

Article

An Intelligent Self-Validated Sensor System Using Neural Network Technologies and Fuzzy Logic Under Operating Implementation Conditions

Serhii Vladov ^{1,*} , Victoria Vysotska ^{2,3,*} , Valerii Sokurenko ¹, Oleksandr Muzychuk ¹ and Lyubomyr Chyrun ⁴

¹ Kharkiv National University of Internal Affairs, 27, L. Landau Avenue, 61080 Kharkiv, Ukraine; rector_hnuvs@ukr.net (V.S.); o.muzychuk23@gmail.com (O.M.)

² Information Systems and Networks Department, Lviv Polytechnic National University, 12, Bandera Street, 79013 Lviv, Ukraine

³ Institute of Computer Science, Osnabrück University, 1, Friedrich-Janssen-Street, 49076 Osnabrück, Germany

⁴ Applied Mathematics Department, Ivan Franko National University of Lviv, 1, University Street, 79000 Lviv, Ukraine; lyubomyr.v.chyrun@lpnu.ua

* Correspondence: serhii.vladov@univd.edu.ua (S.V.); victoria.a.vysotska@lpnu.ua (V.V.)

Abstract: This article presents an intelligent self-validated sensor system developed for dynamic objects and based on the intelligent sensor concept, which ensures autonomous data collection and real-time analysis while adapting to changing conditions and compensating for errors. The research's scientific merit is that an intelligent self-validated sensor for dynamic objects has been developed that integrates adaptive correction algorithms, fuzzy logic, and neural networks to improve the sensors' accuracy and reliability under changing operating conditions. The proposed intelligent self-validated sensor system provides real-time error compensation, long-term stability, and effective fault diagnostics. Analytical equations are described, considering corrections related to influencing factors, temporal drift, and calibration characteristics, significantly enhancing measurement accuracy and reliability. The fuzzy logic application allows for refining the scaling coefficient that adjusts the relationship between the measured parameter and influencing factors, utilizing fuzzy inference algorithms. Additionally, monitoring and diagnostics implementation for sensor states through LSTM networks enable effective fault detection. Computational experiments on the TV3-117 engine demonstrated high data-restoring accuracy during forced interruptions, reaching 99.5%. A comparative analysis with alternative approaches confirmed the advantages of using LSTM (Long Short-Term Memory) neural networks in improving measurement quality.

Keywords: intelligent self-validated sensor system; neural network; LSTM (Long Short-Term Memory) neural network; fuzzy logic; scale coefficient; intelligent sensor



Citation: Vladov, S.; Vysotska, V.; Sokurenko, V.; Muzychuk, O.; Chyrun, L. An Intelligent Self-Validated Sensor System Using Neural Network Technologies and Fuzzy Logic Under Operating Implementation Conditions. *Big Data Cogn. Comput.* **2024**, *8*, 189. <https://doi.org/10.3390/bdcc8120189>

Academic Editor: Alberto Abelló

Received: 29 September 2024

Revised: 5 December 2024

Accepted: 11 December 2024

Published: 13 December 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

1.1. Relevance of the Research

Modern requirements for automated control and monitoring systems for complex equipment [1–3], particularly helicopter turboshaft engines (TE) [4,5], demand high measurement accuracy and operational speed. The increasing amount of sensor data and the complexity involved in real-time processing require innovative solutions to enhance measurement systems' effectiveness [6]. Neural network technologies [7,8] and fuzzy logic [9,10], when applied to intelligent systems, improve prediction accuracy and decision-making, especially under real-world operational conditions where uncertainty is often present.

Developing intelligent systems incorporating neural networks allows for self-tuning and adaptation to changes in the operation process. Fuzzy logic, in turn, enables imprecise and uncertain data handling, which is crucial for improving such systems' reliability in challenging operational environments. This approach's relevance lies in its ability to

significantly enhance measurement systems' precision and resilience, ensuring their stable performance even under unpredictable and rapidly changing conditions.

1.2. State of the Art

Recent research on intelligent systems using neural networks [7,8,11,12] and fuzzy logic [9,10,13,14] has shown significant advancements. A key focus is the integration of neural networks into monitoring and control systems for complex equipment. Studies [15,16] have demonstrated that recurrent neural networks improve prediction accuracy in changing operational conditions, helping systems adapt and reduce measurement errors.

Fuzzy logic is widely applied to handle sensor data with inaccuracies and uncertainties. It allows systems to operate effectively in uncertain environments, such as fluctuating external conditions or unstable equipment performance [17]. This approach is used in diagnostic and control systems where quick decisions are needed with ambiguous data.

The combination of neural networks and fuzzy logic [18,19] has become increasingly popular due to their complementary strengths. This integration not only boosts accuracy but also enhances system resilience. Research continues to focus on optimizing algorithms and improving computational efficiency, enabling these systems to handle noise, errors, and sudden changes in real-world conditions.

Despite significant advancements in intelligent systems development using neural networks and fuzzy logic [18,19], gaps remain in ensuring high accuracy under uncertainty and noise interference. Existing research often lacks a focus on creating systems that can effectively compensate for errors caused by external factors or real-time sensor parameter fluctuations.

There is a need for an information system based on intelligent sensors, utilizing fuzzy logic for adaptive error compensation. This approach would enhance measurement accuracy and improve system resilience to external influences. Further investigation into mechanisms capable of real-time data correction from sensors, considering influencing factors, is essential to ensure more precise and reliable operation in complex operational environments.

1.3. Main Research Attributes

The **research object** is the dynamic-object intelligent system.

The **research subject** includes the tools and methods of processing dynamic objects from intelligent systems.

The **research aims** are to develop an intelligent system for complex dynamic objects based on fuzzy logic and neural networks, enabling sensor monitoring and the adaptive adjustment of measurements in real time to enhance accuracy and reliability in changing external conditions.

The following practical and scientific tasks were solved to achieve this aim:

1. The development of an intelligent self-validated sensor system is based on the intelligent sensor concept.
2. The intelligent self-validated sensor system controls method development.
3. An intelligent sensor is implemented in a neural network module form.
4. We conduct a computational experiment to evaluate the effectiveness of the proposed intelligent self-validated sensor control system.
5. We conduct a comparative analysis of the results with alternative approaches. Evaluating the results obtained effectiveness.

This research's **main contribution** is developing an intelligent self-validated sensor system for complex dynamic objects, integrating neural networks and fuzzy logic to enable real-time sensor monitoring and adaptive measurement adjustment.

2. Materials and Methods

2.1. Development of an Intelligent Self-Validated Sensor Based on the Intelligent Sensor Concept

The intelligent sensor concept involves the creation of sensor devices with built-in computing capabilities for processing and analyzing data in real-time [20,21]. Such sensors can collect information about the physical or chemical state of the environment and independently interpret these data, identify patterns, and adapt to changing conditions. The intelligent sensor concept considers the external environment factors influencing the intelligent sensor accuracy measurement. An intelligent self-validated sensor requires functional blocks, a minimum set for measuring and control systems, as shown in Figure 1.

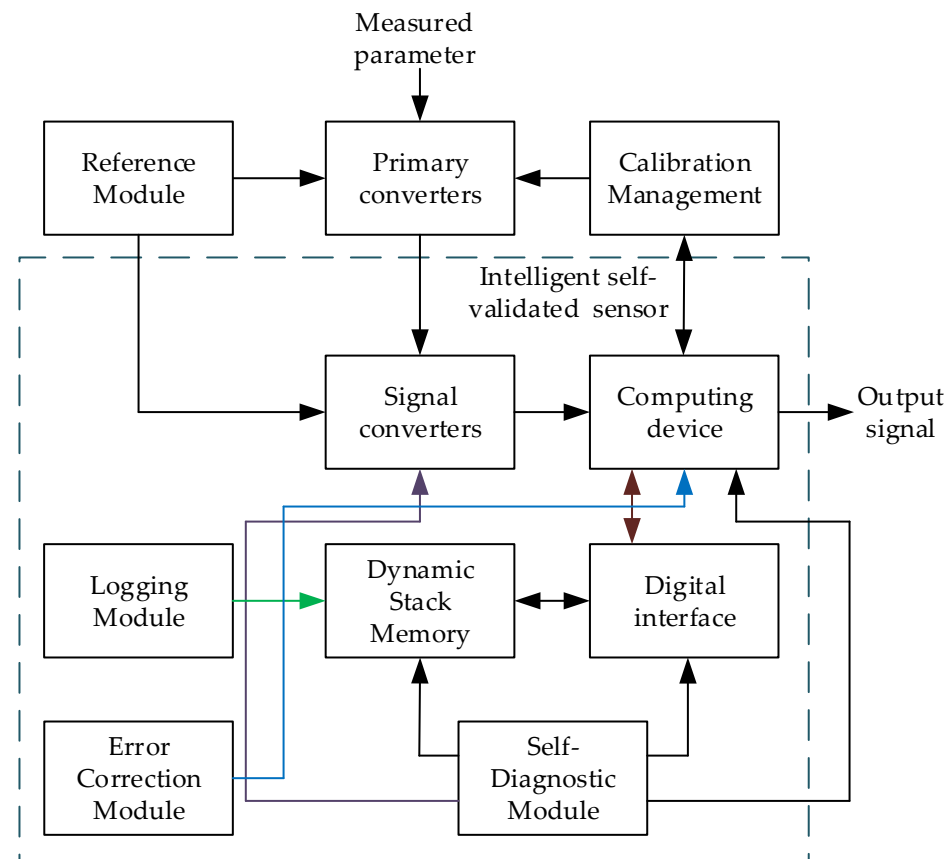


Figure 1. The proposed intelligent self-validated sensor system functional block diagram with dynamic stack memory.

Unlike intelligent sensors [20,21], the proposed intelligent self-validated sensor includes several key functional modules that ensure its operation autonomy. The Self-Diagnostic Module continuously monitors the health and accuracy of all sensor components, identifying deviations in their operation. The Reference Module provides basic reference values with which the current measured data are compared to assess the measurements' accuracy. When errors are detected, the Error Correction Module automatically corrects them, minimizing the failures' impact on the output signals. The Logging Module is used to store the history of all diagnostic activities, including data on errors and their elimination processes, which also simplifies the devices' analysis and quality control. The Calibration Management Unit automatically calibrates the sensor, ensuring the measurements' accuracy under changing operating conditions.

A distinctive feature of the functional block diagram of the intelligent self-validated sensor (Figure 1) compared to that of the traditional one [20,21] is the dynamic stack memory introduction in the authors' interpretation [22], proposed in the neural network model for solving the identifying complex dynamic objects problem in unsteady modes.

The dynamic stack memory mathematical model developed by these authors' group in [22] considers memory segmentation, which allows optimizing its use by creating new segments when overflowing and merging them when the data amount decreases.

Dynamic stack memory efficiently controls temporarily required data, allowing the memory amount to be adapted to current computing tasks in real time. Unlike static memory, dynamic stack memory optimizes resource use, minimizing the cost of unused memory, which is especially important for systems with limited resources. Compared to random access memory (RAM), it has an advantage in data localization since the stack structure simplifies access to the last written elements, which improves performance in sequential signal processing tasks. Unlike register memory, stack memory can process significantly more temporary data without overloading the processor.

The sensor's main characteristic is a function of the following form [20,21]:

$$x = F(U, I), \quad (1)$$

where x is the value measured by the sensor, F is the measured value characteristic, and U and I are the electrical values associated with its calculated value at the sensor output.

