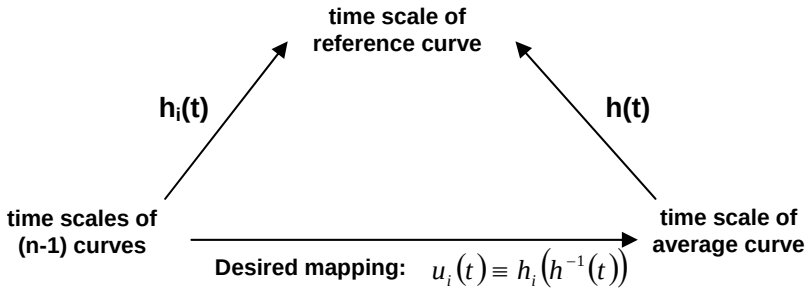

Figure 10 – Creation of shift functions onto the average time line

As it has been said, this procedure is used to create the desired average surgery for our project. However, there are no details given on how to implement the averaging of curves in the publication by [11]. Therefore, the implementation of the given theory was one major work part of this thesis. Furthermore, the computational realization posed some problems because of different frameworks

for this thesis and the work of Wang, Gasser et al. This difference lies within the distinct nature of the curves for which an average is being sought.

In [11], the curves represent the growth of shoulder widths from individuals aged 2-20. Due to this common background, all curves are of the same length and the data is continuous. Curves in this thesis, however, represent surgeries of different length and the functions can only adopt discrete values.

In the following paragraphs, it will be dealt with how to overcome the resulting difficulties. First, it will be demonstrated how a timing average could be created for curves of different lengths. After that, it will be shown how a structural average with average length can be created for curves of different length. Then, the implementation of the algorithm will be explained and the calculated average surgery will be interpreted and evaluated.

3.5.2 Creation of an interpolated average warp path

Once (n-1) warp paths have been obtained, step (2) of the algorithm requires them to be averaged. For curves of different length, this only makes sense if they have common boundaries. As the algorithm proposes, all warp paths are scaled to the unit square so that the horizontal and the vertical time axis range from 0 to 1. Still, an average cannot be calculated because K, the number of points on the warp path, cannot be determined a priori and thus is different for each warp path. The reason for this was described in 3.2. In the implementation, this causes problems, because we cannot simply create an average for each point on the warp path. A simple workaround for this problem would be to interpolate all (n-1) warp paths and then compute an average. But interpolation is not possible because of warp path segments with infinite slope.

In order to be able to interpolate, infinite slope segments have to be avoided. The interpretation of infinite slopes allows us to avoid them by using the average of the infinite segment as its representative. Infinite slope segments appear when there is a certain event in one curve B that does not occur in the other curve A. However, since the curve B can be expected to resemble curve A before and after this unmatchable event, the DTW tries to match the outlier event from B onto the time axis of A by compressing its whole time span onto only one time point of A. This causes the least distance between the two curves because now, similar events before and after the outlier can be regularly matched again.

Two reasons justify taking the average as a representative for an infinite slope segment. First, the single time point in A is not matched with the whole time span of the outlier anymore, but only with the time average. The time matching is thus not lost but it is only coarsened. Second, the matching of the time points in A before and after the special time point are both left unchanged. Figure 11 illustrates this argumentation.

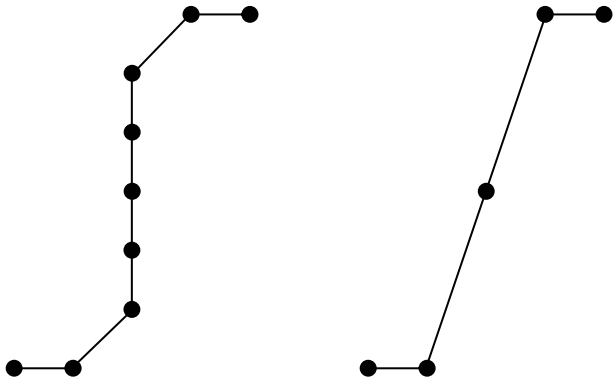

Figure 11 – Infinite slope segments in the warp path have to be replaced during the averaging process.

Moreover, when the sampling rate is high enough, taking the average results in a warp path segment with a slope that is not infinite but very high. The matching behaviour of the DTW is thus preserved satisfactorily.

In the next step, a method is introduced with which such infinite segments can be partially recovered.

3.5.3 Reconstruction of a discrete average warp path

After averaging the infinite slope segments, the warp paths can be interpolated in order to have the same number of equidistant data points in the horizontal direction. The average warp path can then be easily calculated according to step (4) of the algorithm. However, the resulting time mapping is now discrete and equidistant in the horizontal direction but more or less continuous in the vertical direction. But for a correct mapping of all other (n-1) curves in step (3) and (4), we need to reconstruct an average warp path with discrete and equidistant sampling points both in the horizontal and in the vertical direction. The node values on the horizontal time axis should have the range [0 … t_{ref}] and the nodes on the vertical axis should have the range [0 … t_{avg}].

In order to achieve this, a grid was created in which each interpolated warp path point hit one field of the grid (cf. Figure 12). During implementation, this grid was realized by a matrix with the dimensions $\left(t_{ref}+1 \times t_{avg}+1\right)$. The row and column index of the matrix field that was hit by a sample point was calculated by a simple division calculation. Because of the sampling theorem, however, it had to be taken care that the number of interpolated points was at least twice the maximal number of nodes on the warp path axes.

$$n_{nodes} \overset{!}{\geq} 2 \cdot \max\left(t_{ref}, t_{avg}\right)$$

For sufficient redundancy, five times the maximal number of nodes was chosen for the implementation.

Once the so-called hit-matrix H is filled completely, each field of H contains the number of interpolated warp path points that lie inside its boundaries. This matrix can then be backtracked in a similar fashion as during the backtracking step of the DTW algorithm. However, this time it is not the trench of lowest values through the distance matrix L which is backtracked, but it is the crest of highest hit numbers in the hit-matrix H (see figure Figure 12).

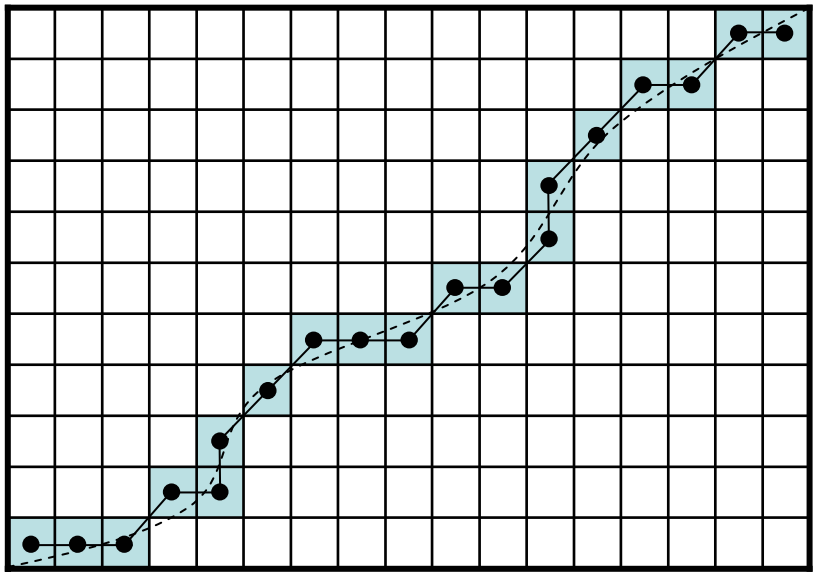

Figure 12 – The interpolated average warp path (dashed) can be restored (dots with solid line) by sampling it with a two dimensional grid (highlighted fields).

There are two main disadvantages to this method:

- It creates sharp corners in the warp path where the interpolated points run by very closely to an intersection of two matrix fields.
- Warp path segments with infinite slope will be created where the interpolated points describe a rather smooth but steep curve.

Nevertheless, this method has some advantages which justify its use:

- The method is computationally efficient.
- It creates a warp path with the exact number of desired nodes for both time scales, $\left(t_{ref} + 1 \times t_{avg} + 1\right)$.
- Referring to the second disadvantage, it can be argued that segments with steep slopes were created by averaging infinite slope segments in step (2). Therefore, infinite slope segments are restored by this method.
- The backtracking is performed in a similar manner as in which the DTW reconstructs a warp path. This ensures that previous results are preserved. Most importantly, however, we abide by the boundary conditions of the DTW mapping behaviour. With this method, each point of curve 1 is mapped onto at least one point of curve 2 and furthermore, the mapping is monotonous.

3.5.4 Creation of a structural average for discrete curves

The calculation according to step (4) of the algorithm is straightforward and mainly designed for curves with continuous values. In our framework, the average was calculated in the same way, however, this resulted in non-discrete function values of the range [0 … 1]. The original modelling of instrument usage only allowed the two states one and zero, representing whether the instrument is in use or not. For the modelling of an average curve, the range of allowed values was extended.

