White Rose Research Online

eprints@whiterose.ac.uk
https://eprints.whiterose.ac.uk
Universities of Leeds, Sheffield and York

Deposited via The University of Leeds.

White Rose Research Online URL for this paper:
https://eprints.whiterose.ac.uk/id/eprint/219647/

Version: Accepted Version

Proceedings Paper:
Alabdullah, M.M., Abdullah, A., Xie, S.Q. et al. (2024) An Autonomous Calibration for IMUs Using Sensor Architecture and Machine Learning Techniques. In: 2024 30th International Conference on Mechatronics and Machine Vision in Practice (M2VIP). 2024 30th International Conference on Mechatronics and Machine Vision in Practice (M2VIP), 03-05 Oct 2024, Leeds, United Kingdom. IEEE. ISBN: 979-8-3503-9192-3. ISSN: 2996-4156. EISSN: 2996-4164.

https://doi.org/10.1109/m2vip62491.2024.10746229

© 2024 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Reuse
Items deposited in White Rose Research Online are protected by copyright, with all rights reserved unless indicated otherwise. They may be downloaded and/or printed for private study, or other acts as permitted by national copyright laws. The publisher or other rights holders may allow further reproduction and re-use of the full text version. This is indicated by the licence information on the White Rose Research Online record for the item.

Takedown
If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

UNIVERSITY OF LEEDS
University of Sheffield
UNIVERSITY of York

An Autonomous Calibration for IMUs Using Sensor Architecture and Machine Learning Techniques

1st :Meernah Mohamemed Alabdullah
Biomedical Engineering, School of Electronic and Electrical Engineeing
Imam Abdurahman Bin Faisal Univresity, University of Leeds
Dammam, Saudi Arabia, Leeds United kingdom
ml20mmma@leeds.ac.uk

2nd : Ahmad Abdullah
School of Biomedical Sciences
Univresity of Leeds
Leeds, United kingdom
a.abdullah@leeds.ac.uk

3rd : Sheng Quan Xie
School of Electronic and Electrical Engineeing
University of Leeds
Leeds, United kingdom
s.q.xie@leeds.ac.uk

4th : Aiqin Liu
School of Biomedical Sciences
University of Leeds
Leeds, United kingdom
a.liu@leeds.ac.uk

Abstract— Inertial measurement units (IMUs) are extensively used in biomechanical research to develop wearable devices for monitoring biomechanical parameters during rehabilitation and disease progression. The accuracy of these measurements critically depends on the calibration process. Our study introduces an autonomous calibration approach for IMUs using sensor architecture and machine learning models to reduce computational complexity and calibration time for accelerometers, gyroscopes and magnetometers in real-time. Our hybrid algorithm combines adaptive bias and scale factor correction with machine learning techniques. Accelerometer was calibrated using Linear Regression and Decision Trees to handle linear and non-linear complexities, while the gyroscope was calibrated using a Forest Regression model and the magnetometer was proposed to be calibrated using Support Vector Machines. Preliminary results demonstrated high accuracy and stability. The sensor architecture approach achieved a Mean Absolute Error of 0.009g for the accelerometer and 0.011 to 0.018 °/sec for the gyroscope, with an overall standard deviation close to zero. The machine learning approach resulted in an accuracy of 0.009g for the accelerometer and 0.011 to 0.012 °/sec for the gyroscope. The total calibration times were ~ 1.16 minutes for the architecture approach and ~ 9 seconds for the ML-based autonomous calibration approach. This innovative approach demonstrates the potential for real-time applications, enhancing the reliability and efficiency of wearable devices in medical and biomechanical fields.

Keywords— Calibration, Inertial measurement units, IMU, Autonomous, Accelerometer, Gyroscope, Magnetometer, Machine learning, SVM, Decision tree.

I. INTRODUCTION

Inertial Measurement Units (IMUs) have garnered significant attention in medical applications, particularly in areas such as rehabilitation monitoring, gait analysis, and postural assessment [1]–[3]. Their compact, affordable, and accessible nature offers a viable alternative to bulky and expensive devices such as motion capture systems (MOCAP). In addition, IMUs provide real-time measurements across different body segments, enabling continuous monitoring and personalized rehabilitation programs outside clinical settings [4]. IMUs offer reliable and precise data by attempting rigorous calibration processes to minimize errors and optimize performance. Additionally, the required calibration time is critical because a long calibration time affects time-sensitive and high-demand environments such as the ones of medical applications, where extended downtime can lead to significant operational delays and increased costs [5]. IMUs are prone to various deterministic errors, which can be mitigated by adjusting parameters such as the sensor axis misalignment, offsets, inaccurate scale factors, and cross-axis sensitivity [6], [7]. Calibration approaches can be categorized into autonomous and nonautonomous methods. Autonomous calibration refers to self-calibration methods that require minimal human intervention, whereas nonautonomous calibration relies on external equipment and specialized setups [6].