Taking into account the factor influencing the measured value s and the factor determined by the specific sensor characteristic (calibration characteristic) z , Equation (1) is rewritten as follows:

$$x = F(U, I) + s + z. \quad (2)$$

The influencing factor is generally represented as follows:

$$s = F(f(P_k)), \quad (3)$$

where P_k is the influencing variable; $f(P_k)$ is the influencing variable transfer function; $F(f(P_k))$ is the transformation function. In this case, the influencing factor can be either one-dimensional or two-dimensional and is represented as follows:

$$s = \sum_{j=1}^{s_i} p_j \cdot \varphi_j(u), \quad (4)$$

$$s = \sum_{j=1}^{s_i} p_j \cdot \varphi_j(u, P), \quad (5)$$

$$s = \begin{cases} \sum_{j=1}^{s_1} p_{j,1} \cdot \varphi_{j,1}(u), & B = b_1 \\ \sum_{j=1}^{s_2} p_{j,2} \cdot \varphi_{j,2}(u, P), & B = b_2 \end{cases} \quad (6)$$

The calibration characteristic factor taken into account in (2) is represented as follows:

$$z = z(U, I) = \sum_{i=0}^N z_i \cdot x^i, \quad (7)$$

where z represents the coefficients, and x represents the values at the calibration points.

The time drift effect on the sensor readings does not depend on the measured value and is represented as follows:

$$x = F(u) + F_t(t). \quad (8)$$

Using Equations (3), (7) and (8), the general Equation (2) is represented as follows:

$$x = F(u) + F_k(P_k) + F_t(t) + z, \quad (9)$$

where $F(u)$ is the main characteristic of the measured value; $F_k(P_k)$ is the correction taking into account the influencing value; $F_t(t)$ is the correction taking into account the time drift (zero correction depending on the duration of sensor operation); z is the correction taking into account the calibration characteristic (taking into account the zero shift for a specific sensor after testing).

The intelligent self-validated sensor's resulting equation is a conversion equation that considers the converter calibration data, time drift, and the magnitude of the influencing factors. The external environment influencing factors cause an increase in measurement error. Therefore, a search for optimal solutions is carried out to compensate for these errors [23]. The necessary data for compensating influencing factors are generated on stands in the static test mode and recorded in memory as tabular data. Algorithmic compensation for calculating output information from a sensor, taking into account correction factors, consists of the following [23]:

- The parameter value is read from the information from the sensor.
- A refined factor is taken from the correction values table for a given parameter.
- The correction factor (tabular) value for a given parameter is multiplied by the output signal.

The use of fuzzy logic is proposed to consider uncertainties and complex nonlinear dependencies caused by time influence, calibration characteristics, and external factors that affect the accuracy of the measurement [1,3,12]. Fuzzy logic allows describing nonlinear dependencies and uncertainties [5,10,13,14,18], which are difficult to formalize using traditional neural network methods [4,7,8,11,12], providing adaptive error compensation. The fuzzy rules and membership functions used for correcting the sensor output data allow for effectively considering influencing parameters in real time, minimizing measurement errors, and increasing sensor reliability in difficult operating conditions [10,14,18]. For each input parameter $x_i(t)$, fuzzification is performed, which transforms it into the fuzzy value set $\mu_j(x_i(t))$:

$$\mu_j(x_i(t)) = \frac{1}{1 + \left(\frac{x_i(t) - c_j}{\sigma_j}\right)^2}, \quad (10)$$

where $\mu_j(x_i(t))$ is the $x_i(t)$ membership degree to the j -th fuzzy variable with center c_j and width σ_j .

The fuzzy rule set has the following form:

$$\text{if } x_1(t) \text{ is } A_1^k \text{ and } x_2(t) \text{ is } A_2^k \text{ and } \dots \text{ and } x_n(t) \text{ is } A_n^k, y(t) \text{ is } B^k, \quad (11)$$

where A_i^k is a fuzzy set for input parameters, and B^k is the output fuzzy set.

In the developed intelligent self-validated sensor system, a refinement for the scaling coefficient is proposed to compensate for a particular dependency by utilizing fuzzy rule inference algorithms through the redundancy present in term sets. The correction coefficient value is "blurred" across the entire range of the influencing variable, while the scaling correction is computed using fuzzy inference algorithms and fuzzy rule bases.

$$K_{scal} = \begin{cases} x_1 \forall t_3 \leq t \leq t_1 \\ x_2 \forall t_4 \leq t \leq t_2 \\ x_3 \forall t_5 \leq t \leq t_3 \\ x_4 \forall t_6 \leq t \leq t_4 \\ \dots \\ x_{n-1} \forall t_{n-1} \leq t \leq t_{n-3} \\ x_n \forall t_n \leq t \leq t_{n-2} \end{cases} \quad (12)$$

To construct the fuzzy rule base for fuzzy inference systems, input and output variable values are determined as fuzzy numbers. The measured parameter is used as the input variable (Figure 2a). At the same time, the scaling coefficient K_{scal} serves as the output

variable (scaling coefficient values depending on the measured parameter are obtained from sensor calibration results during static tests) [24,25] (Figure 2b).

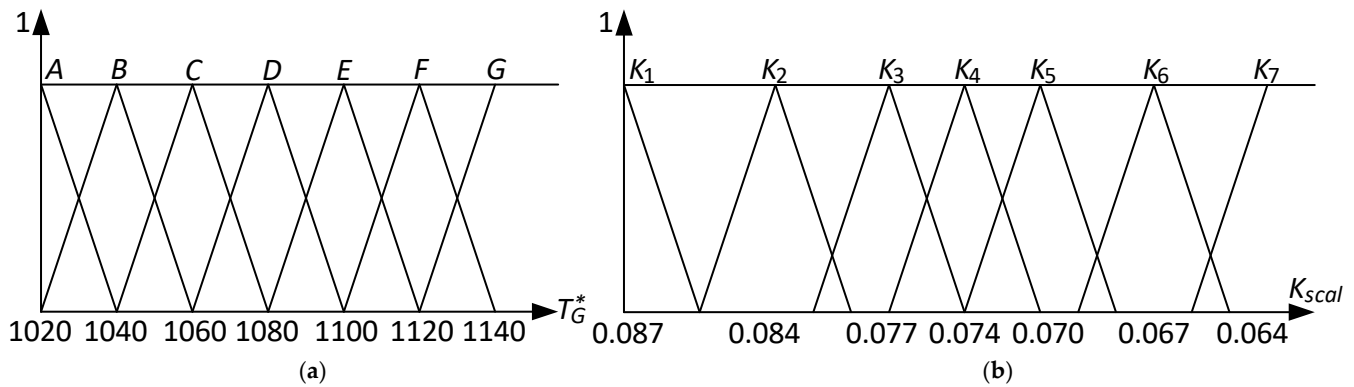


Figure 2. The input variable fuzzy numbers: (a) is the measured parameter, and (b) is the scaling coefficient values.

With each set of measurements, the measured parameter membership to specific fuzzy sets is determined. For example, if the measured parameter falls within the set G range, its membership in other sets is checked. If it belongs only to set G, the scaling coefficient value is selected from set K7. However, if it belongs to set F, the coefficient value is computed based on the input sets' intersection within the K7 and K6 ranges. The resulting scaling coefficient is then added to the correction for sensor temporal instability ΔK_{scal} , measured during the initial operational cycle, and the result K_{scal} is used in solving the onboard navigation task. The scaling coefficient is refined on each subsequent algorithm cycle according to the new measured parameter value.

The aggregation of all rules' outputs is performed according to the following equation:

$$\mu_B(y(t)) = \max_k \min(\mu_{A_1^k}(x_1(t)), \dots, \mu_{A_n^k}(x_n(t))). \quad (13)$$

After aggregation, defuzzification is performed, for example, using the gravity method center:

$$y(t) = \frac{\int y \cdot \mu_B(y) dy}{\int \mu_B(y) dy}. \quad (14)$$

2.2. Development of the Intelligent Sensor Controlling Method

The implementation for monitoring and diagnosing the technical condition of sensors within the measurement system is proposed using two LSTM (Long Short-Term Memory) neural networks, where M_1 and M_2 represent control object models implemented as LSTM networks (Figure 3) [26]. The addition of an intelligent classifier unit with a majority element and a decision-making unit featuring an expert system in the fuzzy logic form enables the monitoring of sensor output values as the control object and the identification the failed element in the circuit. In Figure 3, the following are indicated: $\Delta_1 = Y_{cont. obj.} - Y_{M1}$, $\Delta_2 = Y_{cont. obj.} - Y_{M2}$, and $\Delta_3 = Y_{M1} - Y_{M2}$ are the mismatch vectors between the signal vectors $Y_{cont. obj.}$, Y_{M1} , and Y_{M2} . It is assumed that the failure of two or more units at any given time is unlikely. Therefore, the system may encounter the following events:

- "Sensor failure";
- "Normal operation";
- " M_1 failure", which is a malfunction of the M_1 module or a hardware failure in the neural network, leading to the incorrect computation of Y_{M1} vector components;
- " M_2 failure", which is the Y_{M2} vector components' erroneous computation;
- "Decision-making unit output uncertainty", which indicates a failure in the comparison unit.

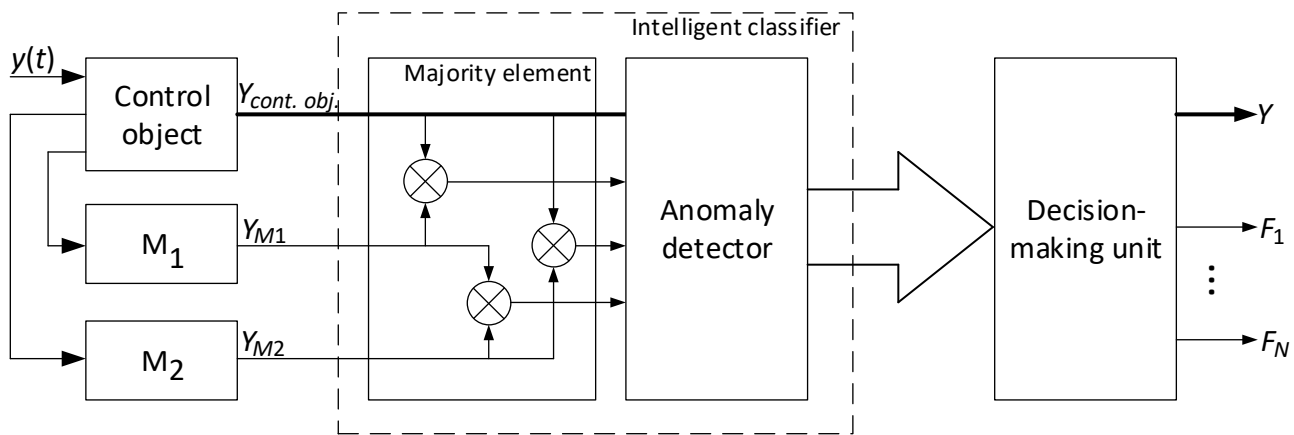


Figure 3. Schematic diagram of a fault-tolerant smart sensor with a majority element and a fuzzy logic decision-making unit.

Table 1 shows the discrepancy in the influence of Δ_1 , Δ_2 , and Δ_3 on the analyzed events using the decision-making unit. The decision-making unit output reaction is $F = 5$. The discrepancies in the component values of Δ_1 , Δ_2 , and Δ_3 are presented as the linguistic variable (term) values: “Small” or “Large”.

Table 1. The discrepancy Δ_1 , Δ_2 , Δ_3 influence the analyzed events that occurred.

