

Wearable Magnetic Induction-based Approach Toward 3D Motion Tracking

Negar Golestani (✉ golestani.negar@gmail.com)

University of Southern California

Mahta Moghaddam

University of Southern California

Research Article

Keywords: 3D motion tracking, magnetics

Posted Date: April 21st, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-441596/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Wearable Magnetic Induction-based Approach Toward 3D Motion Tracking

Negar Golestani^{1,*} and Mahta Moghaddam¹

¹Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, 90089, USA.
*golestani.negar@gmail.com

ABSTRACT

Activity recognition using wearable sensors has gained popularity due to its wide range of applications, including healthcare, rehabilitation, sports, and senior monitoring. Tracking the body movement in 3D space facilitates behavior recognition in different scenarios. Wearable systems have limited battery capacity, and many critical challenges have to be addressed to gain a trade-off among power consumption, computational complexity, minimizing the effects of environmental interference, and achieving higher tracking accuracy. This work presents a motion tracking system based on magnetic induction (MI) to tackle the challenges and limitations inherent in designing a wireless monitoring system. We integrated a realistic prototype of an MI sensor with machine learning techniques and investigated one-sensor and two-sensor configuration setups for motion reconstruction. This approach is successfully evaluated using measured and synthesized datasets generated by the analytical model of the MI system. The system has an average distance root-mean-squared error (RMSE) error of 3 cm compared to the ground-truth real-world measured data with Kinect.

Introduction

Over the past decade, monitoring and recognition of human activities have embraced a growing number of practical usages in a broad range of domains such as healthcare, rehabilitation, sports training, virtual reality (VR) gaming, human-computer interface (HCI) systems, finger tracking, daily life-logging, child and elderly care, and assistance for people with cognitive disorders or chronic conditions [1–7]. Tracking and reconstructing limb movements in 3D space facilitates more detailed evaluation, and it is crucial for the analysis and clinical understanding of complex functional movements. Studying biomechanics of human motion has application in human performance assessment, gesture/posture monitoring, behavioral recognition, gait analysis, and patients' functionality and improvement evaluation during the rehabilitation period [8–10].

There are many solutions for tracking body movement using different monitoring sources [11]. Computer vision-based methods, such as Kinect or optical motion capture (MoCap) system, are the most commonly used techniques that allow users to interact with them and collect data on the user's motion using depth sensors, color, and infrared cameras [12, 13]. However, they inherit computer vision restrictions such as light dependency, coverage limitation, and high computational cost [14, 15]. The MoCap systems require an expensive setup of infrared cameras for tracking reflective markers on an individual's body, which makes them only applicable to the laboratory environment and restricted in physical space. Besides, the markers placement and soft tissue artifacts have a considerable effect on the system accuracy [16, 17]. The RF-based solutions are another motion tracking method capturing data based on wireless signal changes (e.g., Doppler frequency shift and signal amplitude fluctuation) [18]. These methods also suffer from environmental dependency and limitation in the number of detectable gestures due to the high cost of training data collection and the lack of multi-user identification capabilities [19].

Wearable-based solutions are an alternative, cost-effective solution for applications where the optical-based methods are unsuitable. This approach tracks the user's movement based on the sensors readings placed around the human body [20]. The advancement of sensing technologies, miniaturization, embedded systems, and wireless communication systems combined with predictive models for data analysis and detection have made it possible to develop wearable devices working around the human body for continuous physical activity monitoring. Smart devices like smartphones, smartwatches, and fitness bands are becoming widespread for providing valuable insights about an individual's performance and health status. These wearables have multiple embedded physiological, inertial, and ambient sensors that enable multi-modal sensing [1]. Many studies have exploited commercial inertial measurement unit (IMU) devices comprised of accelerometers, gyroscopes, and magnetic sensors, for motion tracking based on wearable sensors. An IMU can be attached to a body segment to estimate its movement in space. By combining multiple of them on adjacent body segments, the kinematics of activities can be determined [20]. For example, [21] presents the development of a smart wearable jumpsuit with multiple built-in IMU sensors for automatic posture and movement tracking of infants. The work in [22] investigates the reliability and validity of IMUs for clinical movement analysis, and [23] presents a single wrist-worn IMU sensor for high-resolution motor state detection in Parkinson's disease. Inertial

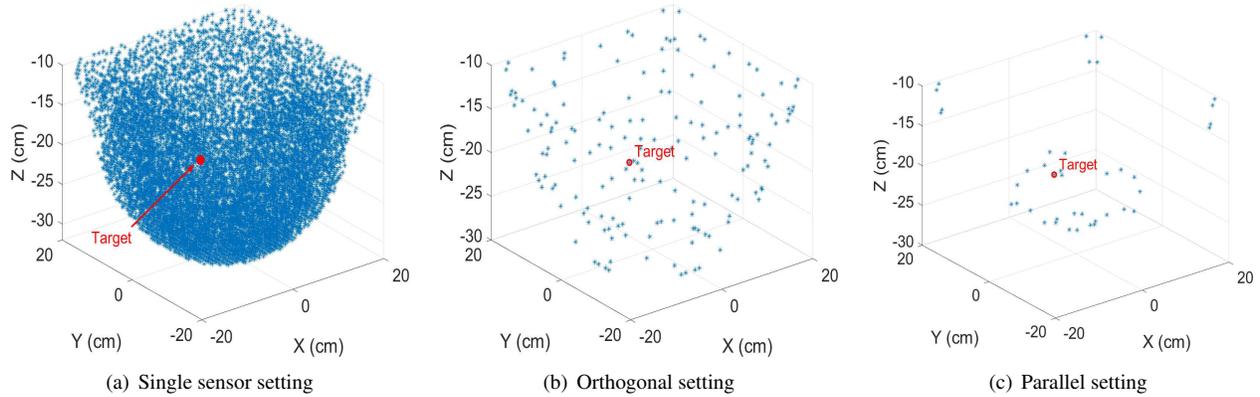


Fig. 1. Data-driven location tracking across all settings for a predefined target point.

sensing can track limb movements by integrating over sensor measurements, though it is subject to drift since the estimation errors caused by the intrinsic noise can grow unbounded with time [1].

In some applications, it is possible to achieve improved accuracy and more specific inferences by fusing the subsets of data collected from those sensors compared to single sensor modalities [24, 25]. Although these devices provide a solution for physiological health monitoring, condition assessment, and medical diagnosis, they still might face challenges. These single-node devices restrict biosensors' placement, while optimizing their position can increase the system's accuracy and robustness in monitoring vital signs (e.g., body temperature and heart rate) [1, 26]. Furthermore, in systems relying on data from a single device, variations in position can significantly affect the performance. In motion tracking applications, a single node wearable is not able to cover the entire body. Therefore it cannot get detailed information about the mobility of an individual's limbs. For example, a smartwatch's inertial sensor cannot capture the movement of the user's legs, limiting the system's ability in classifying activities [1]. A network of distributed wearable devices operating around the human body is an approach that can address these issues.

One of the biggest challenges in wearable-based motion tracking systems is to find the optimum type and number of non-invasive sensors with minimal power consumption to achieve acceptable accuracy and satisfy guidelines and constraints. In [1], we introduced a wireless system based on magnetic induction combined with machine learning techniques to detect a wide range of human activities. We showed that this system can address challenges in terms of power consumption, accuracy, coverage, privacy, and cost. Here, we investigate the capability of the magnetic induction (MI) system in 3D motion tracking and evaluate a prototype device in estimating its motion using trained machine learning-based regressors. We use a calibrated MI model presented in our previous works [27, 27] to generate synthetic MI motion data and train regressors without the need for any measured data. We then evaluate the trained model on real-world measured data.

