



HAL
open science

SPIDAR calibration based on regression methods

M'Hamed Frad, Hichem Maaref, Samir Otmane, Abdelatif Mtibaa

► **To cite this version:**

M'Hamed Frad, Hichem Maaref, Samir Otmane, Abdelatif Mtibaa. SPIDAR calibration based on regression methods. 11th IEEE International Conference on Networking, Sensing and Control (ICNSC 2014), Apr 2014, Miami, FL, United States. pp.679–684, 10.1109/ICNSC.2014.6819707. hal-01016969

HAL Id: hal-01016969

<https://hal.science/hal-01016969>

Submitted on 21 Jul 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

SPIDAR calibration based on regression methods

^{1,2}M'hamed Frad, ¹Hichem Maaref, ¹Samir Otmene, ²Abdellatif Mtibaa

¹ IBISC Laboratory, University of Evry Val-d'Essonne, France

² EμE Laboratory, University of Monastir, Tunisia

{firstname}. {name} @ibisc.fr, abdellatif.mtibaa@enim.rnu.tn

Abstract—In order to obtain accurate position estimation, it is imperative that the SPIDAR, a haptic interface device, is properly calibrated. The driving idea of this work is to derive easy-to-use calibration algorithms that can be used to calibrate our haptic device and to add therefore adaptability to the system behavior. We make use of regression methods and we obtain calibration algorithms as a solution to SPIDAR inaccuracy. The efficacy of the proposed methods is illustrated using experimental data collected from a sensor platform.

Keywords—SPIDAR, Calibration, Characterization, Regression, Neural Network, Support Vector Machine

I. INTRODUCTION

Before launching in a further discussion, let us strike the key word: SPIDAR. The SPIDAR [1], for SPace Interaction Device for Augmented Reality, is a kind of motor string-based haptic display system. The system allows the user to directly manipulate virtual object in the virtual environment. The proposed system is called SPIDAR-8, which has 8 couples of DC motor/rotary encoder distributed on each vertex of a cubic structure. Position of the user can be measured by the length of the strings. The length of a string is known by reading the values from the rotary encoder. Nevertheless, the SPIDAR tracking is highly distorted and inaccurate. The purpose of this paper is to discuss methods for measurement and characterization of the distortion of the SPIDAR position data. These measurements have been performed using an optical tracking solution that provides positions of the SPIDAR's effector. However, this optical tracking system has a defect: tracking-loss. After gathering measurements, various methods can be used to calibrate position data from the SPIDAR to increase accuracy. Several methods are discussed including linear interpolation, neural network and Support Vector Machine Regression (SVR).

The combination of the SPIDAR and the Optical tracking system helps us to make 3D interaction to be more vivid and continuous. In this research paper in due hand, we underline first previous works on tracking system calibration. Then, we develop the theory of regression methods used for solving the SPIDAR calibration problem. Lastly, we evaluate the performances the various calibration methods.

II. RELATED WORKS

Three-dimensional tracker calibration is an extensively studied topic in different machine intelligence communities. The purpose of it is to calibrate data tracker to increase accuracy.

Previous studies [2]–[7] have focused on electromagnetically based tracking systems that are sensitive to the ambient electromagnetic environment. Indeed, Saleh and Kindratenko [2] have applied a calibration method based on neural network for electromagnetic tracking systems. They obtained quite good results compared to other calibration methods like linear interpolation.

Livingston [3] used a mapping grid to calibrate the electromagnetic tracking systems. He restricted the workspace to improve its accuracy. He succeeded to reduce error position to 79%.

Bryson [4], in a technical report, has discussed methods for the measurement and characterization of the static distortion of the position data. The measurement and calibration methods are applied to the Polhemus electromagnetic tracking system

Birkfellner et al [5] developed a hybrid tracking system based on both electromagnetic and optical used in surgery. The purpose of such system is to ensure the tracking continuity as well as to overcome the limits of both systems.