Event	Δ_1	Component Δ_2	Δ_3	Decision-Making Unit Reaction
Normal operation condition	$\Delta_1 = \text{“Small”}$	$\Delta_2 = \text{“Small”}$	$\Delta_3 = \text{“Small”}$	$F_1 = 0$
Failure				
Sensor	$\Delta_1 = \text{“Large”}$	$\Delta_2 = \text{“Large”}$	$\Delta_3 = \text{“Small”}$	$F_2 = 1$
M_1	$\Delta_1 = \text{“Large”}$	$\Delta_2 = \text{“Small”}$	$\Delta_3 = \text{“Large”}$	$F_3 = 1$
M_2	$\Delta_1 = \text{“Small”}$	$\Delta_2 = \text{“Large”}$	$\Delta_3 = \text{“Large”}$	$F_4 = 1$
The decision-making unit uncertainty	$\Delta_1 = \text{“Small”}$	$\Delta_2 = \text{“Large”}$	$\Delta_3 = \text{“Small”}$	$F_5 = 1$
	$\Delta_1 = \text{“Large”}$	$\Delta_2 = \text{“Small”}$	$\Delta_3 = \text{“Small”}$	
	$\Delta_1 = \text{“Large”}$	$\Delta_2 = \text{“Large”}$	$\Delta_3 = \text{“Large”}$	

The fuzzy set characteristic, in this case, is the membership function, which is selected as a triangular function. The logical inference is performed step by step [27]:

- At the fuzzification stage, the membership functions defined on the input variables are applied to their actual values to determine the truth degree for the corresponding fuzzy rule of each premise.
- At the inference stage, the calculated truth value for each fuzzy rule premise is applied to each rule conclusion.

The decision-making block performs fuzzy logical inference using a rule system that links input vectors to the output responses $F_1, \dots, F_5$ of the decision-making block. As an example, consider the formulation of such a rule for the event named “The sensor normal operation condition”:

$$\text{if } (\Delta_1 = \text{“Small” and } \Delta_2 = \text{“Small” and } \Delta_3 = \text{“Small”}) \text{ then } (F_1 = 0, F_2 = 0, F_3 = 0, F_4 = 0, F_5 = 0). \quad (15)$$

For the event “Failure M_1 ” (neural network model M_1), there will be the following rule:

$$\text{if } (\Delta_1 = \text{“Large” and } \Delta_2 = \text{“Large” and } \Delta_3 = \text{“Small”}) \text{ then } (F_1 = 1, F_2 = 1, F_3 = 1, F_4 = 1, F_5 = 1). \quad (16)$$

For other events, the rules are constructed similarly. Let us consider possible solutions (recommendations) when detecting a particular failure:

- When the “Sensor failure” signal appears, you should use signals averaging from the outputs M_1 and M_2 , that is,

$$Y = \frac{Y_{M1} + Y_{M2}}{2}, \quad (17)$$

which allows for a higher accuracy of Y .

- When “Failure M_1 ” or “Failure M_2 ” occurs, the system remains operational, but additional training of the failed neural network is required. The failed neural network is excluded from the voting scheme if the additional training procedure is ineffective. The system remains two-channel.

An anomaly detector is introduced to predict faults based on changes in the output signal relative to expected values. The dissonance metric $D(t)$ is used based on the Euclidean norm:

$$D(t) = \|y(t) - \hat{y}(t)\|^2, \quad (18)$$

and if $D(t)$ exceeds the threshold δ , the system detects an anomaly and generates an alert.

Figure 4 presents a structural diagram for the developed intelligent self-validated sensor system within the measurement system, in which the primary transducer consists of a device that converts a non-electrical signal into an electrical one and a computational block, which includes the following components: an input signal converter, scaling (considering the influencing variable, calibration characteristics, and temporal drift), self-diagnosis (based on a neural network with a decision-making block), and the restoration of lost measurement data in cases of malfunctions or failures.

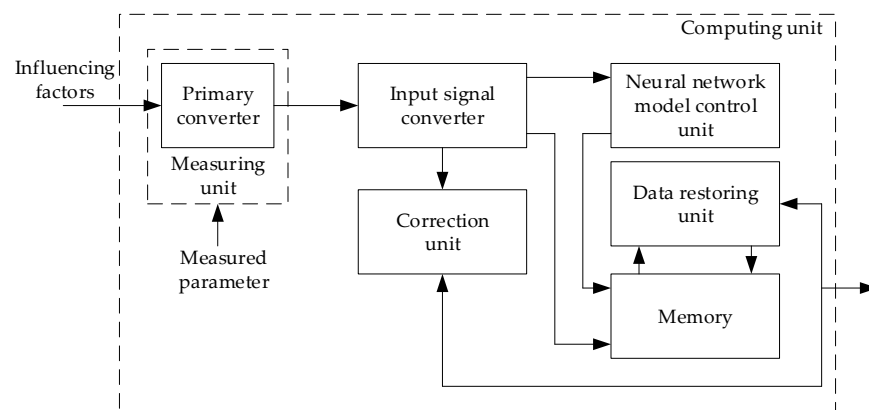


Figure 4. Structural diagram of a multifunctional intelligent sensor as a measuring system part.

2.3. An Intelligent Self-Validated Sensor System Implementation in a Neural Network Module Form

To address the task of monitoring intelligent self-validated sensor system failures, the LSTM network undergoes modifications to include mechanisms for handling anomalies in data. Such anomalies may arise from sensor malfunctions. The modification involves introducing an anomaly detection input module a_t , which influences the state updates within the LSTM network:

$$a_t = \sigma(W_a \cdot x(t) + b(a)), \quad (19)$$

where W_a represents the weight matrix for anomaly detection, $x(t)$ denotes the input signal (sensor data), and a_t indicates the signal indicating whether an anomaly exists (values close to 0 indicate an anomaly, while values close to 1 indicate standard signals).

The state update equations in the LSTM are modified as follows to account for anomalies:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x(t)] + b_f) \cdot a_t, \quad (20)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x(t)] + b_i) \cdot a_t, \quad (21)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x(t)] + b_c), \quad (22)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t, \quad (23)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x(t)] + b_o) \cdot a_t, \quad (24)$$

$$h_t = o_t \odot \tanh(C_t). \quad (25)$$

In this formulation, when an anomaly is detected (a_t is close to 0), the current input contribution to the state change becomes minimal, effectively allowing the LSTM to ignore faulty data. To detect anomalies in sensor data, one can use deviations from expected ranges or statistical methods (e.g., predictions based on previous values). Anomaly detection can be calculated based on current deviations from predicted values:

$$\epsilon_t = |x(t) - \hat{x}(t)|, \quad (26)$$

where $\hat{x}(t)$ indicates the predicted value from the model. If ϵ_t exceeds a specified threshold ϵ_t , then a_t approaches 0, indicating an anomaly:

$$a_t = \begin{cases} 0, & \text{if } \epsilon_t > \epsilon_{threshold} \\ 1, & \text{if } \epsilon_t \leq \epsilon_{threshold} \end{cases} \quad (27)$$

The training algorithm requires adaptation to handle missing data due to sensor failures. The modified LSTM cell is utilized for predicting sensor values, where training occurs on temporal data $x(t)$, and predictions $\hat{x}(t)$ are computed at each time step:

$$\hat{x}(t) = W_{out} \cdot h_t + b_{out}. \quad (28)$$

For training, a modified loss function that accounts for anomalies is applied. The error is ignored if data are identified as anomalous ($a_t = 0$). The primary loss function is the mean squared error (MSE), taking into account L2-regularization:

$$L_t = a_t \cdot \frac{1}{2} \cdot (\hat{x}(t) - x(t))^2 + \lambda \cdot \|W_i\|_2^2, \quad (29)$$

where λ is the regularization coefficient, and W_i denotes the network weights. When $a_t = 0$, the loss function at this step equals 0, preventing the influence of faulty data during training. To prevent overfitting, Dropout is used as one of the most popular regularization methods, which turns off random neurons during training:

$$h_t^{(drop)} = h_t \odot \mathbf{m}_t, \quad (30)$$

where $\mathbf{m}_t$ is a randomly generated mask vector of zeros and ones.

The Adam optimizer is utilized to update weights:

$$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot \nabla_W \cdot L_t, \quad (31)$$

$$v_t = \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot (\nabla_W \cdot L_t)^2, \quad (32)$$

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}, \hat{v}_t = \frac{v_t}{1 - \beta_2^t}, \quad (33)$$

$$W \leftarrow W - \eta \cdot \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}}. \quad (34)$$

After each training cycle, the data are checked for anomalies. If $a_t = 0$ persists for multiple consecutive data points, this indicates a potential continuous sensor failure. In such cases, the system may restore data or notify the operator. Data reconstruction can be

performed based on previously predicted values in the persistent sensor failure event. An additional module is introduced for data reconstruction:

$$x_{reconstructed}(t) = \hat{x}(t), \quad (35)$$

where $\hat{x}(t)$ represents the model's prediction for the current time based on the prior state h_{t-1} .

To improve network stability in dynamically changing data conditions, batch normalization is used:

$$\hat{y}_t = \gamma \cdot \frac{y_t - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} + \beta, \quad (36)$$

where μ_B and σ_B^2 are, respectively, the output data mean value and variance for the batch, while γ and β are the trained parameters.

The proposed approach's scientific novelty lies in the modification of the LSTM network architecture by integrating an anomaly detection module that dynamically adjusts state transitions based on sensor data reliability, significantly enhancing the model's robustness in real-time fault detection scenarios. Unlike traditional methods that discard faulty data or rely on external preprocessing for anomaly detection, this method embeds anomaly handling directly within the LSTM network, allowing the network to adaptively ignore erroneous data and maintain accurate predictions. Furthermore, introducing specialized loss functions that discount anomalies during training and a reconstruction mechanism for sensor failure recovery distinguishes this approach from existing methods, which typically lack integrated fault-tolerant mechanisms and require post hoc data correction.

3. Results

A computational experiment evaluated the proposed intelligent self-validated sensor system's effectiveness for sensor monitoring. The research object chosen for this purpose was the TV3-117 engine [28,29], which belongs to the helicopter TE class and is part of the Mi-8MTV helicopter's powerplant. This research object was selected due to its widespread use in helicopters, long operational history, and rich dataset availability for analysis, allowing for a more precise evaluation of the system's performance in real-world conditions. Additionally, the TV3-117 engine is known for its complex dynamics and high-reliability requirements, making it an ideal platform for testing the intelligent self-validated sensor system effectiveness, which can detect and predict sensor failures during challenging operational modes. This engine selection provides sufficient data variability and the opportunity to test the system across different engine operation modes, including standard and abnormal situations.

A virtual instrument [30] (Figure 5) developed by the authors was employed for the computational experiment. This virtual instrument simulates the helicopter TE sensor system, which comprises three primary sensors that record dynamic thermogas parameters onboard the helicopter [31]:

- The D-2M sensor, which records the gas generator rotor r.p.m. (n_{TC});
- A sensor comprising 14 paired T-101 thermocouples, which records the gas temperature in the front of the compressor turbine (T_C^*);
- The D-1M sensor records the free turbine rotor speed (n_{FT}).

The simulation model was developed in the LabVIEW software (version 7.1) [31] as a virtual instrument simulating the operation of a helicopter engine gas temperature sensor system. The model reproduces the operation of three main sensors, including the D-2M sensor (gas generator rotor speed), the D-1M sensor (free turbine speed), and the T-101 thermocouple (gas temperature in front of the compressor turbine). The model considers the probability of T-101 sensor failure due to difficult operating conditions, making it an essential element for testing the ability of the neural network system to detect and compensate for faults. A generated training dataset, including 256 temperature

measurements in the range of 30 s, was used to train the neural network, ensuring its adaptation to real operating conditions.