Traditionally, IMU calibration has been achieved non-autonomously using turntables and external equipment [8]. Although this method yields accurate calibration results, it is expensive and requires expert users. Consequently, reducing the amount of equipment used and implementing autonomous methods have become prominent research topics. Autonomous calibration approaches can be categorized into software and positioning, deep learning (DL), and fusion techniques. Chao et al. [9] implemented the Particle Swarm Optimization (PSO) algorithm for software and positioning techniques to estimate sensor model parameters, particularly for the gyroscope. This optimization technique significantly enhances accuracy by precisely adjusting the sensor's calibration parameters. Similarly, Jiazhen et al. [10] introduced a Four-Position-and-Three-Rotation (FPTR) calibration sequence, eliminating the need for turntables and reducing calibration time. They integrated bias compensation, linearity calibration, and wavelet denoising to enhance the accuracy of accelerometers and gyroscopes in conjunction with Fiber-Optic Gyroscopes (FOG) IMU. In [11], accelerometers and gyroscopes were calibrated by developing an algorithm that removed biases and optimized outlier awareness. In [7], the authors addressed the non-linear scale factors of accelerometers and gyroscopes, proposing an algorithm that derives these factors effectively to enhance sensor accuracy across a broader measurement range. One of the sensor positioning approaches proposed twelve-position calibration and ellipsoid fitting to calibrate accelerometers, magnetometers, and gyroscopes, enhancing the accuracy of attitude estimation for applications in robotics [12]. Other studies, such as [13], [14], adopted the multi-position approach without precise equipment based on matching specific force and gravity vector magnitudes and aligning the gravity vector with computed values from the gyroscope outputs. In [9], the authors utilized the output vectors of the accelerometers and gyroscopes, calculated angles through integration, and used an inclined surface to determine the rotating heading datum; however, this approach required 60 min to complete all the calibration steps. Across these studies, the gyroscope calibration achieved the highest accuracy of

0.1 (°/sec) [10], while the accelerometer calibration showed a 30% improvement in accuracy [13]. Regarding the DL-based approaches, in [15], a deep convolutional neural network called Calib-Net was employed to dynamically calibrate and correct IMU measurements using sequential gyroscope and accelerometer data, applying dilation convolution for spatiotemporal feature extraction. This method showed better results than the Odometry Neural Network (IONet) introduced in [16], although it took 8.5 min to calibrate the accelerometer and gyroscope. Calib-Net achieved significant accuracy improvements, reducing orientation drift by approximately 30%. Hybrid learning-based methods incorporate Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, and attention CNN-LSTM mechanisms (ACL) to address gyroscope noise, capturing spatiotemporal features and allocating different weights to sequences [17]; this study reported the highest accuracy across the reviewed studies, with average errors of bias instability and angle random walk reduced by 57.1% and 66.7% respectively, compared to the raw gyroscope data. In the context of fusion-based approaches, several notable studies have been conducted. Wen et al. [18] proposed an improved calibration method utilizing the KF-based AdaGrad algorithm, reducing IMU bias errors by 25% in static tests. Sun et al. [19] developed a fast calibration method for strapdown IMU based on HMM/KF, achieving a calibration accuracy within 0.1°. Jafari et al. [20] used a Kalman filter for skew redundant MEMS IMU calibration, reducing bias estimation errors by approximately 50%. Al-Jlailaty et al. [21] reported enhanced calibration accuracy by reducing gyroscope drift errors by 40%. Additionally, KF was used in [21] to benefit from zero-velocity measurement updates and was validated through extensive simulations and real experiments. According to the literature, the fastest time reported for autonomous calibration was 4 sec to calibrate the accelerometer and gyroscope [13].

Although these methods collectively advance the field of IMU calibration using affordable solutions, making it more accessible for research groups and portable instrument applications, several limitations exist. The complexity of DL layers can limit their application in real-time or resource-constrained environments. These relied solely on software without considering alterations to hardware registers. Altering hardware registers can enhance precision and accuracy, optimize performance, ensure persistence, and offer customized settings tailored to specific applications [22]. Additionally, there are limited studies aimed at calibrating accelerometers, gyroscopes, and magnetometers together using one algorithm, which could streamline the calibration process and improve the overall accuracy and consistency of sensor data. The KF calibration approach has limitations, including linearity assumptions, Gaussian noise assumptions, and observability issues [23]. When the IMU errors exceeded certain limits, the KF integration began to degrade, and more complex techniques were required. Therefore, we aimed to develop an autonomous algorithm based on learning approaches and the sensor's architecture to reduce the computational complexity and calibration time for accelerometer gyroscopes. In this study we evaluated the effectiveness of calibration using sensor hardware and assessed the accuracy of calibrated data after implementing adaptive bias and scale factor correction for real-time autonomous calibration. We implemented lightweight machine learning (ML) models in the calibration process to reduce computational complexity.

II. MEHTODOLOGY

A. System Overview

Calibration was performed using a standard setup composed of an IMU (ICM20948 from SparkFun) with sampling rates up to 4kHz, 9kHz, and 100 Hz for accelerometer, gyroscope and magnetometer, respectively. This sensor has an acceleration range of ±16 g, an angular velocity range of ±2000 (°/sec) and a magnetometer with a range of ±49μT. The other component of the setup is the microcontroller. The system was mounted on a rigid box to maintain stable sensors and provide a controlled data collection environment. The calibration algorithm was implemented on the microcontroller which has dual-core processing capability, running at a clock speed of up to 240 MHz and equipped with 520 KB of SRAM. The microcontroller ran a real-time operating system, with calibration algorithms developed using MATLAB (R2021a) and C/C++ with relevant libraries for numerical computations and data handling. The IMU coordinates were set to be aligned with the frame's coordinate system, where the x-axis points in the forward, y-axis right, and z-axis down directions.