Results

Operating Principle

The MI-based communication system is a short-range wireless physical layer that transmits signals by inductive coupling between the wire coils rather than radiating as is done in conventional methods [27, 28]. The transmitter node uses a coil to produce an oscillating magnetic field at a specific frequency. Each sensor node's (receiver) main component is a coil to capture the transmitter's generated magnetic field. According to Faraday's law, the time-varying magnetic field induces a voltage in sensor nodes proportional to the rate of magnetic flux change through their coils. For a predefined coil geometry and operating frequency below 30 MHz, where the environmental effects are negligible, the flux change rate is a function of the sensor coils' position, and orientation relative to the transmitter [1, 27]. The relationship function from spatial data into induced voltage is non-linear and surjective, and the tracking problem objective is to estimate the sensors' positions given the induced voltage measurements.

System Architecture

We used an analytical model of the MI system presented in [1, 27, 29] to calculate the induced voltage at each sensor coil given its position and orientation. This forms the basis of the data-driven backward estimation algorithm that retrieves a node's position using its observed data. It helps assess the system performance under different configurations, such as

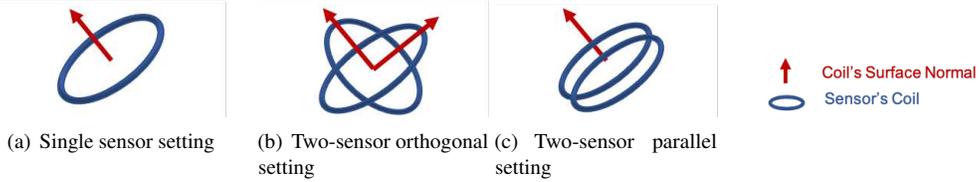


Fig. 2. Schematic representation of configuration settings used in the experiments.

changing the number or arrangement of sensor coils to find the near-optimal setup with acceptable tracking accuracy. Since the model is a function of relative distance and alignment of coils to the transmitter, we transform the coordinate system to locate the new coordinate system's origin at the center of the transmitter coil, with the coil's surface normal oriented in the z-direction. Given the sensors' spatial data, we compute the coordinate transformation matrix and calculate each coil's position and orientation in the new coordinate frame. We explored the node's position $p = (x, y, z)$ with the resolution of 1 cm, and alignment $\hat{n} = (\sin \theta \cos \phi, \sin \theta \sin \phi, \cos \theta)$ with the resolution of 5° as these resolutions are expected to satisfy the accuracy requirements for motion tracking applications [30, 31]. It also provides enough data points within the search domain for comprehensive performance analysis of the system with different configuration settings. The possible solutions, which is a unique single-point in an optimal configuration, are retrieved for a given set of observed data. The domain of the search space is defined as follows to represent the average ranges of distances where sensors can be placed for both

$$\begin{aligned}
 x &\in [-20 \text{ cm}, 20 \text{ cm}] \\
 y &\in [-20 \text{ cm}, 20 \text{ cm}] \\
 z &\in [-60 \text{ cm}, -10 \text{ cm}]
 \end{aligned}$$

male and female subjects relative to an on-body central node on their torso:

$$\begin{aligned}
 \theta &\in [0^\circ, 60^\circ] \\
 \phi &\in [0^\circ, 360^\circ]
 \end{aligned} \tag{1}$$

We studied the performance of an MI sensor (single sensor setting), where the coil can be aligned in any direction. We also adopted two-sensor configurations and investigated different alignment setups. Among these setups, we present the performance analysis of setups where coils' surface normal are aligned in the same direction (parallel setting) or perpendicular to each other (orthogonal setting). Fig. 2 depicts the configuration of sensors in each described setting. In these experiments, the induced voltage measured at the coils is used as input for location estimation. Fig. 1 shows an example result of the data-driven backward estimation algorithm, where each point represents a possible node's position with at least one alignment that can produce the given set of inputs for the defined setting. As the results display, there are many possible solutions for a single sensor setup, and this number reduces by adding another sensor. The sensor voltage data are assumed to be measured with 1 mv accuracy and given as inputs to the algorithm. A comparison between the two-sensor configurations shows that the parallel setting outperforms the orthogonal setting. Although a unique solution cannot be returned as an output, results suggest that the regression methods with proper constraints can meet the minimum required accuracy for the position tracking.

Data Collection

We designed and built an MI sensor for 3D motion tracking (see Methods section), representing the movements by variation in the MI signals received from the transmitter instead of measuring spatial data via conventional sensors such as IMUs. To evaluate the capability of the proposed MI sensor, we employed regression algorithms and investigated their performance on MI sensor's data. Validating and testing machine learning methods is critical and challenging due to the difficulty of collecting realistic valid data and the lack of labeled data. One solution is to create synthetic data for training the model, and here, we used a Variational Auto-Encoder (VAE) model to produce time-series motion data. The MI data corresponding to the generated synthetic movements are then produced using an analytical MI system model [1, 27]. The regressors are then trained on these synthesized data, which removes the need for supervised training measured data. A point to consider is that the MI system model must be calibrated only once to scale the synthetic training data to sensor measurements and tune the regression algorithm (see Methods section). The trained machine learning regressors on the synthetic data are then tested on real-world measurements and reported for comparison.

Table 1. Performance of regression models in motion tracking using synthetic data generated for the different settings.

Model		Distance			X			Y			Z		
		RMSE	MAPE	R2									
Single Sensor	ET	0.028	0.043	0.898	0.05	6.548	0.232	0.035	3.877	0.102	0.031	0.05	0.868
	RF	0.028	0.042	0.9	0.05	6.872	0.25	0.035	3.88	0.125	0.031	0.049	0.871
	KNN	0.029	0.044	0.89	0.052	6.155	0.161	0.037	3.923	0.017	0.033	0.052	0.856
	LightGBM	0.028	0.042	0.9	0.048	6.873	0.29	0.034	3.931	0.173	0.031	0.049	0.872
	MLP	0.03	0.045	0.883	0.049	6.806	0.278	0.034	3.999	0.147	0.034	0.053	0.846
	DT	0.039	0.057	0.807	0.069	10.529	-0.45	0.049	4.05	-0.708	0.043	0.067	0.753
	LR	0.04	0.066	0.796	0.052	8.079	0.177	0.035	4.095	0.138	0.041	0.069	0.769
Orthogonal	ET	0.022	0.029	0.879	0.04	4.897	0.537	0.011	0.851	0.18	0.005	0.007	0.169
	RF	0.023	0.031	0.863	0.043	5.266	0.47	0.031	2.53	0.362	0.026	0.037	0.827
	KNN	0.022	0.029	0.875	0.042	5.749	0.487	0.03	2.137	0.387	0.025	0.034	0.84
	LightGBM	0.026	0.036	0.826	0.05	5.989	0.294	0.035	3.018	0.158	0.029	0.043	0.779
	MLP	0.026	0.037	0.823	0.05	5.239	0.277	0.036	3.191	0.134	0.03	0.043	0.774
	DT	0.033	0.041	0.718	0.062	7.04	-0.096	0.044	2.71	-0.313	0.037	0.049	0.642
	LR	0.035	0.05	0.688	0.053	6.829	0.184	0.037	3.2	0.103	0.038	0.057	0.625
Parallel	ET	0.011	0.018	0.947	0.037	7.539	0.365	0.028	5.184	0.23	0.014	0.025	0.917
	RF	0.011	0.019	0.94	0.037	9.49	0.334	0.028	5.471	0.193	0.014	0.026	0.908
	KNN	0.012	0.021	0.932	0.037	10.543	0.352	0.028	4.855	0.196	0.015	0.027	0.9
	LightGBM	0.012	0.021	0.932	0.04	10.004	0.261	0.03	5.827	0.075	0.015	0.028	0.897
	MLP	0.013	0.022	0.924	0.04	9.251	0.231	0.031	6.268	0.022	0.016	0.03	0.881
	DT	0.016	0.026	0.882	0.053	14.302	-0.338	0.04	5.478	-0.636	0.02	0.035	0.818
	LR	0.021	0.037	0.804	0.041	11.848	0.185	0.031	6.015	0.048	0.023	0.043	0.762