An and Zan demonstrate the advantages of SVM for the identification of non-linear systems against the use of neural networks [6]. They note that learning time and results in generalization are better than those given by neural networks. But their results were for a mono-dimensional system.

III. SPIDAR CHARACTERIZATION

A. The measurement process

For a properly SPIDAR characterization, the three dimensional position must not depend on technical constraints. Therefore, we use a virtual grid composed of a huge number of small cubes, implying that collected SPIDAR data covers the whole SPIDAR workspace. To perform the suggested characterization, two infrared cameras need to be rigidly mounted on the platform used, for instance the big SPIDAR. We call this optical tracking system "ART Tracking".

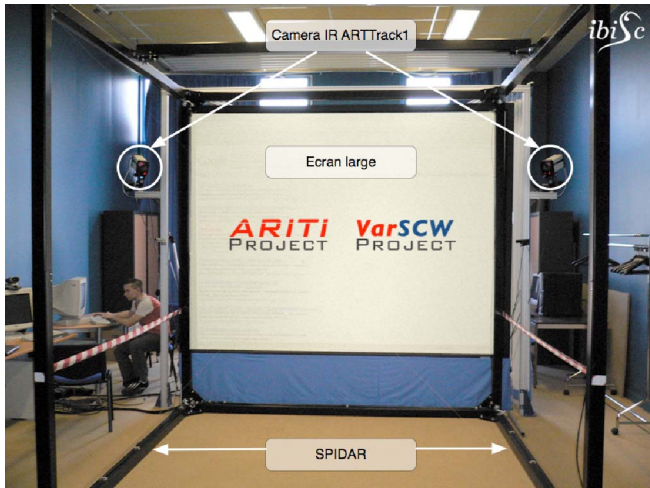


Fig. 1. SPIDAR and infrared camera based optical tracking in our virtual reality platform

Data subsequently needs to be collected while moving the effector in all possible orientations. Whenever the effector collides with a cube, we record the positions given by SPIDAR, the positions given by ART tracking system and also the encoder values. The above measurements were performed in a workspace limited to 1m^3 to obtain good characterization. However, to enable data recording, we must coincide the two referentials. So that, we formulate the relationship between the SPIDAR and ART referential.

$$\begin{aligned} X_{spi} &= X_{art} a_x + b_x \\ Y_{spi} &= Y_{art} a_y + b_y \\ Z_{spi} &= Z_{art} a_z + b_z \end{aligned} \quad (1)$$

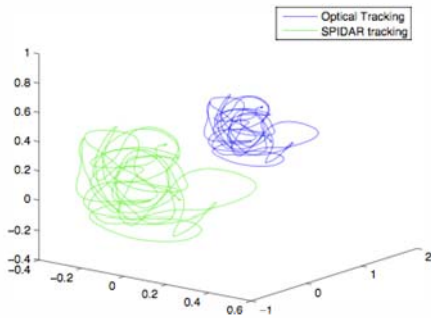


Fig. 2. Relation between the two referentials

B. Three-dimensional structure of the SPIDAR distortion

Distortions can be viewed in several ways. The clearest way of representing of this distortion is with a position error, defined as the returned position by the tracker minus the actual position. The three dimensional graphic below (Fig 3) indicates this distortion. The distribution of absolute error seems to be onion skin distribution in the sense that error is distributed over many spherical planes. The magnitude of absolute error grows as the effector is going far from the center of the measurement volume. In fact, the magnitude of error is longer than 30mm in outside planes.

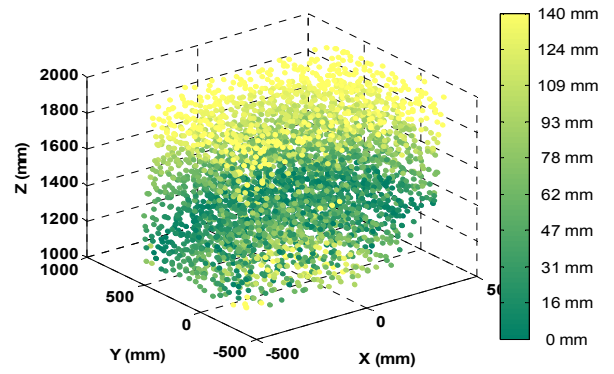


Fig. 3. Absolute Error distribution in the SPIDAR workspace (Dark green is the best).

