A self-calibration algorithm for soil moisture sensors using Deep Learning

Diego Aranda[1*], Alejandro Tapia[1] and Pablo Millán[1]

[1]Department of Engineering, Universidad Loyola Andalucía, Avda. de las Universidades, s/n, Dos Hermanas, 41704, Seville, Spain.

*Corresponding author(s). E-mail(s): daaranda@uloyola.es;
Contributing authors: atapia@uloyola.es; pmillan@uloyola.es;

Abstract

In the current era of smart agriculture, accurate measurement of soil moisture has become a crucial factor for optimizing irrigation systems, significantly impacting efficient water use and crop yields. However, current soil moisture sensor technologies face challenges in terms of accuracy, often leading to inefficient irrigation practices. Calibration of these sensors, critical to improving their accuracy, is hampered by conventional methods that require extensive ground reference data, resulting in costly and impractical processes. This study introduces an innovative self-calibration method for soil moisture sensors, leveraging deep learning capabilities. The method proposed focuses on a novel strategy that requires only two characteristic points for calibration, the saturation and field capacity. Using deep learning algorithms, allowing for effective and accurate self-calibration of the sensors in-situ. This method was tested using a large dataset of simulated erroneous sensor readings generated using simulation software. The results obtained show that the method not only significantly improves the accuracy of soil moisture measurements, with 84.83 % of sensors showing an improvement in accuracy, but also provides a more agile and cost-effective implementation compared to traditional approaches. This advance represents an important step towards more efficient and sustainable agricultural practices, providing farmers with a valuable tool for optimal water and crop management, and highlighting the potential of deep learning in solving complex engineering problems.

Keywords: self-calibration, soil moisture sensors, deep learning

1 Introduction

Soil moisture represents a critical parameter in the field of agriculture, hydrology, and environmental studies, and plays an important role in managing soil health, plant growth, and water resource allocation, among many others. It is through the precise monitoring of soil moisture that the optimization of irrigation systems is achieved, ensuring efficient water use and the minimization of nutrient loss through leaching.

Understanding this fundamental role of soil moisture underlines the need for accurate measurement techniques. The amount of water in the soil is usually quantified in terms of the volumetric water content (usually referred to as VWC), θ, and is formally defined as

$$\theta = \frac{V_w}{V_{wet}} \tag{1}$$

where V_w represents the volume of water contained within a soil sample, and V_{wet} is the total volume of that soil sample [1]. Accurate knowledge of the water content in the soil is essential to efficiently apply irrigation control systems [2]. A wide range of methods and strategies exist for estimating soil moisture, which are generally classified into direct and indirect methods. Direct methods consist of physically measuring soil water content. This method finds frequent application in both research and field analysis, although it typically presents significant drawbacks in terms of duration and efficiency [3]. In this approach the water content is determined through the weight difference of a soil sample after removing its water content in an oven. The sample is usually dried in an oven at a specific temperature, typically around 105°C (221°F), being the water considered to have fully evaporated after a drying period of 24 hours or until weight variation observed is negligible.

On the other hand, indirect methods infer soil moisture on the basis of its influence on other soil or environmental properties. These methods provide real-time insights into soil dynamics and water availability, facilitating decisions in agriculture, land management, and climate modeling. Among the indirect methods, a wide variety of sensors are available, such as capacitive, resistive, frequency-domain reflectometers (FDR), time-domain reflectometers (TDR) or neutron probes, to mention a few.

In the current context of rise of IoT technologies, together with the expansion of smart agriculture, there has been an increasing demand for efficient and effective sensors. However, their effectiveness depends not only on their accuracy, but also on their ease of installation, maintenance and, above all, their cost. Farmers and resource managers are willing to employ "plug-and-play" devices, that are easy to install and operate without much prior technical knowledge. They also need to be easy to maintain to ensure proper functioning over time. The other critical factor is cost, as for these technologies to be accessible and deployed on a large scale, they need to be affordable. In addition, this would enable a much greater spatial representativeness of the terrain as a larger amount of terrain can be held, making the system more resilient to specific factors that may affect its performance [4]. Capacitive and resistive sensors constitute a possible solution to these challenges, given their low-cost, ease of use and low consumption and maintenance requirements [5]. As a matter of fact, during the

last years the demand of these devices has rapidly increased for their use in wireless sensor networks (WSN) [4].

However, although some sensor manufacturers provide calibration equations that relate sensor output to the value of water content, these equations are not universal, and are strongly conditioned by the soil type and environmental context [6]. Calibration refers to the determination of the proper mathematical equations, generally referred to as calibration curves, that relate the raw sensor measurements and the actual soil moisture measured [7]. With independence of type of sensor, calibration is crucial for the correct performance of the sensors, especially in dynamic environmental conditions, where variations in soil properties and climate can introduce significant uncertainties.

The most commonly used method for calibration is to obtain a correlation curve of the raw sensor measurements on soil samples of known water content values, normally measured using the direct or oven-drying method [1, 8–10]. Nevertheless, this process has a number of difficulties, not only related to the large amount of time involved in the calibration process, but also related to soil spatial variability. Different types of soil exhibit unique physical and chemical properties, including variations in texture, organic matter content, electrical conductivity, and mineral composition. In consequence, the relationship between sensor output and the actual soil moisture is highly soil-specific. Therefore, in practice, researchers often find it necessary to establish soil-specific calibrations for each distinct type of soil or soil profile encountered in their study area [5]. In addition, other factors, such as soil compactness, vegetation or the presence of localized heterogeneity, can cause the sensor to yield different outputs for the same level of soil moisture [11].

For this reason, a proper calibration method would not just be soil specific, but also site specific. Obviously, ensuring that the calibration procedure is tailored to the unique characteristics of the specific location of each sensor translate into extremely expensive calibration routines for large networks of sensors, in terms of time and resources, and also into the requirement of performing new calibrations if either the sensors are moved, or even if events occur that could alter the soil properties in the sensor's environment, such as the passage of machinery that can either displace or compact the soil.

Recognizing the aforementioned challenges and costs associated with individualized, site-specific calibration, there is a need for a more agile and adaptable solution. The limitations of traditional calibration techniques, like the oven driving method, are particularly pronounced when dealing with large sensor networks or when dealing with changing environmental conditions. These conventional methods become impractical due to the time, manpower and resources required to re-calibrate them frequently. Against this backdrop, the focus is on a method that retains the rigor of accurate calibration while introducing the flexibility and efficiency needed for large-scale deployments and dynamic conditions.

In this paper, a self-calibration method that enables the fast calibration of sensors deployed at the measurement site it is proposed. This method improves sensor measurements as it conducts an in-situ calibration. It is proposed that by taking into

account two characteristic points of the soil moisture curve: saturation and field capacity, and by being able to predict the behavior of the sensors using a neural network, it is possible to find the correction coefficients of measurement of the sensors. The main contributions of this paper are (i) the development of a novel strategy for soil moisture sensor self-calibration using only two characteristic points, and (ii) the implementation of a Deep Learning-based approach to determinate the field capacity value estimated by the sensors on an automatic basis to self-calibrate the sensors.

The rest of the paper is organized as follows: Section 2 offers an review of related works, including those related to different types of calibrations, Section 3 describes the thresholds or characteristic points of soil moisture, which are then used by the method, Section 4 describes the methodology used for the two-point calibration, including a description of the method to determine saturation and field capacity, Section 5 offers the details of the implementation of the method on a simulated sensor dataset generated by the Hydrus software, finally Section 6 includes the conclusions and future work.