Evaluation

We deployed machine learning regression algorithms to solve the inverse problem of estimating a node’s 3D position (x,y,z) from its sensors’ measurements in meters. The performance of several regression models, including extra trees (ET), random forest (RF), K-nearest neighbors (KNN), light gradient boosting machine (LightGBM), multi-layer perceptrons (MLP), decision trees (DT), and linear regression (LR) is compared using PyCaret [32], an open-source machine learning library in Python. The models are trained on 70% of synthetic data and then scored on the remaining data using the 10-fold cross-validation method. The metrics used for comparison are RMSE, mean absolute percentage error (MAPE), and R-squared (R^2). Before fetching data into the regressors, each feature is standardized individually, and the missing values are substituted with previous non-missing values. The processed data are then divided into fixed-length segments of 2 seconds using the sliding window technique with a 0.1 second step size.

Table 1 summarizes the performance results of all models on the synthetic data for different settings. As the results show, the moving node’s distance and position in the z-direction with respect to the transmitter coordinate frame can be tracked with competing accuracy compare to other methods using wearable sensors (e.g., accelerometer) for motion tracking [20, 31]. Adding another transmitter in a different direction to the presented MI-based system can enable the node’s motion tracking in the new direction and allows 3D positional tracking. The work in [33], for example, has reported results on tracking subjects’ arm motion using smartwatch IMU data. The results show that the system can achieve the highest accuracy when the torso is static with a median error of 8.8 cm. Moreover, [34] presents a framework for reconstructing human motion with the highest accuracy of 6 cm using four 3D accelerometers attached to the user. The work in [35] has proposed utilization of spinning linearly polarized antennas to track translation of an object attached to a passive radio frequency identification (RFID) tag array in 3D and has reported an average error of 13.6 cm. To provide a realistic assessment of real-world performance, we evaluated each of the optimal models’ tracking accuracies on measured data as well. According to the score measures reported on synthetic data, the LightGBM regressor in the single-sensor setting and the extra trees regressor in the two-sensor (orthogonal and parallel) settings outperform other models. Fig. 3 presents the evaluation measures of optimal models using the measured data for each setting. Representative samples of motion tracking in all settings are also displayed in Fig. 4. Our results indicate that the parallel setting with the optimal regression model outperforms other settings on both measured and synthetic MI data.

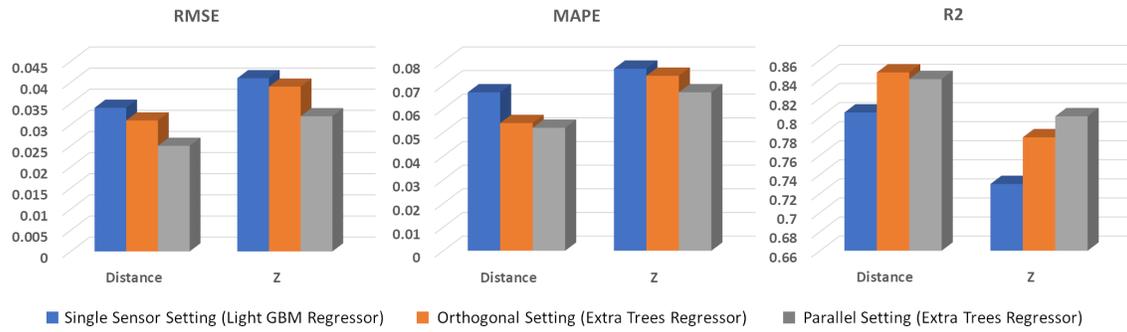


Fig. 3. Tracking performance metrics across all configuration settings on the measured motion and MI data.

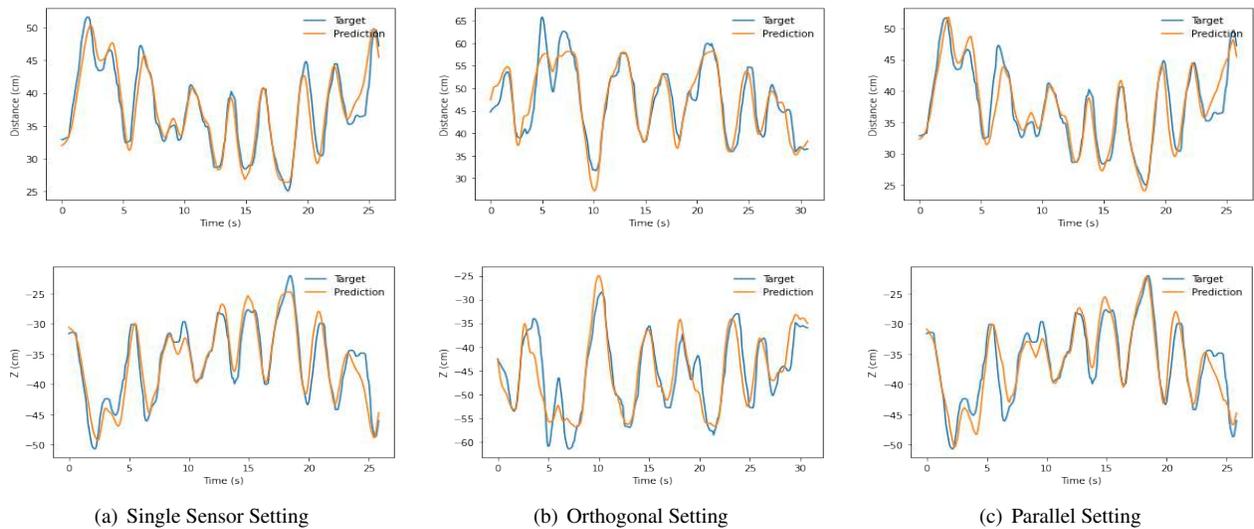


Fig. 4. Distance and motion tracking in the z-direction.

Discussion

We demonstrated an MI-based system to accurately track the motion of receivers with respect to the central transmitter node. We employed an HF RFID transmitter module equipped with a loop antenna. A simple integrated circuit is also used as a receiver, which interfaces with the central node, and records received signals sent from the transmitter using an Arduino. The main focus of this work was to provide a proof of concept for the proposed system, which can also be implemented for real-world applications by proper modifications. For example, the MI coils should be designed to be suitable for wearing on the human wrist, arm, and ankle. Furthermore, a wearable custom-designed central node capable of driving a controlled amount of current at the operating frequency through its coil is required. The receivers should cover the range of about 0.5 m to 1 m with minimum power consumption. The RF output power of the reader we used here is 1 Watt, which can be reduced by designing an efficient sensors capable of sensing lower power signals. For example, [36] presents a transceiver design exploiting the low path loss of Magnetic Human Body Communication (mHBC) communication channels toward ultra-efficient body area networking. The transmitter and receiver respectively require only 7.15 and 4.7 pJ/bit, and their design is a helpful reference for implementing MI transceivers.

