OPEN ACCESS

KJMR

Kashf Journal of Multidisciplinary Research
Vol: 01 - Issue 12 (2024)
P-ISSN: 3007-1992 E-ISSN: 3007-200X
https://kjmr.com.pk

DESIGN AND IMPLEMENTATION OF ERROR ISOLATION IN TECHNO METER

Maria Ashraf
Information Technology Department, The City School, Multan, Pakistan.
Arslan Jalil
Department of Computer Science, Govt. Graduate College, D. G. Khan, Pakistan.
Assad Latif
School of management and engineering North China University of water resources and electric power Zhengzhou Henan, China
Salahuddin
Department of Computer Science, NFC Institute of Engineering and technology, Multan, Pakistan.
Furqan Jamil
Department of Computer Science, National College of Business Administration and Economics, Multan, Pakistan.
DOI: *https://doi.org/10.71146/kjmr150*

Article Info

Abstract

From heavy machinery to air conditioning, everything is getting automated and mechanized. Sensors facilitate automation. Sensors are utilized by industries and farms to collect data. Not only do agriculture and industry rely on data collecting and analysis; many other fields do as well. Sensors determine data accuracy. Typically, sensors fail at risk. Early sensor detection diminishes damage. This study discusses contemporary ways of fault-finding for researchers and professionals. This study uses the Kalman filter because it minimizes mean-squared error when estimating the state of the process. Even if the system being model is obscure, the filter can estimate past, present, and future states. Use a discrete Kalman filter. The discrete and extended Kalman filters are covered in this introduction. In addition, we will analyze a simple physical example using actual numbers.

Keywords: *Kalman Filter, Machine Learning, Isolation, Sensors. Techno meter*

Introduction

As technology becomes better, engineering systems get better. The more intricate the system is, the higher the risk that something may go wrong with it. Techniques known as Fault Detection and Isolation (FDI) need to be utilized in order to locate faults as early as possible, isolate them, and then correct them. This must be done in order to maintain the integrity of the system in the event that something goes awry [1]. It is possible for fault detection, isolation, and identification technology to play a significant role in the improvement of system availability, safety, and reliability, as well as in the reduction of maintenance costs and the risk of catastrophic failures. FDII stands for fault detection, isolation, and identification. Over the course of the past few years, there has been a significant increase in the number of researchers who have focused their efforts on the development of complex fault detection and isolation (FDI) schemes. These schemes are a significant component of an FDII solution and have been the primary focus of the majority of these researchers. The standard operating procedure for an FDII is comprised of a number of significant steps, the most important of which are as follows: (a) the generation of residuals as indicators of faults; (b) the isolation of the faulty actuator, sensor, or component element; and (c) the identification and estimation of the severity of the fault parameter. [2]. Sensors are utilized extensively throughout the process of obtaining the necessary information. Because in the realm of the Internet of things (IoT), the entire system is reliant on sensors and the data that is produced by them. In most cases, however, sensors are the ones carrying out the monitoring, particularly with regard to the monitoring of the environment. Monitoring of aquaculture systems, disease detection, and use in industrial settings. For instance, in an agricultural system, sensors are designed to collect information regarding the level of moisture in the soil, as well as information regarding the nutrients and temperature, in order to determine the state of the crop's health and the quality of its output. Now if you look at the smaller picture or take the example of a smart home, the air conditioner (AC) is turned ON and OFF based on the data that is being provided by a temperature sensor. The use of sensors enables intelligent control and the support of precise agricultural practices. In general, sensors are installed in very harsh environments, which can include high temperatures and even being submerged in water. These conditions can cause the sensors to sustain mechanical damage and make them more prone to faults, which in turn will make them less accurate and unstable. One example of this can be seen in the power field, where the tachometer that measures the speed of the rotor is large installed generators is obviously available for any kind of mechanical injury as well. In the event that the sensors develop faults, the performance of the system could be negatively impacted, which could have catastrophic consequences [3]. Thanks to advances in machine learning algorithms and controllers, autonomous vehicles are now common in big cities. These strategies depend on sensor data. Sensor malfunctions and error propagation can damage the dependability and safety of autonomous cars. Monitoring and prognostics for sensor health are crucial. A sensor health monitoring system tries to predict the performance and dependability of sensors and make decisions on such predictions.[4]. Advanced condition-based maintenance focuses on prognostics and health management. More productivity, maintainability, and dependability are needed. This strategy was established due to rising demand for all three (PHM). Optimized reliability depends on PHM, which uses predictive data. This follows what was said. Compared to model-based PHM systems, these data-driven solutions are easier to use, have a lower false alarm rate, and consume less processing power. The emergence of AI, ML, and DL-based technologies has led to bias against traditional statistical model-based FDI approaches. AI, ML, and DL-based approaches learn deeper than statistical models (FDI)[5].

1.1 Motivation and novelty

- **Industrial perspective**: This strategy has the potential to improve the quality of both operation and maintenance, as well as provide early warning in the event that a piece of equipment fails, preventing accidents from occurring or escalating in severity. Additionally, it makes it simple to make rapid adjustments in the event that something goes wrong. This prevents any long-term damage from occurring and provides the wind energy industry with technological guarantees that it can continue to expand. Because of this, having fault monitoring, fault diagnosis, and fault-tolerant control in the blade pitch system is of the utmost importance. On the basis of the diagnostic observers, a system for finding faults in the blade pitch system and isolating them was developed. In particular, a condition monitoring (CM) method of pitch systems in wind turbines has been emphasised as a way to watch how the blade pitch works. This method makes use of a supervisory control and data acquisition (SCADA) system.
- **Domestic perspective:** A sensor takes the physical activity that needs to be measured and translates it into an electrical equivalent before processing it. This makes it possible for the electrical signals to be easily sent and subsequently processed. The binary information that an object is present or not, as well as the measurement value that has been reached, can be output by the sensor depending on the situation (analog or digital).

In this research objectives are as follows:

- To show drift error, also known as errors, which refers to the change in a sensor's low frequency over time.
- To provide the system with information regarding the state of the sensor.
- Use Kalman filter model (KFM) approach is used to come up with a strategy for error isolation.

