# Online Temporal Synchronization of Pose and Endoscopic Video Streams

Özgür Güler[a], Ziv Yaniv[b], Wolfgang Freysinger[a]
[a]University ENT Clinic, Innsbruck Medical University, Innsbruck, Austria
[b]Imaging Science and Information Systems (ISIS) Center, Dept. of Radiology, Georgetown University Medical Center, Washington, DC, USA

## ABSTRACT

Computer assisted navigation systems that combine real-time endoscopy images with pre-operative volumetric data sets aim at improving the physician's understanding of the underlying anatomical structures. To achieve accurate and safe guidance these systems are required to provide a consistent representation of the physical world. This implies that all data streams are synchronized. In our case, we are dealing with synchronization of tracking data and a video stream obtained by a tracked endoscope. Previously, such synchronization was obtained pre-operatively using phantoms. This type of approach assumes a constant latency between the data streams and is less desirable for clinical use due to the required additional hardware. In this work we describe an online temporal synchronization method. The method is based on the observation that in clinical practice the endoscope is not in constant motion. By identifying corresponding stationary points in the video and tracking streams temporal synchronization can be performed online in a manner that is transparent to the user. Initial evaluation of our approach in a laboratory study has shown that it provides comparable estimates to a phantom based approach we had previously proposed.

**Keywords:** Image-guided therapy, temporal calibration, synchronization

## 1. INTRODUCTION

Image guided surgery (IGS) has significantly improved patient safety during surgical procedures treating the paranasal sinuses and the frontal skull base. Currently, IGS systems are in routine use for such procedures at our hospital [1]. Intra-operatively the imaging modality used in these procedures is endoscopic video, primarily used for functional endoscopic sinus surgery (FESS) in ORL-surgery [2]. More recently, we have mounted a laser onto our 3D-navigated endoscope, providing a non-contact image-processing-based position-sensing [3]. Navigated endoscopy has also been applied to examining the inside of cavities of hollow organs of the body from many different angles using the surface topography after reconstruction from a sequence of monocular endoscopic video [4;5].

Endoscopic video provides a real-time two dimensional representation of the underlying three dimensional anatomy. Computer assisted navigation systems improve the physician's understanding of the underlying anatomical structures by augmenting these images with data derived from three dimensional pre-operative images such as CT and MR. Using this additional information we have successfully implemented "augmented reality" (AR) in routine clinical practice by superimposing positional data and ancillary structures on the live endoscopic video of the operating site. Thus, optimal access paths and anatomical structures such as the arteria carotis interna or the nervus opticus can be displayed [6;7]. Figure 1 is an example of overlaying the pre-operative planned access path on intraoperative video.

A key step in image guided surgery is registration, aligning the pre-operative image space with the patient's physical space. The most common registration approach in clinical use is based on paired-point matching [8-10], and is often complemented by surface registration [11;12]. In daily clinical routine surface registration approaches are preferred due to their intraoperative ease of use [13]. While paired-point based registration is still the most common registration method it suffers from several drawbacks. Its accuracy is dependent on the point features spatial configuration, and on their type. The three most commonly used point features are titanium screw fiducials, skin adhesive fiducials and

Medical Imaging 2011: Visualization, Image-Guided Procedures, and Modeling,
edited by Kenneth H. Wong, David R. Holmes III, Proc. of SPIE Vol. 7964, 79641M
© 2011 SPIE · CCC code: 1605-7422/11/$18 · doi: 10.1117/12.877720

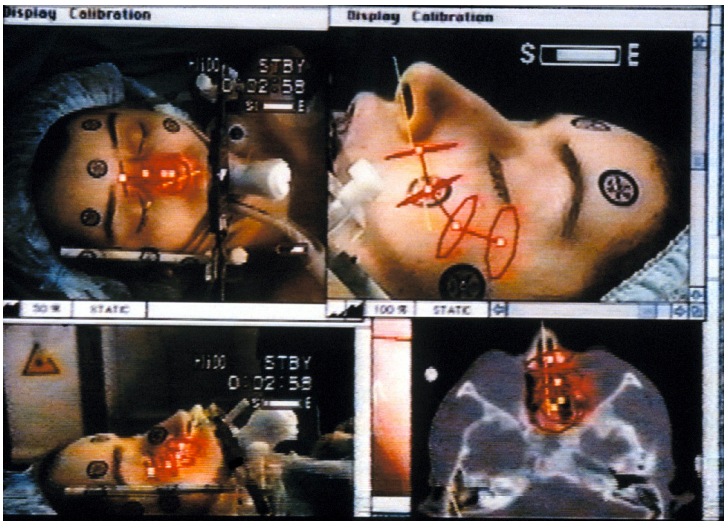



Figure 1.Overlay of preoperative planned access path (bottom right) on intraoperative video.