One implementation approach to reduce the number of nodes with battery is to make the central node serve as both transmitter and receiver. It means that the central unit can broadcast the signal and listen back to the responses reflected from the sensors, similar to an RFID system based on passive (battery-less) tags. In an RFID system, the reader sends an interrogation signal to the transponders, which is also used to energize the tag. The tag activates and sends back its unique identifier (UID) if the received power is higher than its sensitivity [37]. A modulation resistance connected in parallel with the tag antenna switches between two different (usually conjugate matching and a short circuit) load impedances at the clock rate of the signal transmitted from the reader to modulate the backscattered signal [38]. Therefore, the central node can communicate with the tags via a secure near-field link backscattering from them. The amplitude of the demodulated signal is calculated and reported at the reader side by a value proportional to the received signal's power level, known as the received signal

strength indicator (RSSI). A point to consider is that the load modulation is not a practical solution for data transmission in an MI-based motion tracking system. The reason is that the backscattered field, and consequently, the voltage signal received by the reader, switches over two values[13]. The average power returned to the reader is no longer a direct function of distance and misalignment between coils since it varies by the number of zeros and ones in the data stream. Therefore, proper modulation and modifications are required to be able to employ existing RFID protocols.

Here we have compared the relationship between RSSI and MI signals with motion data by recording RSSI data of RFID tags in addition to the MI sensors data. We employed custom air-cored, three-layer copper coils with a 5 cm radius and 34 American wire gauge (AWG) wire diameter as the tag antenna attached to STMicroelectronics ST25DV04K RFID tag. We measured motion and RSSI data reported from the reader for 112 experiments. The calculated average R^2 and the correlation between RSSI and the tags distance from the reader are respectively 0.22 and 0.39. For an MI sensor, the calculated R^2 and correlation averaged over 110 samples are 0.53 and 0.69, respectively. These results indicate that the MI signal has a stronger relationship with its motion compared to a passive tag.

Methods

Hardware Design

The system consists of a transmitter (central) node generating an oscillating signal at 13.56 MHz. We used ISC.LRM1002 long-range RFID reader module [39] attached to ISC.ANT310/310 long-range HF antenna [39] to generate the RF signal. Since we used this setup for RFID measurements presented in the discussion, we used the same transmitter for a better comparison. The receiver node consists of MI sensors. Each sensor includes an air-cored, single-layer copper coil with a 5 cm radius and 10 AWG wire diameter to capture the transmitter's signal and measure the induced voltage. Resistance and self-inductance of the coil measured by vector network analyzer (VNA) at the resonance frequency are $101\ m\Omega$, and $241\ nH$, respectively. To improve the system efficiency, we have employed resonant inductive coupling attached to the coil. The tuning circuit can be as simple as a capacitor to tune the frequency or be a Π or T matching circuit to tune the frequency, control Q-factor, and match input and output impedances for higher power transfer [40]. Here, we used a $560\ pF$ capacitor parallel to a trimmable capacitor with the adjustable range of $3\text{-}10\ pF$ to tune the circuit to resonance accurately.

The transmitted AC signal attenuates as a function of distance and alignment of the node with respect to the transmitter antenna. To track the signal's amplitude changes, we used an envelope detector consisting of an IN5817 Schottky diode, a resistor of $1\ k\Omega$, and a capacitor of $1\ nF$. The envelope detector's output, which is the resistor's voltage, is measured by an Arduino Nano (ATmega168) microcontroller. The resolution of ADC (analog pin A1) is 10 bit for a defined measurement range. Fig. 5 depicts an MI sensor components.

Measurements

We employed a Microsoft Kinect v2 to capture the 3D position and alignment of the transmitter and the MI sensor node. The Kinect sensor consists of a depth camera, an RGB camera, and a microphone array sensor. The RGB camera and depth camera respectively provide a 1920×1080 color image, and 512×424 depth image at 30 frames per second with a resolution of a few millimeters in measure range between 0.5 m to 4.5 m [41]. The depth stream provides the sensor's distance to every point within its area of coverage. As the cameras have different pixel resolutions and are not perfectly aligned, three coordinate spaces

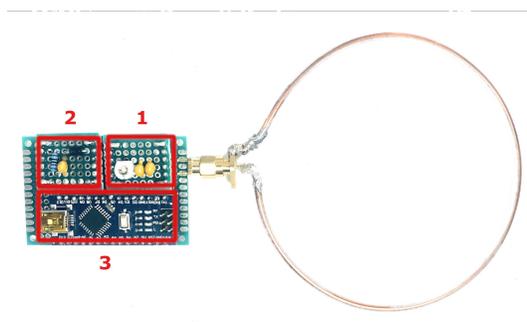


Fig. 5. The MI sensor prototype hardware contains:
 1) Variable capacitor for frequency tuning
 2) An envelope detector
 3) Arduino microcontroller for measurement.

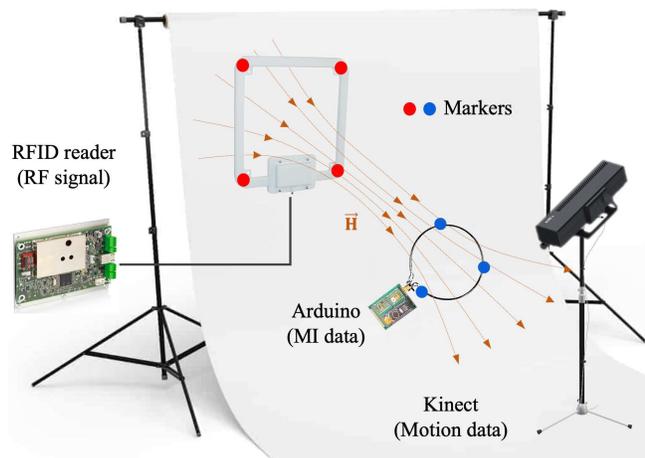


Fig. 6. Schematic representation of measurement setup.

and types are defined: color space point (x_c, y_c) , depth space point (x_d, y_d) , and camera space point (x_w, y_w, z_w) , representing a point in the color images, depth images, and real-world, respectively. The software development kit (SDK)'s mapping function can be used to map a point from one coordinate space to another.

We used colored markers to facilitate motion tracking of the devices and developed a video processing algorithm analyzing the color frames to locate pixels corresponding to the target color. The transmitter antenna and the MI node are labeled with distinct colored markers and placed in front of a white background. A threshold range is set for each color to extract pixels with the color value within the defined range. The detected pixels are classified to N_m clusters, where N_m is the number of markers, using K-means clustering methods. Then, the connected neighboring pixels of each cluster are grouped. Since the markers are colored foam balls, the circle with the minimum area enclosing each set is calculated, and the largest region is given as the target circle. The next step is mapping color to camera space to find the corresponding spatial location of each extracted color pixel. The result is a list of 3D real-world points mapped from the target circle's pixels, and each marker's location is computed by taking the median over all the calculated values. This process repeats for each new color frame that Kinect captures.