2 Related Works

In the literature, self-calibration generally refers to the calibration strategies which do not rely on a controlled environment, and that must be performed on the field, when the sensor is already in operation after deployment. The different approaches on which self-calibration can be performed are generally classified in terms of the availability of ground truth data [12]:

- **Non-Blind Calibration:** Utilizes known reference data or *ground truth* along with sensor measurements for parameter adjustment. For example, readings from high-quality sensors could be used as the reference.
- **Semi-Blind Calibration:** A middle-ground approach that uses some ground truth data but not to the extent as in Non-Blind Calibration. Accurate or approximate information about the location of some nodes could be used for calibration.
- **Blind Calibration:** No ground truth is required. The system self-adjusts based on sensor measurements and inherent relationships among them. Discrepancies between closely located sensors can be used for adjustment.

Several works have been carried out on calibration methods for soil moisture and air quality sensors that could be considered non-blind. These works often employ data-driven methods, including machine learning or deep learning methods such as dense neural networks, multivariate adaptive regression splines (MARS), Gaussian process regression (GPR), feedforward back propagation neural network (FFBP-NN), etc, to find the relationship between the sensor output and the true value, always using a reference ground true, which can be that of a sensor reference or some direct measurement method, as in the case of oven drying for humidity measurement [6, 13–16].

In the field of semi-blind calibration, numerous studies focus on soil moisture estimation by using a combination of data sources. These include "true" data from high-precision reference sensors, as well as less precise data such as satellite readings,

results from not very precise sensors, and predictions from numerical models [17–21]. This combined approach allows for a nuanced calibration process, leveraging precise and less precise data to improve sensor performance. These works use deep learning and machine learning methods like Support Vector Machine (SVM), convolutional long short-term memory (ConvLSTM) and long short-term memory (LSTM).

Regarding blind calibration methods, there are works in the literature that address the problem using different methodologies for sensor calibration. In [22] it is showed that as long as the sensors slightly oversample the signals of interest, unknown sensor gains can be perfectly recovered. The calibration parameters are recovered with a projection of the measurement vector to the orthogonal complement of the measurement space, where the ground truth data vector is located. In [23], this idea is extended with a method that uses total least squares (TLS) estimation to provide a solution that is robust to misspecification of the signal subspace.

Other research investigating mobile sensors takes approach of data collected from various sensors within the same location. In [24] introduces an approach for blind calibration in densely deployed mobile sensor networks. The proposed scheme leverages mutual calibration relationships between sensors to enhance measurement accuracy. By utilizing interactions among sensors, the scheme significantly improves calibration precision compared to traditional calibration methods. In [25] it is introduced the Density Guided Calibration (DGC) scheme for calibrating nonlinear mobile sensors using piecewise linear functions. It utilizes the idea that sensors moving in the same area accumulate similar true values over time, offering a straightforward optimization-based solution to the nonlinear calibration problem. These methods may be constrained in less dense networks or sensors with limited movement.

Some other works takes advantage of sparse signals. In [26] the authors explores the problem of calibrating sensors when the measurements are distorted by unknown gains, assuming the signals are sparse. The paper presents convex optimization solutions to estimate unknown gains and sparse signals jointly. Similarly, in [27] sparse signals are assumed. The authors propose a blind calibration method that relies on measures performed on a few unknown but sparse signals. The method reparameterizes the problem to allow for a convex formulation and introduces a normalization constraint to avoid trivial solutions. This enables the use of off-the-shelf algorithms to numerically solve the calibration problem.

Other methods used for blind calibration range from probabilistic models to deep learning technique In [28] it is proposed a Multi-Output Gaussian Process (MOGP) in a two-phase approach: learning and calibration. MOGP models the sensor signal and helps to estimate drift, especially in long-term trends. In [29] proposes a deep learning method called Projection-Recovery Network (PRNet) for blind calibration of sensor networks. PRNet projects drifted sensor data to a feature space and uses a deep convolutional neural network to recover drift-free measurements.

Much of the previous work is challenging to apply to soil moisture sensors due to the non-sparse nature of humidity, which can be considered as a continuous signal. Moreover, the variability of humidity estimation by sensors due to environmental or sensor wear factors indicates that the signal subspace is not always constant over time. Additionally, the signals are influenced not only by gain, but also by an offset. Among

other factors, it should be noted that the phenomenon may not have been oversampled due to great spatio-temporal variation of humidity and limited sensor deployment and also that there may be a lack of communication between sensor.

3 Characteristic soil moisture thresholds

The strategy proposed is based on the identification and analysis of certain phenomena involved in the dynamics of water in porous soils, here defined as characteristic soil moisture thresholds, that are strongly related to the soil properties. These situations, schematized in Fig. 1, are the soil saturation, field capacity, wilting point and dry soil:

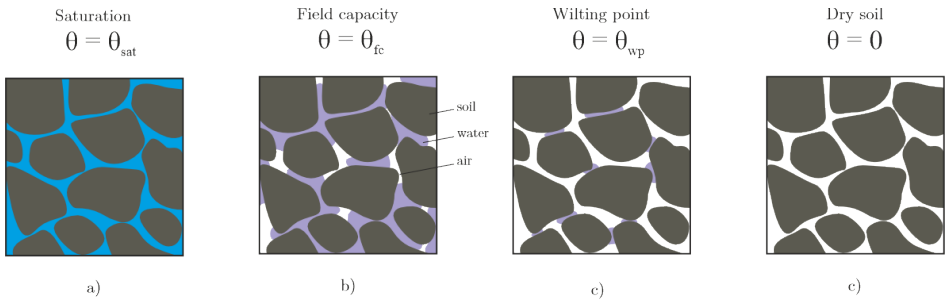

Fig. 1 Scheme of the characteristic VWC values of a soil: a) Saturation, b) Field Capacity, c) Wilting Point and d) dry soil.

Soil saturation is referred to as the threshold moisture value, θ_{sat}, at which all the soil pores are completely filled with water (Fig. 1-a). This threshold value, θ_{sat}, is a constant of the soil parameters, and thus can be considered well known for the soil under study, which means that, if saturation is observed to happen, a relationship between the raw moisture reading and the known saturation moisture constant can be established. In practical terms, detecting saturation corresponds to the identification of an upper limit in soil moisture readings, that remains consistent even during a substantial water input. This allows an automatic identification through a exhaustive scrutiny of raw moisture readings.

The field capacity (FC) is a vital hydraulic characteristic of soils. After excess water has drained away from saturated soil under the action of gravity, a certain amount of water is retained by the water-soil tensional forces (Fig. 1-b), being the VWC approximately constant [30]. The value of this moisture, θ_{fc}, can be determined by direct field observations of the soil water content after irrigation or rainfall for 2 to 3 days until the soil water content is unchanged. It can also be determined by looking at the soil matric potential which represents the energy required to extract water from the soil to a reference state, which is typically free or saturated water and is usually measured in units of pressure. Commonly the potentials at which the soil is at the FC are considered to be - 1/20, - 1/10, and - 1/3 bar in the UK, the Netherlands, and the USA, respectively [31]. As in saturation, FC constitutes an intrinsic characteristic of the particular soil under investigation, and thus can be considered to be a known constant of the soil under study. Consequently, when soil FC is encountered, it enables

the establishment of a relationship between the raw measured moisture values and the well-established FC moisture constant.

Despite of the gravity forces not being able to remove the moisture retained at the FC, the evapotranspiration caused by plant roots during photosynthesis is capable of extracting an additional amount of water from the soil, up to a threshold θ_{wp}, named permanent wilding point, where the roots are no longer capable. It is relevant to note that this θ_{wp} not only depends on the soil properties (such as the texture), but also on the plant [32]. Finally, through oven drying, the soil can be completely dried , reaching $\theta = 0$, where all the soil pores are filled with air (Fig. 1-d).

