Temporal Calibration in Multisensor Tracking Setups

Manuel Huber   Michael Schlegel   Gudrun Klinker*

Technische Universität München
Fakultät für Informatik
Boltzmannstraße 3
85748 Garching b. München, Germany

Abstract

Spatial tracking is one of the most challenging parts of Augmented Reality. Many AR applications rely on the fusion of several tracking systems in order to optimize the overall performance. While the topic of sensor fusion has already seen considerable interest, most results only deal with the integration of particular setups.

A crucial part of sensor fusion is the temporal alignment of the sensor signals, as sensors in general are not synchronized. We present a general method to calibrate the temporal offset between different sensors by applying the normalized cross correlation method.

Keywords: sensor fusion, calibration, tracking, ubiquitous tracking, synchronization

1 Introduction

In order to correctly combine data from two tracking sensors it is necessary to know the exact temporal relationship between data acquired from the different sources. We call such sensors to be synchronized or temporally calibrated.

Synchronization can either be achieved by hardware means or by logical means on the sensor data. For hardware synchronization the acquisition of sensor data is triggered by a central clock, connected to all participating sensors. Logical synchronization depends on correctly attaching timestamps to each sensor measurement. The fusion algorithm then either has to correctly infer measurements such that only observations regarding the same point in time are fused or accommodated for sequential measurements. A general concept on how to achieve a correctly operating system is described in [7] and involves a push/pull dataflow architecture.

The temporal calibration problem so far is mostly only solved for particular hardware setups. Mostly components are either hardware synchronized or the lag between different sensors is tuned in software by experimental means. Drawbacks of these approaches are that common off-the-shelf hardware often lacks hardware synchronization interfaces and that dynamic sensor fusion as proposed in [5] requires methods for automatic adjustments.

In this paper we present a general method to calibrate the temporal offset between different sensors.

The data from two rigidly connected sensors measuring corresponding spatial relationships are compared by computing a similarity measure which determines the level of agreement between the two sensors, which is to be maximized (for an example see figure 1(a)). We use normalized cross-correlation as the similarity measure which has to be preceded by a suitable projection to transform the individual multidimensional sensor data into one dimensional signals.

*e-mail: {huberma, schlegem, klinker }@in.tum.de

IEEE International Symposium on Mixed and Augmented Reality 2009
Science and Technology Proceedings
19 -22 October, Orlando, Florida, USA
978-1-4244-5419-8/09/$25.00 ©2009 IEEE

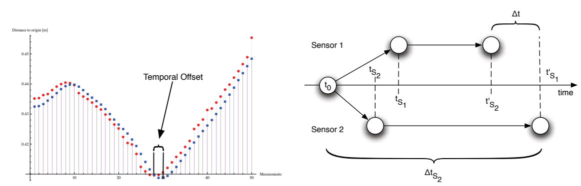


Figure 1: (a) Data of two different senors; (b)Schematic visualization of different points in time

2 Related Work

One of the major remaining questions in dynamic reconfigurable tracking setups (such as [5]) is how to dynamically account for sensor synchronization. The Ubitrack framework so far accounts for unsynchronized sensors by utilizing a Push/Pull dataflow architecture [7] which depends on the correctness of timestamps associated with sensor measurements. The negative influence of lag on the general usability of AR applications is generally agreed upon (see for example [3] or [8]).

Also in [2] and [1] a sensor synchronization scheme is discussed for the application of calibrating inertial sensors and vision based tracking. Their approach relies on detecting abrupt movements in both the camera image as well as the inertial tracker. In [4] it is indicated that the employed camera and inertial tracker are synchronized via a common clock source that triggers both sensors. Such a setup using hardware synchronization currently seems to be the most common case, but in general is prohibitive in ubiquitous tracking scenarios.

3 Approach

Consider an event happening in the real world at time t_0, two sensors S_1, S_2 sense this event and register it at times t_{S_1} and t_{S_2}. These are in general different observation times of the same event because every type of sensor needs a different amount of time for the internal signal processing. Some more delay will be caused by the operating system or by sending the data over a network to a second workstation. Assuming the measurements arrive in the tracking software at times t'_{S_1}, t'_{S_2}. All these sources of time delay can be reduced to one single delay for each sensor $\Delta t_{S_1} = t'_{S_1} - t_0$ and $\Delta t_{S_2} = t'_{S_1} - t_0$. For the sensor fusion it is only necessary that all sensors are temporally aligned relative to each other, so the offset to the unknown true point in time t_0 is not relevant. To align the sensor data it is sufficient to determine the temporal offset $\Delta t = t'_{S_1} - t'_{S_2}$ (see figure 1(b)).

The basic idea to find the temporal offset, is to shift one sensor data successively by discrete time offsets and compare both streams with a similarity measure. We investigated the cross correlation coefficient as a similarity measure, which is quite common in the field of signal processing. Let T_{S_1}, T_{S_2} be the sets of all timestamps $t \in T_{S_1} \cup T_{S_2}$ where measurements of either S_1 or S_2 respectively were taken. We define the two data time series as $X = \{x_t : t \in T_{S_1}\}$

and $Y = \{y_t : t \in T_{S_2}\}$ as the actual sensor data x_t, y_t from sensors S_1 and S_2 respectively at the individual timestamps. Note that the timestamps at which S_1 and S_2 acquire data, do in general not agree and thus T_{S_1} and T_{S_2} are distinct.

We can thus define Pearsons's correlation coefficient of these signals as

$$\rho_{X,Y} := \frac{Cov(X,Y)}{\sigma_X \sigma_Y}.$$

Note that the correlation is only defined for scalar valued time series. In order to be applicable to high dimensional tracking data a suitable dimensionality reduction has to be performed. For our experiments, we chose to implement a straightforward projection onto an arbitrary, but constant axis.