1. Literature Review

In previous studies, an investigation was conducted into the process of implementing a NumPy module for a Kalman Filter in the computer programming language Python. The Kalman Filtering technique can be broken down into two distinct stages: the Prediction stage, and the Update stage. Following an evaluation of each stage, a coding structure in the form of a matrix-input/output function is then allocated to it. This article demonstrates how a Kalman Filter can be used to solve the challenge of localizing mobile devices in wireless networks. They presented an illustration of utilizing this kind of filter for the goal of localization in wireless networks, whereas we demonstrated a technique that involved two steps. The subsequent step of this process will entail the installation of certain additional Bayesian filters. These filters include the Extended Kalman Filter, the Unscented Kalman Filter, and the Particle Filter. It is feasible that a third phase of estimating smoothing will be adopted in the future [14]. This research presents a novel method for the defect detection, isolation, and identification/estimation (FDII) of gas turbine engines, which can be applied to either a single sensor or many sensors simultaneously. Fault detection, isolation, and identification/estimation are what are meant by the abbreviation FDII. Our technique is based on a hierarchical multiple-model structure, and the generated Hybrid Kalman Filter (HKF) acts as the detection filter for that structure. It also acts as the foundation for our methodology. The HKF is still capable of capturing the nonlinearities of the system despite the fact that linear Kalman filters are utilised in the analysis. This is achieved by combining a piecewise linear (PWL) model with a nonlinear on-board engine model (OBEM) in order to cover the complete operating range of the engine. This is referred to as covering the entire engine's operating range. Because of this, the HKF is able to take into account the nonlinearities of the system. Although each individual operating point corresponds to a greater operating range, in comparison to the multiple linear Kalman filter (MLKF), the method that we have proposed requires a lower overall number of operating points. This is despite the fact that the total number of operating points is lower. Another notable addition made by this work is the incorporation into our suggested sensor FDII system of the consequences of health parameter degradations. This is an important step forward in the research. This was accomplished by upgrading the reference baselines for OBEM health parameters, which enables it to be possible to eliminate false alarms and inaccurate defect detections. OBEM [2]. Autonomous driving research [4] requires precision and dependability. Faulty sensors cause certain algorithms to fail. Errors kill. Automated vehicles must predict problems. Machine learning models are trained using real-world data and sensor simulations. This study reveals flaws. Forecasting, isolating, and identifying multi-fault multi-sensor autonomous cars. Our system uses two effective DNN topologies. Researchers provide a health index based on a sensor defect detection system. Combining the stated architecture with deep learning techniques enables self-driving cars to find and separate problems. Applying DNN models to multi-sensor control systems led to major discoveries. Even when the models were subjected to several types of faults, such as the drift fault, these discoveries displayed great isolation and identification accuracy for a range of sensors. This was true despite many model errors. Depending on the problem and sensor, the multi-class DNN fault identification system's performance ranged from 73% to 100%. After determining that the sensors are healthy, the signals are transferred to the system that checks their health. This process repeats until no sensors are faulty. The sensor fault detection system was used to build an HI measure by considering three ways the issue could worsen. This is done to make an educated guess about the sensors' status. The TFT network uses the HI measure to forecast how the sensors will behave in the future and identify potential problems. Quantile loss measures how well anything in this study performs. P10, P50, and P90 quantiles lost 0.0315, 0.0611, and 0.0299, respectively [4]. In research [5], accurate and reliable autonomous driving is needed. Faulty sensors can cause algorithm failure. Killing errors. Automated automobiles must predict problems. Machine learning algorithms are trained using real-world and synthetic sensor data. This investigation shows flaws. Identifying, forecasting, and locating errors in multisensory autonomous cars it uses two DNN architectures. Researchers provide a sensor fault-based health index. Combining the above architecture with deep learning allows self-driving cars to discover and categories problems. DNN models used to multi-sensor control systems revealed fresh information. These results showed remarkable isolation and identification accuracy for a range of sensors, even with faults like the drift fault. Despite model errors, this was correct. Depending on the issue and sensor, the multi-class DNN fault identification system's performance ranged from 73% to 100%. Once the sensors are in good condition, the signals are transferred to the system that evaluates them. This process will continue until no sensors are broken. The sensor fault detection system was used to create an HI measure by considering the three scenarios in which the problem could worsen. This allows an educated guess about the sensors' status. The TFT network uses HI to forecast sensor activity and identify potential faults. Quantile loss can be used to evaluate study results. P10, P50, and P90 lost 0.0315, 0.0611, and 0.299. In this study [15], Data-driven subsets will be constructed to isolate sensor problems. This increases the isolation's reliability. Using linear regression, erroneous signals and primary residuals can be estimated simultaneously. Both jobs are simultaneous. L1-regularized LSE estimates model parameters and sparseness. Raising the regularization weight did this. This method lets you build a string of residuals generators, each with its own fault sensitivity. The residuals generators' fault sensitivity allows this. Then, residual selection is used to help isolate the damaged sensor by boosting fault sensitivity. This approach recovers certain structured residuals. Strong, online, recursive. The Bayesian Filter calculates the residuals' distance from the nominal fault directions to evaluate the likelihood of a sensor fault. This technique was developed to improve modelling clarity. Six aircraft sensors were utilized to evaluate a defect isolation strategy. Multiple P92 Tecnam flights were investigated. This challenge was completed using aero plane data. LASSO's detection and isolation results are comparable to or better than LS-in RB's typical noise. This happens even when both systems are running normally. Our method beats LS-RB in noisy sensor situations. Few LASSO residual repressors. Because our method employs fewer sensor data than LS-RB, it's less sensitive to sensor noise. Our solution leverages lower fault detection thresholds while maintaining the same fault probability ratio (FPR). This is crucial for real-time fault isolation. Early process problem diagnosis and plant safety and dependability are key to continued development. Oil exploration companies must monitor their Geophone string sensors to avoid losses (SG-10). Methods are being developed that will enable for earlier detection of process problems than typical limit and trend checks based on a single process variable. Utilizing low-computational power devices such as the Raspberry Pi 4, in this [1] research, a number of pattern recognition strategies are tested for their effectiveness in identifying drift errors in sensor readings, and the results are analyzed and compared. Pattern recognition research is being conducted on a wide variety of different approaches, some of which include the Support Vector Machine (SVM), Artificial Neural Networks (ANNs), and K-Nearest Neighbor (KNN) classifiers. These are just a few examples. The findings of this investigation relied on the information that was acquired from (SG-10). In order to determine the four characteristics, an analysis was carried out on sensors that were either healthy or damaged, and these sensors were included in both the training data and the testing data (resistance, noise, leakage, and tilt). Offline testing on trained models has been done in order to detect drift problems in the performance of sensors utilizing models. The performance of the sensor had to be evaluated through the course of these tests. When determining which of the many algorithms were the most effective, we took into account not only their specificity and sensitivity but also their accuracy. The KNN algorithm managed to obtain an accuracy rate of 98 percent, the ANN algorithm managed to get an accuracy rate of 98.3 percent, and the SVM algorithm managed to achieve an accuracy rate of 62 percent. The results indicated that the ANN and the KNN classifiers performed noticeably better than the others when measured against the datasets that were investigated (SVM).

2. Proposed Methodology

Putting an end to **faulty reading** has become a necessity. The quality of data-driven sensor for predicting **accurate**

reading can enhance the overall research and safety process, thereby ensuring that many individuals sensor can lead better accuracy. Thus, Machine Learning (ML) enters the picture. We will employ Kalman Filter algorithm to isolate **faulty** reading as it generates accurate and straightforward predictions. In the following stage, known as Data visualization, we did statistical analyses on our Dataset, such as Power vs. Speed, Cluster Map, etc.