4 Two points-based self-calibration approach

This section introduces the general methodology for the blind calibration of soil moisture sensors, in which the identification of two characteristic moisture thresholds allows to build a set of linear equations that permit the determination of the experimental calibration coefficients.

4.1 Mathematical framework

The approach is based on the initial assumption that the drift error in uncalibrated sensors can be modeled using a linear relation as follows:

$$Y = \alpha \hat{Y} + \beta \tag{2}$$

where Y and $\hat{Y}$ are, respectively, the actual soil moisture and the soil moisture observed by the sensor, and α and β are the experimental coefficients whose estimation is the main objective of the algorithm. Assuming saturation is reached and the field capacity condition is met after a certain period, the equations below can be written, with alpha and beta remaining constant over time:

$$Y_{sat} = \alpha \hat{Y}_{sat} + \beta \tag{3}$$

$$Y_{fc} = \alpha \hat{Y}_{fc} + \beta \tag{4}$$

where $\hat{Y}_{sat}$ and $\hat{Y}_{fc}$ are the sensor uncalibrated measurements at saturation and field capacity, respectively, while Y_{sat} and Y_{fc} represent the actual values of soil moisture at saturation and field capacity. If all of these values are known, the coefficients alpha and beta can be calculated by solving the system formed by the equations 3 and 4. The values of Y_{sat} and Y_{fc} can be estimated or measured for the soil under consideration using high precision sensors in the field or under laboratory conditions. Although $\hat{Y}_{sat}$ can be easily determined by carefully evaluating the sensor readings after a significant level of irrigation and/or precipitation, after which saturation is clearly evidenced by the establishment of a constant moisture level, the determination of $\hat{Y}_{fc}$ is more challenging, as it involves a drying process that generally requires several days without any water supply (such as irrigation or precipitation). Not only this is time-consuming and costly due to the high price of the required equipment (such as high accuracy tensiometers), but it is impractical for field applications during production.

The proposed self-calibration method is designed to rapidly and effectively predict the calibrated output $\hat{Y}_{fc}$ by leveraging a deep learning framework. This method utilizes a sequence of raw, uncalibrated sensors readings as input to the model. A significant benefit of this technique is its efficiency in time; it eliminates the need to wait for the soil to reach field capacity before estimating $\hat{Y}_{fc}$. Instead, the model only requires analysis of the r uncalibrated sensors readings collected at the moment when the soil transitions from being fully saturated to starting its moisture loss. Subsequently, by estimating the missing value $\hat{Y}_{fc}$ within the system of equations presented in 3 and 4, it becomes feasible to determine the coefficients alpha and beta. This process effectively corrects the drift error, significantly improving the system's accuracy once these coefficients are determined.

Figure 2 depicts the scenario showing, as an example, the time series of actual moisture and readings from a sensor that has experienced drift in a soil layer. The figure shows the saturation plateaus alongside the actual and estimated moisture values measured by the sensor, including the actual and estimated field capacity values. The sensor measurements used as inputs to the model for predicting field capacity are highlighted in red. To obtain this input sequence, the algorithm needs to identify the moment when the saturation plateau ends, evaluating a sequence of measurements from a rolling window of length $q + r$, where q is the window size for the saturation detection and r is the window size for FC detection. The analysis is performed in three stages: (i) the identification of the saturation through the analysis of the first q measurements in the window (allowing the determination of $\hat{Y}_{sat}$), (ii) the identification of the abrupt moisture decrease during s measurements after the saturation end, and (iii) the prediction of $\hat{Y}_{fc}$ by utilizing the subsequent r readings after saturation concludes as inputs to the deep learning mode. A general flowchart of the algorithm is shown in Fig. 3.

4.2 Detecting the end of the saturation plateau

The identification of the moment when the saturation plateau ends is critical to obtain the values of $\hat{Y}_{sat}$ and the r-length sequence necessary to predict $\hat{Y}_{fc}$. This is performed through the analysis of the $q + s$-sized rolling window (Fig.2) through the verification of a set of numerical criteria. Given the simplicity of the saturation phenomena, which is characterized by a series of constant moisture readings that do not vary under water supply, a set of three conditions have been determined through trial and error. The moment of interest is therefore considered to have just happened if the following three conditions are simultaneously satisfied:

1. **Negligible moisture variation in time:** This condition was designed to identify periods where the soil moisture reaches a stable plateau, indicating moisture equilibrium. It is quantitatively assessed by comparing the average rate of moisture change to a predefined threshold, τ_1. This is implemented using forward finite differences with the first q readings as:

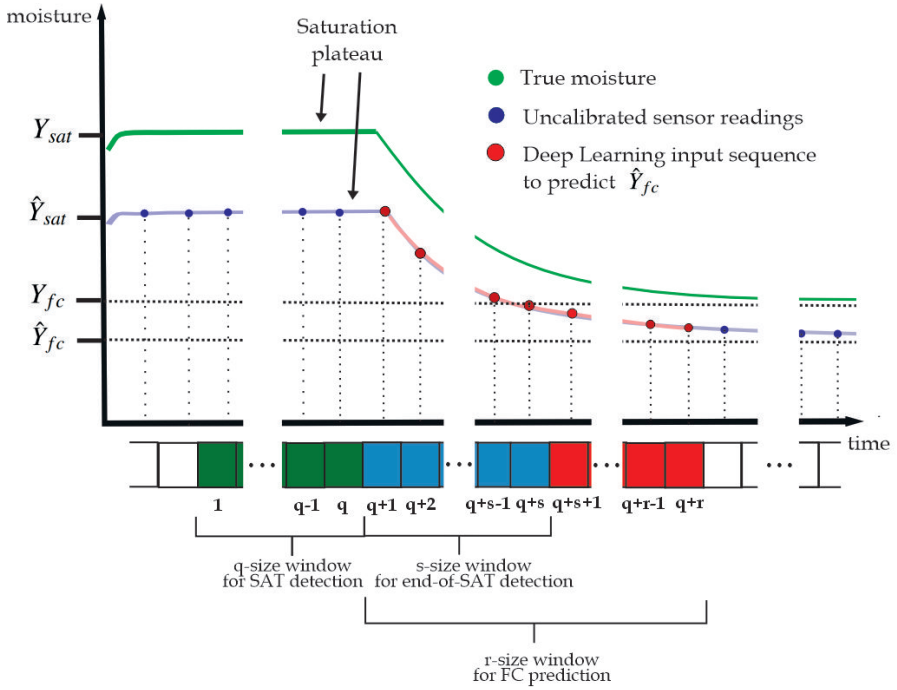

Fig. 2 Time series for actual moisture and the readings from an uncalibrated sensor. It also visible the sequence used to predict $\hat{Y}_{fc}$ and the time windows for determining the moment at which the saturation plateau ends.

$$\frac{1}{q}\sum_{i=1}^{q-1}\frac{|\hat{Y}_{i+1}-\hat{Y}_i|}{\delta t} \leq \tau_1 \tag{5}$$

being δt the time between readings. This condition ensures that, during the saturation, the soil moisture does not experience significant fluctuations, thereby manifesting a near-zero first derivative. By taking the absolute value and averaging over a range, this condition offers robustness against minor signal disturbances.

2. **Identification of Sudden Moisture Decrease Post-Saturation:** This criterion is formulated to pinpoint the precise moment at which a notable decline in soil moisture occurs following a saturation phase. It enables the determination of when the moisture transitions from a state of saturation to the onset of a drying period, by utilizing the following condition:

$$\frac{\hat{Y}_{q+1}-\hat{Y}_q}{\delta t} > \tau_2 \tag{6}$$

where τ_2 represents a specific threshold that defines a significant rate of moisture reduction, signaling the cessation of the saturation phase. The difference in moisture levels between two consecutive readings, divided by the time interval δt, is compared against τ_2 to establish the commencement of moisture decrease.