Average values between [0; 1] can be interpreted as quasi-probabilities. All n surgeries of the set are synchronized onto the average surgery. Thus, for one time point t_k ($t_k = [0 \ldots t_K]$) of the average surgery, there is always at least one corresponding instrument vector from each of the actual surgeries. Let us assume that at t_k, n_k out of the n surgeries had a certain instrument in use. Then, the value for this instrument in the average will be

$$f_0(t_k) = \frac{n_k}{n}.$$

For example, the average of our three test surgeries featured values of [0; 1/3; 2/3; 1]. These values are not a probability in the axiomatic sense, but they give an idea of how well an instrument could be synchronized by the algorithm. Furthermore, they reflect how well the surgeries correspond to each other at t_k.

3.5.5 Preliminary results

The creation of an average warp path and an average surgery was implemented as described in the paragraphs above. As an example case, $OP1_{test}$ and $OP3_{test}$ were warped onto $OP2_{test}$. $OP2_{test}$ thus served as the reference surgery according to step (1) of the algorithm. Figure 13 shows the two warp paths $h_1(t)$, $h_3(t)$ and the resulting average warp path.

It should be remarked that horizontal and vertical path segments that are common to both warp paths are preserved in the average.

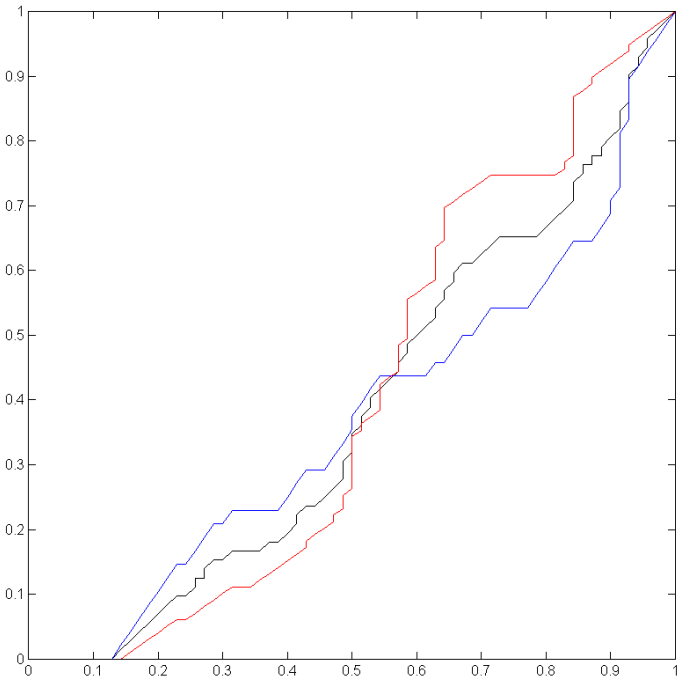

Figure 13 – $h_1(t)$ (blue), $h_3(t)$ (red) and average warp path (black)

Figure 14 shows the average surgery that was calculated for the three surgeries in the test set. As it has been mentioned before, it features instrument representations with the values [0; 1/3; 2/3; 1].

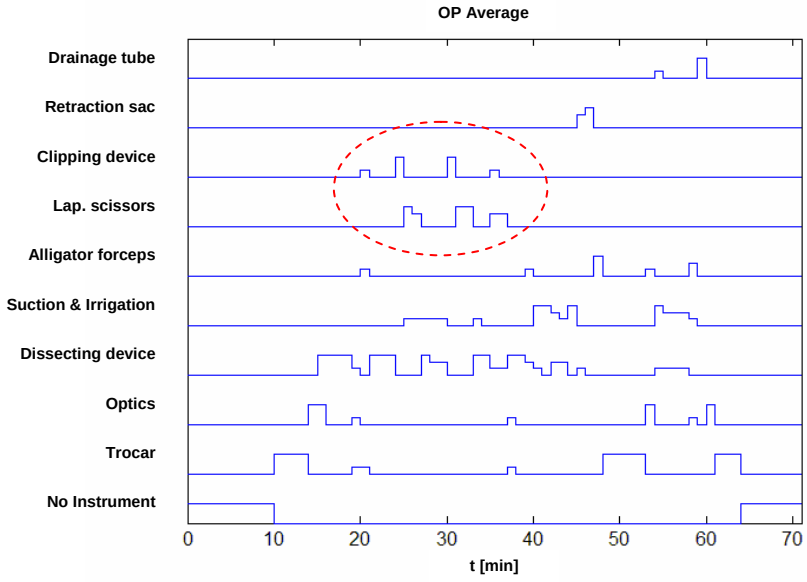

Figure 14 – Average surgery created out of the three test surgeries

One exemplary sequence in the average curve is marked in the figure. Since this is a good illustration of how well the average curve represents the actual curves, the following paragraph will shortly go into detail and explain what this curve segment implies.

It can be seen that in the average, the correct order and number of vessel clipping and subsequent vessel cutting has been preserved. With a quasi-probability of 1, there are two peaks for the clipping device and two peaks for the laparoscopic scissors. This corresponds perfectly to the real LC workflow as described in 2.4. Once again, this should not be interpreted as an axiomatic probability of 1, but it should be inferred that this is a chain of events which occurred in all three test surgeries. However, in the first surgery, some clipping was performed before this event sequence. During the third surgery, the branching of a vessel required the surgeon to conduct a third clipping. In the average curve, this is represented by two small peaks with a quasi-probability of 1/3, stating that these events only occurred in one of the three surgeries. This already demonstrates that a certain workflow is reflected in the average surgery and that abnormalities in the workflow are faded out in the average. In order to be able to deduct the workflow, however, the average has to preserve all important events as good as possible.

In order to receive a good average curve, the method described so far can be further improved by weighting the instruments. This will be described in the following section.

3.6 Automatic weighting of surgery instruments

3.6.1 Motivation for instrument ranking

As it has been mentioned already, the laparoscopic instruments used in a minimally invasive surgery relate to the workflow status of the surgery. Taking a closer look at the data of several LCs, however, leaves the impression that some instruments directly correlate with a certain event while other instruments can lead to ambiguous implications. For example, the usage of a clipping device clearly indicates that a vessel is being clamped. The usage of a dissecting device, on the other hand, can suggest that a vessel is being dissected, that the gallbladder is being detached from the liver or that the detached gallbladder is being put into the retraction sac. Especially the latter case demonstrates the idea that the event of retracting the gallbladder is better reflected by the usage of the retraction sac than by the usage of the dissecting device. In other words, if the DTW preferred synchronizing certain instruments over others, this might lead to a better synchronization of surgeries in general. This is especially important when creating an average, where overlaying many surgeries might lead to blurry edges in the average instrument tracks.

The one-dimensional representation of surgery data in 3.1.2 already featured such a biased ranking of instruments. As it was denoted there, the ranking relied on a subjective perception of instrument importance. A more objective ranking is introduced here.

3.6.2 Iterative weight updating algorithm

An objective weighting method is proposed by Kassidas et al. in their paper on the synchronization of batch trajectories using Dynamic Time Warping [9]. The so-called batch trajectories, which in fact are multi-dimensional curves like ours, represent chemical processes where the development of each process variable over time forms one dimension of the batch. They have to be synchronized for better comparison of batch histories and for the application of statistical analysis. In particular, the synchronization significantly improves the results of a certain chemical analysis technique called Multiway PCA/PLS.

In the background of [9], some variables are more suited for DTW synchronization than others. Suitable variables for example feature a monotonic development. In contrast to that, heavy noise, for example, impairs other variables to such an extent that they become unsuitable for synchronization. Therefore, it is suggested to give a higher weight to appropriate variables during the filling of the DTW distance matrix. The weight is adjusted in an iterative process.

The first iteration starts with an equal weight for all variables and an average curve is generated. Subsequently, each variable from the set is compared to its averaged, synchronized equivalent. Those variables which could be better synchronized with the average will then receive a higher weight for the next iteration, while badly synchronized variables, e.g. noisy ones, will receive a lower weight.

This principle is adopted for the calculation of our surgical average. The evaluation of synchronization quality is calculated according to the following formula,

$$W(j,j) = \left[\sum_{i=1}^{n} \sum_{k=0}^{t_{avg}} \left[OP_i(k,j) - f_0(k,j) \right]^2 \right]^{-1},$$

where W stands for a diagonal matrix with the j-th diagonal element representing the j-th instrument from our instrument vector. OP_i is the i-th surgery from our set of n recorded surgeries. $f_0(k)$ is the calculated average surgery with k being the timestamp of the range [0 … t_{avg}]. The difference between each instrument of each surgery and the respective instrument of the average is thus summed up and squared. Taking the reciprocal value ensures that an instrument with little overall difference is weighted higher in the next iteration.

Before the weights can be used, however, they are normalized so that the sum of all weights equals the number of instruments n_{Instr}:

$$W\left\{ N / \left[\sum_{j=1}^{N} W(j,j) \right] \right\}.$$

According to [9], the number of iterations has to be found empirically.

3.6.3 Weighting results

Figure 15 shows the development of instrument weights after ten iterations of calculating an average curve. As it has been assumed, some variables are weighted to be more important in contrast to others. For example, the suggestion that the retraction sac should be weighted higher than the dissecting device is clearly affirmed by the iterative procedure. However, the clipping device was ranked rather high by the intuitive assessment in 3.1.2 while it loses weight during the objective calculation here.