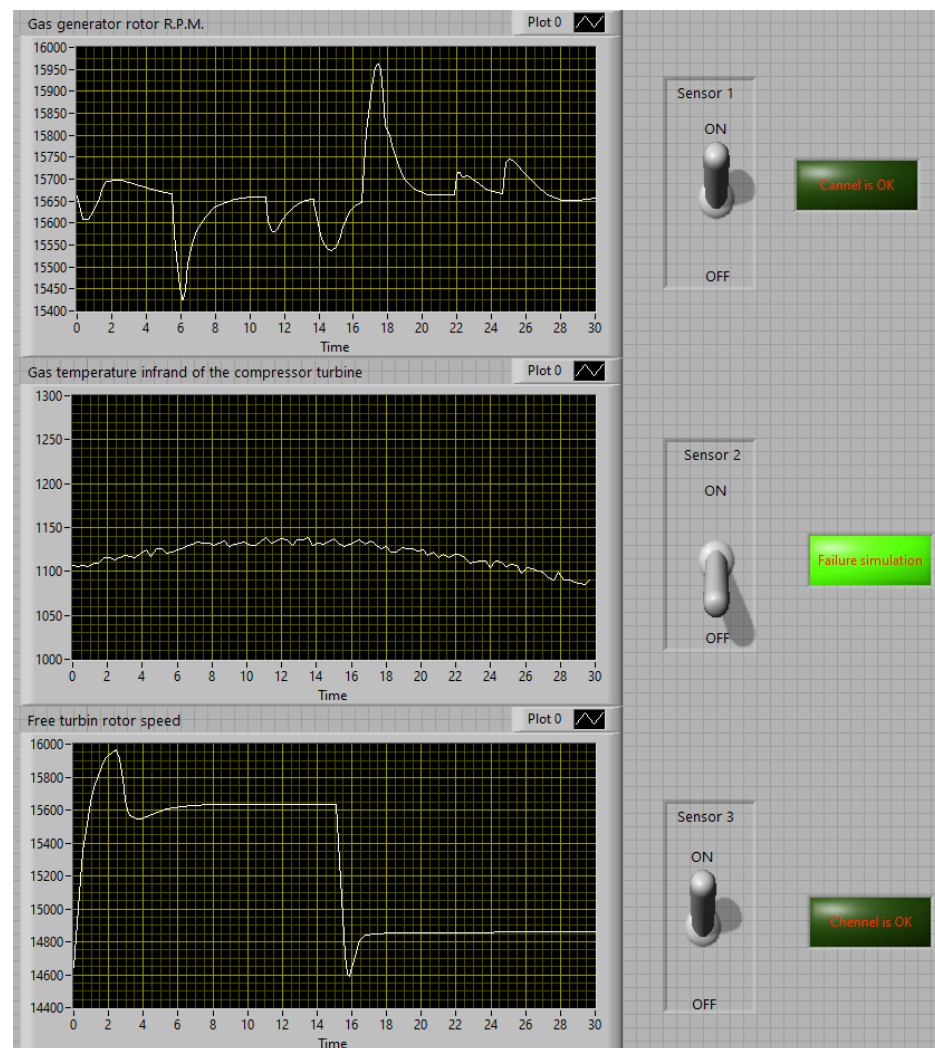


Figure 5. A virtual instrument within the LabVIEW software environment, emulating a system incorporating three sensors.

The diagrams presented in Figure 5 show a fragment of the changes in the TV3-117 engine n_{TC} , T_G^* , and n_{FT} values in the nominal operating mode under normal operating conditions [31]. The data were obtained by an official authors' request of the authors' team to the Ministry of Internal Affairs of Ukraine. They were received within the implementation framework of the project "Theoretical and applied aspects of the aviation sphere development" (registered under number 0123U104884).

It is assumed that the D-2M and D-1M sensors function correctly, while the 14-paired-thermocouple T-101 sensor carries a failure probability. This sensor, responsible for measuring gas temperatures in the front of the compressor turbine, is susceptible to malfunction due to the harsh operating environment, potentially leading to inaccurate data or a complete signal loss. The failure probability is critical in evaluating the system's ability to detect and manage sensor faults in real time. Thus, the diagram of the gas temperature signal in the front of the compressor turbine is taken as a training sample (Figure 6). To form the training dataset in Figure 6, 256 values of gas temperature in the front of the compressor turbine were selected in the time interval from 0 to 30 s with a 0.117 s step (Table 2).

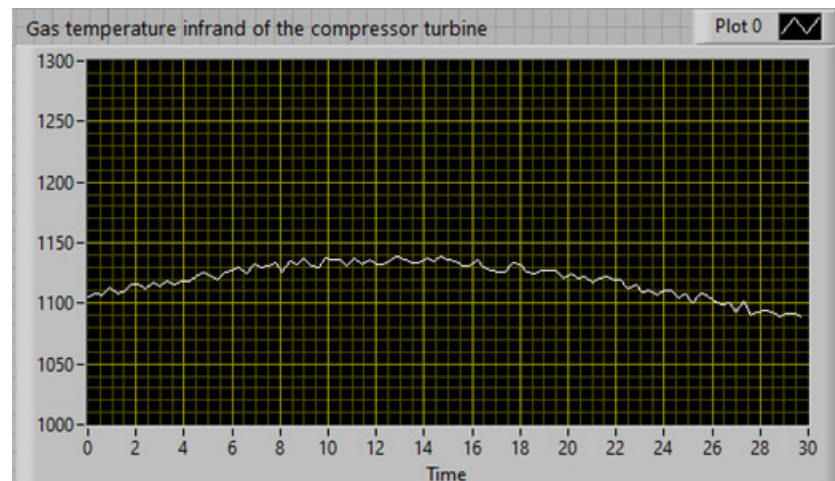


Figure 6. A diagram of the gas temperature signal in the front of the compressor turbine.

Table 2. Parameter values of the gas temperature in the front of the compressor turbine T_G^* : training dataset fragment (authors' research).

Number	1	...	38	...	84	...	122	...	179	...	214	...	256
Value	1105	...	1116	...	1130	...	1131	...	1123	...	1104	...	1104

The normalization of the training dataset parameter values was carried out as follows:

$$\tilde{x}_i = \frac{x_i - x_{i \min}}{x_{i \max} - x_{i \min}}. \quad (37)$$

Here, $\tilde{x}_i$ represents a dimensionless quantity within the interval $[0; 1]$, while $x_{i \min}$ and $x_{i \max}$ correspond to the minimum and maximum values of the variable x_i . The normalized training dataset is subsequently presented in Table 3.

Table 3. Normalized values for the gas temperature parameter in the front of the compressor turbine T_G^* used as a training dataset fragment (authors' research).

Number	1	...	38	...	84	...	122	...	179	...	214	...	256
Value	0.971	...	0.981	...	0.993	...	0.994	...	0.987	...	0.970	...	0.970

The homogeneity of the training dataset (Table 2) was evaluated using the Fisher–Pearson [32] and Fisher–Snedecor [33] tests, as described in [34,35]. According to these tests, the gas temperature in the compressor turbine front dataset of the TV3-117 engine is homogeneous, as the calculated values for the Fisher–Pearson and Fisher–Snedecor criteria fall below their critical thresholds: $\chi_{T_G^*}^2 = 4.643 < \chi_{critical}^2 = 6.6$ and $F_{T_G^*} = 2.116 < F_{critical} = 2.58$. To evaluate the dataset's representativeness, cluster analysis (k-means [36,37]) was employed in [34,35]. Random sampling split the dataset into training and test subsets in a 2:1 ratio (67 and 33%, corresponding to 172 and 84 elements, respectively). The cluster analysis results (Table 3) identified eight distinct groups (classes I...VIII), confirming the clusters' presence and demonstrating the similarity between the testing and training datasets (Figure 7). Based on these findings, the optimal dataset sizes for gas temperature in the compressor turbine front of the TV3-117 engine were determined as follows: 256 elements (100%) for the full (training) dataset, 172 elements (67%) for the control dataset, and 84 elements (33%) for the test dataset.

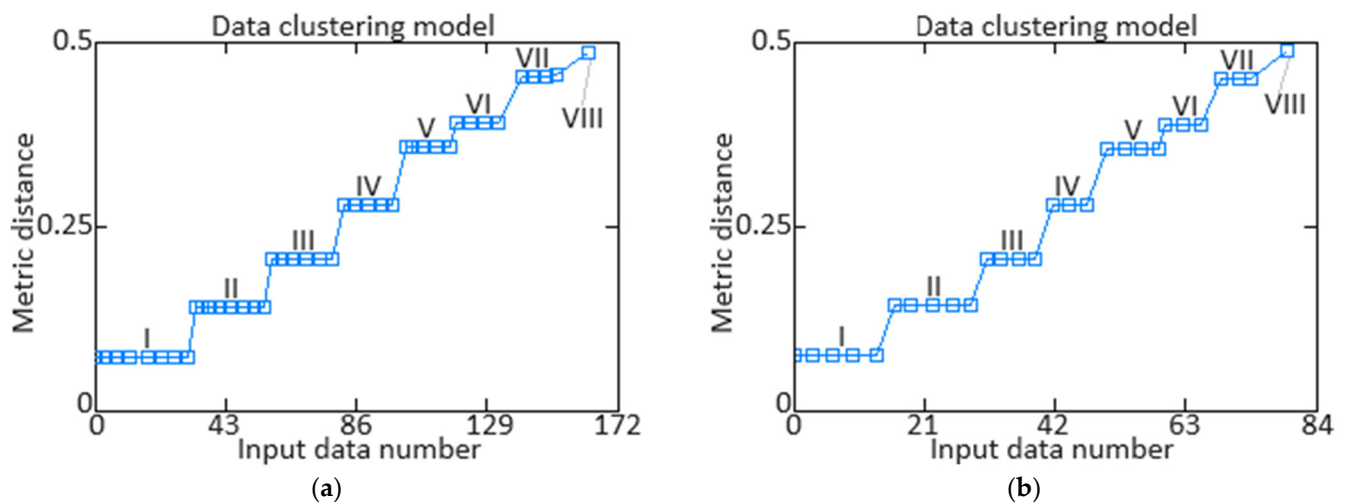


Figure 7. Cluster analysis results: (a) is the training dataset of the parameter T_G^* , and (b) is the parameter test dataset T_G^* .

At the initial computational experiment stage, error compensation for the influencing variable is performed. In this study, as an example, the case is considered where the current temperature equals 1130 K (Figure 8). In this instance, the input variable corresponds to a truth degree of 0.4 for set G and 0.6 for set F . The corresponding sub-conditions are applied in K_6 and K_7 , leading to two fuzzy sets. The output variable “scaling coefficient”, determined by the center-of-area method for the membership function values, results in $K_{scal} = 0.083$ (Figure 9).