3.1. Research Design

We have analyzed the **COVID19 Disease** patient dataset with the appropriate data preprocessing. In the preprocessing phase, we examined the missing values and noisy data. Then, many ML models were trained and implemented. Predictions are made using Logical Regression, K-Nearest Neighbor, Decision Tree, **Random Forest**, Support Vector Machine, etc. Each stage is described in greater detail below.

3.2. Data Set Collection

We obtained the Dataset from *Estimation of Sensor Fault Using Machine Learning*; which is already done by university fellow. The dataset includes **02** features (Speed (pu), Power(pu)). Our Dataset contains **99** samples in total. Figure 01 demonstrates the details of a dataset.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 99 entries, 0 to 98
Data columns (total 2 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   Power(pu)  99 non-null     float64
 1   Speed(pu)  99 non-null     float64
dtypes: float64(2)
memory usage: 1.7 KB
```

Figure 1 Dataset Detail

3.2.1 Data Description

There have been numerous disagreements over the description of Faulty reading information in prior studies. As previously established, there are multiple different types of metadata employed in past studies. We have utilized the metadata description here.

	Power(pu)	Speed(pu)
count	99.000000	9.900000e+01
mean	0.748811	1.046415e+00
std	0.000861	4.732783e-01
min	0.747251	1.605541e-08
25%	0.748073	1.000000e+00
50%	0.748893	1.000000e+00
75%	0.749589	1.000001e+00
max	0.750144	4.000000e+00

Figure 2 Describe Dataset Attributes

Figure 02 displays our Dataset's information in terms of the total number of records (**99**), determines the mean, standard deviation, minimum and maximum values, and **25** percent, **50** percent, and **75** percent of the Dataset.

3.3 Null Value

This stage requires identifying missing values for each attribute of a dataset. Below, **Figure 4** shows that Power and Speed attributes have no null values in them