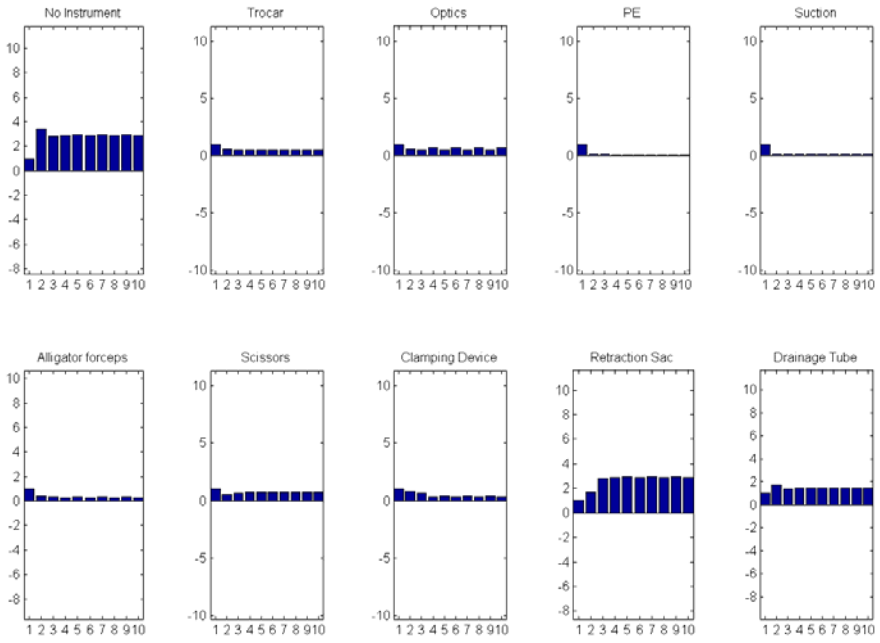

Figure 15 – Weight updating during ten iterations of calculating an average surgery

In the average curve, this leads to small but significant changes. Figure 16 shows two magnified segments of the average curve. The left image shows the average as calculated with an equal weighting for all instruments and the right image shows the average after the tenth iteration of instrument weighting.

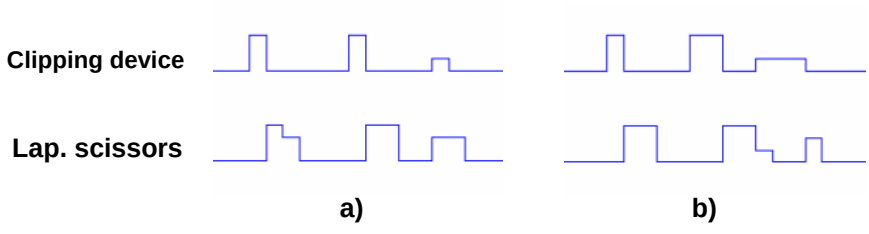

Figure 16 – Clipping phase in an average curve created with (b) and without (a) instrument weighting

The curves represent the workflow phase of clipping and cutting of the two main vessels of the gallbladder. Due to the distributed weighting which led to Figure 16 b), the first usage of scissors has been better synchronized in all three surgeries. Therefore, the peak has a quasi-probability of one which does not drop. The same applies to the second peak. Another significant change is that in the left image, the third clipping and cutting takes place at the same time. This cannot be done in reality. In the right image, the clipping is performed before cutting. Therefore, this progression corresponds better to the actual workflow.

Although these changes are very small, they demonstrate a certain improvement of synchronization results. It can be assumed that for more than three surgeries and for finer time resolutions, the automated weighting procedure will have a more significant impact on the creation of an average curve. Accordingly, this will lead to a better representation of an average workflow.

All in all, both suggestions have been confirmed, namely that the weighting of instruments helps in finding a better average, as well as that the weighting should be conducted by an objective method for optimal results.

4 Results

This chapter is dedicated to the description of the workflow recovery method and to the results obtained with the final dataset.

First, 4.1 describes how workflow phases are assigned to the average curve. The time relation between the average surgery and each surgery from the dataset can then be used to retrieve the workflow phases for each surgery.

Second, all algorithms are applied onto the real dataset in 4.2. To begin with, it is described how the final dataset was obtained. Then, the results are presented and evaluated. This includes the weighting of instruments, the creation of an average curve, the performance of workflow recovery with our method and the performance of signal controlling using the average curve. A GUI is used to analyze and display the results in an intuitive and understandable manner.

4.1 Recovery of Workflow

4.1.1 Definition of workflow phases

The discussion in 3.5 and 3.6 showed that a structural average preserves synchronizable events very well if the associated instruments are weighted appropriately. These are the events which also have the highest significance for our method of workflow recovery. Therefore, the recovery of the workflow is very straightforward once a representative average curve has been generated.

In the structural average, the beginning and the end of a characteristic segment is marked and the time within is declared as a certain workflow phase, depending on the events taking place within this period of time. This can be described as a classification $t_k \mapsto wp(t_k)$, where t_k is the timestamp of the workflow and wp is a mapping of the timestamp onto a set of m workflow phases $WP = \{wp_1, wp_2, \ldots, wp_l, \ldots, wp_m\}$. t_k lies inside of a workflow phase l, if it lies between the beginning and end time, $t_{l,b} \leq t_k < t_{l,e}$. The workflow of an arbitrary surgery can be retrieved by using its warp path onto the average curve. For the n curves which form the average curve, this warp path is the inverse shift function of step (3) in the DTW algorithm (cf. 3.5): $wp\left(t_k = u_i^{-1}(t)\right)$. For every new curve, the workflow phase can be retrieved after calculation of a warp path w_i onto the average, and also taking its inverse, $wp\left(t_k = w_i^{-1}(t)\right)$.

It is important to note that these workflow phases have to be identified manually by inspecting the average curve. Therefore, this identification has to be performed by an individual with insight into the general workflow of the inspected surgical procedure, e.g. by a surgeon or an engineer. As discussed in the last chapter of this thesis (cf. 6), one might think of an automated recovery of the workflow for the future. Until this is possible, the manual identification of the workflow by a medically trained person is the most reliable and fastest method.

Breaking down the average of the three test surgeries into workflow phases is skipped for the moment. Instead, the next chapter will describe the collection of more detailed real-life surgery data. The recording of more instruments and the sampling of instrument vectors with 1 Hz enables us to give a more detailed and finer segmentation of workflow steps than for the test surgeries. In the next section, a summary of all results with the algorithm and the workflow recovery will be given.

4.1.2 Controlling context-sensitive events

The workflow can be recovered by assigning surgery phases to the time line of the average curve. Using the same method enables us to control certain events which are strongly related to the current workflow step.

In our framework of a LC, two examples shall be used to demonstrate this concept. These two events systematically occurred at the same workflow-related moments during all surgeries. The first event is the controlling of the OR lights. They are usually switched on at the beginning and at the end of the surgery as well as during the retraction of the gallbladder, but they are switched off during laparoscopic activity. The second event which can be controlled is the calling of the next patient. In all surgeries, this occurred after having clipped and cut the second main vessel of the gallbladder. The methodical recurrence of these events allows for modelling them with similar classifications as in 4.1.1.

The result of event controlling will be presented in a GUI which was developed for this project. This will be further explained in section 4.2.3.

4.1.3 Effect of workflow changes

An interesting challenge for a workflow recovery system is the special case of abrupt changes in the workflow. In the following paragraph, the results of a synchronization attempt will be shortly described. It should be noted that this is no formal description of the problem but a mere demonstration of the effect on our system.

Possibly the most significant change in workflow during a laparoscopic cholecystectomy occurs when the procedure has to be changed to open surgery. As it has been described in section 2.4, this happens in less than 5% of all LCs. Nevertheless, such an event should be detected by the system.

During the observation of several surgeries in the course of this project, a change to open surgery occurred in one of ten observed cases. The change to open surgery was decided during the vessel dissection phase of the surgery. Due to severe

adhesions, neither the artery nor the bile duct could be dissected or clipped. The surgeon's decision to switch to open surgery had multiple effects on the events in the OR. For example, all instruments including all trocars were immediately pulled out of the patient's body, the lights were switched on and the OR team exchanged the set of surgical instruments. Unfortunately, this surgery was not recorded with video data so that no instrument vectors could be retrieved. Therefore, the change to open surgery will be "simulated" in order to demonstrate the effects.

The next section will describe the recording of six LCs at the surgical site of our clinical partner. In order to simulate the effects of such a change in workflow, the sixth surgery will be modified to a similar workflow as during the aforementioned change to open surgery. At an arbitrary point during the vessel dissection, all instrument variables will be set to zero for the rest of the curve. The other five surgeries will be used to calculate an average surgery. Then, the original sixth surgery OP_6 and the modified surgery OP_6* are synchronized with the average.

For our system, the abrupt change in workflow means that the former instrument vectors are hard to match with subsequent feature vectors of the surgery. This is well reflected in the DTW distance matrix L. Figure 17 shows 3D plots of the DTW matrix for OP6 and OP6*. The steep rising of accumulated distance for OP6* indicates that the workflow has been changed in a way that complicates any further synchronization significantly. The overall sum of accumulated distances in the DTW matrix L could be an indicator for this effect. Accordingly, appropriate methods can be designed and implemented into the system so that severe workflow changes can be detected.