The characterization of SPIDAR leads us to different problems. On one hand, SPIDAR has several defects that distort position measurements. On the other hand, SPIDAR interface is a black box. So, we have no idea on the mathematical model used to compute effector position.

IV. CALIBRATION METHODS

This paper presents a regression-based approach for solving the SPIDAR's calibration problem. The calibration can be done in several ways. We will try linear and non-linear regression methods: Polynomial, Neural Network and Support Vector Machine In statistics, regression is composed of techniques for modeling the relationship between a response variable and one or more explanatory variables [7]. The goal is to find a useful function $\hat{f}(x)$ for the regression function $f(x)$ that estimates effector position in terms of value of observed responses. The following discussion summarizes the main outlines of each regression method.

A. Linear regression

The linear regression model is a set of regression parameters that has the following form:

$$f(X) = \beta_0 + \sum_{j=1}^n X_j \beta_j \quad (2)$$

Typically we have a set of training data $(x_1, y_1) \dots (x_N, y_N)$ from which to estimate the parameters. Each $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})^T$ is a vector of feature measurements for the i^{th} case. The least square principle will be utilized to derive estimation of the regression parameters.

$$RSS(\beta) = \sum_{i=1}^N (y_i - f(x_i))^2 \quad (3)$$

The linear regression model includes the important class of polynomial regression models in which the relationship between the independent variable and the dependent variable is modeled as an n^{th} order polynomial. It allows

finding the polynomial function that properly fits a given set of data.

B. Neural Network

A neural network is a two-stage regression or classification model, typically represented by a network diagram as in figure 4. For regression, typically $K = 1$ and there is only one output unit Y_1 at the top [8].

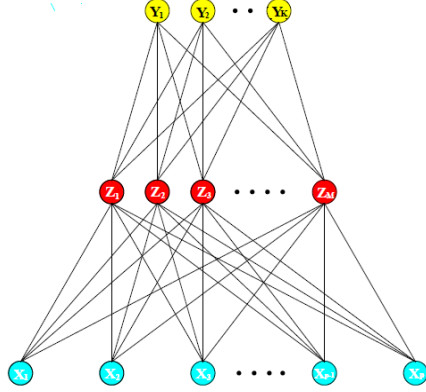


Fig. 4. A Schematic of a single hidden layer, feed-forward neural network [8]

The neural network model has unknown parameters, often called weights, and we seek values for them that make the model fit the training data well. We refer the complete set of weights by θ , which consists of:

$$\{\alpha_{0m}, \alpha_m; m=1, 2, \dots, M\} \quad M(p+1) \text{ weights}$$

$$\{\beta_{0k}, \beta_k; k=1, 2, \dots, K\} \quad K(M+1) \text{ weights}$$

For regression, we use sum-of-squared errors as our measure of fit (error function):

$$R(\theta) = \sum_{k=1}^K \sum_{i=1}^N (y_{ik} - f_k(x_i))^2 \quad (4)$$

The generic approach to minimize $R(\theta)$ is by gradient descent, called back propagation. In the back propagation method, each hidden unit passes and receives information only to and from units that share a connection. However, back-propagation can be very slow, and for that reason it is usually not the method of choice [8].