B. Calibration Framework and Data Acquisition

Figure 1 shows the conceptual framework that outlines the overall algorithm for IMU calibration. This framework integrates adaptive bias correction and ML models. The implemented algorithm relied on collecting raw IMU data autonomously without additional equipment. Raw gyroscope data were collected while the sensor was stationary for specific periods. This approach ensures accurate bias and drift estimation by minimizing external influences on sensor readings. The accelerometer was calibrated using a multi-position static method first introduced by Lotters et al. [24], where the sensor was oriented in six different positions [25]. This orientation was adopted to capture the effect of gravity on each axis, providing comprehensive coverage for estimating and correcting the scale factors and offsets. For simplicity, it was assumed that the sensor had a linear response; the errors were constant across the measurement

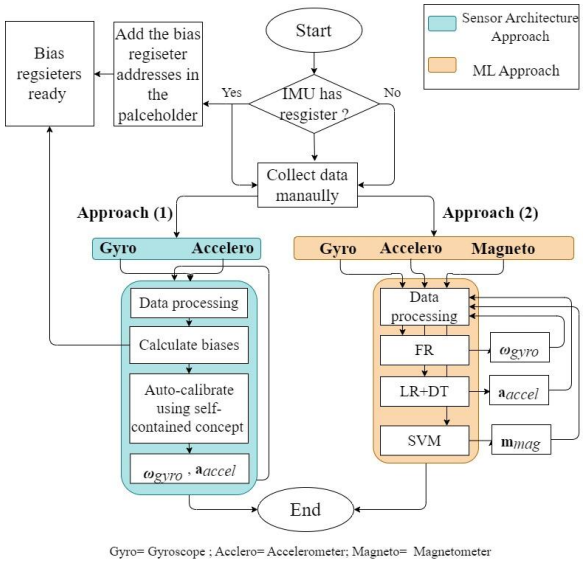


Figure 1. Our proposed auto calibration algorithm pipeline.

range and the gravitational force was the only external acceleration acting on the sensor. In contrast, the magnetometer raw data were collected by waving the IMU in a spherical shape in the air to cover all possible directions, exposing the sensor to a wide range of magnetic field orientations. A calibration framework was developed to provide a high-level overview of the calibration process used to align and correct the gyroscope, accelerometer, and magnetometer data.

C. Sensor's Architecture Approach

This approach relied on assigning calculated biases to the registers of the sensor. The bias registers are typically available for the accelerometer and gyroscope but not for the magnetometer; this is because the magnetometer measures the Earth's magnetic field, which can vary significantly based on location and environmental factors [26]. Thus, this approach was focused on the gyroscope and accelerometer. The detailed algorithm is illustrated in Table 1. The auto dynamic calibration was achieved by implementing an adaptive bias and scale factor correction that also works to enhance the accuracy and stability of the raw data without introducing delays by ensuring that any gradual changes in the bias due to temperature variations or other environmental factors were accounted for. This approach was activated when the buffer became filled with historical raw data collected from the accelerometer and gyroscope. Then, the raw data was averaged for each axis over a specified historical size. These average values represent the dynamic biases, which were then subtracted from the sensor's readings.

Table 1. Sensor's architecture autonomous calibration approach.

Algorithm 1.1. Gyroscope & accelerometer autonomous calibration

1. Determine initial parameter estimates using real-time data collection
   a) Initialize $\hat{g}_{x,0}$, $\hat{g}_{y,0}$, $\hat{g}_{z,0}$, $\hat{a}_{x,0}$, $\hat{a}_{y,0}$, $\hat{a}_{z,0}$ to zero.
   b) Collect gyroscope readings following the prescribed orientations then the average is computed for each orientation to determine the biases and scale factor

$$\hat{g}_{x,0} = \frac{1}{n}\sum_{i=1}^{n} f_{x,i} \ , \hat{g}_{y,0} = \frac{1}{n}\sum_{i=1}^{n} f_{y,i} \ , \hat{g}_{z,0} = \frac{1}{n}\sum_{i=1}^{n} f_{z,i}$$

$$\hat{a}_{x,0} = \frac{1}{n}\sum_{i=1}^{n} r_{x,i} \ , \hat{a}_{y,0} = \frac{1}{n}\sum_{i=1}^{n} r_{y,i} \ , \hat{a}_{z,0} = \frac{1}{n}\sum_{i=1}^{n} r_{z,i}$$

$$\hat{s}_{i,0} = \frac{2xACC_SENSITIVTY}{r_{i,pos} - r_{i,neg}} \quad \text{for } i \in \{x, y, z\}$$

Where, $\hat{s}_{i,0}$ is the calculated scale factor and r_i is the posittive and negative readings