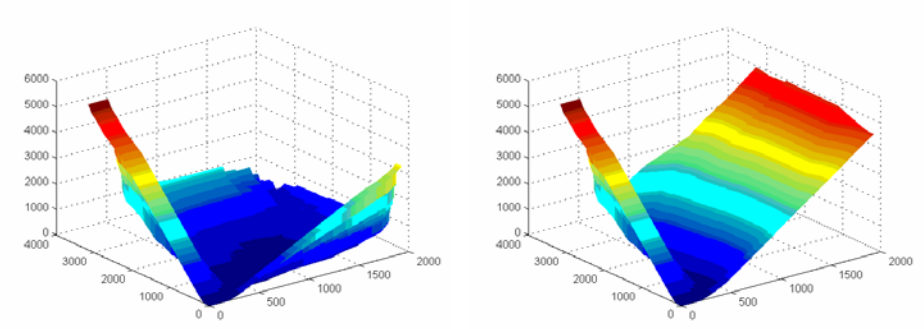

Figure 17 – Synchronization of the sixth surgery from the real data set onto an average created out of the other five surgeries. The DTW matrix reflects normal synchronization with OP6 (left) and steeply increasing accumulated distances for OP6* (right) when a significant change in workflow occurs.

4.2 Real life Application

So far, the DTW algorithm, the creation of an average curve and the workflow recovery method have been conceptualized and implemented. In this section, the developed system will be applied onto real-life data. First, the source of data will

be described, then the results will be analyzed in short and finally, a GUI will be explained where all results can be displayed simultaneously and in a comprehensive manner.

4.2.1 Acquiring of real-life data

The real-life data was taken from laparoscopic cholecystectomies which were again conducted under Prof. Feussner and his surgical team at the Klinikum Rechts der Isar, Munich. During the project time of this thesis, the data of six procedures could be acquired. However, for further publications on this topic, the CAMP chair aims at collecting the data of ten to fifteen surgeries.

Instead of manual protocols, several signals were recorded in the OR this time. The signals were analyzed offline at the institute after the surgeries were finished. In the following, a table will be given with all recorded signals, the signal types, the recording devices and their specific purposes.

Signal	**Signal Type**	**Recording device**	**Purpose**
External videos	2-3 video streams	2-3 HDTV cameras	• Tracking the number and position of surgical staff • Information on OR lighting
Laparoscopic Instrument in use over time	Discrete states	Manual detection from laparoscopic video	• Signal for synchronization • Identification of surgery phase and progress
HF mode	Audiostream	Wireless microphone attached to HF generator	• Detection of cutting and coagulation phases • Additional dimension to the instrument vector
Laparoscopic video	Videostream	Video switch at laparoscope device	• Illustration of laparoscopic surgery • Manual identification of instrument vector

Table 3 – Signals recorded in the OR for final datasets

It has to be noted that for the actual surgeries, the sampling rate for instrument vectors was raised from one measurement per minute to one measurement per

second. Furthermore, the number of recorded instruments was increased from ten to seventeen. As it was described in 2.4, cutting the umbilical port and inflating the abdomen with CO2 marks the beginning of relevant workflow to us and the first abdominal suture marks the end.

Figure 18 shows an exemplary instrument curve of the first surgery, OP1.

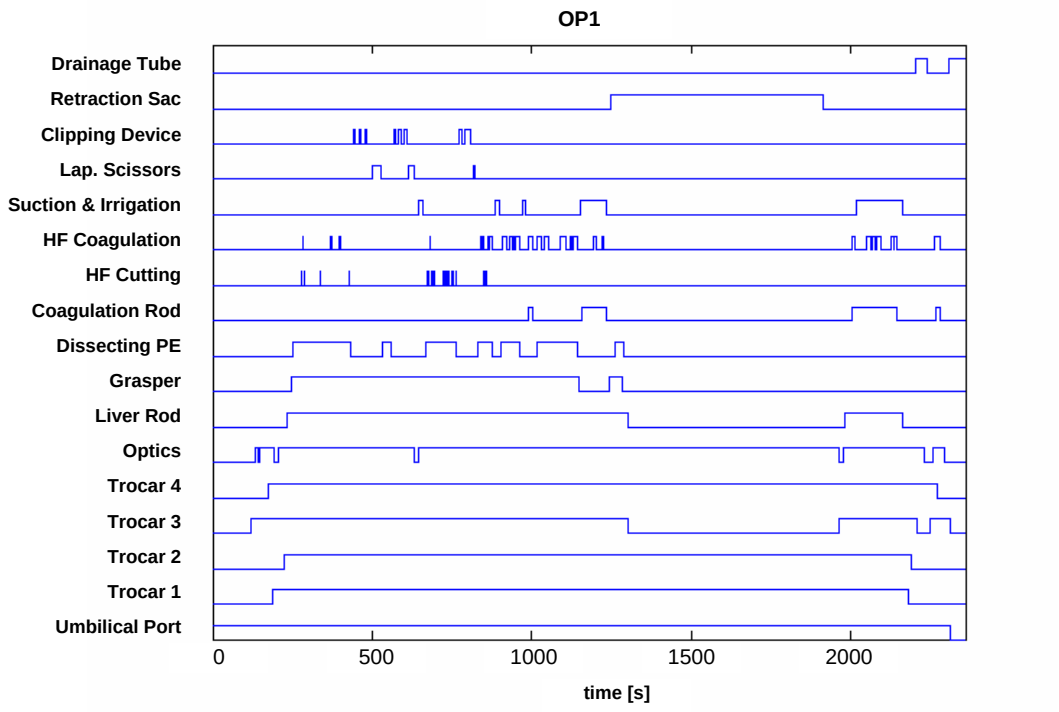

Figure 18 – Multi-dimensional signal representation of OP1, the first of six real data surgeries recorded at the Klinikum Rechts der Isar.

4.2.2 Results with the developed algorithms

The synchronization towards an average curve was performed with only five of the six surgeries. The sixth surgery was then synchronized with the average in order to analyse how well the workflow could be retrieved for unknown curves.

The average curve was calculated with 10 iterations of weight updating. First, the development of four exemplary weights will be shown, and then the average surgery including 14 workflow phases is presented and discussed.

Instrument weighting Figure 19 shows the development of weights after ten iterations of the DTW synchronization algorithm. After initializing every weight with the value one, the weight is distributed among the instruments. This happens, as predicted, according to their workflow relevance. “Trocar 3” and the “drainage tube” receive high weightings since their usage and non-usage reflect distinctive workflow phases. The clipping device is of course particular for the clipping phase, however, it could not be synchronized as well as the other two signals and thus is not weighted as high. The lowest weighting is assigned to the dissecting

PE. This corresponds with the subjective assessment of its low workflow relevance.

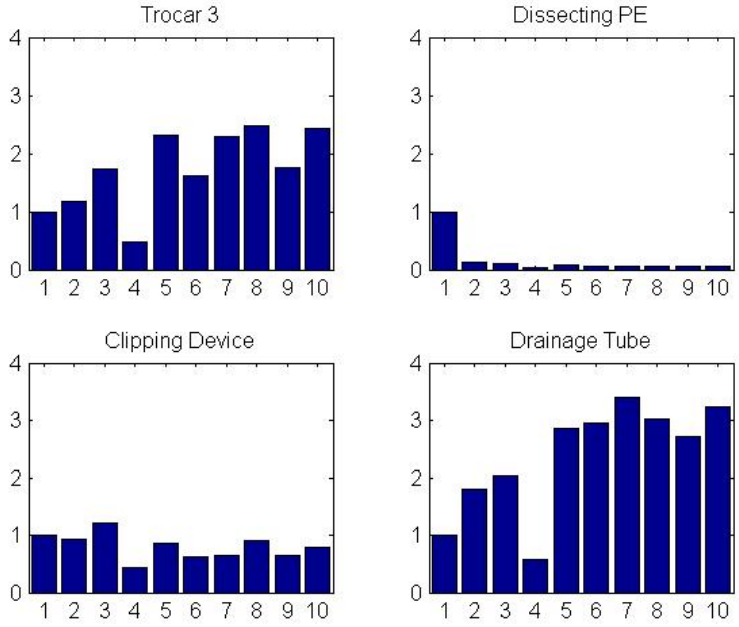

Figure 19 – Weight development for four instruments after ten iterations of average curve calculation

Average curve and workflow phases Figure 20 shows the average curve after ten iterations of weight adjustment. Just as in 3.5, it features quasi-probabilities for the usage of each instrument over time. Remarkably, despite the fact that five surgeries were used this time, there are still some events which could be synchronized so well that some instrument curves still rise from zero to one and vice versa at important workflow steps. Using these significant curve landmarks, 14 workflow phases were identified and labelled within the average curve.