The Levenberg-Marquardt optimization approach is a nonlinear optimization method using descent gradient. This method improves the classical gradient descent algorithm. This optimization reduces significantly the algorithm convergence time (equation 5).

$$\begin{aligned} w_{i+1} &= w_i - \mu \Delta E(w) \\ \Delta E(w) &= \frac{\partial E}{\partial w} \end{aligned} \quad (5)$$

C. Support vector Regression

Statistical learning-based kernel methods are rapidly replacing other learning methods especially neural network as a preferred tool for machine learning. Here, we develop a

statistical learning-based approach using Support Vector Machine algorithm for solving SPIDAR calibration problem [9]. Therefore, it is important to show SVM adapted for regression with a quantitative response [8]. We first underline the linear regression model:

$$f(x) = x^T \beta + \beta_0$$

To estimate β we consider minimization of:

$$H(\beta, \beta_0) = \sum_{i=1}^n V(y_i - f(x_i)) + \frac{\lambda}{2} \|\beta\|^2 \quad (6)$$

$$\text{Where } V_\varepsilon(r) = \begin{cases} 0 & \text{if } |r| < \varepsilon \\ |r| - \varepsilon, & \text{otherwise} \end{cases}$$

This is an " ε -insensitive" error measure, ignoring errors of size less than ε .

If $\hat{\beta}, \hat{\beta}_0$ are the minimizers of H , the solution function can be shown to have the form:

$$\hat{\beta} = \sum_{i=1}^N (\hat{\alpha}_i^* - \hat{\alpha}_i) x_i \quad (7)$$

$$\hat{f}(x) = \sum_{i=1}^N (\hat{\alpha}_i^* - \hat{\alpha}_i) \langle x, x_i \rangle + \beta_0 \quad (8)$$

Where $\hat{\alpha}_i, \hat{\alpha}_i^*$ are positive and solve the quadratic programming problem:

$$\min_{\alpha_i, \alpha_i^*} \varepsilon \sum_{i=1}^N (\alpha_i^* + \alpha_i) - \sum_{i=1}^N y_i (\alpha_i^* - \alpha_i) + \frac{1}{2} \sum_{i,i'=1}^N (\alpha_i^* - \alpha_i)(\alpha_{i'}^* - \alpha_{i'}) \langle x_i, x_{i'} \rangle$$

Subject to constraints:

$$\begin{aligned} 0 &\leq \alpha_i, \alpha_i^* \leq \frac{1}{\lambda} \\ \sum_{i=1}^N (\alpha_i^* - \alpha_i) &= 0 \\ \alpha_i \alpha_i^* &= 0 \end{aligned}$$

Where:

$$\begin{aligned} \langle x_i, x_{i'} \rangle &\text{ Kernel function} \\ \varepsilon &\text{ Insensitive measure} \\ \frac{1}{\lambda} &\text{ Regularization parameter} \end{aligned}$$

V. EXPERIMENTAL RESULTS AND DISCUSSIONS

In this section, we present the results of three calibration methods: Linear regression, Neural Network and Support Vector Regression.

A. Calibration using polynomial regression

The calibration via polynomial regression tries to fit the data via an order n polynomial in x , y , and z . These polynomials are found via least-squares methods. Polynomial of order 1 was tried using 4096 set of measurements.

If $p_x(\hat{p}_x)$, $p_y(\hat{p}_y)$ and $p_z(\hat{p}_z)$ and are the components of the actual (calibrated) effector position:

$$\begin{pmatrix} \hat{p}_x \\ \hat{p}_y \\ \hat{p}_z \end{pmatrix} = \begin{pmatrix} 0.7204 \\ 0.7286 \\ 0.7564 \end{pmatrix} \begin{pmatrix} p_x \\ p_y \\ p_z \end{pmatrix} + \begin{pmatrix} -44.1471 \\ 7.6676 \\ 338.7982 \end{pmatrix}$$

The graph below (Fig 5) shows that the first degree polynomial regression wasn't very successful to reduce the error amount. Besides, the magnitude of errors remains higher than 30 mm. The absolute error distribution is nonlinear, whereas the polynomial calibration method used is a linear approach. Hence, calibration will be successful only in partial areas where the SPIDAR behavior is linear. It is apparent that the simple regression fails as a viable calibration method.