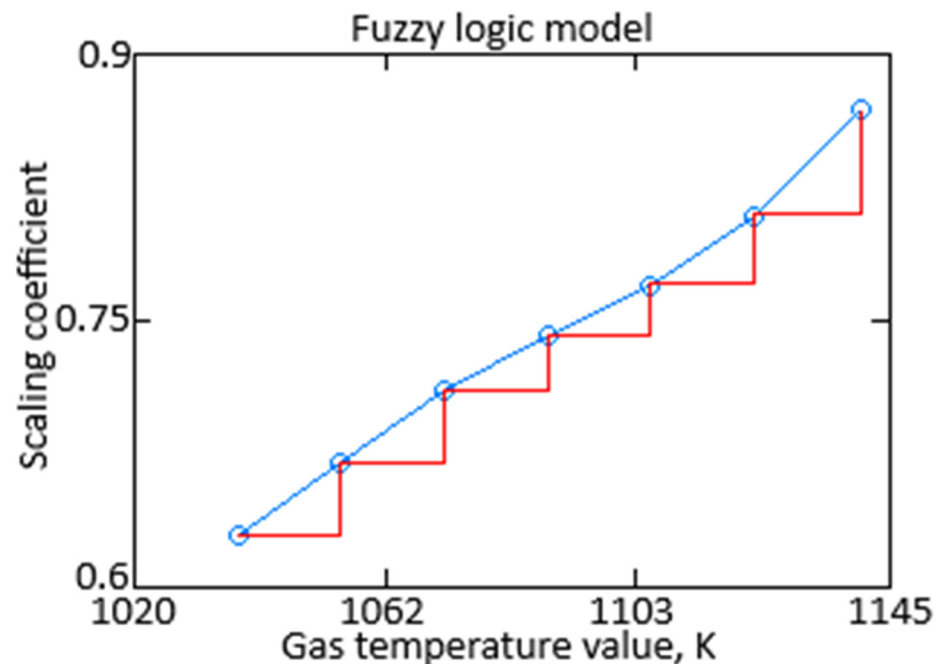


Figure 8. Two approaches comparing results when refining the scaling coefficient K_{scal} : (red curve) using a correction table; (blue curve) applying fuzzy logic.

At a temperature value of 1130 K, the tabular value for the correction scale factor will be 0.081, and that using the fuzzy logic device was 0.0832. The systematic error for the current moment of the sensor operation time was 0.0022.

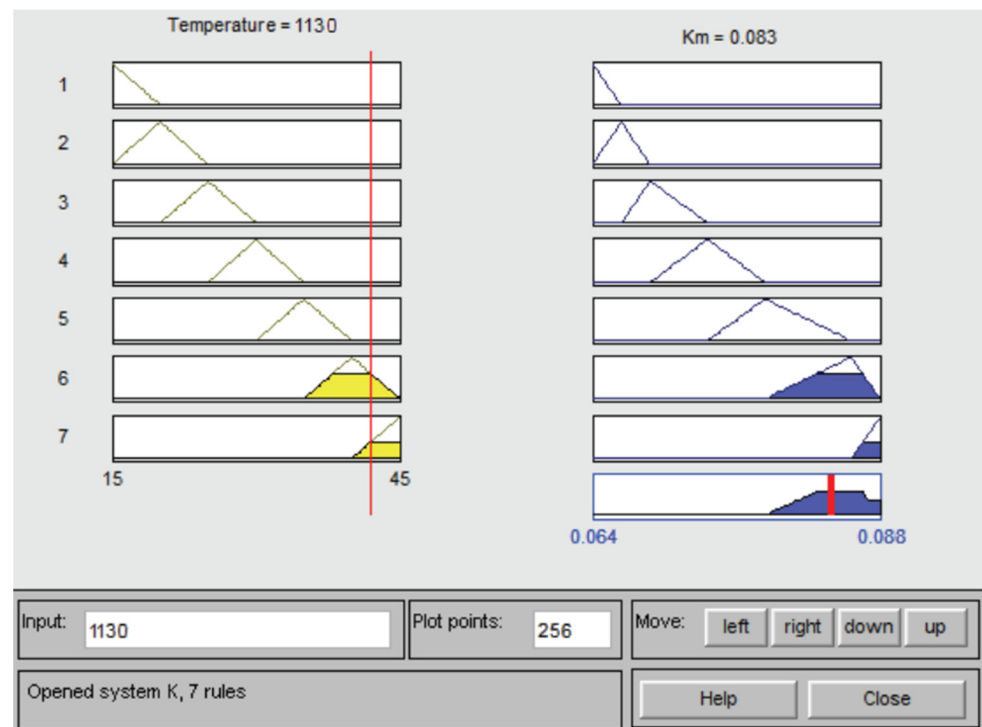


Figure 9. Scheme of the scaling coefficient value derived using fuzzy logic.

At the same time, for the gas temperature in the front of the compressor turbine in the intelligent self-validated sensor system, operating under conditions of influencing environmental factors, it is necessary to restore missing measurements during an operation-forced interruption period caused by failures and malfunctions. This study explores the potential use of neural networks (two LSTM networks) for restoring lost measurement data during periods of forced interruption caused by malfunctions or failures (Figure 10). Recording current measurement data on a unique storage device is assumed to be possible. The task is to reconstruct the time series sequence during the interruption, considering the available history of measured values. The first increments after the interruption period are utilized for more accurate data restoration.

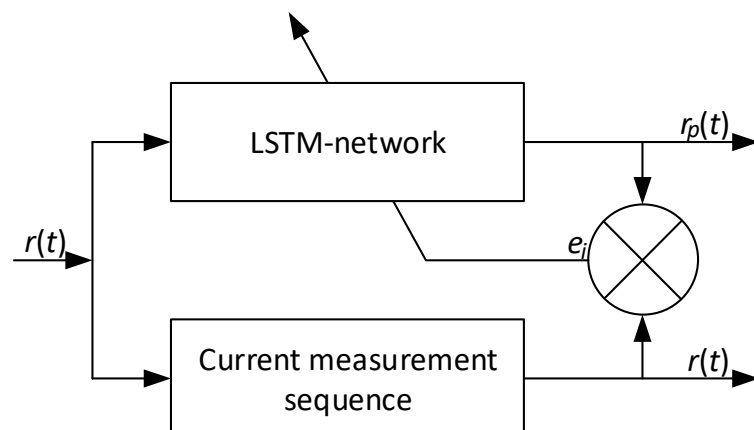


Figure 10. Scheme for restoring lost information using the LSTM network.

The diagram presents an approach for restoring sensor readings during a forced interruption period, where the LSTM network is trained under normal operating conditions of the intelligent self-validated sensor system using the error signal e_i . To restore 102 missing values from the 128th to the 230th time step, the input to the neural network included

128 historical values from the time series before the interruption and 3 values following the interruption. Figure 11 illustrates how the LSTM network with feedforward propagation accurately approximates the apparent acceleration. Additionally, in the outliers or noise presence in the time series, the LSTM network smooths the input values, allowing for precise data restoration.

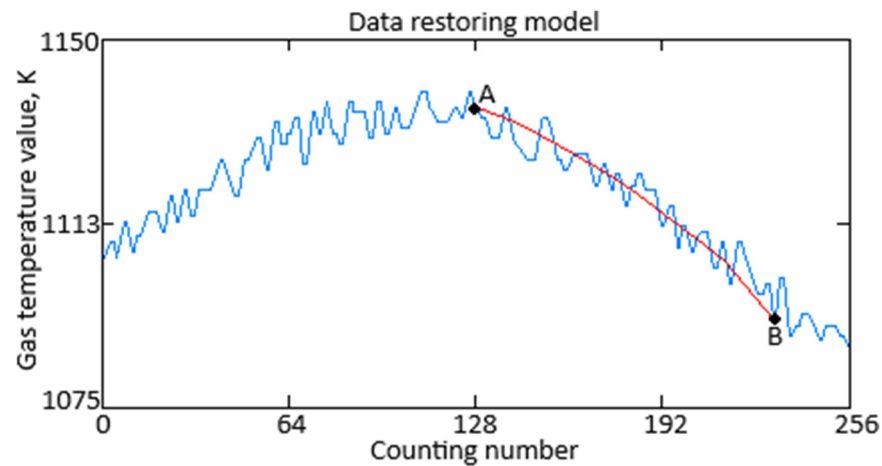


Figure 11. LSTM network operation results in restoring lost information during the work interruption period.

In Figure 11, the blue curve shows the change dynamics in the gas temperature in front of the compressor turbine value under the serviceable sensor condition. The red curve means the data reconstructed by the LSTM neural network in the short-term sensor failure case (point “A” is the sensor failure moment, and point “B” is the sensor’s return-to-serviceability moment). In Figure 11, the LSTM neural network successfully reconstructs the gas temperature in front of the compressor turbine data during the short-term sensor failure period (from point “A” to point “B”). As can be seen, the reconstructed data are smoothed by averaging, eliminating sharp jumps or noise. At the same time, the discrepancies between the reconstructed and original data are minimal and do not exceed 5%, which indicates the neural network model’s high accuracy (to be studied later in the article).

To validate the model, several metrics were utilized, including accuracy, precision, recall, *F1*-score, and AUC-ROC (area under the ROC curve) [38–41]. The accuracy metric is applied (refer to Figure 12) and is computed at a training epoch t using the following equation:

$$Accuracy = \frac{1}{N} \cdot \sum_{i=1}^N \mathbf{1}(y_i = \hat{y}_i), \quad (38)$$

where y_i indicates the valid label for the i -th example, $\hat{y}_i$ signifies the model’s predicted label for that example, N denotes the dataset examples’ total number (either training or validation), and $\mathbf{1}(y_i = \hat{y}_i)$ is the indicator function that returns 1 if $y_i = \hat{y}_i$ and 0 otherwise.

Restoring sensor accuracy readings during forced interruption periods is achieved at the 99.5% rate through LSTM network application. This high accuracy level indicates the model’s effectiveness in capturing temporal dependencies and reconstructing missing data accurately. The results demonstrate LSTM networks’ potential for reliable information recovery in dynamic measurement systems.

During sensor reading restoration in forced interruption periods, losses (Figure 13) are reduced from 2.5 to 0.5% after 100 training epochs with the LSTM network. This significant decrease in losses highlights the model’s ability to train effectively from the training data, leading to improved accuracy in information recovery. The results underscore the extended training efficacy in enhancing LSTM networks’ performance for critical measurement applications.

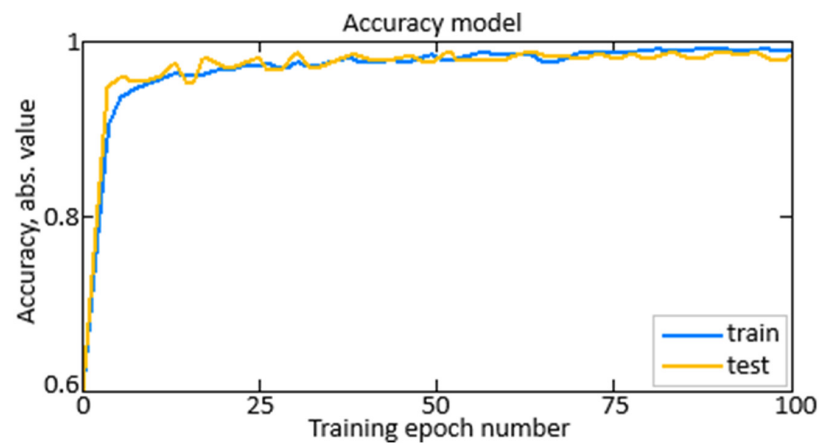


Figure 12. Accuracy metric diagram.

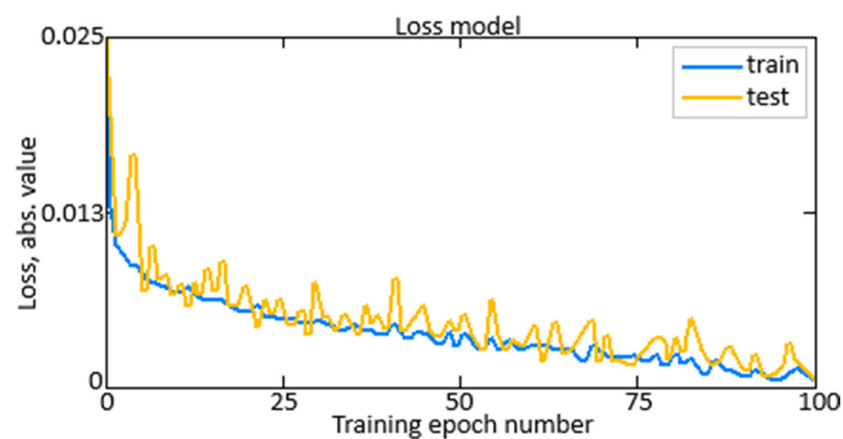


Figure 13. Loss metric diagram.