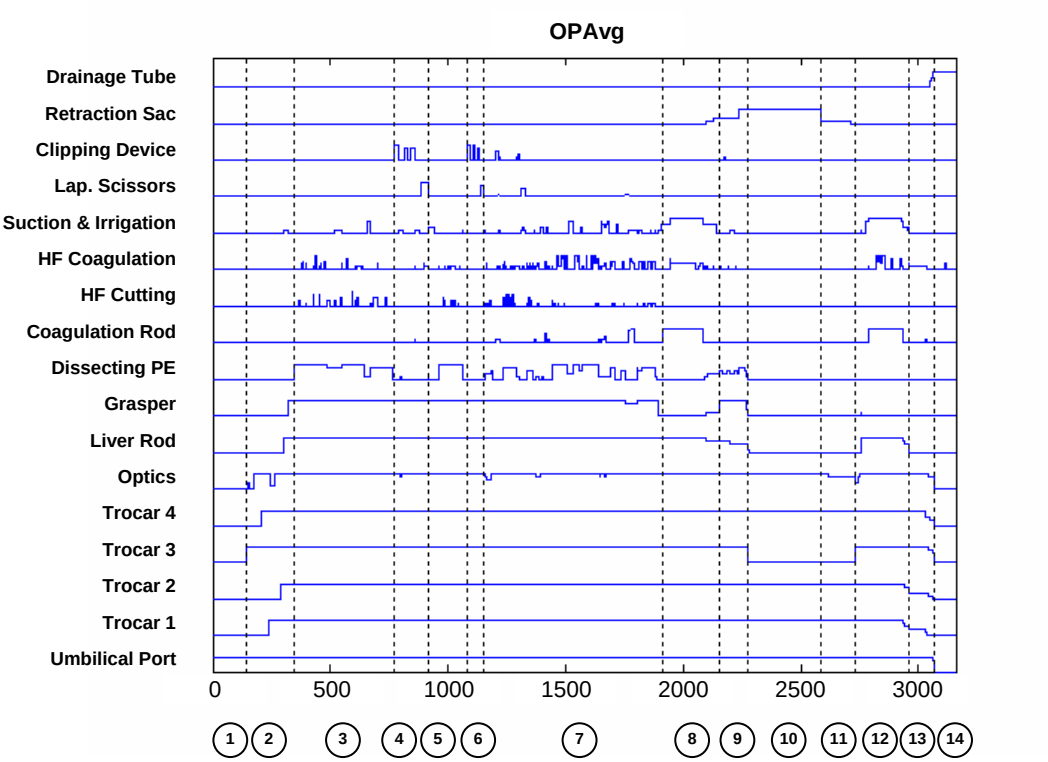

Figure 20 – Average curve over time, after 10 iterations of weight adjustment. 14 workflow phases were identified despite the averaging process, their labels can be found in Table 4.

Performance of workflow recovery The same 14 phases as in the average were looked up within each surgery curve. The start time of each phase was read from the raw surgery data. For each phase, however, a second starting point exists in form of a virtual starting point which is proposed by our system. This virtual point could be calculated by taking the starting point of the respective phase in the average and transforming it onto the timeline of the raw surgery.

The actual starting point was taken as the reference and the virtual point was compared against it. A match of the two times reflects that our system was able to classify the surgery phase correctly. A deviation of up to five seconds was tolerated, because our phase lengths range from 69 to 758 seconds. Five seconds therefore are less than 10 % of the shortest phase which we try to identify. However, every phase for which the starting point had an aberration of more than five seconds was valued as a wrong or inexact classification.

Table 4 shows the workflow phases including the events which triggered their beginning. Furthermore, the reference and the virtual start time are given for each phase and for each surgery. Deviations above five seconds are highlighted. On an average surgery duration of 2924 seconds, the maximum deviation was 157 seconds or 5,7%. Each surgery contained 13 starting points which should be matched (the first of 14 starting point is at time zero for all curves and is therefore ignored). Over six surgeries, an average of 11,3 out of 13 trigger points were

classified correctly. This amounts to a value of 87% for the rate with which our system can classify surgery phases successfully.

#	Phase name	Start time	OP1	OP2	OP3	OP4	OP5	OP6
1	CO_2 inflation	t	0	0	0	0	0	0
		t*	0	0	0	0	0	0
2	Trocar insertion	t	120	120	90	186	157	194
		t*	120	120	90	186	157	194
3	Dissection phase 1	t	253	248	288	419	376	347
		t*	253	248	288	419	376	347
4	Clipping/ cutting 1	t	442	610	951	935	742	697
		t*	442	610	951	935	742	697
5	Dissection phase 2	t	527	727	981	1079	969	775
		t*	527	727	997	1222	1025	776
6	Clipping/ cutting 2	t	566	850	1135	1677	1086	800
		t*	566	850	1135	1677	1086	800
7	Gallbladder detaching	t	631	913	1204	1789	1187	931
		t*	631	913	1235	1792	1112	931
8	Liver bed coag. 1	t	1158	1829	2083	3146	1369	1359
		t*	1158	1829	2083	3146	1369	1359
9	Packaging of gallbladder	t	1245	2051	2285	3636	1553	1483
		t*	1245	2051	2285	3760	1553	1483
10	External gb. retraction	t	1304	2263	2382	3972	1606	1612
		t*	1304	2263	2382	3972	1606	1612
11	External cleaning	t	1912	2870	2446	4065	1769	1684
		t*	1912	2870	2446	4217	1612	1684
12	Liver bed coag. 2	t	1961	3073	2464	4420	1782	1696
		t*	1961	3073	2464	4420	1782	1696
13	Trocar retraction	t	2160	3257	2688	4600	2021	1816
		t*	2160	3257	2688	4632	2023	1821
14	Abdominal suturing	t	2293	3290	2717	4697	2195	1842
		t*	2307	3290	2718	4692	2192	1842
Correct detections:			12/13	13/13	11/13	9/13	10/13	13/13
Success rate total:			11,3/13 or 87%					

Table 4 – Table with phase numbers and respective labels. t is the actual beginning time of the phase, t* is the beginning time as proposed by the developed system. Deviations of more than five seconds are highlighted.

It has to be mentioned that the sixth surgery (OP6) was not used for the averaging process. It was rather synchronized onto the average curve in order to demonstrate the capability of the system to classify unknown curves. The synchronization was performed only once, but the instruments were weighted with the values that were

obtained after ten iterations in the averaging process (as proposed by [9]). Remarkably, all phases of this new and unknown surgery were classified correctly. This test is performed with only one surgery and one average and thus cannot be taken as a general quality measure. But still, it demonstrates the potential of our approach to recover the workflow of unknown surgeries.

Performance of signal controlling As a next step, a demonstration of the controlling capabilities with this approach shall be given. Two of the activities within the operating theatre which are dependent on the workflow stage can be manipulated by our system.

The first signal is the status of the ambience and OR lighting. Both are switched on at the moment when the laparoscope is first inserted into the body. They are completely switched on during phase ten, the external gallbladder retraction. Then, they are turned off again until the end of phase 14. This is the moment when the laparoscope is pulled out of the body for the last time and the abdominal suturing begins. See Table 5 for an evaluation of the controlling performance. The controlling of the lights can be claimed to be as successful as the phase classification.

The second controllable signal is the calling of the next patient. In all nine observed procedures, the surgical nurse called for the next patient shortly after the second main vessel was clipped and cut. This is the event which marks the end of phase six and triggers the beginning of phase seven. Unfortunately, there are no exact timestamps for this event within our recordings. However, the request to prepare the next patient was only roughly tied to the end of phase six. Within the average, this event is detected with a tolerance of 75 seconds. Since the calling of the next patient is not very time critical, our system can be argued to give reasonable suggestions for this task.

#	Light status	Start time	OP1	OP2	OP3	OP4	OP5	OP6
1	On (start of OP)	t	0	0	0	0	0	0
		t*	0	0	0	0	0	0
2	Off (start lapar.)	t	135	134	102	200	187	217
		t*	149	150	102	200	187	236
3	On (ext. gb. retr.)	t	1304	2263	3282	3972	1606	1612
		t*	1304	2263	3282	3972	1606	1612
4	Off (cont. lapar.)	t	1961	3073	2464	4420	1782	1696
		t*	1961	3073	2464	4420	1782	1696
5	On (lapar. finish)	t	2293	3290	2717	4703	2195	1842
		t*	2293	3290	2718	4692	2194	1842
Correct detections:			4/5	4/5	5/5	4/5	5/5	4/5
Success rate total:			4,3/5 or 87%					

Table 5 – Table showing light phases and the moments when the lights are switched on or off. t is the actual switching time in the surgery, t* is the switching time as proposed by our system. Deviations of more than five seconds are highlighted.

4.2.3 GUI concept

The graphical user interface was designed in order to show as much of the collected surgery data as possible. The usual data set of one surgery contains one laparoscopic video and up to three external videos. For each surgery, these videos had to be synchronized with each other. This was achieved by holding a regular stopwatch as a reference timestamp into the camera and later reading out the offsets of each video towards the stopwatch.

The main feature of the GUI is to display the DTW synchronization of two or more videos. One reference surgery is loaded into the GUI and an arbitrary surgery can be loaded and mapped onto the reference. The warp path shall give an idea of how the two time scales are linked to each other. If the average surgery is loaded as the reference, an arbitrary number of surgeries can be loaded and compared against each other and the average.