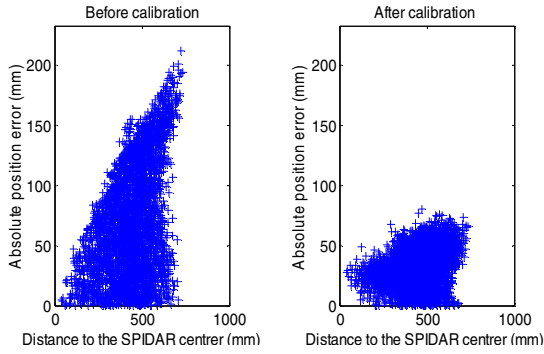


Fig. 5. Absolute error position as a function of distance to the SPIDAR center (before and after polynomial based calibration)

B. Calibration using Neural Network

a) Neural Network implementation

To implement a neural network we need to follow five steps. First, we determine the number of input and output layers. In our work, we use one input layer which represents the position given by SPIDAR. Secondly, we scale all inputs to have mean zero and standard deviation one. This ensures all inputs are treated equally in the regularization process.

$$\begin{cases} N_{spi} = ((P_{spi} - \Gamma_{spi}) / \sum_{spi}) \\ N_{art} = ((P_{art} - \Gamma_{art}) / \sum_{art}) \end{cases}$$

Where

$$\Gamma_{spi} = \gamma_{spi} \cdot J \text{ et } \sum_{spi} = \sigma_{spi} \cdot J$$

$$\Gamma_{art} = \gamma_{art} \cdot J \text{ et } \sum_{art} = \sigma_{art} \cdot J$$

γ_{spi} : Matrix of means of components of the matrix of positions given by the SPIDAR

γ_{art} : Matrix of means of components of the matrix of positions given by ART system.

P_{spi} : Positions given by the SPIDAR

P_{art} : Positions returned by the ART system

J: Identity matrix

N_{spi} : standardized positions of the SPIDAR

N_{art} : standardized positions of the ART system.

Thirdly, we need to estimate the number of hidden layers and units. The choice is guided by background knowledge and experimentation. With one hidden layer, network is able, with a sufficient number of units to approximate any continuous function. To avoid computation complexity, we will use one hidden layer.

The optimal number of units in the hidden layer was estimated empirically by testing many networks with different number of units. The absolute error is plotted as a function of the number of units in the hidden layer (Fig 6). The plot allows identifying the most efficient network and thus the number of units. Indeed, the number of units is equal to five.

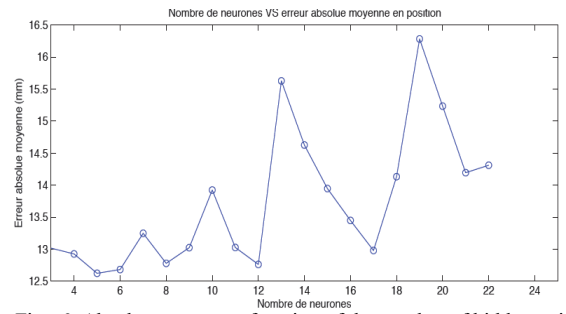


Fig. 6. Absolute error as a function of the number of hidden units

Fourthly, we use the Bipolar Sigmoid activation function to move from the input layer to the hidden layer. The second activation function is chosen to be linear.

Lastly, the learning problem that we consider in this paper is categorized as a supervised learning approach. Hence, we use a Levenberg-Marquardt algorithm, measurements of mean squared error illustrate the learning behavior. The squared error is plotted as function of epochs. The error stabilizes after a few epochs. The error value at the 51 epoch is in the range of $10^{-1.73}$ mm.

b) Neural Network performances

We see from the graphs that the calibration via neural network was successful to reduce the magnitude of the absolute error from an almost the whole workspace (less than 30 mm). We note that the positions returned by the neural model are very similar to those given by ART tracking system.

The magnitude of absolute error is plotted as a function of distance to the center of SPIDAR. Obviously, error is less sensitive to distance. The errors at distances higher than 400 mm were successfully corrected by calibration. This plot shows the efficacy of neural network. So we can easily note that the distribution of absolute position error is much more uniform after calibration. Absolutely, neural network has strongly reduced position error, thus increased the former accuracy.

