Automatic Calibration of Magnetic Motion Trackers using a Bayesian-Neural Cascade

Piperakis Romanos, Saito Suguru, Nakajima Masayuki

Graduate School of Information Science & Engineering, Tokyo Institute of Technology
E-mail: {romax, nakajima}@img.cs.titech.ac.jp, suguru@pi.titech.ac.jp

1 Introduction

Magnetic trackers are the dominant motion capturing technology for computer graphics applications. But even though they have many advantages versus their mechanical, optical and sonic counterparts, they are amenable to electromagnetic interference. Conventional calibration procedures are tedious and very time consuming, since they require the manual collection of measurements on a relatively dense three-dimensional grid inside the tracking area. Despite this labor-intensive procedure, this type of calibration cannot compensate for every type of distortion resulting in limited accuracy and even motion discontinuities in noise contaminated environments. We propose a Bayesian-neural approach that allows the automatic calibration of magnetic motion capturing systems, using motion tracking data from a short, free-form tracking of a calibration pole.

1.1 Background

In the last 20 years, magnetic tracking has emerged as the preferred way of measuring motions in computer graphics applications, because it offers a simple, elegant way of measuring position and orientation of heads, hands, the entire body and other objects in free space. Other tracking technologies may offer faster performance for a single sensor but the update rate and accordingly the dynamic performance, degrades inversely proportionally each time another sensor is tracked.

The main problem associated with magnetic motion capture systems is their sensitivity to electromagnetic interference and the presence of metal objects in the vicinity of the motion tracking area. Even though pulsed DC systems have been developed to limit the sensitivity of the conventional AC trackers to nearby conductive metals, this technology is unable to compensate for the distortion effects of ferrous metals and external electromagnetic interference (EMI).

In real-world settings it is often the case that the floors and ceilings of the motion capture studios contain steel bar reinforcement, which inevitably interferes with the motion capturing process, inducing measurement errors. Another source of interference comes from the magnetic signatures of electric and electronic devices. In most cases even if precautions are taken to remove large metallic objects away from the immediate tracking area, the interference coming from the studio’s construction materials and its power lines, cannot be so easily counteracted.

Another problem when working with magnetic motion capturing systems is that of the field coverage region. Depending on the specific implementation of the capturing system, if during a motion capturing session the sensor exits the region where the field is strong enough for reliable motion tracking, then the sensor measurement will likely become stuck to the last reliable measurement; that is, the exit point from the operational tracking region. Upon re-entry of the sensors in the operational tracking region, the sensors will regain the ability to detect their coordinates and consequently the sensor reading will jump from the previously stuck position to the new point of re-entry. The intermediate ‘stuck’ sensor positions can be classified as outliers.

This effect could also be regarded as a type of distortion; one for which we would require a correction procedure or at the very least a detection algorithm but for which existing calibration procedures are unable to address.

2 Related Work

To control the metallic or electric distortion problem, especially for the AC tracking technologies that are more susceptible to it, special calibration techniques have been developed. The most frequently applied is called “mapping and compensation” and includes the manual collection of hundreds of data points (mapping) to determine the amount of distortion in the operational area. This collection of points is usually performed with the help of a vertical pole with the sensors placed on it at fixed intervals and orientation. This calibration pole is then consecutively positioned on each point of a dense rectangular grid drawn on the tracking area’s floor. The data points collected in this manner, span a
three dimensional, regular rectangular grid. These data points are then used to form a correction (compensation) map that is applied to the sensed signals.

While this procedure is relatively effective, it is time consuming and costly. Further it assumes that the field transmitter remains stationary and that no metallic objects move or are introduced into the area at any time after the mapping. Also, changes in the power distribution or changes in the operation of electric devices cannot be taken into account. In this sense, the corrections achieved with this type of calibration are ineffectual, for instance, if there is a need to periodically relocate the field transmitter, if a metal object is introduced into the motion-capture area or even if a monitor in the vicinity of the tracking area is turned on.