The GUI also has some additional features:

- The average allows displaying a workflow phase as meta-information in the status bar of the GUI.
- A workflow-related clock helps to give an idea of the progress of each displayed surgery and how long each procedure actually took.
- Using the play button, the reference surgery can be watched in regular speed, while all compared surgeries try to match the workflow of the reference surgery through their warp paths. Optionally, using the sliders of each surgery, an arbitrary point of the surgery can be chosen and viewed, while again all other surgeries display synchronized contents.
- For each surgery, a drop-down menu enables the viewer to watch the surgery either from the laparoscopic view or from one of the three external views of the surgery.
- A top-view scheme of the OR is offered as an additional view option. The positions of the surgeon and the assisting team were tracked with binary markers attached to their shoulders. Using the position data, the movement of the surgical team can be observed. The DTW synchronization can help in this view in order to watch common events simultaneously and thus discover weaknesses in the OR layout (not yet implemented upon submission of this thesis).

The GUI was implemented using C++ and QT 4.0. The implementation of the GUI was not part of this bachelor thesis. It was entirely realized by the working student Ralf Stauder at the CAMP chair. He was also the project partner responsible for the position tracking of the surgical personnel. The concept of the GUI was drafted by the whole project team.

A screenshot of the GUI is displayed in Figure 21. Unfortunately, at the time of writing this thesis, the GUI was not yet fully finished and only the comparison of two surgeries at a time was realized. The screenshot shows the synchronized entry of the clipping device into the body of the patient.

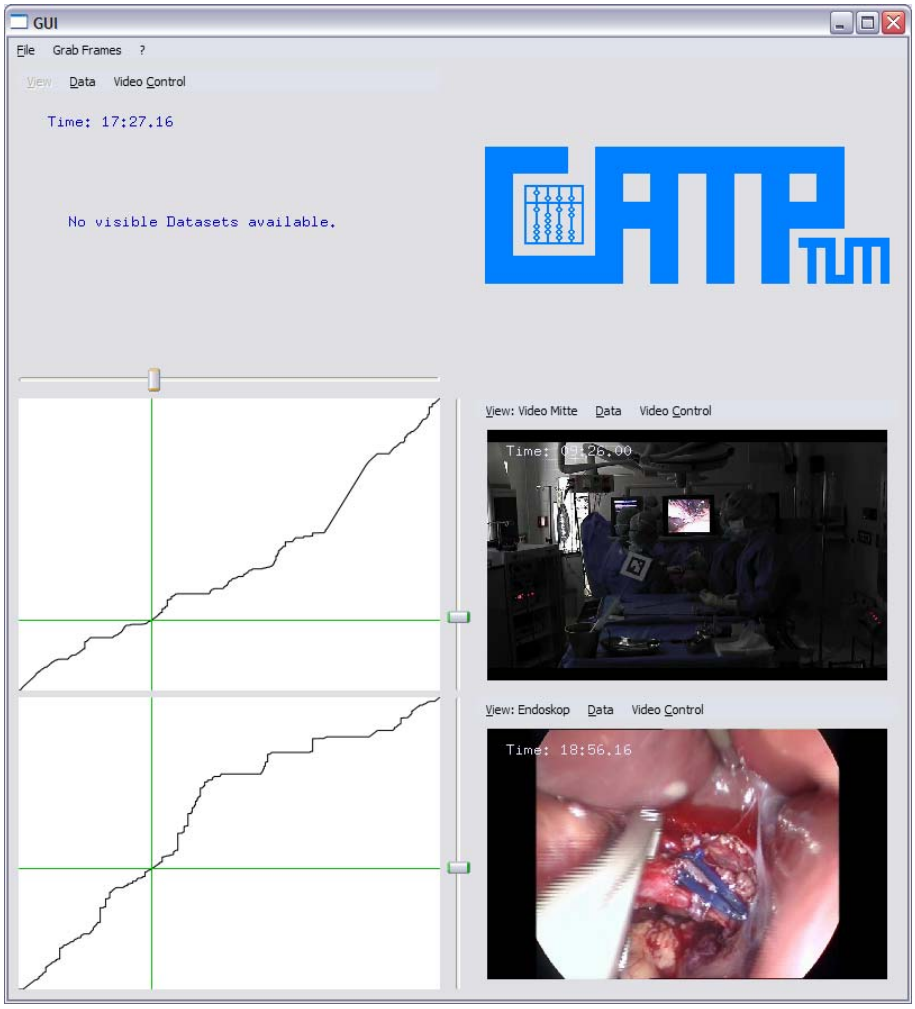

Figure 21 GUI screenshot showing two surgeries (second and third row) synchronized to an average (top row). For each surgery, an arbitrary view can be chosen. Here, the upper surgery is shown from external view and the lower surgery is shown from the viewpoint of the laparoscopic camera. This screenshot shows the synchronized event of clipping one of the two gallbladder main vessels. Although this event took place at entirely different moments in both surgeries (9:26 and 18:56 minutes), they are matched to the average by their respective DTW time warp paths (left column) and displayed simultaneously.

5 Discussion

This chapter serves as a summary of all results obtained during this thesis. 5.1 gives a preliminary conclusion of the results. The strengths and drawbacks of the proposed method are then discussed in 5.2. In 5.3, two other approaches by different research groups are evaluated. Their methods incorporate workflow recovery with Petri Nets and Hidden Markov Models. These are compared to the proposed system. Eventually, a final conclusion is given in 5.4. Again, the advantages and disadvantages of our system are outlined. This time, however, they are evaluated in contrast to the other described methods. A suggestion is given on how to merge the discussed methods in order to obtain more capable workflow recovery systems.

5.1 Preliminary conclusion

In this thesis, a system has been developed for the recovery of workflow during minimally-invasive surgeries. Our approach was confined to recover the workflow of only one kind of surgery, namely a laparoscopic cholecystectomy.

The basic concept of our system is to synchronize several data curves where each curve represents one surgical procedure. The synchronization is capable of matching common events in surgeries although they occur at different times. Once several curves have been synchronized, an average curve is created. Workflow phases are assigned to the average curve and by linking unknown surgeries to the average, workflow phases can be recognized for all new surgeries. The signal used in our context is the usage of laparoscopic instruments over time. The synchronization is achieved with the DTW algorithm and several enhancements of the DTW which allow synchronizing an arbitrary number of trajectories.

Using five surgeries to create an average curve, we could identify 14 workflow phases with 13 characteristic trigger events within the average. Using an additional “unknown” curve, we could achieve an overall phase recognition rate of 87%. The recognition rate for the unknown surgery reached 100%, meaning that all 14 phases for the unknown surgery could be classified correctly.

5.2 Strengths and drawbacks of the proposed method

Since this thesis marks a first step towards the recovery of workflow, the proposed method naturally has some disadvantages. Two main disadvantages could have been identified and shall be mentioned here:

- Especially for larger sets of surgery data, the creation of an average curve requires large computation times.
- The DTW approach can only enable the system to identify workflow sequences with a fixed order.

Apart from these disadvantages, the given method poses two challenges which require careful handling. Only if these two requirements are met, good recognition results can be expected:

- A representative, high-quality average curve has to be found which preserves the workflow steps very well.
- Characteristic trigger events within the curves have to be found in order to define recognizable workflow phases.

On the other hand, recovering the workflow with the DTW algorithm also has some advantages over other methods:

- A multitude of signals can be used for synchronization. Signal types may include continuous, discrete or logical signals. Arbitrary events can be included, like the exertion of force, the development of temperature, the spatial movement in 3D or the instrument usage, as in our case. As long as appropriate distance functions for the DTW algorithm are defined, the events will be synchronized accordingly.
- Events can be synchronized by mere processing of the signal data. No knowledge of the underlying workflow is required.
- The DTW achieves excellent synchronization results. This is well reflected in the GUI.

5.3 Concept and performance of other approaches

Apart from workflow synchronization with the DTW, there are many other approaches dealing with this issue. Two other relevant approaches shall be shortly introduced. The first method is based on Petri Nets and the second employs motion segmentation using Bayes classifiers and Hidden Markov Models.

Petri Nets Maruster et al. [12] proposed the automated discovery of workflow models from hospital data. Using numerous virtual workflow protocols, their method could analyze the underlying processes and automatically create a formal description of the workflow.

The advantage to this approach is that the theory of Petri Nets is well-known and widely used in order to model processes. The main conveniences of Petri Nets lie within their sound mathematical foundation, the graphical representation and their powerful analysis techniques and tools which have continuously grown in number and quality over the past decades [13]. Petri Nets model processes by connecting places and transitions. In a workflow context, transitions represent workflow activities and connecting places model casual dependencies between activities.

Maruster et al. use the Petri net formalism in order to model the administrative patient handling in a hospital. The evaluation is performed from a business-related viewpoint. A mathematical model was derived automatically from raw workflow data, the so-called patient logs. However, the approach only handled virtual data and in one special case, a so-called “non-free-choice” workflow net could not be reconstructed. Nevertheless, there have been further developments in the field which deserve further investigation.

Hidden Markov Models The group around Hager et al. ([14] and [15]) took efforts in automatically detecting and segmenting robot-assisted surgical motions, mainly for the purpose of surgical skill assessment.