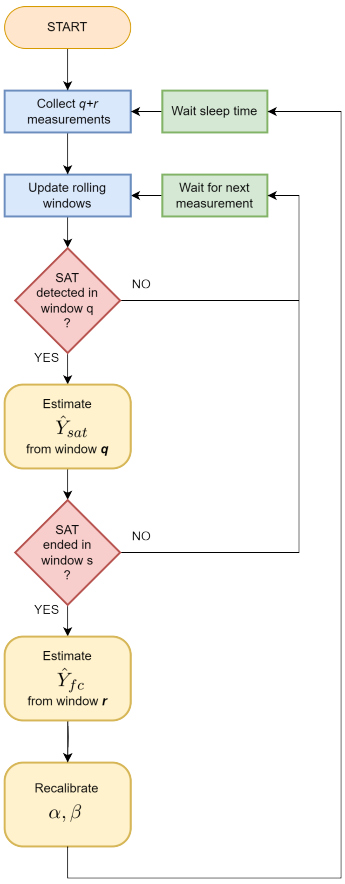

Fig. 3 General flowchart of the self-calibration algorithm

3. **Ensuring Sustained Moisture Decrease Post-Saturation:** This criterion verifies that a continuous reduction in soil moisture occurs immediately following the end of the saturation plateau. It is crucial to distinguish between a genuine, ongoing decrease in moisture levels and transient, isolated fluctuations. This is mathematically expressed as:

$$\hat{Y}_{i+1} - \hat{Y}_i < 0 \qquad \text{for } q < i < q + s \tag{7}$$

Here, the condition mandates that the difference in consecutive moisture readings should be negative, indicating a decrease, across a defined sequence of readings post-saturation. The indices i run from the point immediately after saturation, q, extending through the next s measurements, to ensure that the observed decline in moisture is not merely episodic but a sustained trend.

These three conditions, when met collectively, indicate a very specific behavioral pattern: a saturation plateau followed by a noticeable decrease in soil moisture levels. The thresholds τ_1 and τ_2, as well as the parameter s, are typically determined on a

purely empirical basis. Although they may require adaptations on field applications, due to environmental conditions or sensor characteristics, its determination is direct.

4.3 Determination of Field Capacity

This section outlines the process for estimating the field capacity of the sensors. Once the point of interest mentioned in section 3 is detected, measurements are collected from the sensors and passed to the neural network. Together with data on soil texture and sensor depth, the network predicts the estimated field capacity value of the sensors. First, the sensor data acquisition process is described, followed by details on the deep learning model used.

4.3.1 Data acquisition

As data is acquired from the sensors, the saturation detector algorithm continuously analyze the data until it identifies the point of interest, namely the moment when the curve starts to decline after reaching saturation. Assuming the presence of a sensor in each of the p depth layers in the soil, the algorithm proceeds to collect the next r humidity readings from these p sensors following the detection of saturation. This procedure is depicted in Fig. 4.

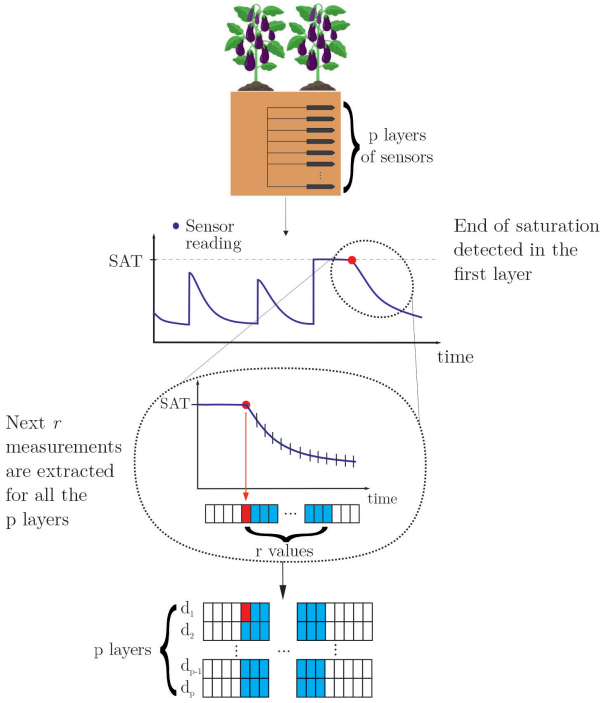

Fig. 4 After the end point of the saturation plateau is detected in an irrigation/precipitation event, the next r measurements are extracted from the p layers (blue painted blocks).

After obtaining moisture data sets from the sensor layers, they are associated with information about sensor depths and soil's textural composition, specifically the percentages of silt, clay, and sand in the soil. These data are then fed into the deep learning model to predict the FC value estimated by the sensor ($\hat{Y}_{fc}$).

4.3.2 Deep Learning model

In this subsection, the Deep Learning model used is described, which is composed of two interconnected components: (i) a 2D convolutional neural network (CNN) and (ii) a fully connected neural network (FCNN). The reason behind this architecture lies in its capacity to capture both temporal and spatial features within the input sequences is expected from the CNN, followed by the ability to integrate these extracted features with additional relevant information in the FCNN, the additional features, which are the soil texture represented by the sand, silt and clay percentages and the depth in cm of the layers into analysis. In the subsequent sections, a deeper dive will be taken into the architecture's specifics, and insights into its training process and predictive performance will be provided.

The architecture being considered combines a Convolutional Neural Network (CNN) with a fully connected neural network (FCNN) to predict soil moisture using sensors data. The CNN takes in r measurements from p distinct sensor layers. Each measurement then goes through convolutional layers designed to extract relevant features from the sensor data.

A CNN is a type of deep learning model specifically designed for the processing of grid-like data, such as images or, in this case, sequential sensor data. The typical CNN architecture features convolutional layers that serve as the network's core. Each layer utilizes a set of kernels to slide over input data—known as convolution operation—and generate feature maps. During training, the filters are automatically modified by the network to capture critical data aspects, including edges, textures, or, as is expected in this case, temporal and spatial patterns in sensor data sequences. In addition to convolutional layers, CNNs often include pooling layers, which serve to progressively reduce the spatial dimension of the data, thereby decreasing the number of network parameters and computations and helping to make the network less sensitive to the exact location of features within the input data. In this application, it was chosen to not use pooling layers due to the already reduced dimensionality of the data.

Post convolutional feature extraction, the resultant feature maps are flattened and concatenated with additional features. These additional features include extrinsic data relevant to the prediction task, which is not encapsulated in the sensor data, as the depth (cm) of the layers and the textural composition of the soil. The combined feature vector then serves as input to a FCNN.