anatomical landmarks. Each of these point features has its advantages and disadvantages. Titanium screws are accurately localized and do not shift but they are intrusive. Anatomical landmarks on the other hand are not intrusive, but it is often hard to accurately localize them. Skin adhesive fiducials provide slightly better localization than anatomical landmarks yet are less intrusive than the screws. Unfortunately, they sometimes shift or fall off during procedures. Apart from the initial registration, in some cases there is a need for re-registration due to movement of the dynamic reference frame attached to the patient. In these cases it is often difficult to localize the point features on the patient as they can be covered due to sterilization reasons or when fiducials are used they sometimes shift or may even fall off.

Aiming to overcome the problems encountered with paired-point methods, we have recently investigated the use of A-mode ultrasound for intraoperative surface point acquisition [14;15]. The acquired surface is then registered to a pre-operatively segmented surface. Another potential approach to surface acquisition is to use the images acquired by endoscopic video [16]. This approach only utilizes the endoscopic images for surface acquisition and does not take advantage of the tracking system which is part of the IGS system. We are currently investigating the use of tracked endoscopic video as a means for intraoperative surface acquisition.
As a first step we perform a spatial calibration of our endoscopic camera relative to a dynamic reference frame rigidly attached to the endoscope. We then use both the tracking data and the images to acquire the surface intraoperatively.

This approach is valid if the latency between the acquisition of tracking information and intra-operative imaging is negligible, or if the imaging apparatus is stationary. In many procedures this is not the case, intra-operative imaging is performed in a dynamic manner, and the latency between the two data streams, video and tracking, is discernable. This is primarily due to the difference in the size of the transferred data. With the increased use of high definition endoscopic systems we expect to see larger temporal differences between the two data streams, making synchronization even more critical for accurate 3D reconstruction from tracked endoscopic images.

As a consequence, temporal synchronization is required. Offline pre-operative temporal calibration of video streams and tracking data has been previously presented in [17-19], synchronizing tracking data and ultrasound data. An underlying assumption of all these methods is that the latency between the two data streams is constant throughout the procedure. In addition they require the use of calibration phantoms which is less desirable in a clinical setting.

We have developed an online temporal calibration method that repeatedly estimates the latency time throughout the procedure in a manner that is transparent to the user. The method is based on the observation that in a clinical setting the endoscope is not in constant motion, rather there are short periods where it is stationary. It is thus possible to synchronize the two data streams by identifying these stationary points.

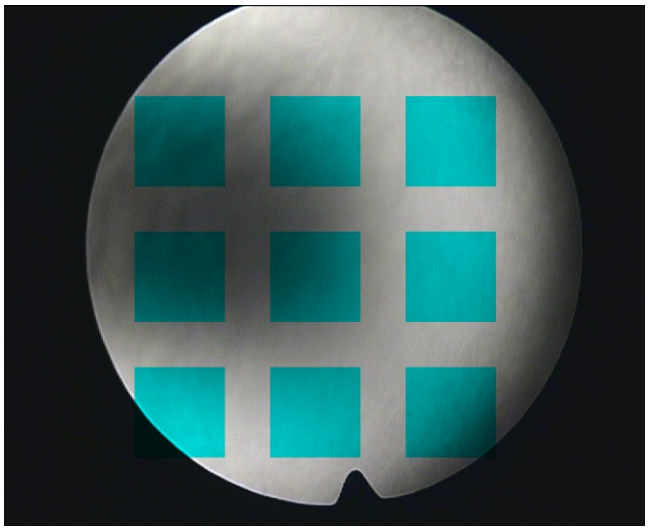


Figure 2. Subset of pixels used for image comparison overlaid onto endoscopic image.

This paper describes our online method for temporal synchronization between two data streams, poses obtained by a tracking system and video obtained by a tracked endoscope. By synchronizing these two data streams we ensure that the information presented to the physician consistently reflects the underlying physical situation. We also show that 3D-reconstruction from endoscopic video yields better results with temporal calibration.

## 2. MATERIALS AND METHODS

To facilitate temporal synchronization we require access to images and tracking data from multiple time points. To this end, we have implemented our algorithm using a modified version of the Image-Guided Surgery Toolkit [20]. The toolkit supports image acquisition with a video component that includes a ring buffer holding images from multiple time points. Using a similar approach we have added a ring buffer to the toolkit's tracking component to provide us with access to tracking data over time.