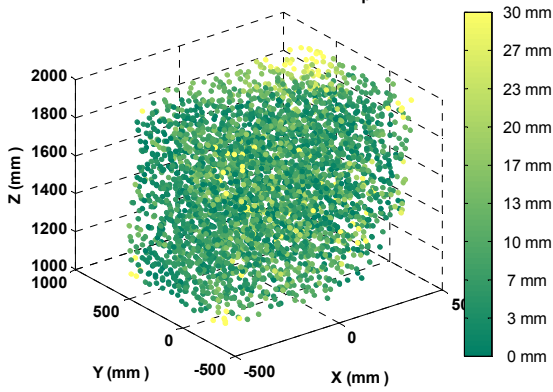


Fig. 7. Absolute Error distribution in the SPIDAR workspace after neural network calibration

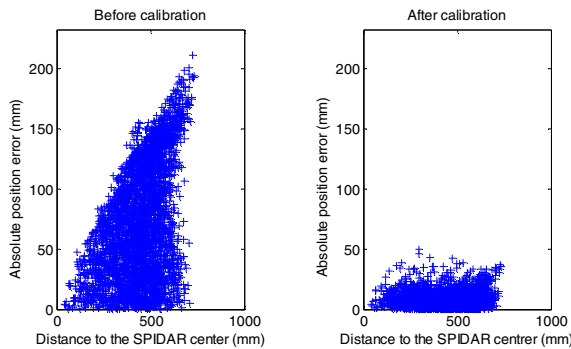


Fig. 8. Absolute position error as a function of distance to the SPIDAR center (before and after neural network based calibration)

To have better idea of the extent position error, we draw the corresponding histograms.

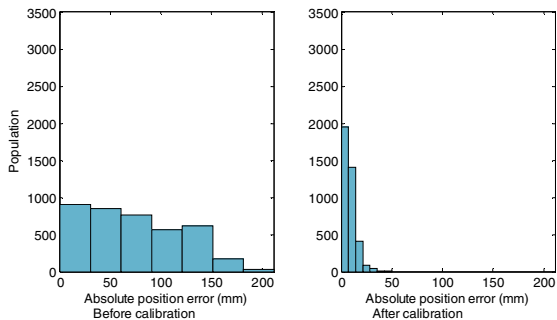


Fig. 9. Histogram of absolute position errors before and after neural network based calibration

The first histogram is spread and indicates bad position accuracy. However, the second is left shifted underlying the good accuracy of neural network results.

Neural network presents reliable standard deviation and mean error values as shown in the table 1. In fact, before calibration, the mean absolute error reaches 72.8593 mm and the standard deviation is 47.0641 mm, which means that the position given by the SPIDAR suffers from a bad accuracy and an important statistical dispersion. After calibration using neural network, the mean absolute error is below 8.1258 mm and the standard deviation is equal to 6.2656 mm, meaning a higher accuracy and a lower statistical dispersion.

TABLE1. Standard deviation and the mean values before and after calibration

| Absolute Error | Raw | NN |
|--------------------------|---------|--------|
| Mean (mm) | 72.8593 | 8.1258 |
| Standard derivation (mm) | 47.0641 | 6.2656 |

After testing neural network with training data, we need to demonstrate its fidelity by measuring its response to measurements that it has never encountered. For this, we have created two databases to evaluate the performance of our network in generalization.

Histograms of position errors and the following tables 2 and 3 show that the neural network keeps good performance with data generalization for both databases mentioned above (database 1 and 2). The standard deviation and the mean values generated by the neural network are very good.