The metrics precision, recall, and *F1-score* are essential for evaluating information restoration performance during the forced interruption of sensor readings. Precision measures the accurate proportion of restored readings among all predicted readings, indicating the model's ability to avoid false positives. Recall assesses the precise proportion of restored readings among all actual readings, reflecting the model's capability to recover valid values. The *F1-score*, as the harmonic mean of recall and precision, provides a balanced measure of the model's overall effectiveness in accurately restoring sensor data. Precision, recall, and *F1-score* are defined as follows:

$$\begin{aligned}
 \text{Precision} &= \frac{TP}{TP+FP}, \\
 \text{Recall} &= \frac{TP}{TP+FN}, \\
 \text{F1-score} &= \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.
 \end{aligned} \tag{39}$$

In this context, the terms *TP*, *TN*, *FP*, and *FN* represent the following: *TP* (true positives) is the restored sensor readings' correct number that was missing during the interruption; *TN* (true negatives) is the correct number of identified readings that were not missing, indicating accurate predictions for non-interrupted periods; *FP* (false positives) is the incorrect number restored readings that were predicted as missing but were present, indicating false alarms in the restoration process; *FN* (false negatives) is the number of actual missing readings that were not restored, reflecting missed opportunities to recover essential data during the interruption.

The obtained values of *precision* = 0.983, *recall* = 1.0, and *F1-score* = 0.991 indicate a highly effective restoration of sensor readings during forced interruption periods. The

precision value reflects the model's ability to accurately predict missing readings with minimal false positives. At the same time, the perfect *recall* demonstrates its capability to restore all missing data without omissions. The high *F1-score* further underscores the balanced model performance, confirming its reliability in restoring critical sensor information in dynamic conditions.

The results were compared (Table 4) with two alternative approaches: (1) an identical proposed intelligent self-validated sensor system in which the LSTM network is replaced by a GRU (gated recurrent unit) neural network and (2) an approach based on linear neural networks (for example, a multilayer perceptron) [42,43].

Table 4. Comparative analysis results.

Metric	Proposed Approach	Alternative Approach 1	Alternative Approach 2
Accuracy	0.995	0.958	0.724
Precision	0.983	0.951	0.716
Recall	1.0	1.0	0.873
F1-score	0.991	0.975	0.901

As shown in Table 4, replacing the LSTM network with a GRU network results in minor declines in quality metrics, ranging from 3.75 to 4.85%. In contrast, using linear neural networks (multilayer perceptron) is not feasible for restoring sensor task readings during forced interruption periods, as the quality metric values decrease significantly—by up to 27.3%.

This study conducts ROC analysis [44,45] to assess the model's capability in accurately differentiating between classes (such as regular helicopter TE operation and sensor failure), enabling a balance between sensitivity analysis (true positive rate) and specificity (1—false positive rate) at various classification thresholds. The ROC curve is constructed by adjusting the model's classification threshold and evaluating its ability to discriminate between classes. It is a diagram representing the relationship between the false positive rate (FPR) and the true positive rate (TPR). The TPR represents the proportion of correctly identified positive instances among the total positive cases. The FPR refers to the ratio of incorrectly classified negative instances to the total number of actual negative cases. The FPR and TPR are defined as follows:

$$TPR = \frac{TP}{TP + FN}, FPR = \frac{FP}{FP + TN}. \quad (40)$$

The AUC-ROC is a quantitative measure indicating the likelihood that the model assigns a higher score to a randomly chosen positive instance than a randomly chosen negative one. The area beneath the ROC curve is expressed as follows:

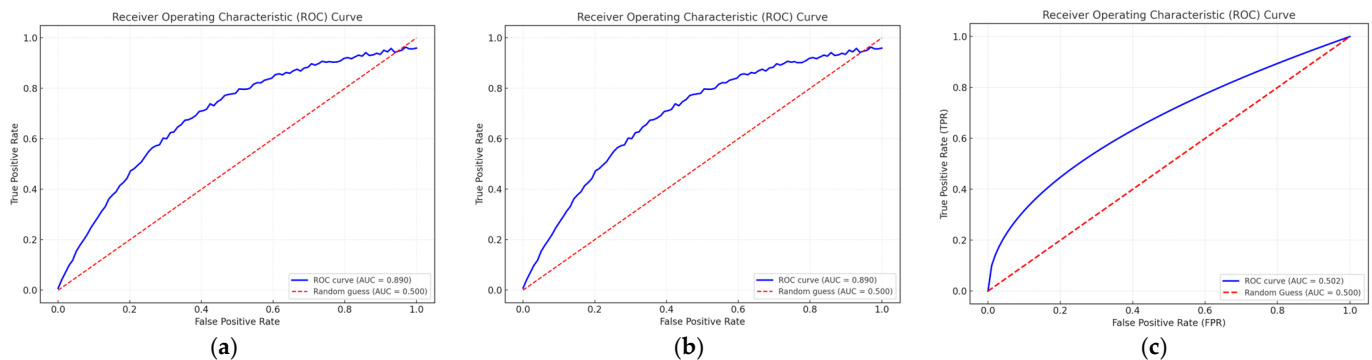
$$AUC - ROC = \int_0^1 TPR \cdot (FPR^{-1}(t)) dt, \quad (41)$$

where *TPR* (true positive rate) is sensitivity, FPR^{-1} is the false positive rate (1 is specificity) inverse function, and *t* is the classification threshold value, varying from 0 to 1.

For ROC analysis within three approaches (the developed intelligent self-validated sensor system based on an LSTM network, alternative approach 1 (replacing the LSTM network with a GRU network), and alternative approach 2 using linear neural networks [42,43]), true positive and false positive rates were calculated for each category and method, followed by the creation of corresponding ROC curves. This procedure involved defining binary classification for each category and isolating it from the others. The ROC analysis results are presented in Table 5. Figure 14 clearly illustrates the variation in the areas beneath the AUC curve.

Table 5. The comparative analysis results (authors' research).

Metric	The Developed Intelligent Self-Validated Sensor System Based on an LSTM Network	Alternative Approach 1 (Replacing the LSTM Network with a GRU Network)	Alternative Approach 2 (Using Linear Neural Networks [42,43])
True Positives	95	90	3
True Negatives	5	10	8
False Positives	280	273	298
False Negatives	21	27	112
True Positive Rate	0.836	0.824	0.573
False Positive Rate	0.0125	0.0132	0.0369
False Negative Rate	0.0113	0.0126	0.0405
AUC-ROC	0.890	0.883	0.502

**Figure 14.** The AUC-ROC curve: (a) the developed intelligent self-validated sensor system based on an LSTM network; (b) alternative approach 1; (c) alternative approach 2.

Based on the obtained AUC-ROC metric values, it was noted that the intelligent self-validated sensor system using an LSTM network and alternative approach 1 (replacing the LSTM network with a GRU network) provide high accuracy with minimal false alarms. The application of alternative approach 2, utilizing linear neural networks [42,43], results in reduced accuracy and the highest number of false alarms.

4. Discussion

This article presents an intelligent self-validated sensor system for dynamic objects based on the intelligent sensor concept (see Figure 1), which involves the autonomous collection, processing, and analysis of data in real time with the ability to adapt to changes in environmental conditions and compensate for measurement errors to enhance system accuracy and reliability.

Unlike conventional intelligent sensors [20,21], the proposed intelligent self-validated sensor integrates advanced functional modules to ensure autonomous operation and enhanced reliability. The Self-Diagnostic Module performs continuous health monitoring and the accuracy verification of all sensor components, promptly identifying operational deviations. The Reference Module supplies baseline values for comparison with real-time measurements, enabling the precise assessment of measurement accuracy. Upon detecting errors, the Error Correction Module intervenes automatically, mitigating their impact on output signals. The Logging Module archives the diagnostic activity history, including error occurrences and resolution processes, facilitating streamlined device analysis and quality assurance. The Calibration Management Unit ensures the sensor's automatic recalibration to maintain measurement precision under fluctuating operating conditions. A key innovation in the proposed sensor architecture is the incorporation of Dynamic Stack Memory [22], adapted from neural network models designed to address complex dynamic object identification in unsteady regimes. This memory model introduces segmentation to optimize memory allocation by dynamically creating new segments during overflow and

merging them when data demands decrease. Unlike static memory, dynamic stack memory adapts resource utilization in real time, reducing overhead from unused memory, which is critical for resource-constrained systems. It also outperforms random access memory (RAM) in sequential data processing tasks by simplifying access to recently stored elements, enhancing performance. Furthermore, in contrast to register memory, stack memory manages larger amounts of temporary data without imposing an excessive load on the processor, making it indispensable for real-time signal processing applications.

An analytical Equation (9) is obtained, describing the main characteristic of the intelligent self-validated sensor, which incorporates correction accounting for the influencing quantity, correction accounting for temporal drift (zero correction depending on sensor operating time), and correction based on calibration characteristics (considering zero offsets for a specific sensor after testing). It significantly improves measurement accuracy, reduces the impact of external factors on results, and ensures the long-term stability of sensor operation by adapting its characteristics to operational conditions and individual features of each device post calibration.

This article applies fuzzy logic algorithms to compensate for errors due to the influencing quantity to refine the scaling factor that corrects the dependence of the measured parameter. It is implemented using fuzzy inference algorithms based on the redundant information contained in the fuzzy sets (see Figure 2). The correction factor value is distributed across the entire range of the influencing quantity, and the scaling factor adjustment is calculated using fuzzy inference algorithms and a set of fuzzy rules as in (10) and (11).

This article proposes implementing the monitoring and diagnosis of the technical condition of sensors in the intelligent self-validated sensor system using two LSTM (Long Short-Term Memory) neural networks (see Figure 3). Introducing an intelligent classifier with a majority element and a decision-making block based on an expert system utilizing fuzzy logic enables the monitoring of the output sensor parameters as an observed object and identifying faulty components in the system.

A computational experiment assessed the effectiveness of the proposed intelligent self-validated sensor system. The TV3-117 engine, widely used in Mi-8MT helicopters due to its extensive operational history and abundant data availability, was the research object, allowing for an accurate evaluation of the system's performance under real conditions.

A virtual instrument (see Figure 5), developed by this team, was employed for the computational experiment. It stimulates the helicopter TE sensor system, including three key sensors that register dynamic thermogas parameters: the D-2M sensor for measuring the gas generator rotor r.p.m., the T-101 sensor with 14 paired thermocouples for registering gas temperatures in the compressor turbine front, and the D-1M sensor for measuring the free turbine rotor speed. It was assumed that the D-2M and D-1M sensors operated correctly, while the T-101 sensor was likely to fail. The training sample (see Tables 2 and 3) was based on the gas temperature in the compressor turbine front signal data (see Figure 6).

Compensation for the error due to the influencing quantity was performed in the first phase of the computational experiment. For example, the case where the current temperature is 1130 K was considered (Figure 8). Here, the input variable leads to a truth degree of 0.4 for set G and 0.6 for set F . These sub-conditions are used in K_6 and K_7 , forming two fuzzy sets. The output variable "scaling coefficient", calculated using the gravity method center for membership functions, yields a value of $K_{scal} = 0.0832$ (see Figure 9).