In all experiments we used our open source intraoperative navigation system open4Dnav [21] based on the modified IGSTK library. We currently use Open4Dnav under the Linux operating system, but it is designed as a platform independent application. An additional module for video and tracking stream data analysis was implemented in open4Dnav. The software was run on an Intel Core® 2 Duo 2,33 GHz, 4 GB RAM. Tracking data was acquired with an optical tracking system, Polaris from Northern Digital Inc. (Waterloo, Ontario, Canada). The endoscope we used is model 5520 from Richard Wolf GmbH (Knittlingen, Germany). Images were acquired from the endoscope over firewire.

### 2.1 Synchronization Method

We have observed that in clinical practice an endoscope is not in constant motion, rather, there are short periods where it is stationary. To synchronize the image and tracking we need only detect corresponding stationary points in both data streams.

Intra-operatively a background process constantly checks if the tracking data corresponds to a stationary point. This is done by calculating the speed of the tracked reference frame attached to the endoscope, $v = \Delta s/\Delta t$. Where $\Delta s$ is the difference between the current and previous positions and $\Delta t$ is the difference between the corresponding time stamps. It should be noted that in IGSTK all transformations are associated with a time stamp. To ensure that the endoscope is indeed stationary we compare the maximal speed observed for the previous n=6 transformations, $v_{filtered,}$ to an empirically obtained threshold. If $v_{filtered}$ is less than our threshold we have a stationary point and store its time stamp, $t_{trackerStationary}$.

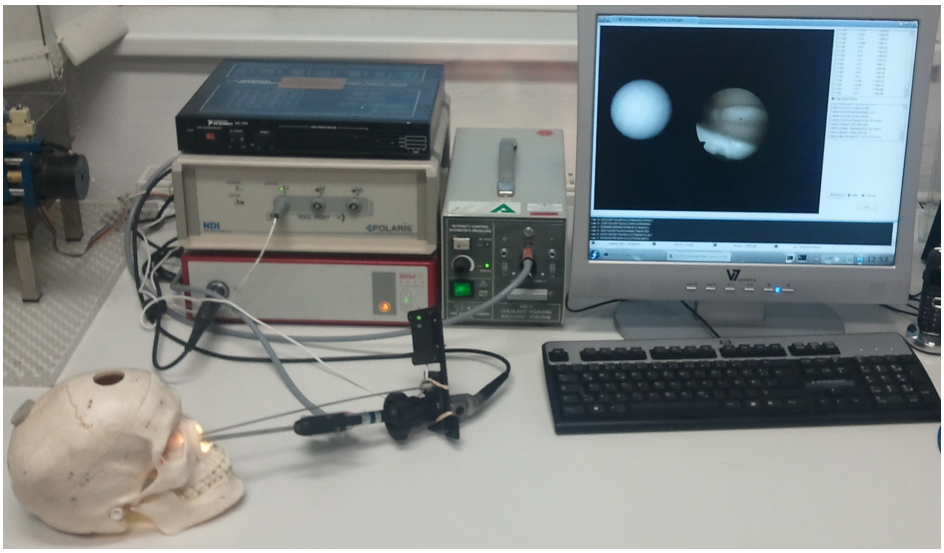


Figure 3. Setup used to evaluate actual refresh rates as compared to requested ones. The endoscope was moved inside the nasal cavities of a plastic skull for approximately 30s. Application shows the live video and position of the endoscope visualized as a white sphere.

Independently another process compares each newly acquired image to the previous image. While the images are acquired in RGB color space we only use the image intensities for our comparison. In our case we compute the Y channel from the YUV color space representation of the image. For computational efficiency we use the sum of absolute differences (SAD) operator as our similarity measure. In addition we use a subset of the image pixels, as shown in Figure 2. Again, an empirical threshold value is used to identify the beginning of a stationary image sequence. To ensure that the endoscope is indeed stationary we compare the maximal SAD value observed for the previous n=6 frames, $SAD_{filtered,}$ to an empirically obtained threshold. If $SAD_{filtered}$ is less than our threshold we have a stationary point and we store its time stamp $t_{videoStationary}$.

We obtained our temporal and intensity thresholds by acquiring tracking data while the endoscope was in a stationary position for approximatly 30 seconds, yielding 972 transformations and 423 video frames. After calculating the mean, $\mu$, and standard deviation, $\sigma$, of the speed and SAD values from the transformations and images, the thresholds were set to $\mu+3\sigma$. Thus we can ensure that we take into account the measurement noise when detecting a stationary point.

Once we detect a stationary point in the tracking and video streams we compute the lag as, $t_{timelag} = t_{videoStationary} - t_{trackerStationary}$. Note that we assume that the video stream is always lagging behind the tracking data.