2. Repeat the following steps for continuous calibration in the "loop" function
   a) Collect gyroscope readings $f_{x,t}$, $f_{y,t}$, $f_{z,t}$ for calibration and collect accelerometer readings $r_{x,t}$, $r_{y,t}$, $r_{z,t}$
   b) Apply low-pass filter to obtain filtered reading for gyroscope $(\tilde{f}_{x,t}, \tilde{f}_{y,t}, \tilde{f}_{z,t})$ and for accelerometer $(\tilde{r}_{x,t}, \tilde{r}_{y,t}, \tilde{r}_{z,t})$, the general equation is:

$$\tilde{v}_{i,t} = \propto v_{i,t} + (1 - \propto)\tilde{v}_{i,t-1}$$

Where, $\tilde{v}_{i,t}$ is the filtered readings for gyroscope and accelerometer; $\propto$ is the low-pass filter coefficient; $v_{i,t}$ is the sensor's raw readings; $\tilde{v}_{i,t-1}$ is the previous filtered readings.

   c) Apply dynamic bias correction to update $\hat{g}_{x,i}$, $\hat{g}_{y,i}$, $\hat{g}_{z,i}$ and $\hat{a}_{x,i}$, $\hat{a}_{y,i}$, $\hat{a}_{z,i}$

Algorithm 1.1. Gyroscope & accelerometer autonomous calibration

$$\hat{g}_{x,i} = \frac{1}{n}\sum_{j=1}^{n} \bar{f}_{x,j} \ , \hat{g}_{y,0} = \frac{1}{n}\sum_{j=1}^{n} \bar{f}_{y,j} \ , \hat{g}_{z,0} = \frac{1}{n}\sum_{j=1}^{n} \bar{f}_{z,i}$$

Correct $(r_{x,t}, r_{y,t}, r_{z,t})$ as follows:

$$\tilde{r}_{i,t} = \begin{cases} \text{sign}(r_{i,t}) \text{if} |r_{i,t}| > 1 - \in \text{and} |r_{j,t}| < \in \text{and} |r_{k,t}| < \in \{x, y, z\} \\ \text{and } i \neq j \neq k \end{cases}$$

Where, $\tilde{r}_{i,t}$ is the corrected readings and $r_{i,t}$ is the raw readings.

   d) Apply calibration adjustments for gyroscope $f_{i,j}$ and accelerometer $r_{i,j}$, assuming gyroscope scale factor =1

$$\hat{v}_{i,j} = \begin{cases} f_{i,j} - \hat{g}_{i,j} \\ (r_{i,j} - \hat{a}_{i,j}) \, xScale_i \end{cases} ; \ i, j \in \{x, y, z\}$$

   e) Update parameter estimates $\hat{g}_i$, $\hat{a}_i$ based on the collected readings from each axis $i \in \{x, y, z\}$

$$\hat{p}_i = \hat{p}_i - \propto (v_{i,t} - \hat{p}_i),$$

where $\hat{p}_i$ is the updated parameter estimate ; $v_{i,t}$ is the raw da and $\propto$ is the learning rate/update coefficient.

   f) Write updated biases into gyroscope registers (if it is available)
   g) Repeat continuously within the loop function

D. Autonomous calibration Using ML Techniques

The calibration of the accelerometer using a hybrid approach of Linear Regression (LR) and Decision Trees (DT) was initiated when the sensor did not have bias registers. Table 2 shows the working process of the proposed algorithm. This algorithm was developed by collecting raw data from each axis orientation; then, these raw data were used as features to train the model. Then, they were split into training and testing sets, with 80% of the data used for training and 20% for testing, using MATLAB. After the LR model was trained, the intercept and coefficient of each axis were extracted for later use in the deployment stage on the microcontroller's firmware. Then, the DT model was trained with a single tree, and the mathematical rules were extracted.

Table 2. Accelerometer ML hypered approach.

Algorithm 1.2. Accelerometer LR-DT autonomous calibration

1. Determine initial parameter estimates using real-time data collection; Initialize $\hat{a}_{x,0}$, $\hat{a}_{y,0}$, $\hat{a}_{z,0}$ to zero.

2. Repeat the following steps for continuous calibration in the "loop" function
   a) Collect accelerometer readings $a_{x,t}$, $a_{y,t}$, $a_{z,t}$
   b) Apply low-pass filter to obtain filtered reading $\tilde{r}_{x,t}$, $\tilde{r}_{y,t}$, $\tilde{r}_{z,t}$
   c) Predict the calibrated values using the linear regression model for each axis $i \in \{x, y, z\}$

$$\hat{c}_{i,0} = coef_{i,j} r_{j,t} + intercept_i$$

   d) Apply decision tree to the filtred raw data from each axis $i \in \{x, y, z\}$

$$treeReults_{i,t} = \begin{cases} -1 & if \ r_{i,t} < \delta_1 \\ 0 & if \ r_{i,t} \geq \delta_1 \ and \ r_{i,t} < \delta_2 \\ 1 & if \ r_{i,t} \geq \delta_2 \end{cases}$$

Where, δ_1 and δ_2 are the upper and lower thresholds of the DT and $r_{i,t}$ is the raw reading for axis i.

   e) Combine LR and DT results to calibrate each axis, by mulitpying by the proper wight (w):

$$cal_{\vec{r},t} = w_1 . (\vec{C}_{\vec{r}} . \vec{r}_t + \vec{I}_{\vec{r}}) + w_2 . treeResults_{\vec{r},t}$$

Where: $cal_{\vec{r},t}$ calibrated values for each axis; $\vec{C}_{\vec{r}}$ is vector of coefficients for each axis; $\vec{r}_t$ vector of raw measurements at time t; $\vec{I}_{\vec{r}}$ intercept for each axis.