Using the Intuitive Surgical’s Da Vinci system, hand motions of a surgeon were tracked by using several variables like position, speed and acceleration. The motion features were then processed using methods like feature normalization and Linear Discriminant Analysis (LDA) in order to reduce data complexity and redundancy. LDA was furthermore used in order to project the features of specific surgical motions into a space where they were better separated from each other.

A Bayes classifier was used in order to segment the features into a set of surgical motions. This method yielded a process recognition rate of around 92% [14]. The employment of Hidden Markov Models (HMMs) allowed for surgical motion segmentation with an accuracy of 86% [15].

HMMs come from the domain of speech recognition. In this domain, HMMs gradually replaced the DTW because of better recognition rates. This fact makes it very interesting to further examine the attributes and capabilities of this approach. A very short overview shall be given here.

HMMs can classify a temporal order of input variable vectors into a set of distinct “vocabulary”. In speech recognition, this vocabulary contains the so-called phonemes. This is the basic set of sounds which a human uses for creating speech in a certain language. Likewise, such a basic vocabulary has to be found for motion detection. This motion vocabulary has to be pre-defined and carefully selected. On the one hand, it must not be too specific, otherwise the classification will be inaccurate. On the other hand, it has to be fine enough in order to fully capture all aspects of the analyzed motion.

The effort that has to be taken in order to find a correct HMM vocabulary can be compared to the efforts which we have to invest in order to find an appropriate average curve and segment it into surgery phases. HMMs, however, have one significant advantage over our approach: different entities of the vocabulary may appear in an arbitrary temporal order. The DTW cannot handle switches in a workflow sequence. On the other hand, the DTW has the advantage that it does

not need any vocabulary before synchronizing two curves. In fact, the DTW can synchronize curves without any knowledge about the underlying workflow. Furthermore, the DTW does not only categorize actions, but it actually synchronizes them over the whole time axis. This is especially useful for replay analyses.

5.4 Final conclusion

Our DTW-based approach has great potential to monitor and recover distinct surgical workflows. This can only be guaranteed under the premise that events within the surgery always follow a specific order. The very first attempt of workflow recovery with the DTW resulted in a recognition rate of 87%.

Although the recognized workflow phases are rather rough, with phase lengths between one and 13 minutes, this is a first step towards satisfying basic requirements of a workflow recovery system as mentioned in the introduction. Due to the synchronization, similar events can be analyzed in a parallel manner. This could be a great help in administrative tasks like analyzing and optimizing the OR layout, the efficiency of surgical staff or the usage of instruments. The system could furthermore be extended to create basic documentation offline and automatically. Apart from that, the performances of two surgeons can be synchronized and compared against each other. If performances of medical students and their supervisors are monitored, our system can serve as a comfortable comparison tool already at this basic stage. This might be a potential application for modern medical education and training.

As mentioned before, the creation of a representative average surgery is crucial to our approach. A recognition rate of 87% still leaves room for improvement and makes further investigations in the DTW-synchronization of multiple curves necessary. One possible source of improvement might be to try out choosing different reference curves for average creation (cf. [10]). Further improvements might be achieved by having more surgeries to choose from and then compile a set of surgeries without any outliers in order to create a sound average. Another idea might be a dynamic weighting of instruments for calculation of the DTW distance matrix, depending on the workflow phase.

As the discussion has shown, there exist other approaches in the field of workflow recovery which are fundamentally different to our system. The recognition rates for these systems are comparable. However, the analyzed processes are often different. In the two examples given above, one interest group is coming from the business side and one is dealing with hand movements for surgical skill assessment. These approaches and our system complement each other in the sense that they compensate each others' weaknesses. Therefore, it is recommended to investigate in these methods as well and try to create hybrid systems with adaptive workflow recovery mechanisms.

6 Outlook

With the outcome of thesis, a first step towards the monitoring and recovery of surgical workflow has been taken. The proposed method is capable of classifying the workflow of a surgery into distinct and subsequent phases. However, the processing of surgical data is so far performed manually. Furthermore, the DTW only achieves optimal results when synchronizing completed recordings of surgeries. Thus, an optimal curve matching can only be guaranteed when it is calculated offline.

Since the offline classification yields promising results, the next step is to realize an online recovery of workflow. Section 4.2.2 showed that unknown curves can be successfully synchronized to an average curve. Thus, workflow phases for unknown surgeries can be obtained. In this respect, it would be highly desirable to perform this classification while the procedure is taking place rather than in an offline manner. Online synchronization is so attractive because many of the mentioned applications and objectives of workflow recovery, including context-specific support of the surgeon and anticipation of next steps only make sense when performed online. The two modelled tasks, switching on and off the lights and calling the next patient, were also evaluated in the face of a possible future scenario in which they are performed live and during a procedure.

Online synchronization, however, creates several new challenges:

- The recognition of instrument vectors has to be performed online. Images from several video recordings made from different perspectives in the OR have to be evaluated. Image processing has to be performed in order to automatically recognize the instruments by their tooltips or some markers attached to them. Information from different videos has to be synchronized and merged appropriately.
- Once the instrument vectors are obtained, the DTW has to be further enhanced for online synchronization. Surgical data has to be synchronized with the average curve although only a part of the surgery information is available.
- In order to make the system reliable, the classification rate has to be further improved. Tasks like context-specific support and controlling of electronic systems depend on a near-perfect identification of workflow steps.

Especially the second point, making the DTW capable of online synchronization, is a big challenge. Although first approaches have been named in [6] and [9], their

methods have to be refined for the given application and further research has to be performed to solve this problem.

Another possible development of this first approach has been mentioned as well. By investigating into workflow recovery theories like Petri Nets and Hidden Markov Models, additional methods can be found and integrated into the given approach. One could think of a hybrid workflow recovery system which classifies coarse workflow steps with the DTW and activates further, appropriate workflow recovery mechanisms in respective surgery phases. By doing this, the surgical workflow can be detected at increasingly fine resolutions. In the example of [15], the system could activate a motion segmentation algorithm based on HMMs during a surgery phase in which the Da Vinci robotic system is used. This would allow identification of the workflow down to single hand movements of the surgeon, while the DTW provides the overall information that the robotic surgery phase is taking place. This example illustrates that combining different workflow recovery mechanism can help to achieve better results.

All in all, the recovery of workflow is one of the crucial tasks towards the long-term goal which has been drafted by experts during discussions in the OR2020 workshop. The ultimate objective, as it has been mentioned in the introduction to this thesis, is the establishment of an intelligent agent with opinionated consultancy inside the OR. Another ambition is the development of an intelligent OR which adapts, supports and controls according to the workflow of the operation.

Up to now, the proposed system is still far from being able to supply such an agent or intelligent OR with a reliable and detailed feedback about the workflow status of the surgery. Nevertheless, a first approach has been accomplished. Since surgical workflow recovery is still a young research field, there is a lot of room for improvement and many theories can be evaluated and integrated into the approach so far. There is a great potential that within near future, a satisfying workflow recovery system can be designed.

7 Reference list

[1] http://www.or2020.org, accessed on: Aug. 10th 2005.

[2] http://www.surgery.usc.edu, accessed on: Jul. 27th 2005.

[3] http://www.laparascope.com, accessed on: Jul. 29th 2005.

[4] C. Herfarth. 'Lean' surgery through changes in surgical work flow. In *British Journal of Surgery, Vol. 90, No. 5, pp. 513–514, 2003.*

[5] M. Dugas, J. Scheichenzuber, and H. Hornung. An Intranet-based System for Quality Assurance in Surgery. In *Journal of Medical Systems, Vol. 23, No. 1, pp. 13-19, 1999.*

[6] T. Sielhorst, T. Blum, and N. Navab. Synchronizing 3D movements for quantitative comparison and simultaneous visualization of actions. In *IEEE and ACM International Symposium on Mixed and Augmented Reality, 2005.*

[7] J. Vogt. Wert der präoperativen Routinediagnostik für die Vorhersage von intraoperativen Schwierigkeiten bei der laparoskopischen Cholezystektomie. *Dissertation, Ruhr University Bochum, Faculty of Medicine & Bethesda Hospital Wuppertal, 2000.*

[8] Factsheet Gallbladder Removal. *Published by BUPA's Health Information Team (British United Provident Association), 2003.*

[9] A. Kassidas, J. F. MacGregor, and P. A. Taylor. Synchronization of Batch Trajectories Using Dynamic Time Warping. In *AIChE Journal, Vol. 44, No. 4, pp. 864-875, 1998.*

[10] T. Gasser and K. Wang. Alignment of curves by Dynamic Time Warping. In *The Annals of Statistics, Vol. 25, No. 3, pp. 1251-1276, 1997.*

[11] T. Gasser and K. Wang. Synchronizing sample curves nonparametrically. In *The Annals of Statistics, Vol. 27, No. 2, pp. 439–460, 1999.*

[12] L. Maruster, W. v. d. Aalst, A. Weijters, A. v. d. Bosch, and W. Daelemans. Automated discovery of workflow models from hospital data. In *Proceedings of the 13th Belgium-Netherlands Conference on Artificial Intelligence (BNAIC 2001), pp. 183–190, 2001.*

[13] W. v. d. Aalst. The application of petri nets to workflow management. In *Journal of Circuits, Systems and Computers, Vol. 8, No. 1, pp. 21–66, 1998.*