### 2.2 Experiments

We evaluated four aspects relating to the temporal performance of our image guided navigation system. The first being the actual refresh rates of our system as compared to the requested ones. The second being the latency between actions in the physical world and the visual feedback given on screen, the third being the effect of temporal synchronization between tracking and video on 3D stereo reconstruction, and the fourth being the long term latency behavior.

In IGSTK based applications using video and tracking streams there exist three separate threads. One is responsible for the main application the other two are running continuously updating the tracked rigid bodies positions and the video frames from the endoscope. The main application, the tracker component, and the video component are controlled by a pulse-generator, which is responsible for updating the rendered scenes. The pulse-generator triggers an update to the view, the tracker and video components according to the frequencies set by the user. Our first experiment compares the actual frequencies obtained by our program with those specified to the components. The 3D view rendering frequency, tracker frequency and video frequency were set to 25Hz, 35Hz, and 25Hz, respectively. We ran our application for 30 seconds and recorded the time for all video frames, transformations and scene updates. The difference between consecutive timestamps were calculated and averaged. The mean value of these differences represents the time needed to get the next scene update, transformation, or video frame. Figure 3 shows our experimental setup.

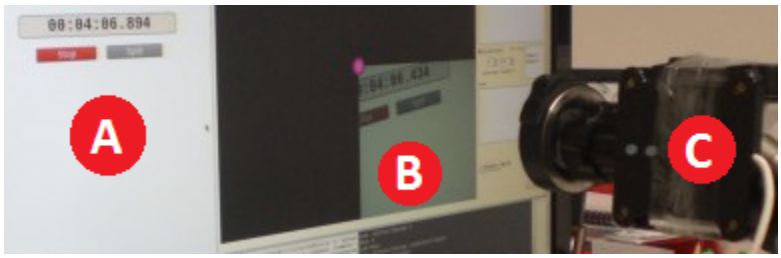



Figure 4. Setup for measuring the delay between capture time and render time. (A) a stopwatch with millisecond precision. The tracked endoscope left (C) captures the stopwatch and the application renders the stopwatch on the same screen (B). Current time on this image is 04:06:894, whereas the rendered scene shows the stop watch in the past witch 04:06:434.

We next evaluated the latency between the physical world and our application's display. For this purpose a clock with millisecond precision was shown on the screen and imaged by the endoscope. Figure 4 shows the setup for measuring the delay between capture time and render time. We readily obtain the latency associated with our scene rendering as both the captured image and the clock are displayed on the same screen. We obtained 20 screenshots and calculated the difference of the two times visible in each screenshot. We averaged these difference values to get the mean latency of rendering the current real world scene.

We then evaluated the effect of synchronization using our approach on 3D stereo reconstruction. We first performed camera calibration[22]. We then acquired two views of a cube with checkerboard patterns on three of its faces with our tracked endoscope. Tracking data was acquired throughout the endoscopes motion. Using the corresponding points in the two images, the known intrinsic camera parameters, and the relative position between the two camera poses obtained from tracking data we performed metric stereo reconstruction. The 3D reconstruction was then aligned to the known cube model using Arun's paired-point rigid registration method. The mean distance between the pairs of points was then calculated. We then compared the stereo reconstruction with and without temporal synchronization. Figure 5. shows our experimental setup.

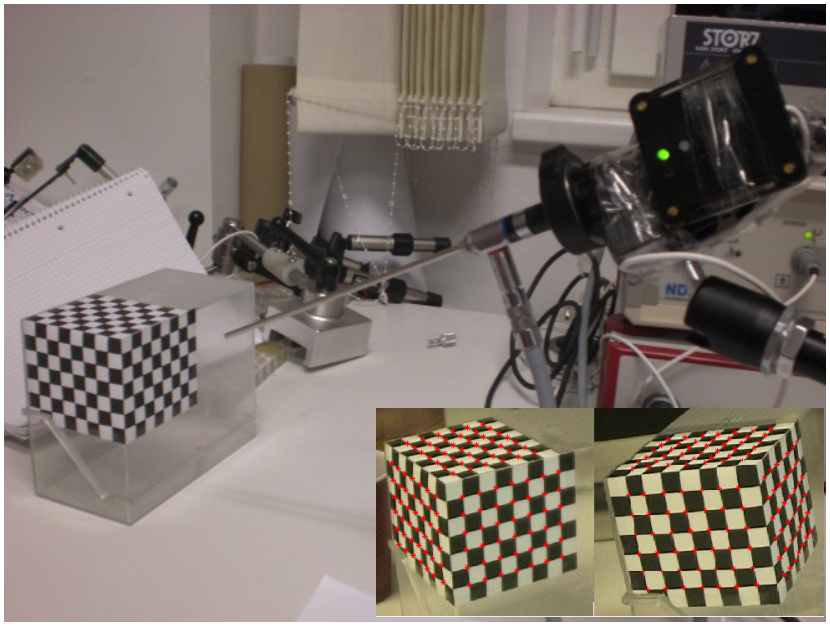


Figure 5. Experimental setup for evaluating the effect of temporal synchronization on stereo reconstruction. Inset shows the detected corners of the checkerboard pattern used for reconstruction.