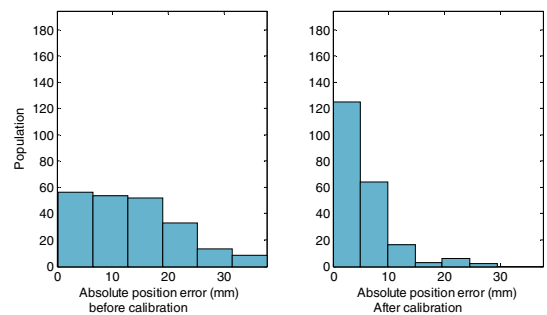


Fig. 10. Histogram of absolute position errors in generalization using the first database (database 1)

TABLE2. Standard deviation and the mean values in generalization using database2

| Absolute Error | Raw | NN |
|--------------------------|---------|--------|
| Mean (mm) | 13.2295 | 5.1523 |
| Standard derivation (mm) | 8.4124 | 5.1177 |

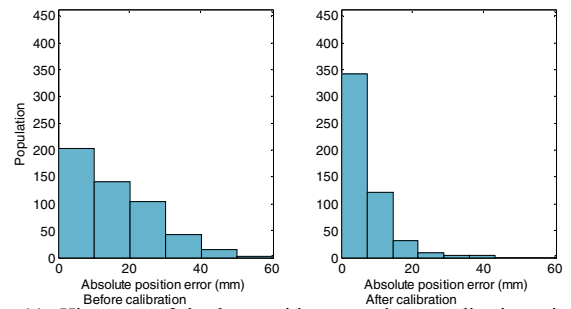


Fig. 11. Histogram of absolute position errors in generalization using the first database (database 2)

TABLE3. Standard deviation and the mean values in generalization using database2

| Absolute Error | Raw | NN |
|--------------------------|---------|--------|
| Mean (mm) | 15.7935 | 6.5667 |
| Standard derivation (mm) | 11.6375 | 6.4320 |

C. Calibration using Support vector regression

In order to study the SVR performances, we draw the absolute position error distribution as previously but after calibration using SVR. As we can see, figure 12 shows that SVR is quite effective and greatly improve the SPIDAR accuracy in comparison with the figure 3. Support Vector Regression performs a good calibration in the whole workspace except in its corners.

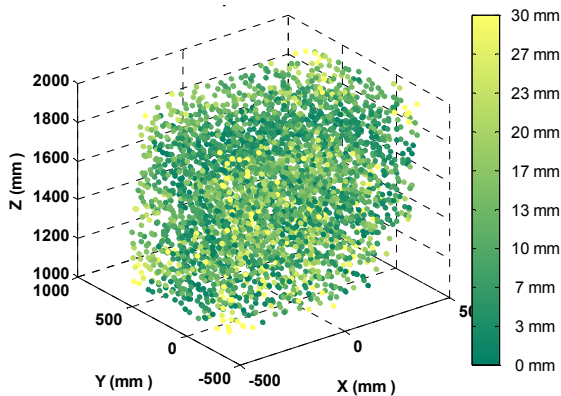


Fig. 12. Absolute Error distribution in the SPIDAR workspace after SVR calibration

Fig.13 shows that after calibration the SPIDAR accuracy is less sensitive of the distance to its center.

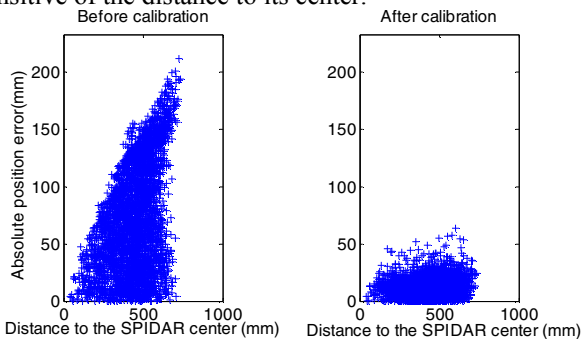


Fig.13. Absolute position error as a function of distance to the SPIDAR center (before and after SVR based calibration)

We also give histogram of position errors before and after using SVR-based calibration.