Table 3 shows the algorithm that was implemented to autonomously calibrate the gyroscope. This algorithm uses Random Forest Regression (FR). First, a synthetic dataset was generated to train the model, which represented various motion patterns to mimic real-world scenarios, such as constant rotation, sinusoidal motion and stationary conditions with low and high noise levels. Then, all the generated datasets were merged into a single dataset. The generated raw data for each axis were used as features to train the model, while the data were split into 70% for training, 15% for validation and 15% for testing. The model's hyperparameters, such as the number of trees and leaf size, were tuned using the validation set based on experimental trials, where a single leaf showed the lowest Mean Square Error (MSE) for each axis. Then, the decision rules were extracted as mathematical equations that can be deployed on the constrained hardware setups.

Table 3. Gyroscope ML autonomous calibration approach.

Algorithm 1.3. Gyroscope FR model autonomous calibration

1. Collect convenient raw data for each axis x,y and z
2. Initialize the buffers for moving average filter
$$\text{xBuffer[i]} = 0, \text{yBuffer[i]} = 0, \text{zBuffer[i]} = 0 \quad \forall_i = 0, .., \text{window size}$$
3. Repeat the following steps within the "loop" function for continuous calibration in real-time:
   a) Collect gyroscope readings
   $$g_{x,t}\ ,\quad g_{y,t}\ ,\quad g_{z,t}$$
   b) Apply low-pass filter to obtain the filtered readings for each axis; N is the size the moving average filter
   $$\tilde{g}_{a,t} = \frac{1}{N}\sum_{i=0}^{N=1} g_{a,t-i}$$
   c) Apply the decision rules to the filtared data to obatined the calibarted data $\omega_x\ , \omega_y, \omega_z$

The ellipsoid fitting approach was adopted [27] to calibrate the magnetometer for hard and soft iron corrections. This approach was handled by collecting the raw data from the magnetometer offline; then the raw data was calibrated using the MATLAB built-in functions. To evaluate the calibration, the Root Mean Square Error (RMSE) was computed for each axis to quantify the difference between the sensor's measurements and the true Earth magnetic field values. Due to their complementary strengths, the calibrated data was used to train two different models, K-Nearest Neighbors (KNN) and Support Vector Machines (SVM). KNN effectively captured local data patterns and non-linear relationships, while SVM excelled in high-dimensional spaces and provided robust global optimization. The extracted features were the raw data of three axes, x, y and z; then they were normalized to scale features between 0 and 1. Then, the dataset was trained randomly into 80% training and 20% testing. Simultaneously, SVM models with Gaussian kernels were trained for each dimension. These models were standardized and cross-validated using five folds to enhance their generalization capabilities. The best-performing model from the cross-validation was selected based on the minimum k-fold loss. Predictions were made on the test set, and RMSE was calculated to assess the model's accuracy. Then, the mathematical models of each training model were extracted as shown in Equations (1) and (2).

$$\hat{y} = \frac{1}{k}\sum_{i=1}^{k} y_i \tag{1}$$

Where $\hat{y}$, is the predicted value, k is the number of neighbors and y_i are the values of the nearest neighbors.

$$f(x) = \sum_{i=1}^{n} \propto_i K(x, x_i) + b \qquad K(x, x_i) = e^{(-\gamma\|x - x_i\|^2)} \tag{2}$$

Where $\propto_i$ are the SVM coefficients, and b is the bias.

E. Performance Evaluation and Robustness Testing

In order to evaluate the performance of the proposed approach, we conducted a series of tests focusing on accuracy, stability, and reliability. These tests were applied to both the accelerometer and gyroscope sensors of the ICM20948 and LSM9DS1 (STMicroelectronics) IMUs. The evaluation was carried out in three trials for each sensor to ensure the robustness and repeatability of the results by averaging each matric for each axis. To assess accuracy, we compared the sensor readings from our proposed autonomous calibration method against the ground truth values, which were zero for the gyroscope at a stationary state and ±1g for the accelerometer when aligned with gravity. The accuracy was quantified as the computed mean absolute error (MAE) between the measured and reference values for each axis using Equation (3). Stability testing was performed by evaluating the consistency of sensor readings over time. We measured the variance of the sensor outputs during a static condition across the three trials, with lower variance indicating higher stability using Equation (4). Reliability was assessed by observing the sensor's performance during repeated measurements under controlled conditions. For this, the sensors were placed in a stationary position for each trial to ensure that external influences were minimized. We specifically measured the standard deviation (SD) of the sensor outputs using Equation (5) to verify that the accelerometer and gyroscope maintained consistent values across multiple trials under these controlled conditions.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} |y_i| - \hat{y}_i \tag{3}$$

Where n is the total number of collected data; y_i is the ground truth values; $\hat{y}_i$ is the predicted measurements.

$$\text{ariance} = \frac{1}{n}\sum_{i=1}^{n} (x_i - \mu)^2 \tag{4}$$

Where n is the total number of collected data; x_i is the i-th value of the dataset; μ is the mean of the collected raw data.

$$\text{SD} = \sqrt{\text{Variance}} \tag{5}$$

III. RESULTS

The performance metrics for the accelerometer and gyroscope calibration using sensor architecture and ML techniques are summarized in Table 4. For the accelerometer calibrated using the sensor architecture approach, the MAE was 0.009g on the x-axis and zero (g) on the y and z-axes. The stability of this method, as indicated by variance, was close to

Table 4.Performance metrics for accelerometer and gyroscope calibration using sensor architecture and ml techniques.