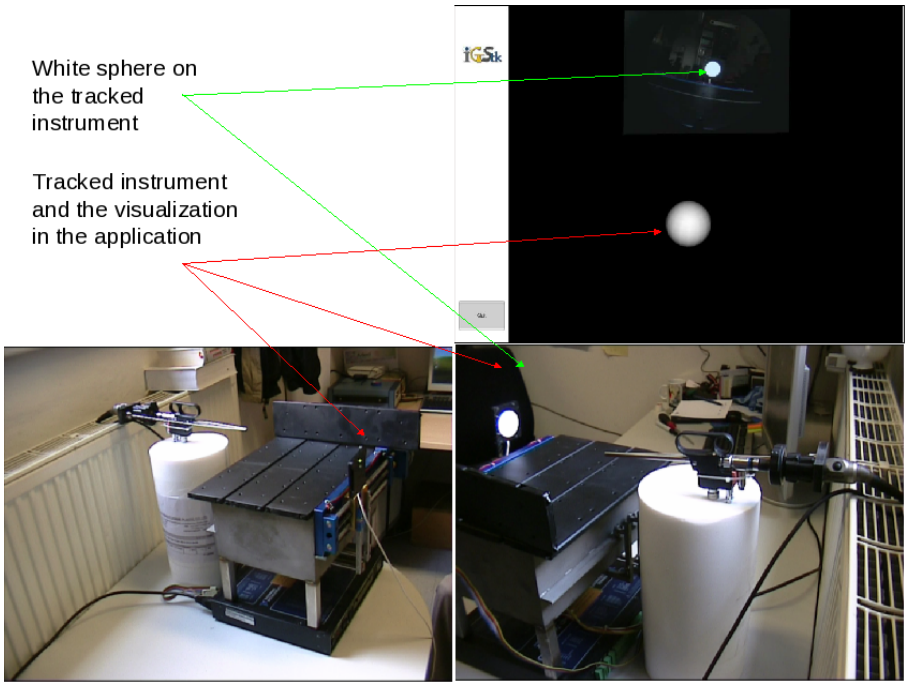



Figure 6. Setup of long term latency assessment

Finally, we evaluated the latency behavior over an extended period of time. Figure 6 shows the experimental setup for the long term experiment. We place a planar circular fiducial marker in front of the camera so that its motion is parallel to the image plane. An IGSTK application shows in one window the video of the white sphere attached to the tracked instrument and the white sphere representing the tracked instrument in space. As the linear stage moves back and forth the white sphere in the video and the white sphere representing the 3D position of the instrument move in the application window. During this motion we take screenshots at a frequency of 10 Hz. Segmenting the centroids of the two white circles provide us with two dimensional position information of the movement following the back and forth motion. As most of the motion is along the linear trajectory we extract a one dimensional (1D) signal from each of the data sets via Principle Component Analysis. Our 1D signal is the projection of each of the point sets onto their principle axis. The data was collected for 2 hours.

## 3. EXPERIMENTAL RESULTS

The actual refresh rates for tracking data acquisition, video acquisition and data rendering were all found to be lower than the user requested ones. For tracking the requested rate was 35Hz and the actual rate was 32.4Hz. For the image capture the requested rate was 25Hz and the actual one was 14.1Hz, and for the rendering the requested rate was 25Hz and the actual one was 15.68Hz. It appears that the video transfer is the bottleneck that limits the system performance.

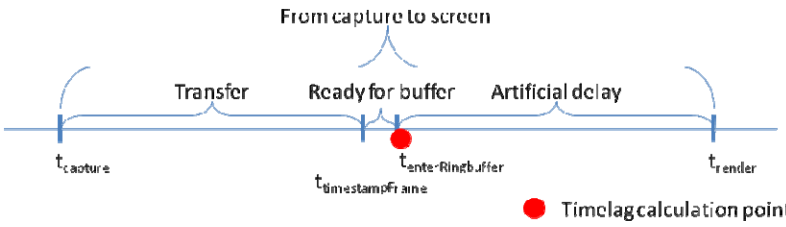



Figure 7. Real world scene captured at time $t_{capture}$ can be rendered at time $t_{render}$. In order to minimize the effect of software implementation, the temporal calibration is done right after transfer time and ring buffer storage.