[14] H. C. Lin, I. Shafran, T. E. Murphy, A. M. Okamura, D. D. Yuh, and G. D. Hager. Automatic Detection and Segmentation of Robot-Assisted Surgical Motions. In *MICCAI 2005, LNCS 3749, pp. 802–810, 2005.*

[15] T. E. Murphy and G. D. Hager. Towards Objective Surgical Skill Evaluation with Hidden Markov Model-based Motion Recognition. *Master Thesis, Johns Hopkins University, Baltimore, Maryland, 2004.*

[16] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. In *IEEE Trans. Acoustics, Speech, and Signal Proc., Vol. ASSP-26, pp. 43-49, 1978.*

8 Appendix

8.1 Medical Terms

Medical term	Explanation
Trocar	Surgical instrument which is used to puncture through the patient's skin. During laparoscopic surgery, the trocar serves as a port through which different instruments like graspers, irrigators or the endoscope can be inserted into the body.
Endoscope	Miniature camera which can be used to inspect cavities inside the human body.
Laparoscope	Slim endoscope which is inserted through an incision in the abdominal wall in order to perform minimal-invasive surgery in the abdominal region
Abdomen	Part of the body that lies between the chest and the pelvis, also called belly or venter
Umbilicus	Navel (also called belly button)

8.2 Protocols of test surgeries

Surgery:		$OP1_{test}$	
Date:		Friday, 15th July 2005	
Time	**Instrument**	**Activity**	**Lights**
08:46	No instrument	Preparation of the patient (disinfection, inflation of the abdomen)	On
08:47	Trocar	Insertion of the trocars	Off
08:50	Optics	Inspection of the surgery site	
08:51	Dissecting device	Dissection of arteria cystica	
08:52	Clipping device	Clipping of arteria	
08:53	Laparoscopic scissors	Cutting of arteria	
08:54	Dissecting device	Dissection of cystic duct	
08:55	Clipping device	Clipping of cystic duct	
08:56	Laparoscopic scissors	Cutting of cystic duct	
08:57	Dissecting device	Both main vessels have been cut → calling of next patient	
08:58	Dissecting device	Detaching the gallbladder from the liver	
08:59	Clipping device	Clipping of a second artery, which became visible during gallbladder detaching	
09:00	Laparoscopic scissors	Cutting of the second artery	
09:01	Dissecting device	Continuing of gallbladder detaching	
09:02	Suction and irrigation device, coagulation rod	Irrigation of bleedings in the liver, coagulation with HF current	
09:04	Retraction sac	Insertion of gallbladder in retraction sac	
09:05	Alligator forceps	Retraction of plastic sac	On
09:06	No instrument	External gallbladder retraction: removal of gallbladder contents and gallstones	
09:10	Trocar	Re-insertion of trocar and optics, final control of liver bleedings	Off
09:11	Suction and irrigation device, coagulation rod	Irrigation of bleedings in the liver, coagulation with HF current	
09:12	Drainage tube	Insertion of drainage tube	
09:12	Optics	Visual inspection of surgery site	

09:15	Trocar	Suturing of drainage tube	On
09:18	No instrument	Retraction of laparoscope and trocars, abdominal suturing	
09:28	No Instrument	Surgery finished	

Surgery:	$OP2_{test}$		
Date:	Friday, 15th July 2005		
Time	**Instrument**	**Activity**	**Lights**
09:46	No instrument	Patient is rolled into OR, disinfection of the abdomen, attachment of pulse measurement, anaesthesia, inflation of abdomen	On
10:01	Trocar	Insertion of trocars	Off
10:06	Optics	Visual inspection of surgery site	
10:07	Dissecting device	Dissection of cystic duct and cystic artery; the cystic duct is very wide with many adhesions	
10:14	Trocar	Cleaning of optics	
10:16	Dissecting device	Dissection of cystic artery	
10:19	Clipping device	Attempt to clamp the artery	
10:20	Laparoscopic scissors	Cutting of artery	
10:21	Dissecting device	Further dissection of main vessels	
10:22	Clipping device	Clipping of cystic duct	
10:23	Laparoscopic scissors	Cutting of cystic duct	
10:23	Laparoscopic scissors	Both main vessels have been cut → calling of next patient	
10:23	Dissecting device	Detaching of gallbladder with HF current	
10:26	Laparoscopic scissors	Cutting of adhesion	
10:27	Dissecting device	Further detaching of gallbladder	
10:30	Suction and irrigation device, coagulation rod	Gallbladder fully detached → irrigation and coagulation of bleedings in the liver bed	
10:33	Retraction sac	Insertion of gallbladder in retraction sac	
10:35	Alligator forceps	Retraction of plastic sac	On
10:36	Trocar	External retraction of plastic sac: removal of gallbladder contents and gallstones	

10:45	Optics	Visual inspection of surgery site	Off
10:46	Suction and irrigation device, coagulation rod	Irrigation and coagulation of bleedings in the liver bed	
10:48	Alligator forceps	Recovery of a loose metal clip	
10:49	Drainage tube	Insertion of drainage tube	
10:50	Trocar	Retraction of trocars	
10:51	No instrument	Abdominal suturing	On
10:56	No instrument	Surgery finished	

Surgery:	$OP3_{test}$		
Date:	Friday, 15th July 2005		
Time	**Instrument**	**Activity**	**Lights**
11:40	No instrument	Patient is rolled into OR, disinfection of the abdomen, attachment of pulse measurement, anaesthesia, inflation of abdomen, height adjustment of monitors	On
11:46	Trocar	Insertion of trocars	Off
11:50	Optics	Visual inspection of gallbladder beneath the liver	
11:51	Dissecting device	Dissection of cystic main vessels	
11:55	Dissecting device	Insertion of an additional, disposable trocar for better access to the gallbladder	
11:56	Alligator forceps	Preparation of the cystic duct	
11:57	Dissecting device	Dissection of cystic artery	
12:01	Clipping device	Clipping of cystic artery	
12:02	Laparoscopic scissors	Cutting of cystic artery	
12:04	Suction and irrigation device	Gallbladder and cystic duct are too swollen → puncturing of gallbladder and extraction of gallbladder contents	
12:06	Dissecting device	Dissection of cystic duct	
12:11	Suction and irrigation device	Irrigation of bleedings and dissection residues	
12:15	Clipping device	Clipping of cystic duct	
12:17	Laparoscopic scissors	Cutting of cystic duct	
12:18	Dissecting device	Dissection of lower cystic duct	
12:22		Irrigation of bleedings	

	Suction and irrigation device		
12:23	Dissecting device	Dissection of lower cystic duct (branch of cystic duct)	
12:24	Laparoscopic scissors	Cutting of large and inaccessible adhesions	
12:25	Clipping device	Clipping of cystic duct branch	
12:28	Laparoscopic scissors	Cutting of cystic duct branch	
12:29	Dissecting device	Detaching of gallbladder from the liver	
12:31	Optics	Retraction of all instruments due to abdominal cramps of the patient	
12:32	Trocar	Activity break, anaesthetic stabilization of the patient	
12:33	Dissecting device	Continuing to detach the gallbladder from the liver	
12:37	Alligator forceps	Recovery of a loose metal clip	
12:38	Suction and irrigation device	Bile leaking from the gallbladder → suction and irrigation	
12:40	Dissecting device	Continuing to detach the gallbladder from the liver	
12:41	Suction and irrigation device	Irrigation of bleedings and suction of bile	
12:42	Dissecting device	Continuing to detach the gallbladder from the liver	
12:43	Suction and irrigation device	Irrigation of bleedings and suction of bile	
12:44	Dissecting device	Continuing to detach the gallbladder from the liver	
12:46	Suction and irrigation device	Irrigation of bleedings and suction of bile	
12:47	Dissecting device	Continuing to detach the gallbladder from the liver	
12:48	Suction and irrigation device	Irrigation of bleedings and suction of bile	
12:49	Dissecting device	Continuing to detach the gallbladder from the liver	
12:50	Dissecting device	Gallbladder fully detached from the liver	
12:51	Retraction sac	Insertion of gallbladder into retraction sac	
12:52	Alligator forceps	Retraction of plastic sac	On
12:53	Trocar	External retraction of gallbladder	
12:54	Optics	Visual inspection of surgery site	Off
12:55	Alligator forceps	Recovery of a loose metal clip	

12:56	Suction and irrigation device	Irrigation of bleedings and vessel stumps	
13:00	Dissecting device	Coagulation of bleedings in the liver bed with HF current	
13:02	Suction and irrigation device	Irrigation of liver bed	
13:03	Drainage tube	Insertion of drainage tube	
13:04	Drainage tube	Retraction of drainage tube due to more bleedings	
13:05	Dissecting device	Coagulation of bleedings in the liver bed with HF current	
13:07	Alligator forceps	Recovery of a loose metal clip	
13:08	Optics	Inspection of surgery site	
13:09	Drainage tube	Insertion of drainage tube	
13:10	Optics	Inspection of surgery site	
13:11	Trocar	Retraction of trocars	
13:12	No instrument	Abdominal suturing	On
13:19	No instrument	Surgery finished	