```
Power(pu)    0
Speed(pu)    0
dtype: int64
```

Figure 3 Null values in Dataset

We have used the most popular Heat Map tool to identify the properties of the Dataset that contain null values. We plotted the bar graph for null values using the Seaborn library for this aim. **Figure 3** depicts the result of the Heat Map. The chart demonstrates that a Power, and Speed have no null values in it.

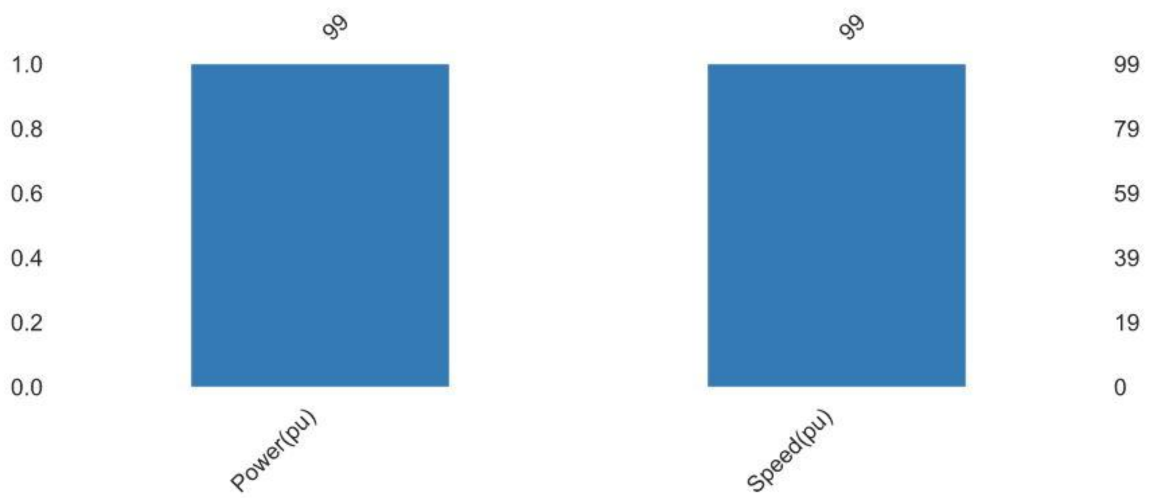


Figure 4 Heat Map Result for Null Values

3.4 Data Profiling

We have utilized the Pandas Profiling Library to describe our Dataset in further detail in terms of Overview, Variables, Missing Value, and Correlations.

3.4.1 Overview

According to the Dataset statistics, **0.0%** of each attribute in our Dataset is without a value. There are **02** attributes in total. The ratio of duplicate rows to total rows is **0.0%.** Our Dataset has one datatype attributes; both attributes are type of numeric.

Overview | Alerts 6 | Reproduction

Dataset statistics

Number of variables	2
Number of observations	99
Missing cells	0
Missing cells (%)	0.0%
Duplicate rows	0
Duplicate rows (%)	0.0%
Total size in memory	1.7 KiB
Average record size in memory	17.3 B

Variable types

Numeric	2

Figure 5 Overall Dataset Statistics

3.4.2 Variables Analysis

In this section, we examine each attribute of the Dataset in detail.

3.4.2.1 Power Attribute

Speed(pu)
Real number ($\mathbb{R}_{\geq 0}$)

HIGH CORRELATION
HIGH CORRELATION
UNIQUE

Distinct	99	Minimum	1.605541×10^{-8}
Distinct (%)	100.0%	Maximum	4.000000143
Missing	0	Zeros	0
Missing (%)	0.0%	Zeros (%)	0.0%
Infinite	0	Negative	0
Infinite (%)	0.0%	Negative (%)	0.0%
Mean	1.046414503	Memory size	920.0 B

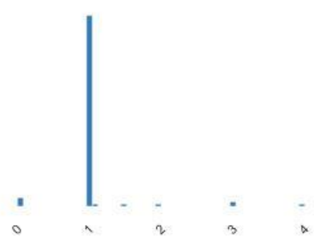
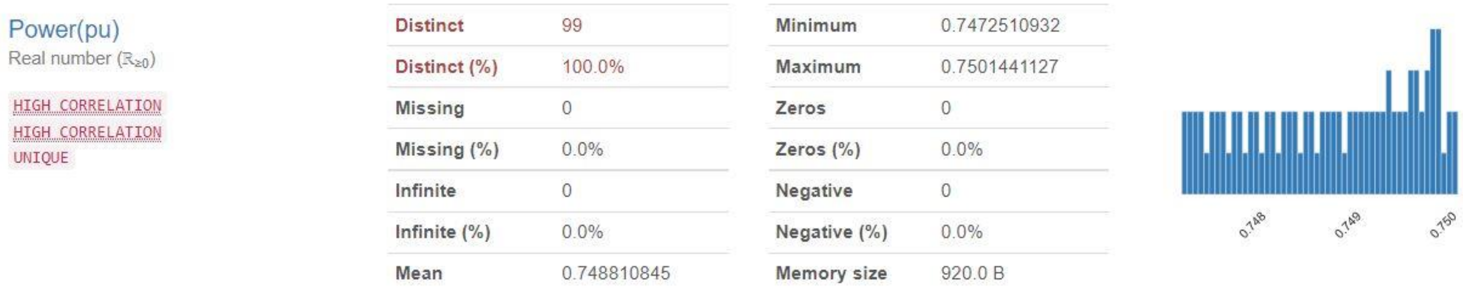


Figure 6 Power Variable Details

Data Profiling indicates that the Power attribute has **no** infinite records. The ratio of distinct age is **100%**. The **0.0%** of missing value in the age attribute. The maximum value is 0.7501441127 and minimum value is 0.7472510932

3.4.2.2 Speed

The Data Profiling displays that the Speed property contains **99** distinct records. The ratio of distinct speed is **100%**. The 0.0% of missing value in resting bps attribute. The maximum value is 1.605541 x 10^{-8} and minimum value is 4.000000143



Distinct	99	Minimum	0.7472510932
Distinct (%)	100.0%	Maximum	0.7501441127
Missing	0	Zeros	0
Missing (%)	0.0%	Zeros (%)	0.0%
Infinite	0	Negative	0
Infinite (%)	0.0%	Negative (%)	0.0%
Mean	0.748810845	Memory size	920.0 B

Figure 7 Speed Variable Details

3.4.2.3 Interactions between Power &

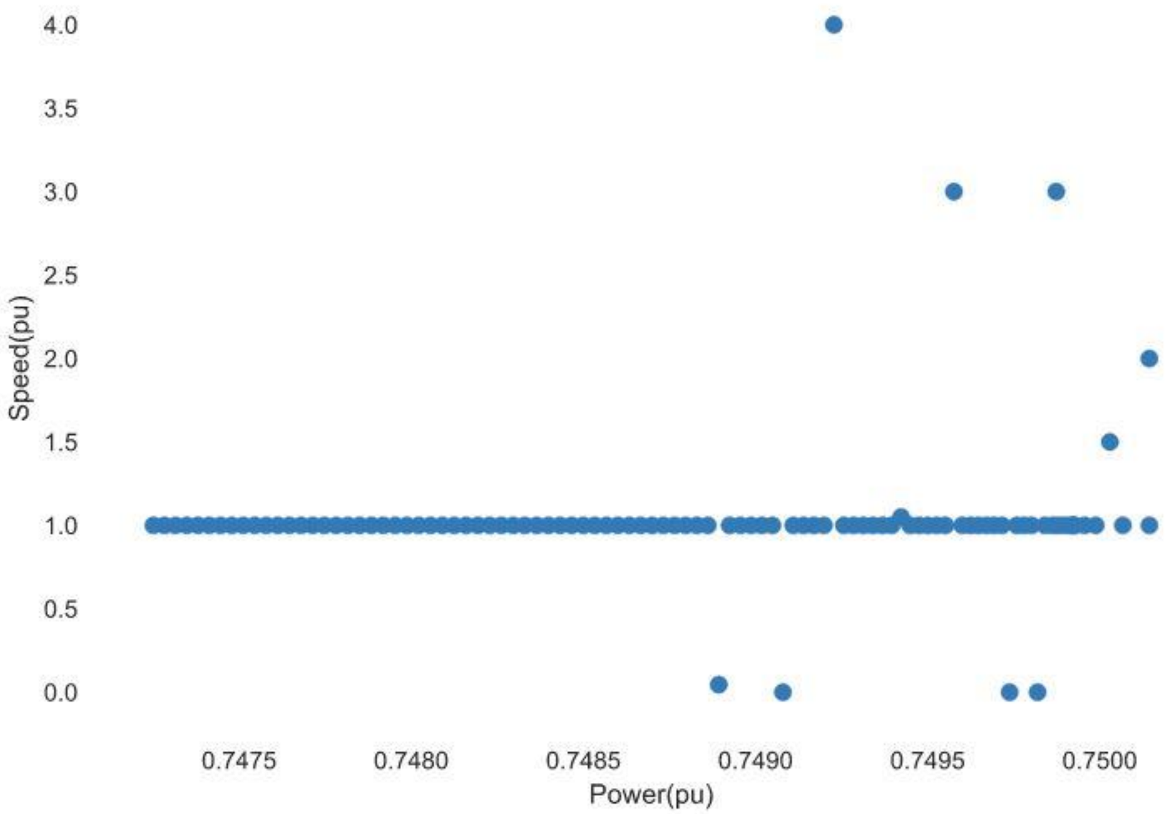


Figure 8 Interaction between Speed and Power Attributes

Speed

The Data Profiling shows that there is some irregular interaction between power and speed as compare to all dataset. These irregular intersections are basically showing faulty reading and outlier in dataset.

3.4.2.4 Phik (φk)

We have utilized **Phik** (φk) to determine the association between **Techno meter** attributes. Phik (φk) is a valuable correlation coefficient calculator. In our Dataset, Speed and Power attribute are linearly dependent. Mean value of speed directly proportional to power show in figure 9.

3.4.2.5 Relation in Attributes Value

The figure 10 shows that there is depended relation between power and speed. This relationship is showing increase in one value also increase in other as well as decrease in one value also decrease in other value.

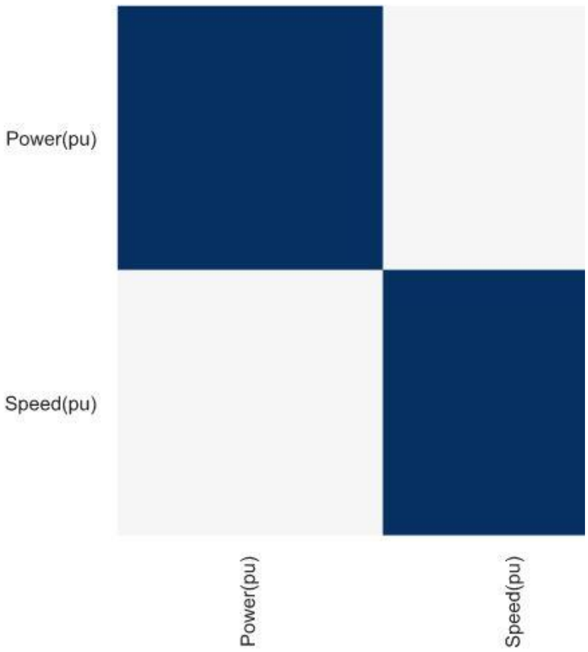


Figure 9 Speed is directly proportional to Power

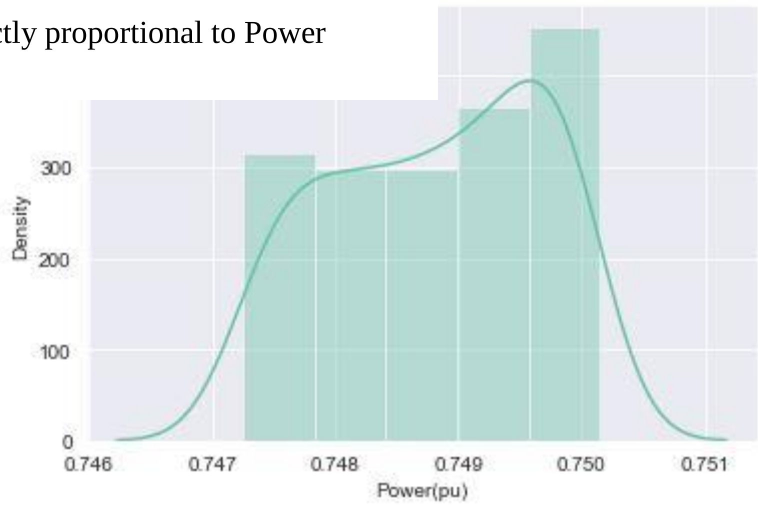
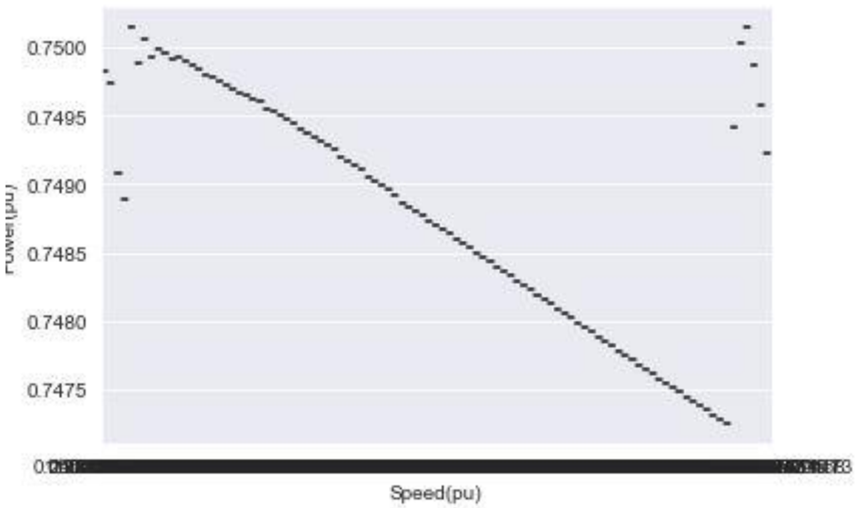


Figure 1 Relation in Attributes Value

3.4.2.6 Outlier

The figure 11 is showing that mostly values of speed and power are between 0.7475 to 0.7500. but still there are nine values which are showing opposite behavior in it

Figure 2: Irregular Behavior in Reading



3. Implementation and Result

This section briefly describes the system's methodology and result. An essential component of this chapter is the technique for isolate techno meter outlier.

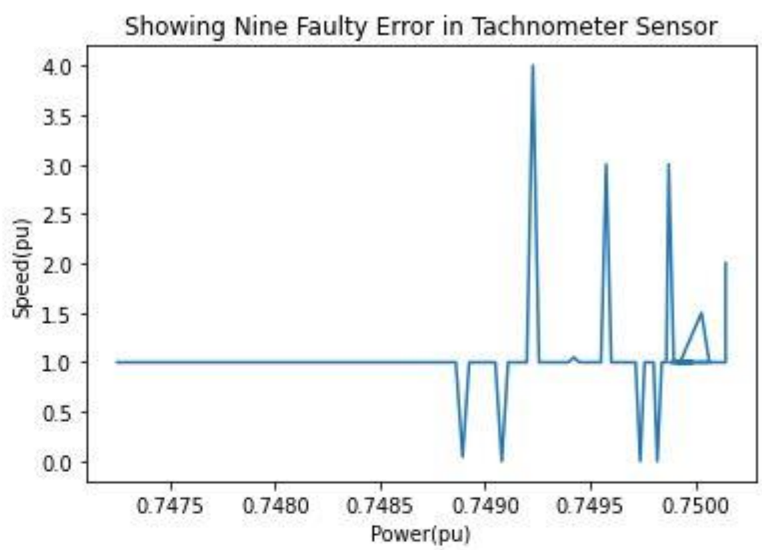


Figure 12 Nine Outlier in Techno meter Dataset

For isolation, we have used Kalman Filter to predict future values for removing bias result in techno meter sensor.