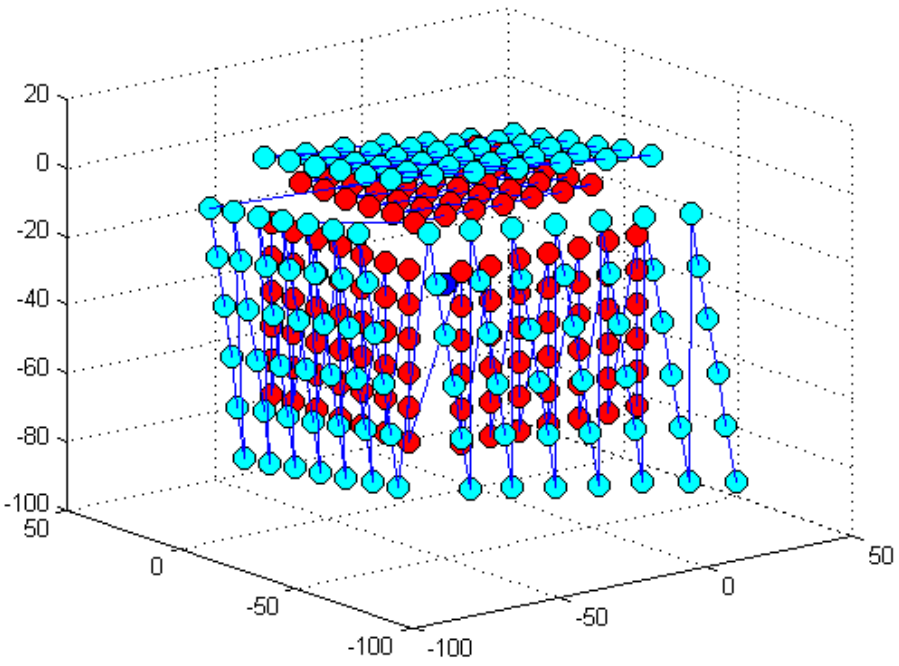
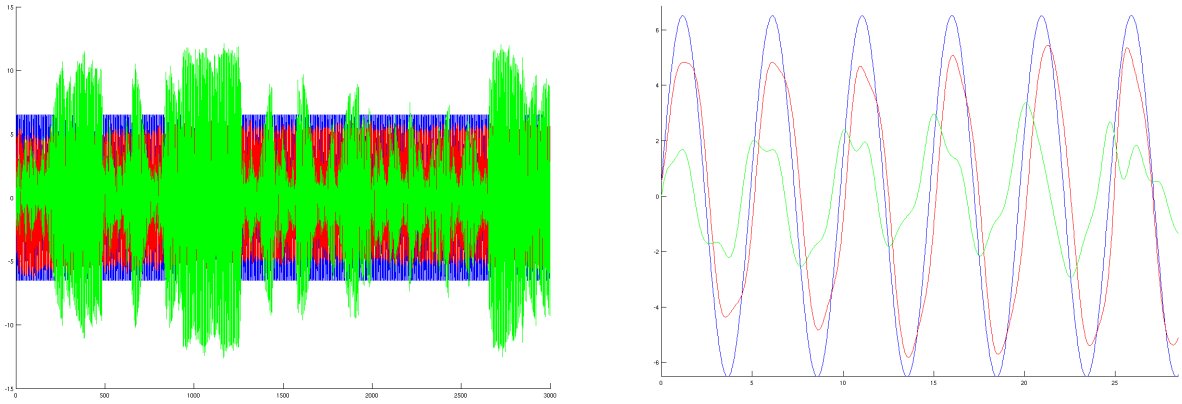



Figure 8. Registered known cube geometry (red dots) with reconstructed cube (cyan dots)

The time delay between capturing a real world scene and rendering it on the display for our setup was 401 +/- 53 ms ($\mu$ +/- $\sigma$). Figure 7 shows the timeline of a video-frame and intermediate stages of video processing. From capture to screen: $t_{captureToRender} = t_{render} - t_{capture} = 401$ ms, Ready for buffer: $t_{readyForBuffer} = t_{enterRingbuffer} - t_{timestampFrame} = 10$ ms, Artificial delay, The renderer gets the third frame in the past from the ringbuffer in order to prevent a conflict between writer and reader thread. According to our previous experiment every ~65.16 ms (15.68Hz) we get a new frame. Setting the delay on the ring buffer to three results in $t_{artificialDelay} = 65.16 \text{ ms} * 3 = 195.48$ ms. Transfer includes frame capturing, transfer over firewire cable, video grabber processing, OS kernel driver proceccing, decoding with OS DV library and IGSTK-video component processing. The transfer time can be calculated with
$t_{transfer} = t_{captureToRender} - t_{artificialDelay} - t_{readyForBuffer} = 195.52$ ms.

