Detection of Power Quality Disturbances

A Major Project Report submitted in partial fulfilment of the requirements for

The award of the degree of

MASTER OF TECHNOLOGY

CONTROL AND INSTRUMENTATION ENGINEERING

Under the supervision of

Neeraj Kumar Bhagat (Associate Professor)

&

Prof. Rajiv Kapoor

By

Shivam Kukrety (2k19/c&i/17)

DEPT. OF ELECTRICAL ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

July 2021

ACKNOWLEDGMENT

Major Project is a golden opportunity of learning and self-development. I consider myself very lucky to have so many people to guide us through to the successful completion of this major project.

This major project is a result of the endeavours and hard work of these wonderful people. Therefore, it is our responsibility to acknowledge them.

I would like to thank my mentor, Mr. Neeraj Kumar Bhagat (Associate Professor) & Prof. Rajiv Kapoor, for his guidance in making this project successful.

Finally I would like to thank our parents and well-wishers for their valuable help and encouragement throughout the project.

Shivam

Shivam Kukrety (2k19/c&i/17)

Delhi Technological University

New Delhi

CERTIFICATE

This is to certify that the report entitled "***Detection of Power Quality Disturbances***" is a bonafide record of the Major Project done by the following student under my supervision and guidance, in partial fulfilment of the requirements for the award of the degree of Master of Technology in Control & Instrumentation Engineering from Delhi Technological University for the term 2019-2021.

Neeraj Kumar Bhagat	Prof. Rajiv Kapoor	Dr. Uma Nangia
Associate Professor	Professor	(HOD)
EE Department	ECE Department	EE Department

DECLARATION

I student of M.Tech (Control & Instrumentation Engineering, 2019-2021 Batch) at Delhi Technological University, hereby declare that the project entitled "***Detection of Power Quality Disturbances***" is the outcome and result of my personal studies, learning and research.

Shivam

Shivam Kukrety (2k19/c&i/17)

CONTENTS

Topic	**Page. No**
Abstract	vi
List of Figures	vii
Chapter 1: Introduction	1
1.1. Overview	2
1.2. Past works	3
Chapter 2: Subspace-Transfer Learning for Power System Disturbance Recognition	5
Chapter 3: Deep Learning & LSTM Architecture	8
Chapter 4: Implementation	10
Chapter 5: Comparison with other latest Techniques	37
Chapter 6: Conclusion: What is the way Forward?	39
Bibliography	40