Approaches towards a more efficient or automated calibration have also been developed for clinical and medical applications, to accommodate the mapping or tracking body parts or organs using magnetic tracking technologies [2] [3] [5]. However, all the current approaches currently involve the use of hybrid technologies. In other words, the only way to circumvent the majority of the motion tracking deficiencies of electromagnetic technologies is by attempting to combine optical, mechanical (inertial), sonic and magnetic motion tracking systems into a single framework [1] [10] [7] [8].

While this remains a valid solution to a lot of problems, it does so at the expense of the added cost and complexity of attaining, setting up and managing the hybrid systems. It also fails to provide a universal solution to one of the most important motion tracking technologies: by attempting to combine optical, mechanical (inertial), sonic and magnetic motion tracking systems into a single framework.

Using as much of this prior knowledge as possible, we could view the problem of calibration as the parameter estimation of a mapping from sensor values to body coordinates. This parameter estimation should by no means be static. Since we don’t know how the field will change due to unpredictable interference or how the sensors will move relative to the actor’s joints, we cannot solve the problem of calibration before analyzing the motion tracking data itself.

This leads us to consider a different paradigm to the calibration problem. If we base our calibration procedure on the entire range of possible motion capture sessions, all the possible combinations of field distortions and different actor bodies, we are losing focus of the problem at hand, which is to acquire the best possible quality of motion capture data given the already acquired mocap session’s measurements. The calibration mapping should thus be defined as a property of an individual session rather than a preprocessing step to the general problem of motion tracking.

As a result this approach aims to use the prior, domain knowledge in conjunction with the specific motion tracked data to produce a clean motion and at the same time detect the average distortion field for the duration of that particular capturing session. Furthermore, such a calibration procedure could be completely automated, boosting efficiency and filtering out inconsistencies in a single framework.

3 Bayesian Framework

If we are to get the most out of a calibration procedure, then it follows that we must take into account all the available domain knowledge we have about motion capturing. This fact is a fundamental axiom of plausible reasoning and is inherently contained in the Bayesian approach to probability theory. In the case of motion capturing of a human actor, this domain knowledge concerns the physical properties of motion itself, the specifications and nature of the motion tracking system, as well as the inherent attributes of the actor’s kinematics. Ideally we would like to incorporate

- Motion continuity. Consecutively detected coordinates have a larger probability of being closer to each other than further.
- Distortion continuity. The electromagnetic field can be taken to be continuous if there are no metal surfaces of appropriate dimensions.
- Tracker specifications. The initial conditions of the calibration can be taken from the ideal performance of the magnetic tracker in the absence of noise, thereby optimizing the parameter search.
- Skeleton constraints. The fixed structure of the human body constraints the possible coordinates that the sensor position can be at any given time.
- Human body kinematic freedom constraints. The human body’s joints have limitations as to the range of rotations they can perform as well as the acceleration of individual limbs. This further constrains the parameter space and can be used to detect and correct calibration inconsistencies.
- Human initial posture. The initial posture of the motion capturing process could provide an approximate starting point for the parameter estimation.

Using as much of this prior knowledge as possible, we could view the problem of calibration as the parameter estimation of a mapping from sensor values to body coordinates. This parameter estimation should by no means be static. Since we don’t know how the field will change due to unpredictable interference or how the sensors will move relative to the actor’s joints, we cannot solve the problem of calibration before analyzing the motion tracking data itself.
3.1 Neural Network Distortion Representation

A key component of any calibration system that attempts to compensate for distortions in the electromagnetic field, is the way it will map the noise contaminated sensor measurements to correct or consistent motion tracked coordinates. This mapping could be represented in a variety of ways. The easiest and more convenient being perhaps a correspondence between two sets of points, the ones the sensors report and the actual coordinates of these sensors in the motion tracking area. An alternative but equivalent representation would be to associate a distortion vector to each point in the sensor coordinate space, which added to a sensor’s reading would yield its calibrated coordinates. Generalizing from the collected set of points to every possible coordinate in the motion tracking region, it is clear that the distortion field can in essence be represented as a vector field.