4.1 Kalman Filter Algorithm

The Kalman filter is an algorithm that, given the measurements that have been gathered over a period of time, can produce estimates of some unknown variables. The usefulness of Kalman filters has been demonstrated in a number of different applications. The shape that Kalman filters take is rather straightforward, and they make light use of computational resources. However, individuals who are not well-versed in estimate theory may still find it challenging to comprehend and utilize the Kalman filters due to the complexity of the subject matter.

4.2. Kalman Filter Proposed Equation

The use of Kalman filters is an advanced approach that provides an effective method for determining the present state of a statistical process in a way that reduces the amount of mean squared error that would otherwise occur. They are an example of a filter, which is a statistical technique for reducing the noise in a sequence of pointwise (or continuous; but pointwise in Kalman filters and in all practical applications) observations in order to generate a value that is relatively close to the actual value for some underlying statistical system.

The use of Kalman filters is ubiquitous despite the fact that they are among the more sophisticated filtering algorithms that are currently in use. They were essential in the computers that were used for the Apollo moon landing, and since then, they have found continuous use in a variety of applications, such as cleaning signals on instruments used by self-driving cars and other similar technologies.

In the language of mathematics, Kalman filters are based on the assumption of a discrete stochastic process of the form:

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1}$$

Which is being measured by a Z_k such that:

$$z_k = Hx_k + v_k$$

W_k and V_k are assumed to be the process noise and the measurement noise respectively, and are assumed to be normally distributed which mean zero and variance QQ and RR, respectively:

$$p(w) \sim N(0, Q)$$
$$p(v) \sim N(0, R)$$

An a priori estimate, denoted by the notation x, and a posteriori estimate, denoted by the notation X_k are distinguished through the use of Kalman filters. The best guess that the filter has for a value before it is observed is referred to as the a priori estimate. On the other hand, a posteriori estimate refers to the filter's best guess after taking into account the information provided by Z_k.

There are error terms and covariance terms associated with these values:

$$e_k^- = x_k - \hat{x}_k^-$$
$$e_k = x_k - \hat{x}_k$$
$$P_k^- = E[e_k^- e_k^{-T}]$$
$$P_k = E[e_k e_k^T]$$

When all of these dynamics are taken into account, the equation that the Kalman filter use to actually simulate the true value is as follows: $x^\wedge{}_{kx}$:

The $z_k - H\hat{x}_k^-$ the difference between the value that was observed, Z_k, and the value that the Kalman filter predicted in the absence of the observation is referred to as the residual. term is the **residual**. The n x m matrix that is referred to as the **gain** is denoted by the word KK. K is selected in such a way that it reduces a posteriori error covariance, also known as P_k. It is possible to arrive at an analytical solution for the value of K by first executing a few equation substitutions, then assigning K=0, and last solving for its trace equaling 0. (e.g. finding a minimal point). There is a closed-form solution to this problem:

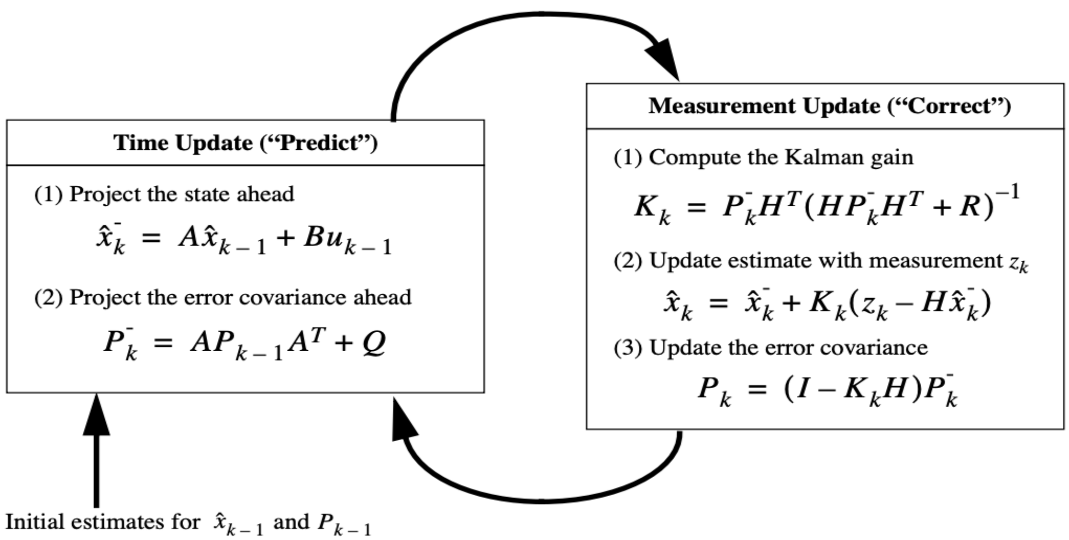
$$K_k = \frac{P_k^- H^T}{HP_K^- H^T + R}$$

This solution has the following revealing properties:

$$\lim_{R \to 0} K_k = \frac{1}{H} = H^{-1}$$
$$\lim_{P_k^- \to 0} K_k = 0 \implies \hat{x}_k = \hat{x}_k^-$$

In other words, as the measurement error covariance gets closer and closer to 0, the residual is given a smaller and smaller weight, while the measurement value Z_k is given a larger and larger weight. In the opposite direction, the predictor Hx_k has a higher level of credibility, whereas Z_k has a lower level of credibility. This equation's main purpose, then, is to strike a balance between the amount of faith it places in the value that has been seen (or measured) and the amount of faith it places in the value that has been modelled. Because of this, Kalman filters have shown to be extremely helpful in many different contexts. They are able to modify the relative worth of prediction against observation in an online manner since their design incorporates a limited kind of model accuracy feedback, which gives them the opportunity to do so.