The analytical model requires the center and alignment of the transmitter and receiver coils/antennas as inputs to estimate the induced voltage. To determine a coil's surface normal, at least three markers ($M_i : i \in \{1, \dots, N_m\}$ with $N_m \geq 2$) are required. Hence, we used four red and three blue markers to track the transmitter antenna and the MI sensor node. The center of each device is calculated by averaging over its markers' location $c = \sum_{i=1}^{N_m} M_i$, and its surface normal is also calculated by the cross product of vectors passing through the markers: $\hat{n} = \vec{v}_1 \times \vec{v}_2$ where $\vec{v}_1 = M_1 - M_2$, $\vec{v}_2 = M_1 - M_3$. We applied the median filter, a non-linear digital filtering technique, to remove noise and spikes in the extracted location and alignment data.

The induced voltage, V_{ind} , at the MI sensors is measured for 30 seconds via Arduino by using a Python script that controls the recording in order to synchronize Kinect's motion data and Arduino's measurements. The sampling frequency is 100 Hz, and the reference voltage range is 0 V to 5 V, which results in the quantization interval of 5/1024 V. The data streams of the node's MI sensors are recorded and used as inputs for the regression model to estimate the device's location. The sampling rate of motion data recorded by Kinect and the sensors' data are different. Therefore, all recordings are resampled with a sampling interval of 100 ms, which also handles the missing sample values. The measurement setup of experimental measurements is presented in Fig. 6.

Synthetic Data

A VAE is based on the auto-encoder architecture and is composed of encoder and decoder networks. The encoder compresses the data into a lower-dimensional space called the latent space representation. The decoder decompresses the reduced representation code to reconstruct the original data. The VAE learns the probabilistic interpretation of these networks and generates new samples using different latent variables as input. Consider dataset $\{x^{(i)}\}_{i=1}^N$ that consists of N i.i.d. samples of some variable \mathbf{x} . VAEs assume that the data are generated by a random process with continuous latent variable, and each latent variable \mathbf{z} is related to its corresponding observation \mathbf{x} through likelihood $p_\theta(\mathbf{x}|\mathbf{z})$, where p_θ is a probability distribution with parameters θ . This probabilistic interpretation of the decoder can decode a latent (hidden) representation code into a distribution over the observation. Similarly, the encoder network returns a latent code sampled from the posterior density distribution $p_\theta(\mathbf{z}|\mathbf{x})$ given a sample from the data space [42]. While both prior $p(\mathbf{z})$ and likelihood $p(\mathbf{x}|\mathbf{z})$ can be formulated exactly, the posterior $p(\mathbf{z}|\mathbf{x})$ requires an intractable integral over the latent space. Hence, an approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$ closest in Kullback-Leibler (KL) divergence to the actual, intractable posterior distribution is considered. The approximate posterior is parameterized by variational parameters ϕ , and the training objective is a tractable lower bound to the log-likelihood [43]:

$$\log p(x) \geq \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\log \frac{p_\theta(\mathbf{x}, \mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})} \right] = \mathcal{L}(\mathbf{x}; \theta, \phi) \quad (2)$$

and can be equivalently written as:

$$\mathcal{L}(\mathbf{x}; \theta, \phi) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} [\log p_\theta(\mathbf{x}|\mathbf{z})] - \mathcal{D}_{KL}(q_\phi(\mathbf{z}|\mathbf{x}) || P_\theta(\mathbf{z})) \quad (3)$$

On the right-hand side of equation (3), the first term, reconstruction error, represents the likelihood of the model reconstructing the input data. The second term, variational regularization term, is the KL divergence and makes the approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$ to be close to $p_\theta(\mathbf{z})$. The $\mathcal{L}(\mathbf{x}; \theta, \phi)$ is a lower bound on the log probability of data $p_\theta(\mathbf{x})$, which is called evidence lower bound (ELBO). Maximizing ELBO with respect to the model parameters θ and variational parameters ϕ respectively maximizes the marginal probability $p_\theta(\mathbf{x})$ and minimizes the KL divergence [42].

We trained the model using the sensors' motion data tracked by the Kinect to produce synthetic time-series samples. After training the model, new time-series data can be generated by sampling from latent space \mathbf{z} with normal distribution parametrized by the mean and the variance [43]. The generated data include the motion of the coils' center and alignment in 3D space for a predefined sensor setting. We synthesized angular variables θ and ϕ to calculate the corresponding coil's surface normal \hat{n} that

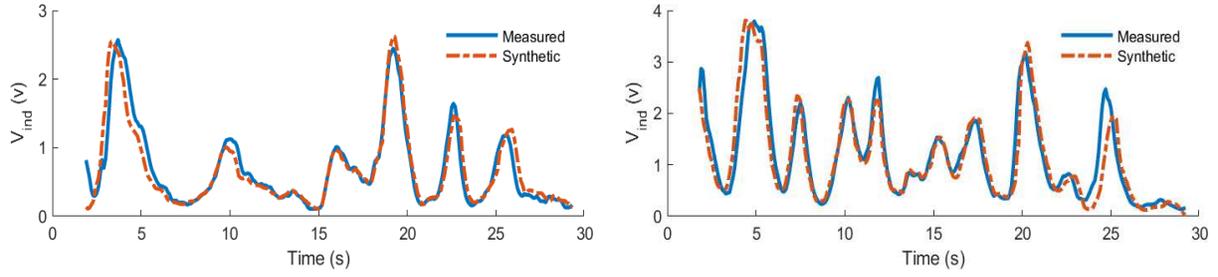


Fig. 7. The measured and simulated induced voltage at an MI sensor during two arbitrary movements, such that both relative alignment and location of the coil varies.

can be defined as $\hat{n} = (\sin \theta \cos \phi, \sin \theta \sin \phi, \cos \theta)$, where the variables θ and ϕ can take values in the range of 0-90 and 0-360 degrees, respectively.

To generate training data for the motion tracking algorithm, we synthesized the induced voltage at the MI sensor using the two-port network model of the MI system [1, 27]. The circuit model represents a forward model of the system estimating the sensor measurements given node motion data. To evaluate the circuit model's accuracy in synthesizing MI data, we fetched the captured motion data by the Kinect as inputs and estimated the corresponding induced voltage at the sensors. The data streams corresponding to MI sensors are simulated for each motion sample. The circuit model is calibrated by finding the scale and bias of the synthesized data with respect to the measurements. Considering s_i and m_i as the generated synthetic data and measurements corresponding to a motion sample, the scale a and bias b can be calculated as follows:

$$a = \frac{1}{N_s} \sum_{i=1}^{N_s} \frac{\sigma_{si}}{\sigma_{mi}} \quad (4)$$

$$b = \frac{1}{N_s} \sum_{i=1}^{N_s} \mu_{si} - \frac{\sigma_{si}}{\sigma_{mi}} \mu_{mi} \quad (5)$$

where μ_{si} , σ_{si} , μ_{mi} , σ_{mi} represent the mean and standard deviation of synthetic data and measurements corresponding to i th motion sample from N_s samples.

We have performed the experiment for 220 motions, including spatial translation and rotation ($N_s=220$). Fig. 7 shows the measured and simulated sensors' data during their movement, picked from the evaluation dataset after calibrating the model. The average normalized root-mean-squared error (NRMSE) and cross-correlation of the synthesized and measured data for all experiments are 12% and 0.91, respectively. It should be noted that the reported metrics consider not only the MI system model inaccuracy but also the error associated with the Kinect-based marker tracking algorithm and Arduino measurements. The measured motion data samples are also used for training VAE to generate synthetic motion data.