To represent this vector field we can use a generalization of the 3D object representation approach of [9] with three outputs, one for each distortion coordinate. The new neural network architecture can be seen in figure 1, where the size of every layer is shown and again the R inputs can be feature functions of the input space. These feature functions can be chosen according to the shape characteristics of the field we wish to represent. This choice could thus include some of our prior knowledge about the form of the magnetic field and the distortion we expect given the system’s configuration, inside the motion tracking studio.

The advantages of using a neural network representation for the distortion field are numerous. For one, we make a tremendous economy in the size of the representation. Instead of storing hundreds of sampled points and interpolating for other points based on the local neighborhood of samples, we have a compact analytic representation, which can be evaluated directly, for any point in the tracking region. This also reveals the computational advantages of such an approach. Instead of having to find the nearest sampled points to each sensor position in order to interpolate for each sensor measurement, the distortion can be found with a forward pass of the neural network directly for that point.

Another point where the neural network representation yields superior results is the expressive power. By using custom feature functions as inputs to the network we can generalize to a better approximation of the distortion field, given the same sampled points. To see how that is possible it is enough to point out that the generalization will be a non-linear interpolation of the sampled points. The exact function used for the interpolation will be specified by the combination of the feature functions and the sigmoid activation functions of the hidden layer. That combination will in turn be the result of the training procedure and thus will have emerged as the locally optimum way of interpolating through the given sampled points.

4 Pole Calibration Algorithm

Turning our attention to the problem of calibration, in the scope of this paper and given the vast extend of this topic and the need for concrete results to verify the correctness of our proposed approach, the problem of human motion calibration had to be simplified. The developed algorithm, consequently, attempts to address the problem of automatic calibration using, not the motion tracking data of a human actor, but a free hand movement of a calibration pole. The concepts and problem formulation for that simplified problem will then be used to identify ways of addressing the general problem.

The procedure for obtaining the data to be used for calibration, uses a pole mounted with sensors in fixed length intervals. Figure 2 shows the placement in detail for N + 1 sensors evenly spaced by a distance of L. The calibration pole is then placed in a fixed position inside the motion tracking area and the motion tracking is initiated. The user picks up the calibration pole and moves it around the motion tracking region freely, as shown in figure 3. After
Figure 3: The calibration procedure starts with the user picking up the calibration pole and moving it around the motion tracking region. As is apparent from the sequence, the trajectory of the pole is free although the speed should be kept relatively slow to minimize interpolation errors.

Figure 4: The effect of a calibration pole trajectory across the outlier boundary. The sensor actual position is denoted by a blue square, the actual trajectory by a dashed red line, the sensor detected position by a red square and the detected trajectory by a blue dashed line. Notice the jump from the point of exit to the point of re-entry, denoted by a blue arrow.

most parts of the region have been swept by the calibration pole, a procedure which should not take more than a minute, the motion tracking is stopped. The data collected are then analyzed to determine the distortion field and the calibrated motion capture of the pole at the same time.

4.1 Tracking Range
During the motion tracking of the calibration pole, as the pole gets moved around the tracking area, the sensors sample a certain volume of the capturing region. Since the bounds of the tracking region are not visible to the user or performing actor, there is a relatively high chance of passing them during all but the most static motion capture sessions. The points residing outside the tracking region \( V_t \) but inside the sampled volume \( V_s \) will yield outliers as discussed in previous paragraphs. The points sampled there will consequently report the last point on the boundary of \( V_s \) and \( V_t \) where the sensor exited \( V_t \). This phenomenon is graphically shown in figure 4, where the red dashed line denotes the actual sensor trajectory, the blue points correspond to the placement of the sensor on the calibration pole, the red points signify the reported coordinates from the sensors in the absence of any other type of interference, and the red arrows give the difference between the sampled points and the reported points or else the distortion of the field in the sampled points. That figure also explains the jump effect from the points of exit from the tracking region, to the points of re-entry. In the figure this instantaneous jump effect is shown as a blue arrow and clearly shows the type of inconsistencies and discontinuities that the outlier effect imposes.