Sensor	Approach	Accuracy (MAE)			Stability (Variance)			Reliability (SD)			Taken time	Model
		x	y	z	x	y	z	x	y	z		
Accelerometer (g)	**Sensor architecture**	0.009	0.000	0.000	0.004	0.000	0.000	0.027	0.000	0.000	1 min	ICM20948
Gyroscope (°/sec)		0.011	0.012	0.018	0.000	0.000	0.000	0.005	0.002	0.002	9.6 sec	
Accelerometer (g)	**ML technique**	0.009	0.009	0.006	0.000	0.000	0.000	0.010	0.013	0.008	2.8 sec	
Gyroscope (°/sec)		0.011	0.012	0.011	0.000	0.000	0.000	0.003	0.003	0.003	2.8 sec	
Accelerometer (g)		0.006	0.008	0.004	0.000	0.000	0.000	0.006	0.006	0.005	5.6 sec	LSM9DS1
Gyroscope (°/sec)		0.000	0.000	0.000	0.000	0.000	0.000	0.001	0.001	0.000		

zero across all axes. The measured reliability by SD was 0.027g for the x-axis and zero (g) for the y and z-axes. The calibration time was ~1 minute. The gyroscope using the same approach achieved an MAE ranging from 0.011 to 0.018 °/sec across different axes. Variances were minimal, close to zero (°/sec), indicating high stability. The SD ranged from 0.002 to 0.005 °/sec, suggesting reliable measurements. The calibration time for the gyroscope was ~9.6 sec. When employing the ML approach, the accelerometer achieved an MAE of 0.009g for the x and y-axes and 0.006g for the z-axis. The variances were close to zero (g). The SD ranged from 0.008g to 0.013g, demonstrating high stability. The calibration time was significantly reduced to ~2.8 sec. For the gyroscope using the ML approach, the MAE was 0.011 to 0.012 °/sec across all axes. Variances remained minimal (0.000 °/sec). The SD was consistently ~0.003 °/sec across all axes. The calibration time was ~2.8 sec. For the accelerometer using the ML approach with the LSM9DS1 sensor, the MAE was 0.006g on the x-axis, 0.008g on the y-axis, and 0.004g on the z-axis. The variances were close to zero (g). The SD ranged from 0.005 to 0.006g, indicating high stability. The calibration time was ~ 5.6 sec. The performance comparison between the k-NN and SVM models shown in Table 5 for magnetometer calibration revealed that the SVM model achieved lower RMSE values across all axes. The SVM model's RMSE was 0.016 µT for the x and y-axes and 0.018 µT for the z-axis, compared to the k-NN model's RMSE of 0.032 µT across all axes. Additionally, Figure 2 highlights that the SVM model can identify and represent the boundary of the data distribution accurately to autonomous calibrate the magnetometer. On the other hand, Table 6 shows the ellipsoid calibration approach for the magnetometer with RMSE values of 1.207 µT for the x-axis, 1.176 µT for the y-axis, and 1.200 µT for the z-axis, indicating effective performance.

Table 5. Comparison performance between k-NN and SVM based on RMSE (μT) metric

Model	x	y	z
k-NN	0.032	0.032	0.031
SVM	0.016	0.016	0.018

Table 6. RMSE (μT) metric to evaluate the calibrated magnetometer using ellipsoids approach

x	y	z
1.207	1.176	1.200

IV. Discussion

This study introduced an innovative approach to the autonomous calibration of IMUs using a hybrid combination of sensor architecture and ML techniques. The main objectives were to improve the accuracy, stability, and reliability of IMU measurements while minimizing

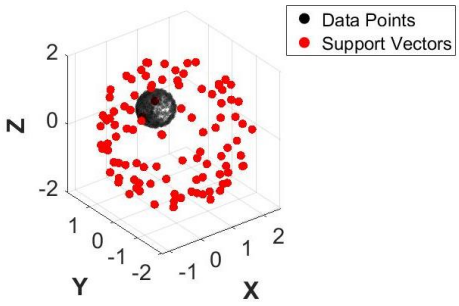


Figure 2. SVM performance visualized Vs. calibrated raw data.

computational complexity and calibration time. Chao et al. [9], used the PSO algorithm for gyroscope calibration and required 60 min to achieve an accuracy of ~0.01 °/sec. Similarly, Calib-Net [16] introduced a DL-based method for calibrating low-cost IMUs, and they achieved a calibration time of around 8.5 min. However, this study did not provide specific numerical accuracy values for gyroscope and accelerometer measurements. Additionally, a study on in-situ [28] gyroscope calibration based on accelerometer data achieved a gyroscope RMSE of ~ 0.03 °/sec (x-axis), 0.02 °/sec (y-axis), and 0.03 °/sec (z-axis) with calibration times of around 5 sec per axis. In comparison, our hybrid approach achieved similar accuracy demonstrated in [9] and improved accuracy compared to the one obtained in [28], ranging from 0.011 to 0.018 °/sec, in significantly reduced calibration times ~ 1.16 min for the sensor's architecture approach and ~9 sec for the ML-based autonomous calibration, highlighting the efficiency and practical applicability of our approach for real-time applications. Our findings suggest that the proposed approach can effectively balance complexity, accuracy and time. Furthermore, the low values in performance matrices underscore our approach's accuracy, stability and reliability. Specifically, the consistent improvements in accuracy, stability, and calibration time observed for the LSM9DS1 sensor validated the effectiveness of our approach and highlighted the versatility and robustness across different sensor models. On the other hand, the initial results obtained for autonomously calibrating magnetometer indicate the potential efficiency of using the SVM model. However, it needs to be validated on real-time data to assess its performance.