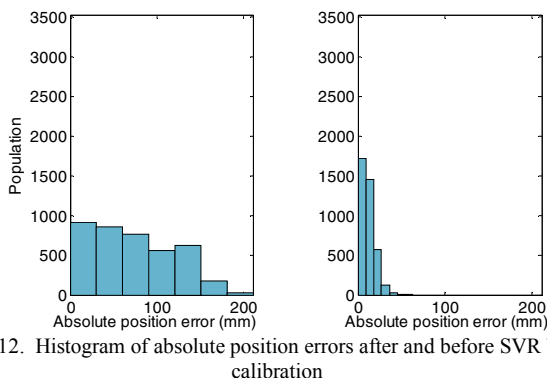


Fig. 12. Histogram of absolute position errors after and before SVR based calibration

As we can see with figure 12 and table 4, after calibration using SVR the mean absolute error is below 11, 4690 mm and

the standard deviation is equal to 8.1342 mm, meaning a higher accuracy and a lower statistical dispersion

TABLE4. Standard deviation and the mean values after and before SVR calibration

| Absolute Error | Raw | SVR |
|--------------------------|---------|---------|
| Mean (mm) | 72.8593 | 11.4690 |
| Standard derivation (mm) | 47.0641 | 8.1342 |

We make a comparison between different methods used for solving the calibration SPIDAR problem. This table presents the performances of each method. It is clearly that calibration using neural network gives the best results.

TABLE5. Standard deviation and the mean values obtained for each calibration method.

| Absolute Error | Raw | NN | SVR | SR |
|--------------------------|---------|--------|---------|---------|
| Mean (mm) | 72.8593 | 8.1258 | 11.4690 | 33.0689 |
| Standard derivation (mm) | 47.0641 | 6.2656 | 8.1342 | 16.1724 |

VI. CONCLUSIONS AND FURTHER WORK

This paper proposed a method of detailed characterization of the SPIDAR output using a virtual calibration grid. It presented also various regression methods for solving the SPIDAR calibration problem and their evaluation.

In future work, we plan to extend the volume of workspace to improve learning by increasing the number of measurements and to test the robustness of different methods used for calibration. We intend also to develop an hybrid tracking system that help us to make 3D interaction to be more interactive as well as more accurate and continuous.

REFERENCES

- [1] Seahak Kim, Masahiro Ishii, Yasuharu Koike, and Makoto Sato, "Development of tension based haptic interface and possibility of its application to virtual reality", Proceedings of the ACM symposium on Virtual reality software and technology, pp. 199–205, 2000.
- [2] T Saleh, V Kindratenko, and W Sherman, "On using neural networks to calibrate electromagnetic tracking systems", Proceedings of the IEEE Virtual Reality annual international symposium, Jan 2000.
- [3] Livingston (1997) Magnetic tracker calibration for improved augmented reality registration. Presence: Teleoperators and Virtual Environment 6(5):532–546.
- [4] Birkfellner W, Watzinger F, Wanschitz F, Enislidis G, Truppe M, Ewers R, Bergmann H (1998) Concepts and results in the development of a hybrid tracking system for cas. Lecture Notes in Computer Science: Medical Image Computing and Computer-Assisted Intervention 1496/1998:343,
- [5] J An, Q Yang, and Z Ma, "Study on svm on-line function regression method for mass data", Machine Learning and Cybernetics, vol. 5, pp. 2773–2777, Jan 2007.
- [6] M Zhang, W Yan, and Z Yuan, "Study of nonlinear system identification based on support vector machine", Proceedings of the International Conference on Machine Learning and Cybernetics, 2004, vol. 5, pp. 3287–3290, Jan 2004.
- [7] X Yan, X Gang Su, "Linear Regression Analysis: theory and computing", 2009, British Library Cataloguing-in-Publication Data.
- [8] T Hastie, R Tibshirani, J Fiedman, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction", 2008, Springer Series in Statistics.
- [9] P Boudoin, H Maaref, S Otmame, M Mallem, "SPIDAR Calibration using Support Vector Regression", 2nd International Conference on Image Processing Theory Tools and Applications (IPTA), pp. 500-505, July 2010.