To evaluate the effect of temporal synchronization on 3D stereo reconstruction we first established a ground truth. We acquired two images with a stationary endoscope. In this manner the temporal latency had no effect on the reconstruction. The mean error for this reconstruction was 9.44mm. We then performed reconstructions using pairs of images acquired by the tracked endoscope with and without applying our temporal synchronization. Without the synchronization the mean reconstruction error was 15.66mm. With the temporal synchronization it was 14.65mm. Figure 8 shows the reconstruction obtained with the temporally synchronized data.

To evaluate the latency behavior over time we set the linear stage to run in a sinusoidal speed at a fixed frequency for two hours. The result is shown in Figure 9.



(a) (b)

Figure 9. (a) Sinusoidal motion of the linear stage over 2 hours and (b) zoomed view. Blue curve is the input data to the stage, red curve is the extracted actual motion of the white circle on the DRF and the green is the difference signal between the two.

Figure 10 shows the autocorrelation plot of the difference signal, calculated using video trajectory and tracker trajectory. The exponential decay of the autocorrelation function indicates a band-limited white noise. Moreover, as can be seen from Fig 9, the measured autocorrelation function of our setup is the superposition of a sinusoid and a first-order-Markov component in the autocorrelation function. As the DRF is being moved from left to right periodically (ideally with a saw-tooth function, in reality with a sinsusoidal movement) this corresponds to the high-frequency oscillatory pattern.

## 4. DISCUSSION AND CONCLUSION

We have presented a new online method for estimating the lag between the tracking and video streams used in image guided navigation. This type of approach is more appropriate for clinical use as compared to currently used phantom based calibration methods. Existing temporal calibration approaches suffer from two major drawbacks. First the interaction between video imaging device, tracking system, and PC is arbitrary, due to different priorities and scheduling times i.e. processor load of operating system services. Therefore a constant latency time calculated preoperatively may not correspond to the intraoperative situation. Another drawback to existing approaches is the use of temporal calibration phantoms. The use of phantoms introduces additional steps in the surgical intervention which are best avoided. The

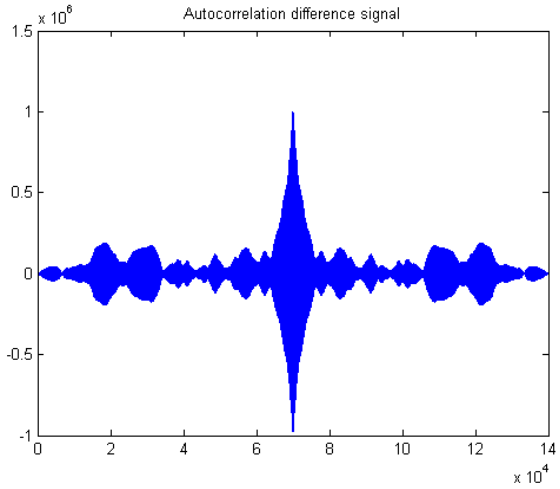



Figure 10. Autocorrelation signal from the difference signal calculated using video trajectory and tracker trajectory.

online temporal calibration tool is part of open4Dnav and runs in the background. In this manner the surgical intervention is not modified by additional calibration steps.

A difference between actual and set frequency was expected. Running tracker, video acquisition and 3D scene in parallel decreased the frequency but it does not effect the latency determination.

The detection of stationary positions using the speed of the tracked endoscope and the SAD value for the endoscope video is reasonable. The success of a stable latency estimation between tracker and video depends on hardware and operating system specific behavior which introduces white noise and makes latency calculation imprecise.

The need to synchronize video and transformation data is due to the differences between the devices, the data formats, and the amount of data transferred, making a software implementation a difficult task on a non real time operating system. We thus conclude that hardware controlled systems or real-time operating systems are more appropriate for minimizing the latency between video and tracking information.

While the online temporal calibration is slightly less stable than our phantom based approach [19], it is less intrusive and more appropriate for clinical use and it does improve the reconstruction accuracy which is the goal of this work.

**Acknowledgement:** This project is funded by the Austrian Science Foundation under contract 20604-B13.

## REFERENCES

[1] F.Kral, M.Markovicz, G.Göbel, A.R.Gunkel, A.W.Scholz, C.Pototschnig, C.Völklein, W.Thumfart, E.Appenroth, M.Schindler, H.Fischer, and W.Freysinger, "Retrospective analysis of navigated FESS interventions since 1995," *Laryngoscope*, vol. in preparation 2008.