V. Conclusion and Future Work

In this paper, we have presented a simple and effective autonomous sensor architecture-based calibration algorithm and ML models that showed efficacy in calibrating accelerometers, gyroscopes, and magnetometers in IMUs by

reducing both computational complexity and calibration time. Our approach leveraged adaptive bias and scale factor correction to ensure real-time autonomous calibration, thus enhancing the accuracy and reliability of the sensor data. While our results demonstrated the potential of this approach, there are several key points to consider. First, the algorithm's performance was validated using simulated and controlled experimental data, and its effectiveness in diverse real-world scenarios remains to be evaluated. Second, the lightweight ML models used, although less computationally intensive, still required optimization to handle highly dynamic environments or extreme conditions. The hybrid approach for the accelerometer calibration necessitated the adjustment of weights in the hybrid mathematical model to allow the model to handle the autocalibration properly based on the sensor's characteristics. Additionally, retraining the model on different sensors is necessary to fine-tune the calibration rules for the DT model. Furthermore, the sensor's architecture approach requires careful adjustment of the bias registers according to the specific sensors used. Despite these considerations, the proposed calibration approach showed promising results for improving the accuracy, stability, and reliability of IMUs. Notably, the calibration approach enhanced with mathematical models can be applied to different sensors with slight adjustments, underscoring its versatility and broad applicability. Future work will involve extensive real-time testing to validate the robustness and reliability of the proposed method in various operational settings, such as clinical gait analysis and rehabilitation exercises. Additionally, further refinement of the ML models will be pursued to enhance their adaptability and performance under different conditions, such as temperature variation and usage in indoor/outdoor environments. As well as the IMUs post-calibration using our approach will be validated on human subjects during different biomechanical applications to ensure the reliability of the sensor for real-time applications.

REFERENCES

[1] T. Cudejko, K. Button, J. Willott, and M. Al-Amri, "Applications of Wearable Technology in a Real-Life Setting in People with Knee Osteoarthritis: A Systematic Scoping Review.," J. Clin. Med., vol. 10, no. 23, Nov. 2021, doi: 10.3390/jcm10235645.

[2] G. C. Ozmen, B. N. Nevius, C. J. Nichols, S. Mabrouk, C. N. Teague, and O. T. Inan, "An Integrated Multimodal Knee Brace Enabling Mid-Activity Tracking for Joint Health Assessment," in Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, 2021, pp. 7364–7368, doi: 10.1109/EMBC46164.2021.9630526.

[3] M. L. Pollind and R. Soangra, "Mini-Logger- A Wearable Inertial Measurement Unit (IMU) for Postural Sway Analysis.," Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. IEEE Eng. Med. Biol. Soc. Annu. Int. Conf., vol. 2020, pp. 4600–4603, Jul. 2020, doi: 10.1109/EMBC44109.2020.9175167.

[4] D. Kobsar and R. Ferber, "Wearable sensor data to track subject-specific movement patterns related to clinical outcomes using a machine learning approach," Sensors (Switzerland), vol. 18, no. 9, Sep. 2018, doi: 10.3390/s18092828.

[5] A. Saleh, M. Ma, A. Am, N. Ae, and A. Mo, "2 Healthcare Technology Management Administration, King Fahad Medical City, Ministry of Health," vol. 29, no. 12, pp. 2553–2560, 2018.

[6] X. Ru, N. Gu, H. Shang, and H. Zhang, "MEMS Inertial Sensor Calibration Technology: Current Status and Future Trends," Micromachines, vol. 13, no. 6. 2022, doi: 10.3390/mi13060879.

[7] X. Zhang et al., "Low-Cost Inertial Measurement Unit Calibration With Nonlinear Scale Factors," IEEE Trans. Ind. Informatics, vol. 18, no. 2, pp. 1028–1038, 2022, doi: 10.1109/TII.2021.3077296.

[8] Y. Bolotin and V. Savin, "Turntable IMU Calibration Algorithm Based on the Fourier Transform Technique," Sensors, vol. 23, no. 2, 2023, doi: 10.3390/s23021045.

[9] C. Chao, J. Zhao, J. Zhu, and N. Bessaad, "Minimum settings calibration method for low-cost tri-axial IMU and magnetometer," Meas. Sci. Technol., vol. 33, no. 2, 2022, doi: 10.1088/1361-6501/ac3ec2.

[10] J. Lu, L. Ye, J. Zhang, W. Luo, and H. Liu, "A New Calibration Method of MEMS IMU Plus FOG IMU," IEEE Sens. J., vol. 22, no. 9, pp. 8728–8737, 2022, doi: 10.1109/JSEN.2022.3160692.