References

1. Golestani, N. & Moghaddam, M. Human activity recognition using magnetic induction-based motion signals and deep recurrent neural networks. *Nat. Commun.* **11**, 1–11 (2020).
2. Kim, J. *et al.* Wearable smart sensor systems integrated on soft contact lenses for wireless ocular diagnostics. *Nat. communications* **8**, 1–8 (2017).
3. Mukhopadhyay, S. C. Wearable sensors for human activity monitoring: A review. *IEEE Sensors J.* **15**, 1321–1330 (2014).
4. Parizi, F. S., Whitmire, E. & Patel, S. Auraring: Precise electromagnetic finger tracking. *Proc. ACM on Interactive, Mobile, Wearable Ubiquitous Technol.* **3**, 1–28 (2019).
5. Jovanov, E., Milenkovic, A., Otto, C. & De Groen, P. C. A wireless body area network of intelligent motion sensors for computer assisted physical rehabilitation. *J. NeuroEngineering Rehabil.* **2**, 6 (2005).
6. Jones, S. E. *et al.* Genetic studies of accelerometer-based sleep measures yield new insights into human sleep behaviour. *Nat. Commun.* **10**, 1–12 (2019).
7. Kim, J., Campbell, A. S., de Ávila, B. E.-F. & Wang, J. Wearable biosensors for healthcare monitoring. *Nat. Biotechnol.* **37**, 389–406 (2019).

8. Camomilla, V., Bergamini, E., Fantozzi, S. & Vannozzi, G. Trends supporting the in-field use of wearable inertial sensors for sport performance evaluation: A systematic review. *Sensors* **18**, 873 (2018).
9. Ghislieri, M., Gastaldi, L., Pastorelli, S., Tadano, S. & Agostini, V. Wearable inertial sensors to assess standing balance: A systematic review. *Sensors* **19**, 4075 (2019).
10. Prakash, C., Kumar, R. & Mittal, N. Recent developments in human gait research: parameters, approaches, applications, machine learning techniques, datasets and challenges. *Artif. Intell. Rev.* **49**, 1–40 (2018).
11. Ramasamy Ramamurthy, S. & Roy, N. Recent trends in machine learning for human activity recognition—a survey. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **8**, e1254 (2018).
12. Bux, A., Angelov, P. & Habib, Z. Vision based human activity recognition: a review. In *Advances in Computational Intelligence Systems*, 341–371 (Springer, 2017).
13. Zhang, S. *et al.* A review on human activity recognition using vision-based method. *J. healthcare engineering* **2017** (2017).
14. Lin, Q. *et al.* Beyond one-dollar mouse: A battery-free device for 3d human-computer interaction via rfid tags. In *2015 IEEE Conference on Computer Communications (INFOCOM)*, 1661–1669 (IEEE, 2015).
15. Zhang, H.-B. *et al.* A comprehensive survey of vision-based human action recognition methods. *Sensors* **19**, 1005 (2019).
16. Poppe, R. A survey on vision-based human action recognition. *Image Vis. Comput.* **28**, 976–990 (2010).
17. Weygers, I. *et al.* Inertial sensor-based lower limb joint kinematics: A methodological systematic review. *Sensors* **20**, 673 (2020).
18. Zhao, M. *et al.* Rf-based 3d skeletons. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, 267–281 (2018).
19. Wang, C. *et al.* Rf-kinect: A wearable rfid-based approach towards 3d body movement tracking. *Proc. ACM on Interactive, Mobile, Wearable Ubiquitous Technol.* **2**, 1–28 (2018).
20. Filippeschi, A. *et al.* Survey of motion tracking methods based on inertial sensors: A focus on upper limb human motion. *Sensors* **17**, 1257 (2017).
21. Airaksinen, M. *et al.* Automatic posture and movement tracking of infants with wearable movement sensors. *Sci. reports* **10**, 1–13 (2020).
22. Al-Amri, M. *et al.* Inertial measurement units for clinical movement analysis: reliability and concurrent validity. *Sensors* **18**, 719 (2018).
23. Pfister, F. M. *et al.* High-resolution motor state detection in parkinson’s disease using convolutional neural networks. *Sci. reports* **10**, 1–11 (2020).
24. Garcia-Ceja, E. *et al.* Mental health monitoring with multimodal sensing and machine learning: A survey. *Pervasive Mob. Comput.* **51**, 1–26 (2018).
25. Tannous, H. *et al.* A new multi-sensor fusion scheme to improve the accuracy of knee flexion kinematics for functional rehabilitation movements. *Sensors* **16**, 1914 (2016).
26. Ha, S. *et al.* The placement position optimization of a biosensor array for wearable healthcare systems. *J. Mech. Sci. Technol.* **33**, 3237–3244 (2019).
27. Golestani, N. & Moghaddam, M. Theoretical modeling and analysis of magnetic induction communication in wireless body area networks (wban). *IEEE J. Electromagn. RF Microwaves Medicine Biol.* **2**, 48–55 (2018).
28. Golestani, N. & Moghaddam, M. Communication system design for magnetic induction-based wireless body area network. In *USNC-URSI Radio Science Meeting (Joint with AP-S Symposium)*, 49–50 (IEEE, 2017).
29. Golestani, N. & Moghaddam, M. Gain analysis of in-motion magnetic inductive coupling-based systems. In *2020 IEEE International Symposium on Antennas and Propagation and North American Radio Science Meeting*, 1677–1678 (IEEE, 2020).

30. Schepers, M., Giuberti, M., Bellusci, G. *et al.* Xsens mvn: Consistent tracking of human motion using inertial sensing. *Xsens Technol* 1–8 (2018).
31. Roetenberg, D., Luinge, H. & Slycke, P. Xsens mvn: Full 6dof human motion tracking using miniature inertial sensors. *Xsens Motion Technol. BV, Tech. Rep* **1** (2009).
32. Ali, M. *PyCaret: An open source, low-code machine learning library in Python* (2020). PyCaret version 2.1.
33. Shen, S., Gowda, M. & Roy Choudhury, R. Closing the gaps in inertial motion tracking. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, 429–444 (2018).
34. Tautges, J. *et al.* Motion reconstruction using sparse accelerometer data. *ACM Transactions on Graph. (ToG)* **30**, 1–12 (2011).
35. Wang, C. *et al.* Spin-antenna: 3d motion tracking for tag array labeled objects via spinning antenna. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, 1–9 (IEEE, 2019).
36. Park, J. & Mercier, P. P. A sub-10-pj/bit 5-mb/s magnetic human body communication transceiver. *IEEE J. Solid-State Circuits* **54**, 3031–3042 (2019).
37. Buffi, A., Michel, A., Nepa, P. & Tellini, B. Rssi measurements for rfid tag classification in smart storage systems. *IEEE Transactions on Instrumentation Meas.* **67**, 894–904 (2018).
38. Finkenzeller, K. Battery powered tags for iso/iec 14443, actively emulating load modulation. In *RFID SysTech 2011 7th European Workshop on Smart Objects: Systems, Technologies and Applications*, 1–8 (VDE, 2011).
39. Feig company. <https://www.feig-electronics.com/>. Accessed: 2020-09-20.
40. Golestani, N. & Moghaddam, M. Improving the efficiency of magnetic induction-based wireless body area network. In *IEEE International Microwave Biomedical Conference (IMBioC)*, 166–168 (IEEE, 2018).
41. Lachat, E., Macher, H., Landes, T. & Grussenmeyer, P. Assessment and calibration of a rgb-d camera (kinect v2 sensor) towards a potential use for close-range 3d modeling. *Remote. Sens.* **7**, 13070–13097 (2015).
42. Kingma, D. P. & Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).
43. Laptev, N. Anogen: Deep anomaly generator. In *Outlier Detection De-constructed (ODD) Workshop. Available online: https://research.fb.com/publications/anogen-deep-anomaly-generator/(accessed on 20 August 2018)* (2018).