A FCNN is a traditional feedforward Neural Network, wherein each neuron in one layer is connected to every neuron in the subsequent layer. FCNNs are well-suited for tasks involving structured data and are renowned for their capability to capture complex relationships within data. In this model, the CNN is followed by the FCNN, which is responsible for the integration of the features extracted by the CNN with additional relevant information. This information encompasses the depth at which the sensors are situated and details about the textural composition of the soil. The role of

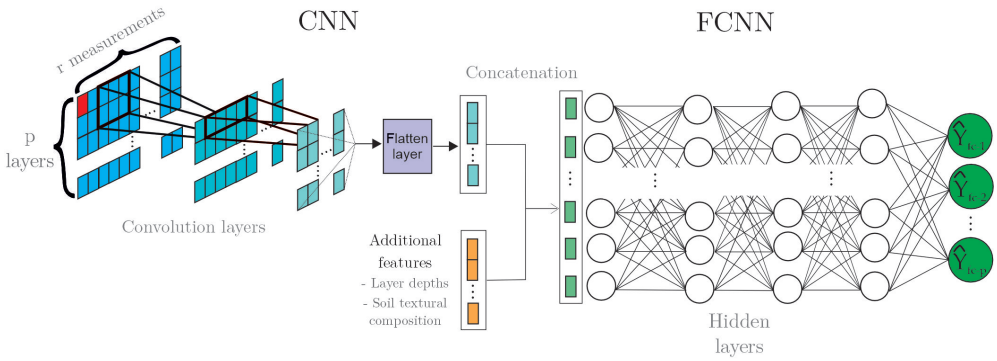

Fig. 5 Schematic representation of the integrated CNN-FCNN architecture for soil moisture prediction. The left segment illustrates the CNN with two convolutional layers processing r measurements from p sensors. The extracted features are then flattened and concatenated with additional features. The right segment depicts the FCNN which receives the concatenated features as input, with multiple hidden layers leading to the final output values $\hat{Y}_{fc,i}$, indicative of the predicted soil moisture at FC in every layer.

the FCNN is to effectively combine these features and make accurate predictions of the field capacity. The FCNN culminates in an output layer where the predicted values, denoted as $\hat{Y}_{fc,i}$, represent the soil moisture in FC estimated by the uncalibrated sensor in layer i. A general schematic of the architecture is shown in Fig. 5.

5 Results

This section provides a summary of the application of the self-calibration algorithm to a data set that simulates drift error sensor measurements in various scenarios. The aim is to analyze the neural network's capacity to predict the field capacity estimated by the uncalibrated sensors, as well as the algorithm's performance in correcting the sensor measurements. First, the simulation software utilized and the generation of the dataset will be described, which will be followed by the presentation and evaluation of the results.

5.1 Numerical generation of temporal series

Hydrus 1D [33] is a computational software for simulating the one-dimensional movement of water, heat, and multiple solutes in variably saturated porous media. It is widely recognized for its application in hydrology, agronomy, and environmental sciences. It employs the Richards' equation for water transport and convection-dispersion equations for solutes, allowing for the simulation of various hydrological scenarios and the assessment of soil dynamics in response to different environmental conditions.

This software is used to generate a diverse and comprehensive dataset for training and validating the neural network and self-calibration algorithm, which consists of a collection of moisture time series for various soil types and depth. To generate these time series, it was also simulated various irrigation and precipitation events to produce

a more realistic dataset and situations where at least the first layer reaches saturation, which is necessary for applying the algorithm. Additionally, the moisture values at saturation and field capacity for each soil type is acquired.

In order to automate the generation of moisture readings for multiple scenarios, a Python-based implementation of Hydrus-1D, Phydrus [34], has been employed. This implementation provided the creation, calibration, and visualization of Hydrus-1D, thus allowing its direct implementation in Python modules.

5.1.1 Spatio-temporal Domain

The spatial domain was delineated as a one-dimensional soil profile extending to a depth of 30 cm. The spatial discretization was set a grid cell size of 1 cm. Measurement points were placed at various depths, ranging from 5 cm to 30 cm, with consistent intervals of 5 cm, having a total of 6 layers. Regarding temporal parameters, the total simulation duration was established at 200 days for each soil type. Data acquisition was set at a high temporal resolution, with a sampling rate of every 15 minutes.

5.1.2 Boundary Conditions

The simulation's upper boundary condition was configured as an atmospheric pressure boundary with a surface layer, thus allowing the water flux across the top boundary to be governed by external conditions, specifically an irrigation/precipitation regimen. This setup permits the accumulation of water on the surface, where the height of the surface water layer can increase due to irrigation or precipitation events. It was observed that this boundary condition secured coverage in all simulated scenarios. Conversely, the lower boundary was assigned a free-draining water flow condition to emulate an unrestricted water flow at the bottom of the soil profile.

5.1.3 Field Capacity Determination

To ascertain the FC values, additional simulations were executed starting from a matric potential of 0 kPa, considered representative of a saturated state. Adhering to the free-draining boundary condition at the bottom, the system was allowed to stabilize until the matric potential reached -33 kPa at the topmost layer, aligning with the FC definition presented in Section 3.3. The moisture level recorded at this juncture was designated as the FC value.

5.1.4 Variability in soil type

In pursuit of comprehensive data to train the network, a total of 989 soil types were simulated. Each soil type was characterized by a textural composition—specifically percentages of sand, silt, and clay—generated randomly and uniformly. The aim behind this approach is to obtain a dataset diverse enough to enable the network to learn the nuances of moisture dynamics across a spectrum of soil types.

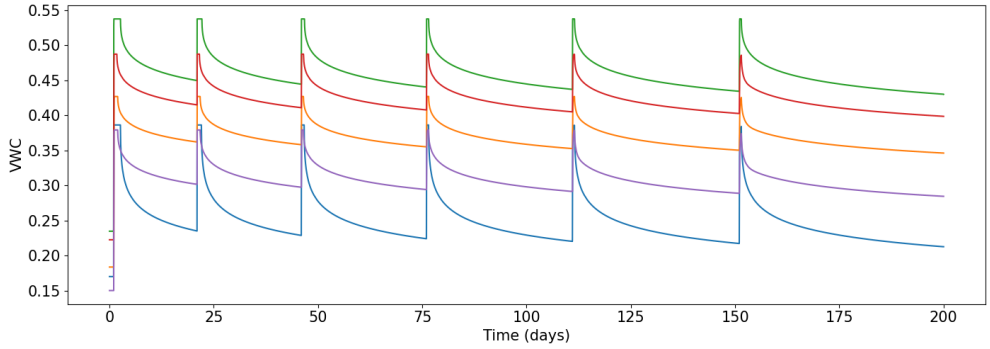

Fig. 6 Time series of moisture generated by Hydrus corresponding to the first layer. The data correspond to 5 different soil types. It is possible to observe the 6 events of irrigation/precipitation.

5.1.5 Irrigation/precipitation patterns

Six irrigation or precipitation events have been defined in each simulation. These events represent a specific period of time when water is added to the soil, either by artificial irrigation or natural precipitation. Each event has a specific flow rate, which refers to the amount of water applied per unit of time, and duration, which causes saturation in at least the first layer of soil. Using varied irrigation and precipitation events can increase the data variability in the simulation.

5.2 Saturation Detection

The algorithm for detecting the end of the saturation plateau was applied to all Hydrus generated time series using the following parameters: $q = 11$, $s = 3$, $\tau_1 = 3 \times 10^{-3}$, $\tau_2 = 5 \times 10^{-3}$. For this application case with synthetic data, the algorithm's parameter selection was done through a trial-and-error approach. This was due to the specific nature of the problem and the absence of a priori knowledge to guide a more formal selection. Although this method identified a set of parameters that effectively detected the saturation endpoint, it is important to note that adjustments may be necessary for its direct applicability in real experimental data contexts. An example of the algorithm implementation is shown in figure 7, where the time series for the first layer of moisture versus time, simulated with Hydrus for a clay soil type is displayed, alongside with its finite difference. The algorithm detected the end of the saturation plate at the green highlighted point. The below figure shows the corresponding finite difference. In the first $q = 11$ points, the mean of the finite difference does not exceed the tolerance τ_1. This is followed by a sharp drop of the finite difference, which exceeds the tolerance τ_2. Finally, the final $s = 3$ points have a negative finite difference.