[11] Z. Zou et al., "Robust Equipment-Free Calibration of Low-Cost Inertial Measurement Units," IEEE Trans. Instrum. Meas., vol. 73, pp. 1–12, 2024, doi: 10.1109/TIM.2023.3234081.

[12] M. Dong, G. Yao, J. Li, and L. Zhang, "Calibration of Low Cost IMU's Inertial Sensors for Improved Attitude Estimation," J. Intell. Robot. Syst. Theory Appl., vol. 100, no. 3–4, pp. 1015–1029, 2020, doi: 10.1007/s10846-020-01259-0.

[13] D. Tedaldi, A. Pretto, and E. Menegatti, "A robust and easy to implement method for IMU calibration without external equipments," in 2014 IEEE International Conference on Robotics and Automation (ICRA), 2014, pp. 3042–3049, doi: 10.1109/ICRA.2014.6907297.

[14] H. Zhang, Y. Wu, W. Wu, M. Wu, and X. Hu, "Improved multi-position calibration for inertial measurement units," Meas. Sci. Technol., vol. 21, p. 15107, Nov. 2009, doi: 10.1088/0957-0233/21/1/015107.

[15] R. Li, C. Fu, W. Yi, and X. Yi, "Calib-Net: Calibrating the Low-Cost IMU via Deep Convolutional Neural Network ," Frontiers in Robotics and AI , vol. 8. 2022, [Online]. Available: https://www.frontiersin.org/articles/10.3389/frobt.2021.772583.

[16] C. Chen, C. X. Lu, J. Wahlström, A. Markham, and N. Trigoni, "Deep Neural Network Based Inertial Odometry Using Low-Cost Inertial Measurement Units," IEEE Trans. Mob. Comput., vol. 20, no. 4, pp. 1351–1364, 2021, doi: 10.1109/TMC.2019.2960780.

[17] Y. Liu, J. Cui, and W. Liang, "A hybrid learning-based stochastic noise eliminating method with attention-Conv-LSTM network for low-cost MEMS gyroscope," Front. Neurorobot., vol. 16, 2022, doi: 10.3389/fnbot.2022.993936.

[18] Z. Wen, G. Yang, and Q. Cai, "An Improved Calibration Method for the IMU Biases Utilizing KF-Based AdaGrad Algorithm," Sensors, vol. 21, no. 15. 2021, doi: 10.3390/s21155055.

[19] F. Sun and T. Cao, "Fast calibration for strapdown IMU based on HMM/KF," vol. 27, pp. 567–570, Apr. 2012.

[20] M. Jafari, M. Sahebjameyan, B. Moshiri, and T. A. Najafabadi, "Skew redundant MEMS IMU calibration using a Kalman filter," Meas. Sci. Technol., vol. 26, no. 10, p. 105002, 2015, doi: 10.1088/0957-0233/26/10/105002.

[21] H. Al Jlailaty, A. Celik, M. M. Mansour, and A. M. Eltawil, "IMU Hand Calibration for Low-Cost MEMS Inertial Sensors," IEEE Trans. Instrum. Meas., vol. 72, pp. 1–16, 2023, doi: 10.1109/TIM.2023.3301860.

[22] Y. Zhai, H. Feng, and Y. Fu, "A progressive online external parameter calibration and initialization method for stereo-IMU system," Ind. Robot Int. J. Robot. Res. Appl., vol. 49, no. 2, pp. 344–358, Jan. 2022, doi: 10.1108/IR-07-2021-0135.

[23] S. Hosseinyalamdary, "Deep Kalman Filter: Simultaneous Multi-Sensor Integration and Modelling; A GNSS/IMU Case Study.," Sensors (Basel)., vol. 18, no. 5, Apr. 2018, doi: 10.3390/s18051316.

[24] J. C. Lötters, J. Schipper, P. H. Veltink, W. Olthuis, and P. Bergveld, "Procedure for in-use calibration of triaxial accelerometers in medical applications," Sensors Actuators A Phys., vol. 68, no. 1, pp. 221–228, 1998, doi: https://doi.org/10.1016/S0924-4247(98)00049-1.

[25] S. Stančin and S. Tomažič, "Time- and Computation-Efficient Calibration of MEMS 3D Accelerometers and Gyroscopes," Sensors, vol. 14, no. 8. pp. 14885–14915, 2014, doi: 10.3390/s140814885.

[26] D. R. Greenheck, R. H. Bishop, E. M. Jonardi, and J. A. Christian, "Design and Testing of a Low-Cost MEMS IMU Cluster for SmallSat Applications," AIAA/USU Conf. Small Satell., 2014.

[27] Z.-Q. Zhang and G.-Z. Yang, "Micromagnetometer calibration for accurate orientation estimation.," IEEE Trans. Biomed. Eng., vol. 62, no. 2, pp. 553–560, Feb. 2015, doi: 10.1109/TBME.2014.2360335.

[28] S. Ranjbaran, A. Roudbari, and S. Ebadollahi, "A Simple and Fast Method for Field Calibration of Triaxial Gyroscope by Using Accelerometer," J. Electr. Comput. Eng. Innov., vol. 6, no. 1, pp. 1–6, 2018, doi: 10.22061/JECEI.2018.797.