[2] W. Freysinger, A. R. Gunkel, and W. F. Thumfart, "Image-guided endoscopic ENT surgery," *Eur. Arch. Otorhinolaryngol.*, vol. 254, no. 7, pp. 343-346, 1997.

[3] F. Kral, Ö. Güler, M. Bickel, J. Puschban, and W. Freysinger, "Live endoscopic video streams augmented for 3D-navigation," in *World Congress on Medical Physics and Biomedical Engineering, September 7 - 12, 2009, Munich, Germany,* 25/6 ed. O. DÃ¶ssel and W. C. Schlegel, Eds. Springer Berlin Heidelberg, 2009, pp. 293-295.

[4] Y. I. Abdel-Aziz and H. M. Karara, "Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry," in *ASP Symposium on Close-Range-Photogrammetry.* American Society of Photogrammetry, Ed. Univ. of Illinois at Urbana-Champain: American Society of Photogrammetry, 1971, pp. 1-18.

[5] M. Truppe, F. Pongracz, O. Ploder, A. Wagner, and R. Ewers, "Interventional Video Tomography.," *Proc. SPIE*, vol. 2395, pp. 150-152, 1995.

[6] W. Freysinger, M. J. Truppe, A. R. Gunkel, and W. F. Thumfart, "Stereotactic Telepresence in ENT surgery," *HNO*, vol. 50, no. 5, pp. 424-432, May2002.

[7] W. Freysinger, M. J. Truppe, A. R. Gunkel, W. F. Thumfart, F. Pongracz, and J. Maierbaeuerl, "Interactive Telepresence and Augmented Reality in ENT Surgery: Interventional Video Tomography.," *LNCS*, vol. 1205, pp. 817-820, 1997.

[8] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-Squares Fitting of Two 3-D Point Sets.," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 9, pp. 698-700, 1987.

[9] C. R. J. Maurer and J. M. Fitzpatrick, "A review of medical image registration.," in *Interactive Image-Guided Neurosurgery.* R. J. Maciunas, Ed. Park Ridge, Ill., USA: AANS, 1993, pp. 17-44.

[10] J. B. A. Maintz and M. A. Viergever, "A survey of medical image registration," *Medical Image Analysis*, vol. 2, no. 1, pp. 1-36, 1998.

[11] P. J. Besl and N. D. Mckay, "A Method for Registration of 3-D Shapes," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239-256, 1992.

[12] Zhengyou Zhang, "Iterative point matching for registrationof free-form curves and surfaces.," *Int. J. Computer Vision*, vol. 13, no. 2, pp. 119-152, 1994.

[13] R. R. Shamir, L. Joskowicz, S. Spektor, and Y. Shoshan, "Localization and registration accuracy in image guided neurosurgery: a clinical study.," *Int. J. CARS*, vol. 5, pp. 45-52, 2009.

[14] G. M. Diakov and W. Freysinger, "Accuracy evaluation of initialization-free registration for intraoperative 3D-navigation.," *Int. J. Computer Assisted Radiology and Surgery*, vol. in print 2007.

[15] G. Diakov, F. Kral, O. Guler, and W. Freysinger, "[Automatic registration of patients with A-mode ultrasound for computer-assisted surgery. Laboratory proof of concept]," *HNO*, vol. 58, no. 11, pp. 1067-1073, Nov.2010.

[16] D. Burschka, M. Li, M. Ishii, R. H. Taylor, and G. D. Hager, "Scale-invariant registration of monocular endoscopic images to CT-scans for sinus surgery," *Med. Image Anal.*, vol. 9, no. 5, pp. 413-426, Oct.2005.

[17] F. Rousseau, P. Hellier, and C. Barillot, "A novel temporal calibration method for 3-D ultrasound," *IEEE Trans. Med. Imaging*, vol. 25, no. 8, pp. 1108-1112, Aug.2006.

[18] G. M. Treece, A. H. Gee, R. W. Prager, C. J. Cash, and L. H. Berman, "High-definition freehand 3-D ultrasound," *Ultrasound Med. Biol.*, vol. 29, no. 4, pp. 529-546, Apr.2003.

[19] Ö. Güler, Z. Yaniv, Freysinger W., and K. Cleary, "Temporal calibration of tracking and image acquisition," *Int. J. CARS*, vol. accepted 2009.

[20] K. Cleary, P. Cheng, A. Enquobahrie, Z. Yaniv eds., "IGSTK: The Book", Signature Book Printing, 2009.

[21] Ö. Güler, "Concept of an open-source 3D-navigation system." Bakk Innsbruck University, 2004.

[22] J-Y Bouguet , Camera Calibration Toolbox for MATLAB, [http://www.vision.caltech.edu/bouguetj/calib_doc/index.html], accessed Jan. 1 2011.