5.3 Neural Network dataset

After generating the time series for soil moisture, the saturation detection algorithm was applied to obtain data sequences for each saturation event. The length of each sequence was set to $r = 48$, which corresponds to 12 hours of measurements at a

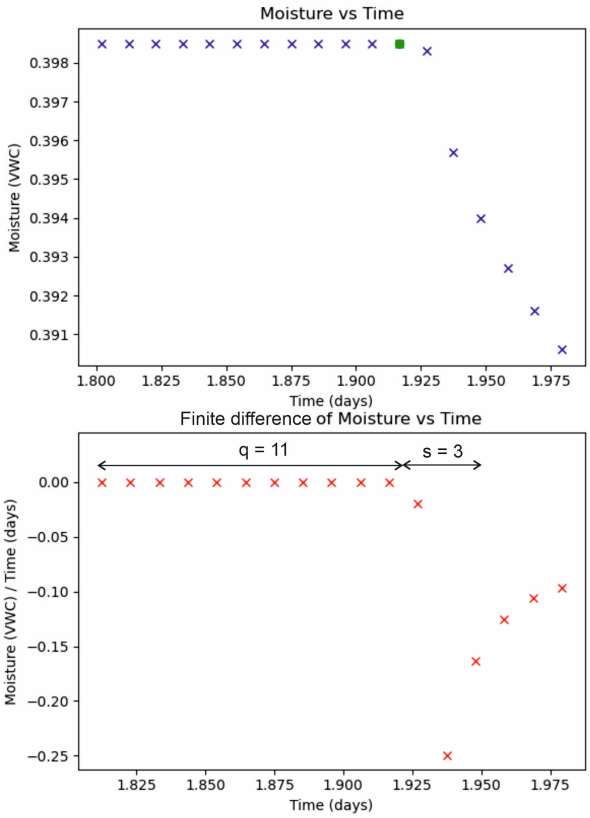

Fig. 7 Moisture time serie simulated by Hydrus for the first layer in a clay type soil, alongside with its finite difference

sampling rate of 15 minutes. Figure 8 presents a graphical example of the process. The sequences of six layers are divided into sets of three layers each, simulating the different combinations of sensor layers that are possible, in the manner of data augmentation. It has been established that in each of these new sets, the first layer should be no more than 10 cm deep. This results in the generation of 16 new sets, as shown in 9.

Finally, to simulate the readings from sensors without appropriate calibration, equation 2 is employed on the generated data. Across each soil type and its six layers, 20 values of α and β are randomly generated to simulate diverse sensor drift errors:

$$\begin{aligned} \alpha &\sim \mathcal{U}(0.9, 1.1) \\ \beta &\sim \mathcal{U}(-0.07, 0.07), \end{aligned} \tag{8}$$

as illustrated in Fig. 10. In the simulations of soil moisture drift errors, values of 20, 30, and 40 for the α and β parameters were evaluated across different soil types and layers. It was observed that enhancements in performance were not significant with values beyond 20, yet computational costs increased. Therefore, the number of α and

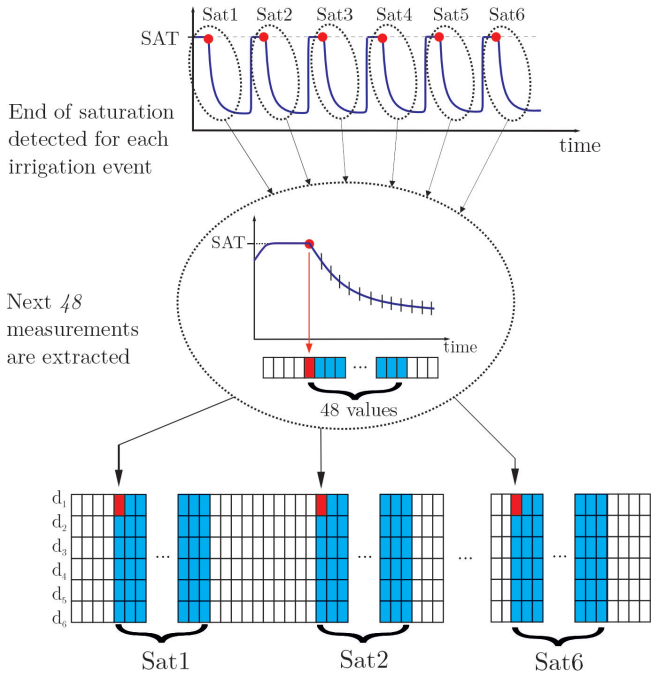

Fig. 8 Algorithm for saturation detection applied to the time series generated by Hydrus-1D, to obtain sequences of length 48 at each saturation time.

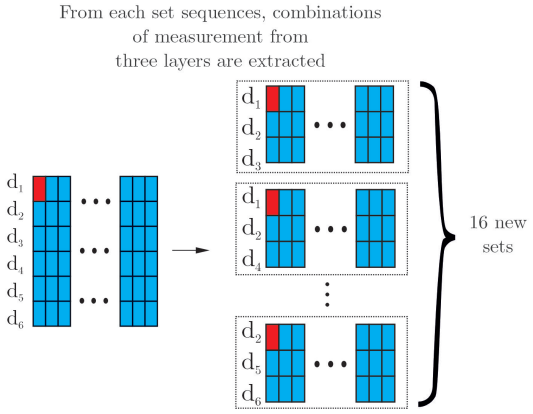

Fig. 9 New 3-layer sets are created by selecting 3-layer combinations from the initial 6-layer set. The condition for these new sets is that the first layer must be either layer 1 or 2.

β values was confined to 20 for each soil type and layer, optimizing performance and computational efficiency.

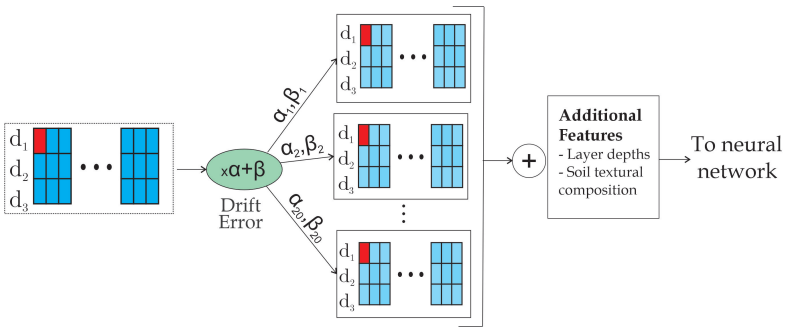

Fig. 10 The error function is applied to each set of 3-layer sequences along with the FC value to simulate the error response of the sensors. Finally, these sets are fed to the neural network.

5.4 Hyperparameter tunning and compilation of the model

In relation to the chosen final architecture to be applied to the generated dataset, it was selected a CNN with 2 convolutional layers followed by a FCN with 4 hidden layers. The model's hyperparameters were selected using a tuning method called hyperband, which is characterized by its efficiency in terms of resource usage, which is particularly useful in this application with a large dataset. In this method, resources are divided into different brackets and multiple configurations are trained at each level with varying amounts of allocated resources. The best configurations, as determined by the validation metric, are promoted to the next level where they are trained with additional resources. An efficient balance between exploration of the hyperparameter space and exploitation of promising configurations is achieved through this approach. The key hyperparameters that can be adjusted include the number of filters, the kernel shapes, the number of units in the dense layers, and the batch size. The table 1 shows the values of these hyperparameters taken to analysis.

Table 1 Hyperparameters considered in the tuning process

Hyperparameter	**Values**
Filters in the first convolutional layer	[2,4,6]
Filters in the second convolutional layer	[2,4,6]
Kernel shape in the first convolutional layer	[1,30] , [1,20] , [1,35]
Kernel shape in the second convolutional layer	[2,10], [2,15], [2,5]
Neurons in the first and fourth hidden layers	[8,16,32,64]
Neurons in the second and third hidden layers	[16,32,64,128]
Batch size	[16,32,64]

The tuning was conducted over multiple trials dividing the dataset into three parts (64%), validation (16%) and testing (20%) and the best observed model have a 4 filters in both convolutional layers, kernel shapes of [1,30] and [2,10] in the first and second convolutional layers respectively, 8 neurons in the first and fourth hidden layers, 16 neurons in the second and third hidden layers and a batch size of 64.