There is a number of problems associated with the inclusion of the outlier points in a neural network training procedure for the distortion field. These are associated with the fact that the training set will contain multiple conflicting associations between points in the outlying area \( V_s - V_t \) and the various points of exit from \( V_t \), as well as the discontinuities from the later to the points of re-entry into the \( V_t \). The result of these bad examples are a compromised training procedure and a bad generalization at least in the vicinity of the outlier boundary with the \( V_t \). Therefore it is instrumental for the efficiency and performance of the algorithm to discard these points prior to the training process. In a later paragraph we will discuss how that can be achieved.

4.2 Bayesian Filtering
Before it is possible to discard the outlying sensor measurements or train the neural network representation with the distortion field, a way of detecting and correcting for that distortion is required. The best possible approach to this problem is to treat it as a problem of plausible reasoning. In other words, given the already identified prior knowledge, the simplified structure of the calibration pole and the entire sequence of the motion tracking data, find what is the most probable state of the field that would yield correct results. From the law of large numbers and the overlapping restrictions from the various aspects of the prior knowledge, it is possible to reach a stable conclusion with a relatively high degree of confidence. In other words, given the linearity of the constraints and sufficient data, the distortion representation will map valid sensor measurements into a linear space. Since linear spaces have the property of preserving angles, the resulting motion capture data for a human character will be correct.
The main concept of the calibration process that was developed, is the propagation of our state of knowledge (probability distribution) about the sensor's correspondence to real points, throughout the motion tracking sequence. In other words given our initial uncertainty, the uncertainty of the starting position and the errors associated with the process and the subsequent measurements, find the most probable trajectory of the sensor's coordinates through the motion tracking area. Unfortunately, this problem can quickly become untractable for nonlinear dynamic processes and restrictions, since even if we begin with a conjugate probability distribution function for the initial and process parameters, the distribution will soon cease to be conjugate. This in turn means that we will no more have a simple closed form for the probability distributions and the complexity of propagating that distribution forward in time will soon become overwhelming.

To avoid the complications and questionable performance issues associated with using local linearizations or approximating sampling techniques, it was decided to use linear process dynamics for the motion of the calibration pole:

\[ X(t+1) = FX(t) + \text{noise}(Q) \]
\[ Y(t) = HX(t) + \text{noise}(R) \]

where

\[ X(t) = [s_0(t), s_n(t), ds_0(t), ds_n(t)]^T \]

is the system's state matrix at time \( t \) and it consists of the sensors at the end of the pole. Equations 1 and 2 then give the update and correction equations of the Kalman filter. Specifically, \( F \) is the system matrix which relates the state \( X \) at-time- \( t \), to that of time \( t+1 \). \( H \) is the observation matrix that relates the observations at time \( t \) to the state at time \( t \) and \( Q \) and \( R \) are the process and measurement error covariance matrices respectively.

Using equation 1 and the fact that

\[ s_0(t+1) = s_0(t) + ds_0(t) \]
\[ s_n(t+1) = s_n(t) + ds_n(t) \]

the exact values of the system and observation matrices, can be easily found.

This formulation of the problem has the additional advantage that since all the model parameters are initialized with a gaussian distribution, the

\[ s_0(t) = s_0(t) + \frac{1}{\pi} (s_n(t) - s_0(t)) - \frac{(y - t)s_0(t) + i s_n(t)}{\pi} \]

where \( s_i(t) \) is the \( i \)th sensor measurement at time \( i \) and \( t = 0, 1, 2, \ldots \), Consequently \( s_0(t) \) and \( s_n(t) \) are moving freely and lie at the two ends of the pole, and the rest of the sensor positions are constrained from the above equation.