Because Kalman filters are recursive, it is not necessary to recompute all of the previous information in order to generate an update. This is one of the reasons why they are favourable from a computational point of view. This provides an advantage over many other filters, most of which require a significant amount of computing work. In this way, they bring to mind recurrent neural networks, despite the fact that Kalman filters are obviously orders of magnitude simpler computationally.



The parameters RR and QQ are critical to the operation of the model and must be set before it can be executed. RR may usually be determined by watching the phenomenon in an offline manner for some time before the Kalman filter was implemented. This should be done before the Kalman filter was implemented. Since it is not possible to directly see QQ, it is a hyperparameter that needs to be controlled instead. It's interesting to note that the research advises use a different Kalman filter in order to tweak QQ in an online manner. In the context of neural learning, this is essentially the same as pretraining.

After applying above mention equations, we have above three reading (**Speed**) in it. Noisy measurement (outlier). Posteri estimate values (**isolate reading**) and truth values (actual values).

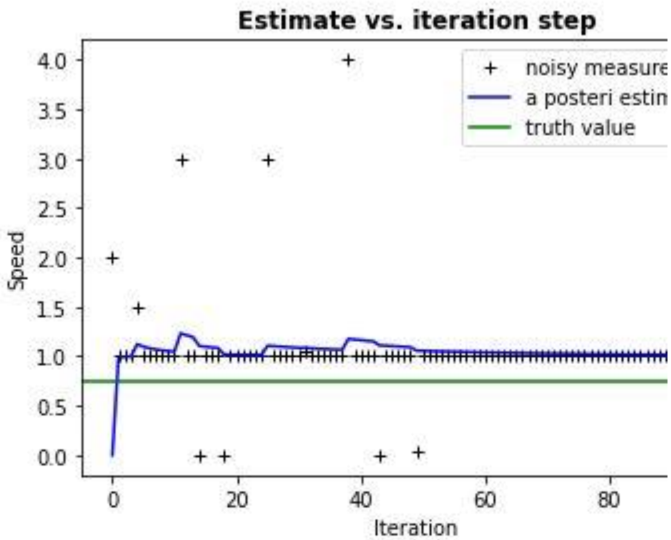


Figure 13 Estimate vs iteration Step (Speed Attribute)

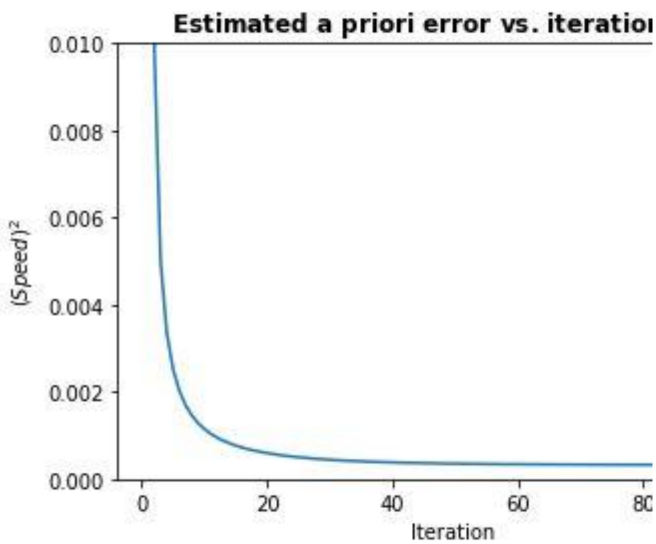


Figure 14 After Isolate Speed Reading (Kalman Filter Algorithm)

After applying above mention equations, we have above three reading (**Speed**) in it. Noisy measurement (outlier). Posteri estimate values (isolate reading) and truth values (actual values). The figure 15 is showing there is no noisy measurement (Outlier) in power attribute

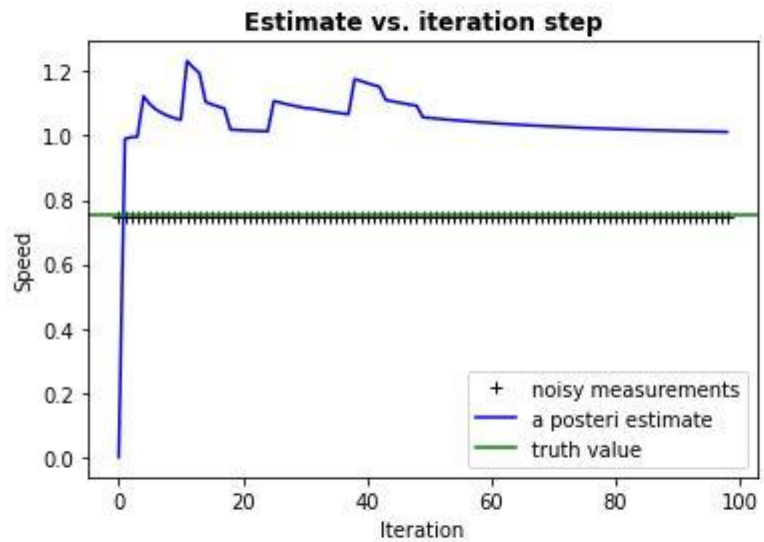


Figure 15 Estimate vs iteration Speed

All the data points in the test data were evaluated and for each power whose speed variated from the predicted results; it was added to faulty data frame. After that we have used Kalman Filter to isolate these nine values. The table 1 is showing the isolated values of nine faulty reading after Kalman Filter.

Table:1 Reading After Isolate

Sr.	Power(pu)	Speed(pu)
1	0.750144	1.00000082625648
2	0.750029	1.0000082826256
3	0.749874	1.00000092625648
4	0.749819	1.00000072625648
5	0.749738	1.00000032625648

6	0.749576	1.00000088225648
7	0.749228	1.00000088225648
8	0.749080	1.00000099925648
9	0.748893	1.00000074525648

5. Conclusion and Future Direction

As the study name suggest we! want to estimate the fault that is occur in the sensor. So, first we! have to get real time data from any application of a sensor. So, as we! are the students of Electrical power engineering we! choose power system and tachometer as a sensor. we! built its Python Simulink model and then isolate faulty data of speed and power. The dataset chooses for this project consists of Power (input feature) and Speed (output feature). As the different values of speed in the dataset changes in micros that created a little drop in accuracy, so that's why when linear model predicts the output. So, for linear prediction we! choose Kalman Filter Algorithm to isolate faulty values of speed by giving the model power feature. So, by that we! get the isolated faulty speed and at last isolated faulty speed was plotted using NumPy, marplot. To deal with unseen examples, a Kalman filter algorithm must be cognizant of this necessity and thus, when it learns it must make conscious and diligent efforts to generalize from examples it has seen. Good generalization from data is a prime activity a learner program must perform. By using Kalman filter we! are able to detect the Faulty values of the speed. In previous works many methods has been introduced the difference in the project. Anomaly detection is the process of identifying unexpected items or events in data sets, which differ from the norm. And anomaly detection is often applied on unlabeled data which is known as unsupervised anomaly detection. Anomaly detection has two basic assumptions: The future of the project has a very good scope now the end of the project is to isolate the fault so if someone wants to do any further work. Students that can work further on this project in more advanced level. we! mainly focus on the isolate the fault techno meter sensor. Now if someone to take this project to more advanced level he can work on How to erase that fault reading for any sensor.