The model was compiled using the Adam optimizer and the mean squared error loss function. The Adam optimizer is a popular choice for deep learning models due to

its adaptive learning rate and momentum. The loss function, $L(y, \hat{y})$, can be defined as the sum of the squared differences between the actual (y) and predicted ($\hat{y}$) FC values, as shown in eq. 9.

$$L(y, \hat{y}) = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \tag{9}$$

5.5 FC prediction

The model performance was assessed by computing the absolute error in predicting the sensor estimation of the field capacity for each sample in the test set. The resulting mean absolute error (MAE) for field capacity prediction was 1.51×10^{-2} m^3/m^3, with a maximum MAE of $2.14 \times 10^{-1} m^3/m^3$ and a minimum of $1.69 \times 10^{-9} m^3/m^3$. Furthermore, the model's effectiveness was evaluated across six distinct layers of soil. The results, as can be seen in table 2, show that the model predictions are relatively consistent across the layers

Table 2 Mean error in FC prediction per layer

Layer	**1**	**2**	**3**	**4**	**5**	**6**
Mean Error ($\times 10^{-2}$)	1.40	1.43	1.46	1.45	1.46	1.40

In addition, a ternary plot, presented in Fig. 11, was utilized to illustrate the distribution of errors in the model's prediction of field capacity across different soil types. A higher prediction error is seen in soils with an intermediate composition of silt and clay. On the other hand, a lower error is observed in soils with a high proportion of either sand or clay.

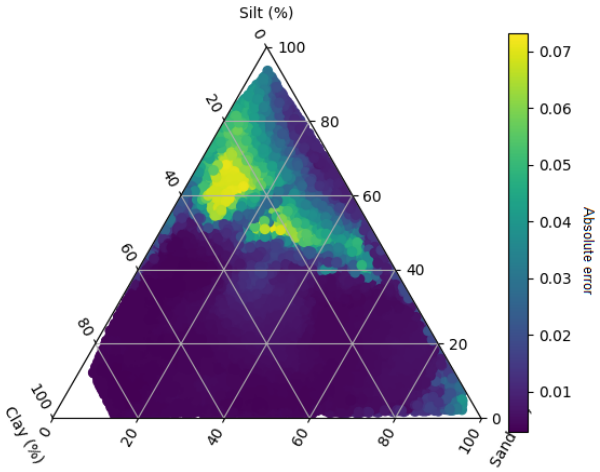

Fig. 11 Ternary plot illustrating the distribution of FC error prediction across various soil types

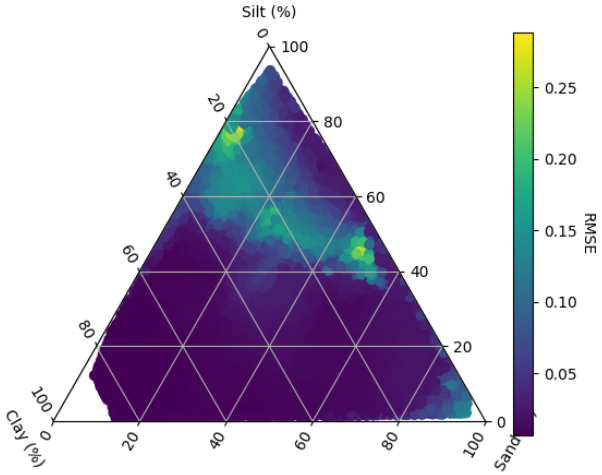

Fig. 12 Ternary plot illustrating the average error distribution of the recovered sensors among various soil types.

5.6 Sensor Error Correction

To evaluate the algorithm's performance in terms of corrected measurements, the root mean square error (RMSE) of the uncorrected and corrected sensor readings relative to the original time series produced by Hydrus, which is considered the ground truth, were calculated and compared. The RMSE values obtained were 0.1372 and 0.0428 for the uncorrected and corrected sensor readings, respectively, which finally resulted in a relative improvement of 68.8%. As for the results for each layer, the mean RMSE for each layer and the relative improvement achieved by the algorithm are presented in table 3.

Table 3 Performance of algorithm in correcting sensors readings per layer

Layer	**1**	**2**	**3**	**4**	**5**	**6**
Sensor RMSE	0.1444	0.1385	0.1381	0.1396	0.1376	0.1395
Corrected sensor RMSE	0.0434	0.0436	0.0438	0.0439	0.0442	0.0448
Relative Improvement (%)	69.9	68.6	68.3	68.6	67.9	67.8

A ternary classification plot was also employed to visualize the distribution of average errors across different soil types, as can be seen in Fig. 12. The algorithm demonstrated superior performance, as reflected by lower RMSE values, on soils with a greater percentage of sand or clay. Conversely, moving towards the regions of the triangle representing higher percentages of silt, there is a rise in RMSE, indicating a reduced corrective capability of the algorithm for these types of soils.

Another metric used to measure the quality of the correction algorithm was defined as follows:

$$Improvement = 1 - \frac{RMSE(Corrected\ reading,\ Hydrus\ moisture)}{RMSE(Sensor\ reading,\ Hydrus\ moisture)} \tag{10}$$

where:

- *RMSE(Corrected reading, Hydrus moisture)* is the Root Mean Square Error between the corrected sensor readings and the Hydrus moisture measurements.
- *RMSE(Sensor reading,Hydrus moisture)* is the Root Mean Square Error between the original sensor readings and the Hydrus moisture measurements.

This metric evaluates the performance of the correction algorithm by comparing the improvement in accuracy of sensor readings after correction. An Improvement value of 1 indicates perfect correction, whereas a value of 0 suggests no improvement. Negative values indicate that the correction process has degraded the quality of the sensor readings.

After applying the correction algorithm to the simulated sensor measurements and evaluating their performance using the previously defined improvement metric, significant results were observed. The metric showed that 84.83% of the sensors experienced an improvement in the accuracy of their moisture measurements. This indicates that the applied correction was not only effective but also widely applicable to most of the evaluated sensors. Although 15.17% of the sensors did not show improvement according to the established metric, this result is valuable in highlighting specific error patterns or sensor characteristics that require further adjustments to the correction method, which are detailed in the following subsection.

5.7 Correlation analysis

To gain a better understanding of the variables that have the most significant impact on the algorithm's performance, a correlation analysis was performed. The variables considered in the analysis were the depths of the sensors layers (*Layer Depth*), the textural composition of the soil represented by the percentages of silt (*%Silt*), Clay (*%Clay*) and sand (*%Sand*), the values of the drift error coefficients alpha (α) and beta (β) and the actual moisture values in saturation and FC of the soils (*Sat* and *FC*). The Spearman's correlation coefficient was selected due to its appropriateness for datasets where the relationship between variables is not linear. This non-parametric method, which assesses monotonic relationships based on ranks rather than values, offers a robust alternative for analyzing the given dataset. The results of the analysis are shown in the table 4.

The data indicate a significantly negative correlation for *%Silt*, suggesting a substantial adverse effect on performance. This is also evident in figure 12, where soils with higher percentages of silt exhibit higher error values. However, it is worth noting that the variable with the highest correlation index is clay, indicating that soils with a higher percentage of clay perform better.