The pole is in reality rotating and translating in space. Nevertheless, the full transform cannot be expressed in a linear equation. For that reason the edges of the pole were taken to move freely but with the restriction that the sensor measurements are to lie between the two edges ((collinear)
linear dynamics of the process and the linear combination of the measurements have also gaussian distributions. In other words the propagation of the gaussian distribution in time, subject to linear dynamics, is also gaussian. This means that we can have a very compact and computationally efficient representation of the exact state of our knowledge in every frame of the motion capturing process.

4.3 Cascade Training & Outliers

If we apply the Kalman filter described above to the collected motion captured data, we acquire the calibrated sensor coordinates of those measurements. This association of points can be used to train the neural network distortion field discussed previously to give a representation of the calibration errors. Having defined the way the distortion errors are derived, we can now reconsider the problem of discarding the outlying points in order to arrive to a more robust training procedure.

Since it is clear that the volume spanned by the sensor measurements is always a sampling of \( V_s \), where the field is stronger than the detection threshold value, these sensor measurements, in a sense, define the bounding volume of \( V_s \cup V_t \).

One easy way to discard "stuck" points would therefore be to find the sensor measurements that reside closest to the surface of the minimum sized polyhedron that includes all the sensor measurements. These measurements would at least include all the points on the boundary surface between \( V_s \) and \( V_t \). Thus to remove the outliers it would suffice to discard the surface points from the training procedure. After discarding these outlying points, the training procedure can continue without the problems already mentioned.

To increase performance, in practice, the convex hull of the sensor measurements was used instead of the exact bounding polyhedron. This enabled a much faster implementation of the test conditions at the expense of some minor inaccuracies that it is believed that the neural network can generalize over.

After the first filtering, the discarding of outlying points and the training of the neural network distortion field representation, further refinements can be performed to increase the quality of the obtained results. To understand why that would be necessary it must be pointed out that the Kalman filter is a linear approximation of the process dynamics and hence some errors are inevitable. Currently, the way this situation is handled is to renormalize the filtered sensor measurements to correspond to a constant length pole. This ad-hoc algorithm however, injects some error into the calibration process.

It is possible to refine the calibration process if we simulate the neural network distortion field with the sensor measurements and refilter the resulting calibrated coordinates with a much smaller process and measurement error. In practice the errors of the iterated Kalman filtering can be taken from the mean square error of the neural network training session. If there are no outliers in the motion tracking data then this process will quickly converge, while if there are unremoved outliers the filtering error will reach and fluctuate around a minimum.

5 Experimental Results

The algorithm described was tested with the use of a “motion star” magnetic pulse motion tracking system by Ascension. One out of the two available field generators was utilized for training, while the other, unconnected field generator was used to provide additional distortion. Furthermore, the active field generator was placed at floor level to provide enough vertical space in order to test for outlying sensor measurements. The calibration pole was mounted with six sensors at constant fixed intervals of approximately 30 cm and placed in front of the active field generator.

Three different motion tracking sessions were taken, where three different users picked up the calibration pole and moved it around the room trying to sweep as much volume as possible. The duration of these sessions was approximately one minute and the motion capturing system had a sampling rate of 30 frames per second. Thus the datasets collected contained ten to twenty thousand points with six sensor readings for each frame.

The data was then preprocessed to convert from inches to meters and filtered using a Kalman filter with initial covariance 0.1, process covariance 0.01 and measurement covariance 0.1 (meters). The neural network employed had six inputs with \( x, y, z, x^2, y^2 \) and \( z^2 \) feature functions and five nodes in the hidden layer. Also the Levenberg Marquardt (LM) training algorithm was used that typically converged in no more than ten epochs, to an average mean square error in the order of \( 10^{-4} \) meters. Each epoch consisting of approximately 60 thousand points, took 15 to 20 seconds on an Athlon 1Ghz PC.