Acknowledgements

The authors would like to thank Dr. Danilo Brizi for his helpful advice on the RFID antenna design discussed in this paper.

Author contributions

N. G. was the main contributor to this work and was responsible for developing and implementing the methods, data generation, and analysis; M. M. supervised the research.

Figures

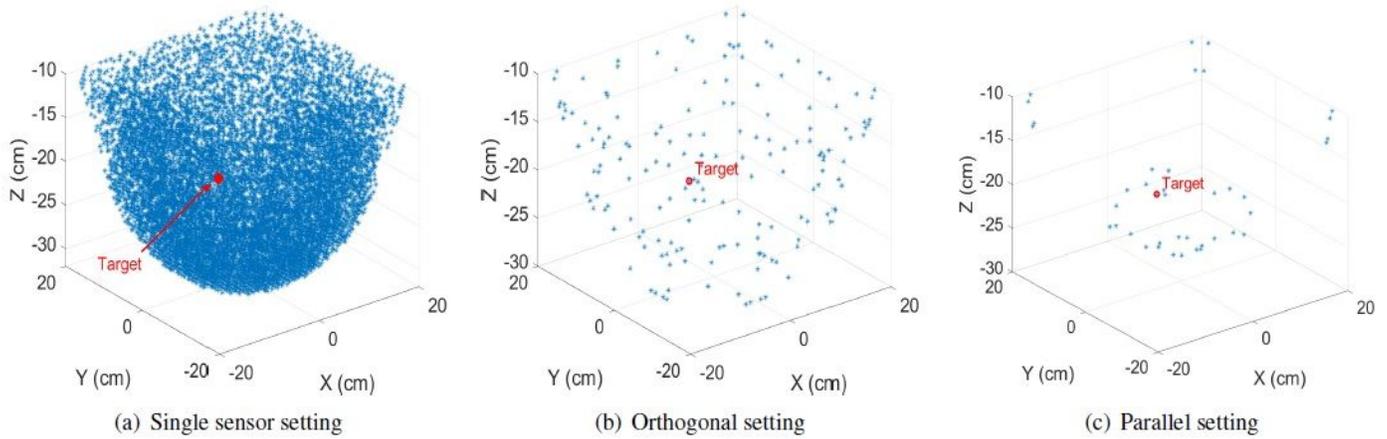


Figure 1

Data-driven location tracking across all settings for a predefined target point.

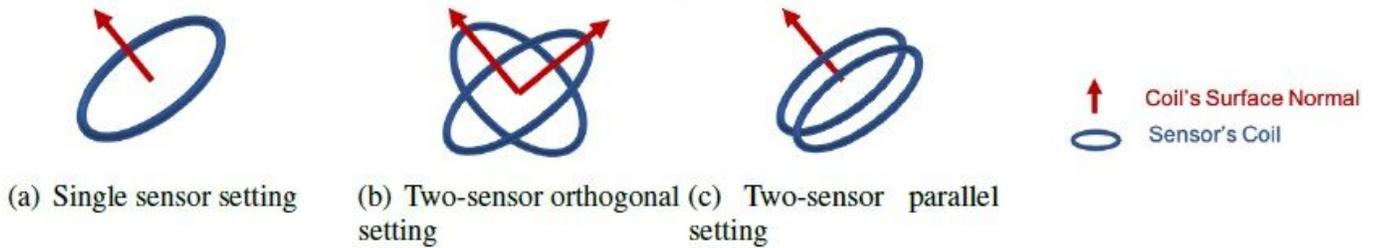


Figure 2

Schematic representation of configuration settings used in the experiments.

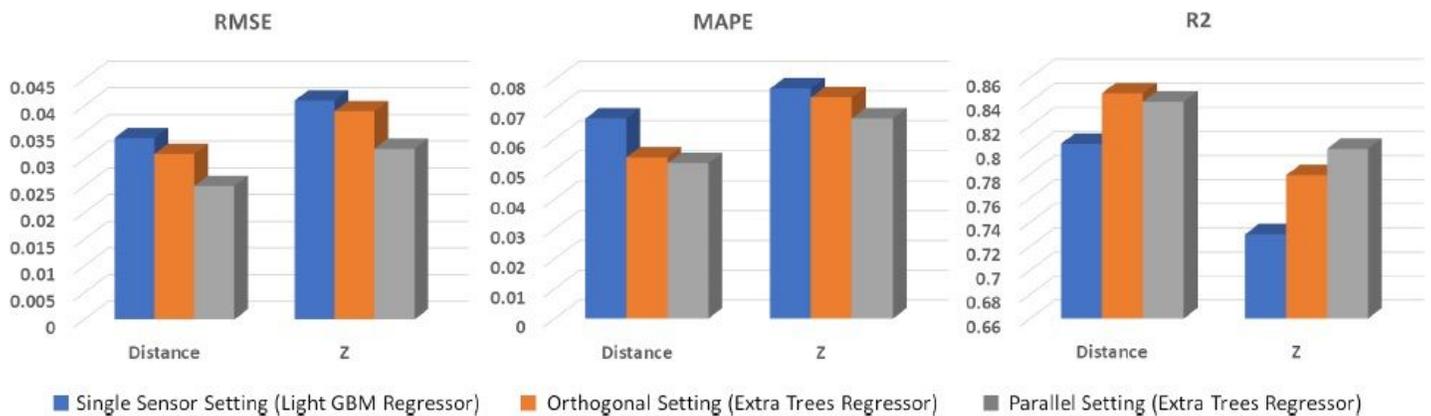


Figure 3

Tracking performance metrics across all configuration settings on the measured motion and MI data.

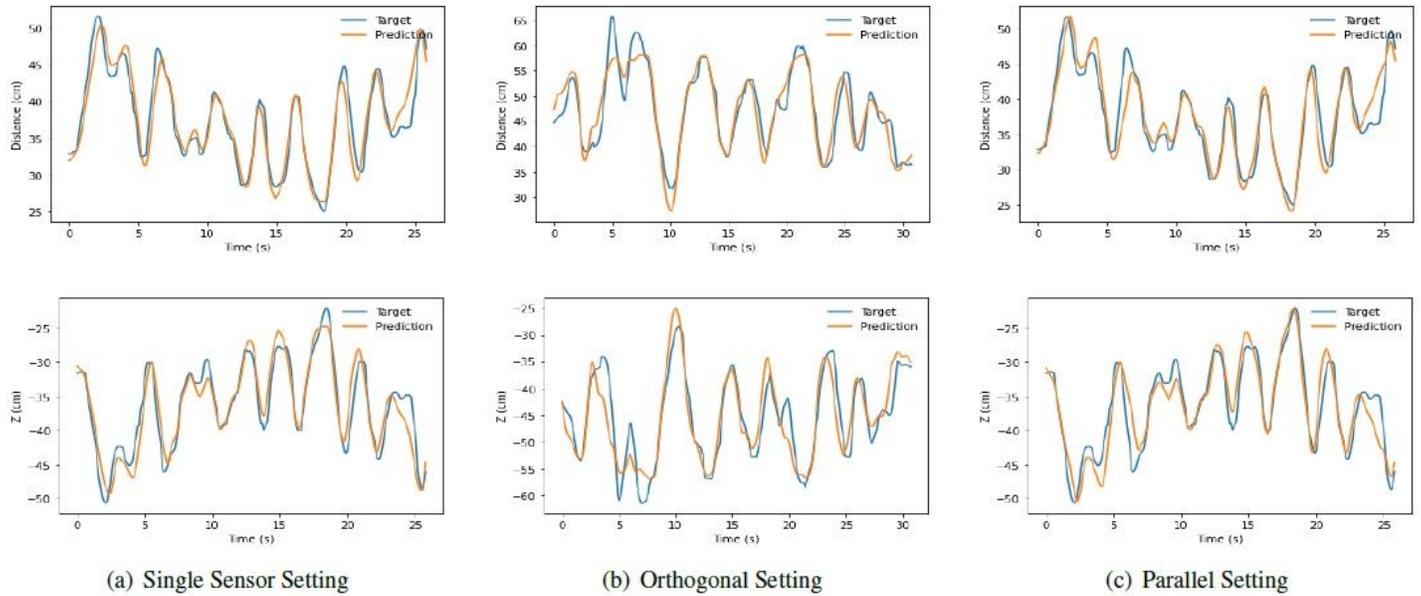


Figure 4

Distance and motion tracking in the z-direction.