Table 4 Spearman's correlation coefficients

Variable	Coefficient
Clay content	0.29030
Field capacity moisture	0.20770
Sand content	0.11640
Saturation moisture	0.06417
α coefficient	0.01010
β coefficient	-0.01101
Layer depth	-0.08298
Silt content	-0.39000

6 Conclusion

In this study, the challenge of self-calibration of soil moisture sensors has been addressed, a critical issue in modern agriculture and water resource management. A deep learning-based self-calibration algorithm has been introduced, enhancing the accuracy of sensors in real-time and at the point of deployment.

The uniqueness of the proposed method lies in its ability to calibrate sensors using only two characteristic points: soil saturation and field capacity. The need for laborious and costly calibration methods has been eliminated, facilitating more efficient and effective implementation.

In section 5, simulation results have been presented, demonstrating the algorithm's capability to significantly improve the accuracy of soil moisture sensors across different soil types and environmental conditions. The self-calibration method has improved the accuracy of 84.83 % of the sensors. However, limitations have been identified in the study, such as the need for further field testing and validation against standard calibration methods. Future research could focus on addressing these limitations and extending the algorithm's applicability to other types of sensors and environmental conditions.

Declarations

Funding: The authors did not receive support from any organization for the submitted work.

Competing interests: The authors have no competing interests to declare that are relevant to the content of this article.

Data availability and access: The datasets used in this work are publicly available in the Zenodo repository: https://doi.org/10.5281/zenodo.10897580. Upon request, any other specific fold used can be provided by the corresponding author.

References

[1] Matula, S., Bát'ková, K., Legese, W.L.: Laboratory performance of five selected soil moisture sensors applying factory and own calibration equations for two soil media of different bulk density and salinity levels. Sensors **16** (2016)

[2] Gantiva-Osorio, M., Pérez-Ruiz, A., Chaparro, J.A.: Comparison of moisture controllers for blueberries based on hydrodynamic parameters of the substrate, 172–176 (2023)

[3] Klute, A.: Methods of Soil Analysis, Part 1: Physical and Mineralogical Methods. American Society of Agronomy: Madison, WI, USA, ??? (1986)

[4] Bogena, H.R., Huisman, J.A., Oberdörster, C.: Evaluation of a low-cost soil water content sensor for wireless network applications. Journal of Hydrology **344**, 32–42 (2007)

[5] Mittelbach, H., Lehner, I., Seneviratne, S.I.: Comparison of four soil moisture sensor types under field conditions in switzerland. Journal of Hydrology **430–43**(1), 39–49 (2012)

[6] Pahuja, R.: Development of semi-automatic recalibration sytem and curve-fit models for smart soil moisture sensor. Measurement **203** (2022)

[7] Adla, S., Rai, N.K., Karumanchi, S.H., Tripathi, S., Disse, M., Pande, S.: Laboratory calibration and performance evaluation of low-cost capacitive and very low-cost resistive soil moisture sensors. Sensors **20**(2) (2020)

[8] Nagahage, E.A.A.D., Nagahage, I.S.P., Fujino, T.: Calibration and validation of a low-cost capacitive moisture sensor to integrate the automated soil moisture monitoring system. Agriculture (Switzerland) **7** (2019)

[9] Aranda, D., Tapia Córdoba, A., Millan Gata, P.: Calibración y caracterización de sensores capacitivos de bajo coste para la monitorización de humedad de suelo. In: XLIII Jornadas de Automática, pp. 479–485 (2022). Universidade da Coruña. Servizo de Publicacións

[10] Parvin, N., Degré, A.: Soil-specific calibration of capacitance sensors considering clay content and bulk density. Soil Research **54** (2016)

[11] Wendroth, O., Nambuthiri, S., Walton, R.J.: Accounting for soil spatial variability in soil water capacitance probe calibration. Vadose Zone **12**(2), 1–14 (2013)

[12] Barcelo-Ordinas, J.M., Doudou, M., Garcia-Vidal, J.: Self-calibration methods for uncontrolled environments in sensor networks: A reference survey. Ad Hoc Networks **88**, 142–159 (2019)

[13] Köksal, E.S., Cemek, B., Artık, C.: A new approach for neutron moisture meter calibration: artificial neural network. Irrigation Science **29**, 369–377 (2011)

[14] Chen, L., Zhangzhong, L., Zheng, W.: Data-driven calibration of soil moisture sensor considering impacts of temperature: A case study on fdr sensors. Sensors **19** (2019)

[15] Si, M., Xiong, Y., Du, S.: Evaluation and calibration of a low-cost particle sensor in ambient conditions using machine-learning methods. Atmospheric Measurement Techniques **13**, 1693–1707 (2020)

[16] Verma, S., Pahuja, R.: Recalibration and performance comparison of soil moisture sensors using regression and neural network characteristic models. Materials Today: Proceedings journal **45**, 4852–4861 (2021)

[17] Ahmad, S., Kalra, A., Stephen, H.: Estimating soil moisture using remote sensing data: A machine learning approach. Advances in Water Resources **33**, 69–80 (2009)

[18] A., Y., Wang, G., Hu, P.: Root-zone soil moisture estimation based on remote sensing data and deep learning. Environmental Research **212** (2022)

[19] Patrizi, G., Bartolini, A., Ciani, L.: A virtual soil moisture sensor for smart farming using deep learning. IEEE Transactions on Instrumentation and Measurement **71** (2022)

[20] Tan, R., Yuan, Z., Xing, G.: System-level calibration for data fusion in wireless sensor networks. ACM Transactions on Sensor Networks **9** (2013)

[21] Lee, C., Sohn, E., Park, J.D.: Estimation of soil moisture using deep learning based on satellite data: a case study of south korea. GIScience and Remote Sensing **56**, 43–67 (2019)

[22] Balzano, L., Nowak, R.: Blind calibration of sensor networks (2007)

[23] Lipor, J., Balzano, L.: Robust blind calibration via total least squares. 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2014)

[24] Lee, B.-T., Son, S.-C., Kang, K.: A blind calibration scheme exploiting mutual calibration relationships for a dense mobile sensor network. IEEE Sensors Journal **14** (2020)

[25] Wang, C., Ramanathan, P., Saluja, K.K.: Blindly calibrating mobile sensors using piecewise linear functions. 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (2009)

[26] Bilen, Puy, G., Gribonval, R.: Convex optimization approaches for blind sensor calibration using sparsity. IEEE Transactions on Signal Processing **62** (2014)

[27] Gribonval, R., Chardon, G., Daudet, L.: Blind calibration for compressed sensing by convex optimization. IEEE Transactions on Signal Processing **62** (2014)

[28] Yang, A., Wang, Pengjun, Huazhong, Y.: Blind drift calibration of sensor networks using multi-output gaussian process, 1–4 (2018)

[29] Wang, Y., Yang, A., Chen, X.: A deep learning approach for blind drift calibration of sensor networks. IEEE Sensors Journal **17**(Number), 4158–4171 (2017)

[30] Li, X., Shao, M., Zhao, C.: Estimating the field capacity and permanent wilting point at the regional scale for the hexi corridor in china using a state-space modeling approach. Journal of Soils and Sediments **19**, 3805–3816 (2019)

[31] Reynolds, C.A., Jackson, T.J., Rawls, W.J.: Estimating soil water- holding capacities by linking the food and agriculture organization soil map of the world with global pedon databases and continuous pedotransfer functions. Water Resources Research **36**, 3653–3662 (2000)

[32] Rai, R.K., Singh, V.P., Upadhyay, A.: Soil analysis. Planning and evaluation of irrigation projects **505** (2017)

[33] Šimůnek, J., Genuchten, M.T., Šejna, M.: Development and applications of the hydrus and stanmod software packages and related codes. Vadose Zone Journal **7**, 587–600 (2008)

[34] Collenteur, R.A., Brunetti, G., Vremec, M.: "phydrus: Python implementation of the hydrus-1d unsaturated zone model. version 0.2.0". Technical report (2019)