If the correlation coefficient $\rho_{X,Y}$ is equal to 1, both signals are identical. In general $\rho_{X,Y} \neq 1$ even if the same type of sensors is used because both measurements will be affected by noise and other kinds of errors. The time-offset of the two sensor signals can be determined by consecutively shifting one signal by small offsets against the other signal and calculating the correlation coefficient for each time-shift.

The task is to find a Δt which maximizes the similarity of both signals. This leads to the following formula

$$\Delta t = \text{argmax}_{\delta t}\{\rho_{X,Y^{(\delta t)}}\},$$

where $Y^{(\delta t)}$ is the signal Y shifted by δt.

4 Evaluation

To validate the method described above we conducted a series of experiments involving different combinations of sensors. Special attention was paid to demonstrate to cover most of the interesting sensor types used for spatial tracking. As such we are mostly concerned with 6DoF pose sensors and subsets thereof such as 3DoF position or 3DoF rotation sensors.

Calibration Results For our evaluation we used different types of tracking devices. We used an *A.R.T.* optical tracking system, an *Xsens MTx* inertial sensor, a *Faro Fusion* coordinate measurement machine (CMM) as well as a *optical square marker tracker* using an off-the-shelf camera.

Table 1 shows the summary of several temporal calibration results obtained for different sensor combinations.

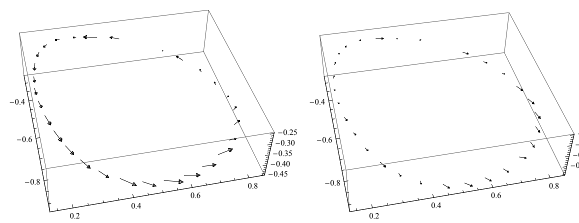
	Position (3DoF)		Rotation (3DoF)	
Combination of Sensors	reg.	unreg.	reg.	unreg.
FARO, A.R.T.	13ms	12ms	—	—
FARO, Xsens	—	—	21ms	19ms
A.R.T., MarkerTracker	54ms	47ms	52ms	50ms

Table 1: Measured temporal offsets

Error reduction To illustrate the effectiveness of the temporal alignment, we analyzed the resultant spatial registration error between two different trackers in both the unsynchronized and the synchronized case.

For this data set the tip of the Faro CMM was moved in a simple circle with moderate speed. The 3DoF position of the tip was recorded both by the Faro system and by the A.R.T. system, which was additionally transformed into the Faro coordinate frame. Figure 2(a) shows the error vector between measurements from the A.R.T. system and corresponding points measured by the Faro system during the movement. The root mean square (RMS) error in this case is 13mm. From the direction of the vectors the movement of the marker ball is clearly visible, which indicates a distinctive lag between the two sensor systems.

Figure 2(b) shows the same error vectors with the difference that the timestamps were corrected according to the calibration value determined. In this plot the direction of the error vectors no longer corresponds to the direction of the movement and the RMS has been reduced to 6mm. The remaining errors mostly stem from calibration errors and sensor noise.



(a) (b)

Figure 2: Error vector between measurements from A.R.T. and the Faro system during movement; (a) without temporal alignment; (b) with temporal alignment

5 Future Work

Integration as an online recalibration tool requires further thought on the classification of pathological input cases. While the offset calibration method as described above produces proper results for general input cases, there are also pathological inputs (such as no motion at all or fast, perfectly periodic movements) which can lead to meaningless calculations. An online method thus would have to decide whether the current input is sufficient (e.g. exhibits enough entropy) to allow for reliable calibration.

There are further possibilities to apply cross correlation to multidimensional data, such as *canonical correlation analysis* [6]. However, first investigations did not yield any major improvements over the methods described above. Nevertheless a more thorough comparison could be performed.

References

[1] M. Aron, G. Simon, and M. Berger. Use of inertial sensors to support video tracking. *Computer Animation and Virtual Worlds*, 18(1):57–68, 2007.

[2] M. Aron, G. Simon, M. Berger, I. LORIA, and F. Nancy. Handling uncertain sensor data in vision-based camera tracking. In *Third IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 58–67, 2004.

[3] R. Azuma and G. Bishop. Improving static and dynamic registration in an optical see-through hmd. In *SIGGRAPH*, pages 197–204, 1994.

[4] G. Bleser and D. Stricker. Advanced tracking through efficient image processing and visual–inertial sensor fusion. *Computers and Graphics*, 2008.

[5] M. Huber, D. Pustka, P. Keitler, E. Florian, and G. Klinker. A System Architecture for Ubiquitous Tracking Environments. In *Proceedings of the 6th International Symposium on Mixed and Augmented Reality (ISMAR)*, Nov. 2007.

[6] B. Johansson. Multidimensional signal recognition, invariant to affine transformation and time-shift, using canonical correlation. Master's thesis, Linköping University, SE-581 83 Linköping, Sweden, October 1997.

[7] D. Pustka. Construction of data flow networks for tracking in augmented reality applications. In *Proc. Dritter Workshop Virtuelle und Erweiterte Realität der GI-Fachgruppe VR/AR*, Koblenz, Germany, September 2006.

[8] G. Welch and G. Bishop. SCAAT: Incremental tracking with incomplete information. In T. Whitted, editor, *SIGGRAPH 97 Conference Proceedings*, Annual Conference Series, pages 333–344. ACM SIGGRAPH, Addison Wesley, Aug. 1997.