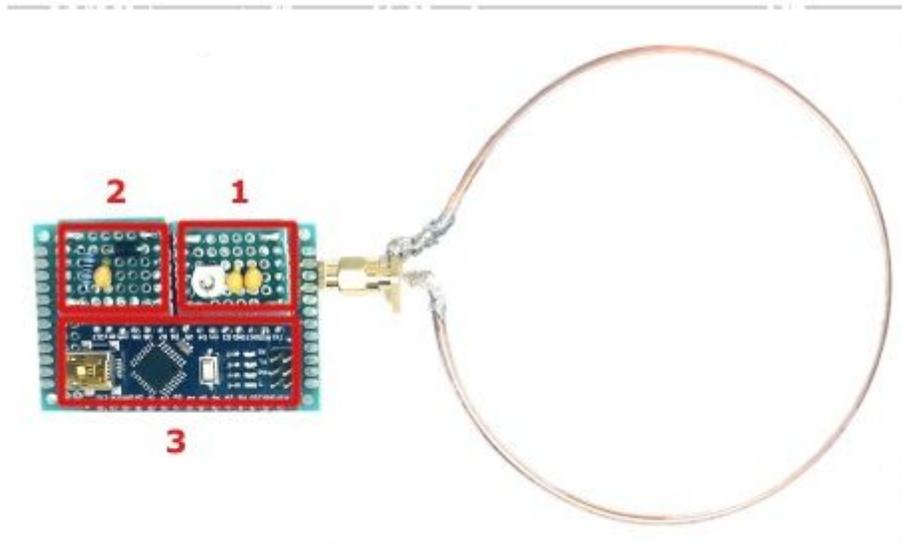


Figure 5

The MI sensor prototype hardware contains: 1) Variable capacitor for frequency tuning 2) An envelope detector 3) Arduino microcontroller for measurement.

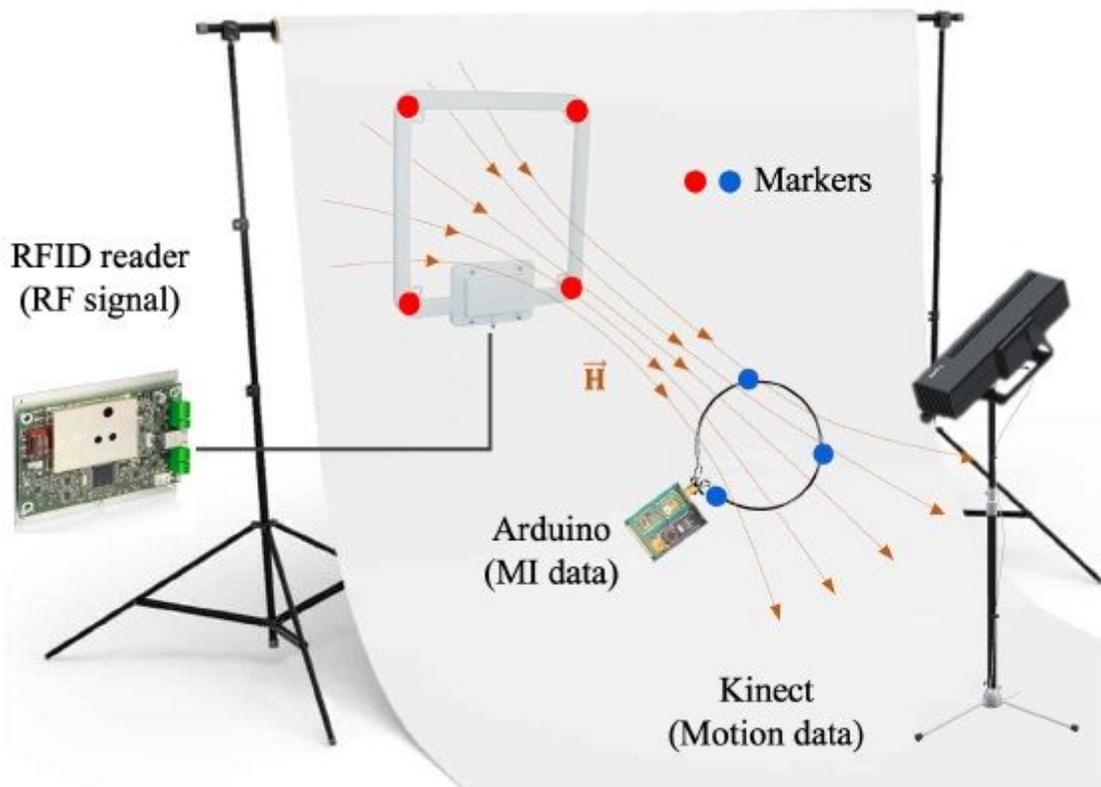


Figure 6

Schematic representation of measurement setup.

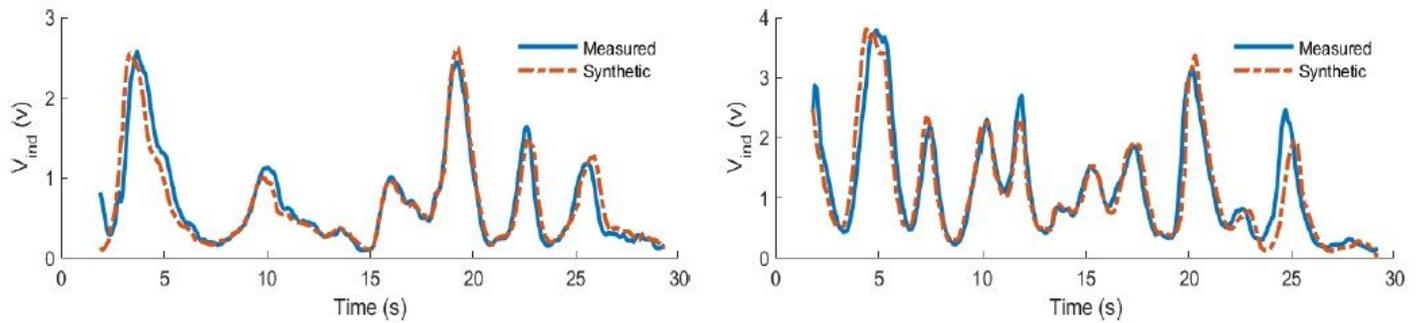


Figure 7

The measured and simulated induced voltage at an MI sensor during two arbitrary movements, such that both relative alignment and location of the coil varies.