The data was checked for outliers by discarding all the surface points of the sensor measurement convex-hull, shown in figure 7, and the remaining points were used to train the neural network distortion field. Next the measurement points were simulated by the distortion neural network and the calibrated points were filtered again with the reduced mean square error of the LM training method as the new measurement covariance.
The resulting distortion field, is shown in Figure 6 while Figure 7 displays both the convex hull and the distortion field. From these figures we can see that the distortion is very small close to the field generator and increasing as we move away from it. Notice also the interference from the inactive field generator on the left side of the figures. As a means to evaluating the correctness of our proposed method it must be noted that the resulting sensor measurements are collinear and equally spaced, as imposed by the geometry of the calibration pole. Furthermore, the distance of consecutive sensor coordinates is equal to the spacing of the actual sensors and the observed pole trajectory are consistent with the tracking procedure. Figure 8 shows a part of the trajectory of the calibration pole in a 100 pose resolution. The observed, filtered and smoothed sensor coordinates are displayed with different colors.

Finally, it must be noted that the filtering of the calibrated points by a Kalman filter ensures that the calibration procedure is customized to the properties of the specific motion capture data. The requirement of a local, customized calibration procedure for each and every specific motion tracking session, is thus met.

6 Conclusions & Future Work

With the current implementation of the algorithm, it is clear that it deals with off-line motion capturing. The motion data have to be first collected before they can be used. Nevertheless, if the algorithm is used just to provide the distortion field calibration, then subsequent measurements can be corrected with respect to static interference factors on-line. In this scenario, the new points collected will have to be simulated (forward pass) from the neural network distortion field in order to provide the calibrated points. The only error that might result from this type of calibration procedure is a small global scale error, which is irrelevant to the application of human character motion tracking since the angles between body segments are preserved. This form of calibration however, is not customized to each specific motion tracking session any more, nor can it correct for detected outlying data and transient interference. Therefore it sacrifices some of the merits of the described approach and retains only the automatic and more accurate distortion field representation.

The probabilistic approach to calibration for individual motion captured datasets has been presented as a generalized framework, only a subset of which has been currently implemented. As such, there is a lot of extensions and enhancements that could be included in this section.
The main motivation behind the initial calibration framework approach was to automate the acquisition of clean, consistent, human animation data from magnetic motion tracking systems. As a first step, the calibration process was greatly simplified and automated by the use of a calibration pole. Nevertheless, the final target of the developed framework is the human body structure and not a calibration pole. The later was used to display the merits of the approach to a simple use case and to provide evidence for the accuracy and ease of the method. Consequently, a natural way forward is to extend the approach to more than one rigid segment, in order to represent the human skeleton articulation structure. While doing so does not impose any theoretical difficulties, it would be preferable to research ways of dealing with the limitations of the proposed method in the simple case of a calibration pole before generalizing to more complex articulated structures.

### 6.1 Problems & Limitations

A limitation of the current approach is a result of the simplifications needed to efficiently filter the data. More specifically, the fact that the constant length of the pole was not included in the process dynamics but instead was left to be derived from the noisy data and was re-normalized at each iteration. To alleviate this limitation, the use of non-linear process dynamics and the application of some linearization technique around each frame is currently being considered. The success of such an approach however, relies heavily on certain properties of the probability density function used, in conjunction with the way it is propagated by the non-linear dynamic process. Hence at this stage the question of whether using such an approach or the much similar extended Kalman filter, remains unanswered.

### 6.2 Alternative Bayesian Models

Some other methods of calibrating the mocap data are also being considered. For example factor HMM or custom Bayesian networks could perhaps match the problem description better but their implementations could prove much more demanding. Consequently the various trade-offs between accuracy, performance and complexity have to be carefully considered and tested before committing to a different model. Especially in the general case of a custom Bayesian network that explicitly incorporates all the prior information about the calibration procedure, it would be necessary to use Gibbs sampling or markov chain monte carlo methods to approximate the various probability distributions of interest. The sampling, however, associated with both these approaches, introduces additional concerns related to the convergence and burn-in time of valid results as well as imposing a severe computational overhead in order to be used in every motion capture frame. The most appropriate model is therefore still to be determined.

### References