This study examines the application of neural networks (two LSTM networks) for restoring lost measurement information during forced interruptions caused by failures or malfunctions (see Figure 10). To recover 40 missing values from the 55th to the 95th cycle, 55 values from the time series history before the interruption and 3 values after it are fed into the neural network. Figure 11 demonstrates that the feedforward LSTM network accurately approximates the change in acceleration. The network smooths the input data in terms of outliers or the presence of noise, aiding precise information restoration.

The sensor accuracy reading recovery during forced interruptions reaches 99.5% (see Figure 12) due to the LSTM network application, indicating the model's effectiveness in

considering temporal dependencies and accurately restoring data. Losses during data recovery (see Figure 13) decrease from 2.5% to 0.5% after 150 training epochs of the LSTM network. This significant drop in losses confirms the model's capability to learn effectively and enhance data recovery accuracy, emphasizing the importance of prolonged training for improving LSTM network performance in critical measurement systems.

A comparative analysis of the results was performed with two alternative approaches: 1—replacing the LSTM network with a GRU network in a similar intelligent system; 2—using linear neural networks (multilayer perceptron). The evaluation was based on metrics such as accuracy, precision, recall, *F1-score*, and AUC-ROC (see Table 4). It was established that replacing the LSTM with a GRU leads to a decrease in performance metrics by 3.75–4.85%. Meanwhile, employing linear neural networks for sensor reading recovery during interruptions proves ineffective, as quality metrics drop to 27.3%. According to the obtained AUC-ROC metric values (see Figure 14), the intelligent self-validated sensor system utilizing the LSTM network and alternative approach 1 (LSTM replacement with GRU) demonstrates high accuracy and minimal false alarms. In contrast, using alternative approach 2, based on linear neural networks, results in decreased accuracy and increased false alarm rates.

A key advantage of the obtained results is the system's adaptation to changing conditions and compensation for measurement errors, enhancing accuracy and reliability. The presented analytical Equation (9) describes the main characteristic of the intelligent sensor, encompassing corrections based on the influencing quantity, temporal drift, and calibration characteristics, significantly improving measurement quality and ensuring device stability post calibration. The fuzzy logic used to compensate for errors due to the influencing quantity allows for refining the scaling factor that corrects the measured parameter dependence. Fuzzy inference algorithms based on redundant information in fuzzy sets contribute to a more precise distribution of correction factor values. Additionally, monitoring and diagnosing the sensor conditions using two LSTM neural networks have been implemented, enabling the effective control of output parameters and fault identification.

The practical significance of the developed intelligent self-validated sensor system lies in enhancing the accuracy and reliability of measurements through intelligent self-validated sensors that adapt to changes in external conditions and compensate for errors [46]. The system ensures autonomous data collection and processing in real time, minimizing human error and decreasing the response speed to changes in the objects' state, which is critical for dynamic systems such as aviation engines [47]. Using neural networks to monitor sensor conditions allows for early fault detection, reducing maintenance costs and preventing accidents [48]. Furthermore, the system can be adapted to various fields, including aviation, automotive engineering, and energy, providing high-precision measurements and the analysis of dynamic processes [49,50]. Implementing intelligent self-validated sensor concepts, along with the use of neural networks and fuzzy logic, represents a step toward modern technologies in measurement systems, opening new aspects for further research and development [51,52].

Despite the results, the system has limitations, such as the necessity for high-quality data for neural network training and sensitivity to noise in the data [53]. Future research prospects include improving training algorithms, developing more robust models for operation under external noise conditions, and exploring the possibility of integrating the system into complex dynamic environments [54]. Additionally, it is advisable to consider applying other types of neural networks and hybrid approaches to enhance the overall system efficiency.

5. Conclusions

This article describes creating a smart system for monitoring dynamic objects, leveraging an intelligent self-validated sensor system to autonomously gather, process, and analyze data in real time while adapting to environmental changes and correcting measurement inaccuracies. An analytical equation was used to describe the intelligent self-validated

sensor system's main characteristics, accounting for corrections related to the influencing variable, time drift, and calibration characteristics. These approaches significantly enhance measurement accuracy, minimize the impact of external factors, and ensure long-term stability in sensor operation. Fuzzy logic technology addresses errors in the influencing variable by refining the scaling factor that aligns the measured parameter with its fluctuations. By employing fuzzy inference algorithms and utilizing redundant data from term sets, the correction factor is effectively distributed across the full range of the influencing variable, ensuring enhanced accuracy and stability in the measurement results—essential for the intelligent self-validated system's optimal performance.

This article also proposes the implementation of diagnostics and monitoring for the sensors' technical condition using LSTM neural networks. Integrating a majority-element classifier and a decision-making block based on fuzzy logic allows for monitoring sensor output parameters and identifying faulty system elements. A computational experiment conducted on the TV3-117 engine, providing high accuracy in evaluating system performance, confirms the proposed methods' effectiveness in actual operational conditions.

The analysis of the results indicates that the accuracy in recovering sensor readings during forced interruptions reaches 99.5%, thanks to LSTM networks. Long-term training significantly reduces data recovery losses, demonstrating high model learnability. A comparative analysis with alternative approaches, such as GRU and linear neural networks, shows that the LSTM system delivers superior accuracy metrics and minimal false alarm rates, highlighting its advantages in monitoring and diagnosing sensors' technical condition.

The practical significance lies in the system's applicability in the aviation, automotive engineering, and energy sectors, opening new research avenues. This research establishes a foundation for future studies to improve training algorithms and develop robust models capable of operating in noisy environments, further advancing intelligent self-validated sensor systems. The proposed methodologies also open opportunities for exploring their integration in diverse applications, such as aviation, automotive engineering, and energy, to enhance system reliability and performance.

Author Contributions: Conceptualization, S.V.; methodology, S.V.; software, V.V.; validation, V.V. and L.C.; formal analysis, S.V.; investigation, V.V. and L.C.; resources, V.S. and O.M.; data curation, V.V.; writing—original draft preparation, S.V.; writing—review and editing, V.V. and L.C.; visualization, S.V. and V.V.; supervision, V.S. and O.M.; project administration, V.S. and O.M.; funding acquisition, V.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Acknowledgments: This research was supported by the Ministry of Internal Affairs of Ukraine through the project "Theoretical and applied aspects of the development of the aviation sphere" under Project No. 0123U104884. The research was carried out with the grant support of the National Research Fund of Ukraine, "Methods and means of active and passive recognition of mines based on deep neural networks", project registration number 273/0024 from 1/08/2024 (2023.04/0024).

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

LSTM network	is the Long Short-Term Memory network;
TE	is the turboshaft engine;
RAM	is the random access memory;
MSE	is the mean squared error;
LabVIEW	is the laboratory virtual instrumentation engineering workbench;
AUC-ROC	is the area under the ROC curve;
GRU	is the gated recurrent unit neural network.

References

1. Coutinho, F.; Lizarralde, N.; Mendes, M.; Bostrom, R.; Silva, T.; Couto, M.; Lizarralde, F. iWAAM: An Automated System for Monitoring and Control of Wire-Arc Additive Manufacturing. *IFAC-PapersOnLine* **2023**, *56*, 6576–6581. [\[CrossRef\]](#)
2. Yuan, Y.; Au, F.T.K.; Yang, D.; Zhang, J. Active Learning Guided Automated Cable Force Monitoring Based on Modified S-Transform. *Measurement* **2024**, *224*, 113880. [\[CrossRef\]](#)
3. Sæternes, E.H.; Thune, A.; Rustad, A.B.; Skeie, T.; Cai, X. Automated Parameter Tuning with Accuracy Control for Efficient Reservoir Simulations. *J. Comput. Sci.* **2024**, *75*, 102205. [\[CrossRef\]](#)
4. Vladov, S.; Yakovliev, R.; Bulakh, M.; Vysotska, V. Neural Network Approximation of Helicopter Turboshift Engine Parameters for Improved Efficiency. *Energies* **2024**, *17*, 2233. [\[CrossRef\]](#)
5. Vladov, S.; Yakovliev, R.; Hubachov, O.; Rud, J. Neuro-Fuzzy System for Detection Fuel Consumption of Helicopters Turboshift Engines. *CEUR Workshop Proc.* **2024**, *3628*, 55–72. Available online: <https://ceur-ws.org/Vol-3628/paper5.pdf> (accessed on 19 August 2024).
6. Paek, E.S.; Fan, T.; Finch, J.D.; Choi, J.D. Enhancing Task-Oriented Dialogue Systems through Synchronous Multi-Party Interaction and Multi-Group Virtual Simulation. *Information* **2024**, *15*, 580. [\[CrossRef\]](#)
7. Su, Y.; Wang, M.C.; Liu, S. Automated Machine Learning Algorithm Using Recurrent Neural Network to Perform Long-Term Time Series Forecasting. *Comput. Mater. Contin.* **2024**, *78*, 3529–3549. [\[CrossRef\]](#)
8. Riedel, H.; Mokdad, S.; Schulz, I.; Kocer, C.; Rosendahl, P.L.; Schneider, J.; Kraus, M.A.; Drass, M. Automated Quality Control of Vacuum Insulated Glazing by Convolutional Neural Network Image Classification. *Autom. Constr.* **2022**, *135*, 104144. [\[CrossRef\]](#)
9. Liu, X.; Du, J.; Cheng, X.; Zhu, Y.; Ma, J. An Adaptive Shift Schedule Design Method for Multi-Gear AMT Electric Vehicles Based on Dynamic Programming and Fuzzy Logical Control. *Machines* **2023**, *11*, 915. [\[CrossRef\]](#)
10. Yao, K.-C.; Lin, C.-L.; Pan, C.-H. Industrial Sustainable Development: The Development Trend of Programmable Logic Controller Technology. *Sustainability* **2024**, *16*, 6230. [\[CrossRef\]](#)
11. Mystkowski, A.; Wolniakowski, A.; Idzkowski, A.; Ciezkowski, M.; Ostaszewski, M.; Kociszewski, R.; Kotowski, A.; Kulesza, Z.; Dobrzański, S.; Miastkowski, K. Measurement and Diagnostic System for Detecting and Classifying Faults in the Rotary Hay Tedder Using Multilayer Perceptron Neural Networks. *Eng. Appl. Artif. Intell.* **2024**, *133*, 108513. [\[CrossRef\]](#)
12. Facchini, L.; Betti, M.; Biagini, P. Neural Network Based Modal Identification of Structural Systems through Output-Only Measurement. *Comput. Struct.* **2014**, *138*, 183–194. [\[CrossRef\]](#)
13. Alateeq, M.; Pedrycz, W. Logic-Oriented Fuzzy Neural Networks: A Survey. *Expert Syst. Appl.* **2024**, *257*, 125120. [\[CrossRef\]](#)
14. Wu, Y.-J.; Chen, M.-C.; Hwang, W.-S.; Cheng, M.-H. Dynamic Routing Using Fuzzy Logic for URLLC in 5G Networks Based on Software-Defined Networking. *Electronics* **2024**, *13*, 3694. [\[CrossRef\]](#)
15. El-Shafei, E.; Elsayed, W.M.; Elwahsh, H.; Alsabaan, M.; Ibrahim, M.I.; Elhady, G.F. Deep Complex Gated Recurrent Networks-Based IoT Network Intrusion Detection Systems. *Sensors* **2024**, *24*, 5933. [\[CrossRef\]](#) [\[PubMed\]](#)
16. Li, L.; Lu, Y.; Yang, G.; Yan, X. End-to-End Network Intrusion Detection Based on Contrastive Learning. *Sensors* **2024**, *24*, 2122. [\[CrossRef\]](#) [\[PubMed\]](#)
17. Fracczak, L.; Podsedkowski, L.; Kobierska, A. Data Fusion Using Fuzzy Logic Techniques Supported by Modified Weighting Factors (FLMW). *Int. J. Fuzzy Syst.* **2015**, *18*, 72–80. [\[CrossRef\]](#)
18. Ishaque, M.; Johar, M.G.M.; Khatibi, A.; Yamin, M. A Novel Hybrid Technique Using Fuzzy Logic, Neural Networks and Genetic Algorithm for Intrusion Detection System. *Meas. Sens.* **2023**, *30*, 100933. [\[CrossRef\]](#)
19. Dawy, I. The Most General Intelligent Architectures of the Hybrid Neuro-Fuzzy Models. *Am. J. Artif. Intell.* **2018**, *2*, 1–6. [\[CrossRef\]](#)
20. Kibrete, F.; Engida Woldemichael, D.; Shimels Gebremedhen, H. Multi-Sensor Data Fusion in Intelligent Fault Diagnosis of Rotating Machines: A Comprehensive Review. *Measurement* **2024**, *232*, 114658. [\[CrossRef\]](#)
21. Mohammadi, H.; Thirunarayan, K.; Chen, L. CVII: Enhancing Interpretability in Intelligent Sensor Systems via Computer Vision Interpretability Index. *Sensors* **2023**, *23*, 9893. [\[CrossRef\]](#)
22. Vladov, S.; Banasik, A.; Sachenko, A.; Kempa, W.M.; Sokurenko, V.; Muzychuk, O.; Pikiewicz, P.; Molga, A.; Vysotska, V. Intelligent Method of Identifying the Nonlinear Dynamic Model for Helicopter Turboshift Engines. *Sensors* **2024**, *24*, 6488. [\[CrossRef\]](#) [\[PubMed\]](#)
23. Wang, Y.; Zhang, X.; Hu, H. Continuous User Authentication on Multiple Smart Devices. *Information* **2023**, *14*, 274. [\[CrossRef\]](#)
24. Castellões, T.d.O.; Rizol, P.M.S.R.; Nascimento, L.F.C. Association between Premature Birth and Air Pollutants Using Fuzzy and Adaptive Neuro-Fuzzy Inference System (ANFIS) Techniques. *Mathematics* **2024**, *12*, 2828. [\[CrossRef\]](#)
25. Tightiz, L.; Yoo, J. Quantum-Fuzzy Expert Timeframe Predictor for Post-TAVR Monitoring. *Mathematics* **2024**, *12*, 2625. [\[CrossRef\]](#)
26. Vladov, S.; Vysotska, V.; Sokurenko, V.; Muzychuk, O.; Nazarkevych, M.; Lytvyn, V. Neural Network System for Predicting Anomalous Data in Applied Sensor Systems. *Appl. Syst. Innov.* **2024**, *7*, 88. [\[CrossRef\]](#)
27. Chander, G.P.; Das, S. Chaotic Rao3 Based Adaptive Neuro-Fuzzy Inference System to Solve Global Infrastructure Project Selection Problem. *Appl. Soft Comput.* **2024**, *165*, 112046. [\[CrossRef\]](#)
28. Gebrehiwet, L.; Nigussei, Y.; Teklehaymanot, T. A Review-Differentiating TV2 and TV3 Series Turbo Shaft Engines. *Int. J. Res. Publ. Rev.* **2022**, *3*, 1822–1838. [\[CrossRef\]](#)
29. Catana, R.M.; Dediu, G. Analytical Calculation Model of the TV3-117 Turboshift Working Regimes Based on Experimental Data. *Appl. Sci.* **2023**, *13*, 10720. [\[CrossRef\]](#)