REFERENCES

1. Springer Singapore. doi: 10.1007/978-981-13-1280-9.
2. Khan, S. R., Raza, A., Shahzad, I., & Ijaz, H. M. (2024). Deep transfer CNNs models performance evaluation using unbalanced histopathological breast cancer dataset. Lahore Garrison University Research Journal of Computer Science and Information Technology, 8(1).
3. Bilal, Omair, Asif Raza, and Ghazanfar Ali. "A Contemporary Secure Microservices Discovery Architecture with Service Tags for Smart City Infrastructures." VFAST Transactions on Software Engineering 12, no. 1 (2024): 79-92.
4. S. Abhari, S. R. N. Kalhori, M. Ebrahimi, and H. Hasannejadasl, "Artificial Intelligence Applications in Type 2 Diabetes Mellitus Care : Focus on Machine Learning Methods," vol. 25, no. 4, pp. 248–261, 2019.
5. H. Kaur and V. Kumari, "Predictive Modelling and Analytics for Diabetes using a Machine Learning Approach," *Appl. Comput. Informatics*, no. December, 2018, doi: 10.1016/j.aci.2018.12.004.
6. Raza, A., Salahuddin, & Inzamam Shahzad. (2024). Residual Learning Model-Based Classification of COVID-19 Using Chest Radiographs. Spectrum of Engineering Sciences, 2(3), 367–396.
7. A. Mujumdar and V. Vaidehi, "ScienceDirect ScienceDirect ScienceDirect ScienceDirect Diabetes Prediction using Machine Learning Aishwarya Mujumdar Diabetes Prediction using Machine Learning Aishwarya Mujumdar Aishwarya," *Procedia Comput. Sci.*, vol. 165, pp. 292–299, 2019, doi: 10.1016/j.procs.2020.01.047.
8. S. Brian and R. R. B. Pharmd, "Prediction of Nephropathy in Type 2 Diabetes: An Analysis of the ACCORD Trial applying Machine Learning Techniques," no. 317, doi: 10.1111/cts.12647.
9. M. Waqas, Z. Khan, S. U. Ahmed and Asif. Raza, "MIL-Mixer: A Robust Bag Encoding Strategy for Multiple Instance Learning (MIL) using MLP-Mixer," 2023 18th International Conference on Emerging Technologies (ICET), Peshawar, Pakistan, 2023, pp. 22-26
10. Asif, S., Wenhui, Y., ur-Rehman, S., ul-ain, Q., Amjad, K., Yueyang, Y., ... & Awais, M. (2024). Advancements and Prospects of Machine Learning in Medical Diagnostics: Unveiling the Future of Diagnostic Precision. Archives of Computational Methods in Engineering, 1-31.
11. Asif, S., Zhao, M., Li, Y., Tang, F., Ur Rehman Khan, S., & Zhu, Y. (2024). AI-Based Approaches for the Diagnosis of Mpox: Challenges and Future Prospects. Archives of Computational Methods in Engineering, 1-33.
12. J. Li *et al.*, "International Journal of Medical Informatics Establishment of noninvasive diabetes risk prediction model based on tongue features and machine learning techniques," *Int. J. Med. Inform.*, vol. 149, no. August 2020, p. 104429, 2021, doi: 10.1016/j.ijmedinf.2021.104429.
13. S. G. Azevedo *et al.*, "System-Independent Characterization of Materials Using Dual-Energy Computed Tomography," *IEEE Trans. Nucl. Sci.*, vol. 63, no. 2, pp. 341–350, 2016, doi: 10.1109/TNS.2016.2514364.
14. B. Pranto, S. M. Mehnaz, E. B. Mahid, and I. M. Sadman, "Evaluating Machine Learning Methods for Predicting Diabetes among Female Patients in Bangladesh," 2020, doi: 10.3390/info11080374.
15. Shahzad, I., Khan, S. U. R., Waseem, A., Abideen, Z. U., & Liu, J. (2024). Enhancing ASD classification through hybrid attention-based learning of facial features. Signal, Image and Video Processing, 1-14.
16. A. T. Nagi, N. Ayesha, M. J. Awan, and R. Javed, "A Comparison of Two-Stage Classifier Algorithm with Ensemble Techniques On Detection of Diabetic Retinopathy," pp. 9–12, 2021.
17. H. Lu, S. Uddin, F. Hajati, M. Ali, and M. Matloob, "A patient network-based machine

learning model for disease prediction : The case of type 2 diabetes mellitus," 2021.

18. Y. Sun and D. Zhang, "Machine Learning Techniques for Screening and Diagnosis of Diabetes : a Survey," vol. 3651, pp. 872–880, 2019.