30. Vladov, S.; Yakovliev, R.; Vysotska, V.; Nazarkevych, M.; Lytvyn, V. The Method of Restoring Lost Information from Sensors Based on Auto-Associative Neural Networks. *Appl. Syst. Innov.* **2024**, *7*, 53. [\[CrossRef\]](#)
31. Vladov, S.; Scislo, L.; Sokurenko, V.; Muzychuk, O.; Vysotska, V.; Osadchy, S.; Sachenko, A. Neural Network Signal Integration from Thermogas-Dynamic Parameter Sensors for Helicopters Turboshift Engines at Flight Operation Conditions. *Sensors* **2024**, *24*, 4246. [\[CrossRef\]](#)
32. Kim, H.-Y. Statistical Notes for Clinical Researchers: Chi-Squared Test and Fisher's Exact Test. *Restor. Dent. Endod.* **2017**, *42*, 152. [\[CrossRef\]](#)
33. Stefanovic, C.M.; Armada, A.G.; Costa-Perez, X. Second Order Statistics of -Fisher-Snedecor Distribution and Their Application to Burst Error Rate Analysis of Multi-Hop Communications. *IEEE Open J. Commun. Soc.* **2022**, *3*, 2407–2424. [\[CrossRef\]](#)
34. Vladov, S.; Shmelov, Y.; Yakovliev, R.; Petchenko, M.; Drozdova, S. Neural Network Method for Helicopters Turboshift Engines Working Process Parameters Identification at Flight Modes. In Proceedings of the 2022 IEEE 4th International Conference on Modern Electrical and Energy System (MEES), Kremenchuk, Ukraine, 20–22 October 2022; IEEE: Piscataway, NJ, USA. [\[CrossRef\]](#)
35. Vladov, S.; Shmelov, Y.; Yakovliev, R.; Stushchankyi, Y.; Havryliuk, Y. Neural Network Method for Controlling the Helicopters Turboshift Engines Free Turbine Speed at Flight Modes. *CEUR Work. Proc.* **2023**, *3426*, 89–108. Available online: <https://ceur-ws.org/Vol-3426/paper8.pdf> (accessed on 10 September 2024).
36. Hu, Z.; Kashyap, E.; Tyshchenko, O.K. GEOCLUS: A Fuzzy-Based Learning Algorithm for Clustering Expression Datasets. *Lect. Notes Data Eng. Commun. Technol.* **2022**, *134*, 337–349. [\[CrossRef\]](#)
37. Babichev, S.; Krejci, J.; Bicanek, J.; Lytvynenko, V. Gene expression sequences clustering based on the internal and external clustering quality criteria. In Proceedings of the 2017 12th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT), Lviv, Ukraine, 5–8 September 2017; IEEE: Piscataway, NJ, USA. [\[CrossRef\]](#)
38. Andriushchenko, K.; Rudyk, V.; Riabchenko, O.; Kachynska, M.; Marynenko, N.; Shergina, L.; Kovtun, V.; Tepliuk, M.; Zhemba, A.; Kuchai, O. Processes of managing information infrastructure of a digital enterprise in the framework of the «Industry 4.0» concept. *East.-Eur. J. Enterp. Technol.* **2019**, *1*, 60–72. [\[CrossRef\]](#)
39. Hu, Z.; Dychka, I.; Potapova, K.; Meliukh, V. Augmenting Sentiment Analysis Prediction in Binary Text Classification through Advanced Natural Language Processing Models and Classifiers. *Int. J. Inf. Technol. Comput. Sci.* **2024**, *16*, 16–31. [\[CrossRef\]](#)
40. Morozov, V.V.; Kalnichenko, O.V.; Mezentseva, O.O. The method of interaction modeling on basis of deep learning the neural networks in complex IT-projects. *Int. J. Comput.* **2020**, *19*, 88–96. [\[CrossRef\]](#)
41. Cherrat, E.M.; Alaoui, R.; Bouzahir, H. Score fusion of finger vein and face for human recognition based on convolutional neural network model. *Int. J. Comput.* **2020**, *19*, 11–19. [\[CrossRef\]](#)
42. Alviano, M.; Bartoli, F.; Botta, M.; Esposito, R.; Giordano, L.; Theseider Dupré, D. A Preferential Interpretation of MultiLayer Perceptrons in a Conditional Logic with Typicality. *Int. J. Approx. Reason.* **2024**, *164*, 109065. [\[CrossRef\]](#)
43. Yang, Q.; Tang, B.; Li, Q.; Liu, X.; Bao, L. Dual-Frequency Enhanced Attention Network for Aircraft Engine Remaining Useful Life Prediction. *ISA Trans.* **2023**, *141*, 167–183. [\[CrossRef\]](#) [\[PubMed\]](#)
44. Anfilets, S.; Bezobrazov, S.; Golovko, V.; Sachenko, A.; Komar, M.; Dolny, R.; Kasyanik, V.; Bykovyy, P.; Mikhno, E.; Osolinskyi, O. Deep multilayer neural network for predicting the winner of football matches. *Int. J. Comput.* **2020**, *19*, 70–77. [\[CrossRef\]](#)
45. Marakhimov, A.R.; Khudaybergenov, K.K. Approach to the synthesis of neural network structure during classification. *Int. J. Comput.* **2020**, *19*, 20–26. [\[CrossRef\]](#)
46. Yang, L. Measurement of High Quality Development of Manufacturing Industry Empowered by Big Data Based on Intelligent Sensor Systems. *Meas. Sens.* **2024**, *33*, 101092. [\[CrossRef\]](#)
47. Zhou, B.; Xie, S.; Wang, F.; Hui, J. Multi-Step Predictive Compensated Intelligent Control for Aero-Engine Wireless Networked System with Random Scheduling. *J. Frankl. Inst.* **2020**, *357*, 6154–6174. [\[CrossRef\]](#)
48. Lan, J.; Zhan, S.; Patton, R.; Zhao, X. Runtime Monitoring and Fault Detection for Neural Network-Controlled Systems. *IFAC-PapersOnLine* **2024**, *58*, 258–263. [\[CrossRef\]](#)
49. Yin, Z.; He, X.; Li, H.; Li, Y. Design of Real-Time Measurement and Intelligent Path Selection System for Network Quality under Big Data. *Procedia Comput. Sci.* **2024**, *243*, 51–58. [\[CrossRef\]](#)
50. Wang, Y.; Zeng, J.; Du, P.; Xu, H. Intelligent Gear Decision Method for Vehicle Automatic Transmission System Based on Data Mining. *Intell. Syst. Appl.* **2024**, *24*, 200459. [\[CrossRef\]](#)
51. Liu, B.; Li, S.; Wen, Y.; Xie, Z.; Zhang, M.; Cheng, Z.; Liu, D.; Jia, C.; Sun, F. Papermaking-Inspired Sustainable Triboelectric Sensors for Intelligent Detecting System. *Nano Energy* **2024**, *131*, 110322. [\[CrossRef\]](#)
52. Cheng, Z.-Y.; Sun, Y.; Hu, K.; Li, J.; Lu, T.-F.; Li, R.-J. Deep Learning-Based Intelligent Measurement Methods and System for CMM. *Measurement* **2023**, *221*, 113474. [\[CrossRef\]](#)
53. Dinčić, M.R.; Perić, Z.H.; Denić, D.B.; Denić, B.D. Optimization of the Fixed-Point Representation of Measurement Data for Intelligent Measurement Systems. *Measurement* **2023**, *217*, 113037. [\[CrossRef\]](#)
54. Mustafa Hilal, A.; Al-Wesabi, F.N.; Althobaiti, M.M.; Al Duhayyim, M.; Ahmed Hamza, M.; Kadry, S.; Rizwanullah, M. An Intelligent Deep Learning Based Hyperspectral Signal Classification Scheme for Complex Measurement Systems. *Measurement* **2022**, *188*, 110540. [\[CrossRef\]](#)

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.