19. B. G. Choi, S. Rha, S. W. Kim, J. H. Kang, J. Y. Park, and Y. Noh, "Machine Learning for the Prediction of New-Onset Diabetes Mellitus during 5-Year Follow-up in Non-Diabetic Patients with Cardiovascular Risks," vol. 60, no. 2, pp. 191–199, 2019.
20. Khan, S.U.R.; Raza, A.;Waqas, M.; Zia, M.A.R. Efficient and Accurate Image Classification Via Spatial Pyramid Matching and SURF Sparse Coding. Lahore Garrison Univ. Res. J. Comput. Sci. Inf. Technol. 2023, 7, 10–23.
21. Farooq, M.U.; Beg, M.O. Bigdata analysis of stack overflow for energy consumption of android framework. In Proceedings of the 2019 International Conference on Innovative Computing (ICIC), Lahore, Pakistan, 1–2 November 2019; pp. 1–9.
22. F. Farrokhi *et al.*, "Investigating risk factors and predicting complications in deep brain stimulation surgery with machine learning algorithms," *World Neurosurg.*, 2019, doi: 10.1016/j.wneu.2019.10.063.
23. L HUSSAIN, S., Raza, A., MEERAN, M. T., IJAZ, H. M., & JAMALI, S. (2020). Domain Ontology Based Similarity and Analysis in Higher Education. IEEEP New Horizons Journal, 102(1), 11-16.
24. Farooq, M. U., Khan, S. U. R., & Beg, M. O. (2019, November). Melta: A method level energy estimation technique for android development. In 2019 International Conference on Innovative Computing (ICIC) (pp. 1-10). IEEE.
25. Khan, S. U. R., & Asif, S. (2024). Oral cancer detection using feature-level fusion and novel self-attention mechanisms. Biomedical Signal Processing and Control, 95, 106437.
26. Raza, A.; Meeran, M.T.; Bilhaj, U. Enhancing Breast Cancer Detection through Thermal Imaging and Customized 2D CNN Classifiers. VFAST Trans. Softw. Eng. 2023, 11, 80–92.
27. Dai, Q., Ishfaque, M., Khan, S. U. R., Luo, Y. L., Lei, Y., Zhang, B., & Zhou, W. (2024). Image classification for sub-surface crack identification in concrete dam based on borehole CCTV images using deep dense hybrid model. Stochastic Environmental Research and Risk Assessment, 1-18.
28. Khan, S.U.R.; Asif, S.; Bilal, O.; Ali, S. Deep hybrid model for Mpox disease diagnosis from skin lesion images. Int. J. Imaging Syst. Technol. 2024, 34, e23044.
29. Khan, S.U.R.; Zhao, M.; Asif, S.; Chen, X.; Zhu, Y. GLNET: Global–local CNN's-based informed model for detection of breast cancer categories from histopathological slides. J. Supercomput. 2023, 80, 7316–7348.
30. Khan, S.U.R.; Zhao, M.; Asif, S.; Chen, X. Hybrid-NET: A fusion of DenseNet169 and advanced machine learning classifiers for enhanced brain tumor diagnosis. Int. J. Imaging Syst. Technol. 2024, 34, e22975.
31. IoannisKavakiotis, Olga Tsave, Athanasios Salifoglou, and NicosMaglaveras, "Machine Learning and Data Mining Methods in Diabetes Research", Computational and Structural Biotechnology Journal, vol. 15, pp. 104–116, 2017
32. Khan, U. S., & Khan, S. U. R. (2024). Boost diagnostic performance in retinal disease classification utilizing deep ensemble classifiers based on OCT. Multimedia Tools and Applications, 1-21.
33. Bilal, Mahmood, F., Abbas, K., Raza, A., Khan,M.A., & Khan, P.W. (2019). Three Dimensional Agricultural Land Modeling using Unmanned Aerial System (UAS). International Journal of Advanced Computer Science and Applications (IJACSA) [p-ISSN : 2158-107X, e-ISSN : 2156-5570], 10(1).
34. Khan, M. A., Khan, S. U. R., Haider, S. Z. Q., Khan, S. A., & Bilal, O. (2024). Evolving knowledge representation learning

with the dynamic asymmetric embedding model. Evolving Systems, 1-16.
35. Raza, A., & Meeran, M. T. (2019). Routine of encryption in cognitive radio network. Mehran University Research Journal of Engineering & Technology, 38(3), 609-618.
36. Al-Khasawneh, M. A., Raza, A., Khan, S. U. R., & Khan, Z. (2024). Stock Market Trend Prediction Using Deep Learning Approach. Computational Economics, 1-32.
37. Khan, U. S., Ishfaque, M., Khan, S. U. R., Xu, F., Chen, L., & Lei, Y. (2024). Comparative analysis of twelve transfer learning models for the prediction and crack detection in concrete dams, based on borehole images. Frontiers of Structural and Civil Engineering, 1-17.
38. Hekmat, A., Zhang, Z., Ur Rehman Khan, S., Shad, I., & Bilal, O. (2025). An attention-fused architecture for brain tumor diagnosis. Biomedical Signal Processing and Control, 101, 107221. https://doi.org/https://doi.org/10.1016/j.bspc.2024.107221