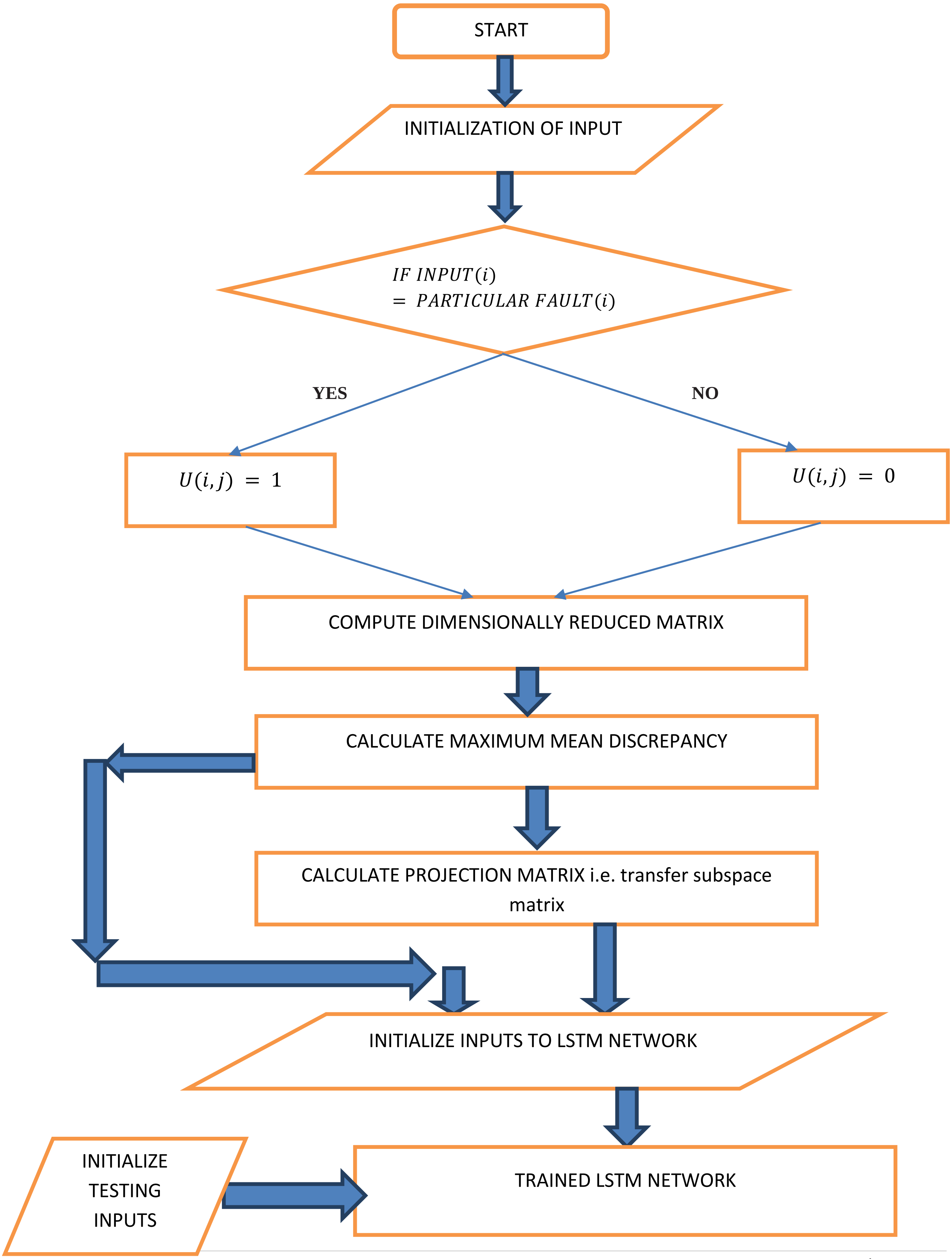
Abstract

Automated and quick fault detection has received quite a lot of importance and some comprehensive studies have been done because of interlinking of varieties of disturbances in the power system. It takes ideal sinusoidal signal as training data aiming to recognize the other different types of faults, it generally involves two problems, i.e., selection and matching between the training and the testing data. Many studies have either studied the two independently or only focusing on selection part with less focus on the matching part of the algorithm. In this paper we propose the algorithm of transfer subspace learning to address the problem of matching which is of considerable importance as how good be the selection if the matching to particular fault is not accurate it will not give desired results. In the experiment we calculate the projection matrix and maximum mean discrepancy matrix to identify the type of fault which has occurred. The experiment so performed on the industrial data verifies our experiment to be workable in the real world situations.

List of Figures

Figure captions	Page no
1. Flowchart	1
2. Sag in power system (with different magnitudes)	11
3. Sag in power system (with different magnitudes)	13
4. Transients in power system	14
5. Harmonics (3^{rd} +5^{th} + 7^{th}) in power system	15
6. 3^{rd} Harmonic in power system	17
7. Sag with SNR = 30%	18
8. Swell with SNR = 30%	20
9. Transients with SNR = 10%	21
10. Harmonics(3^{rd} +5^{th} + 7^{th}) + Sag with signal to noise ratio = 35%	23
11. Harmonics(3^{rd} +5^{th} + 7^{th}) + Sag with signal to noise ratio = 38%	24
12. 3^{rd} Harmonic + Sag	26
13. 3^{rd} Harmonic + Swell	27
14. Transients + Swell	29
15. Transients + Sag	30
16. 3^{rd} Harmonic + Transients	32
17. Harmonics(3^{rd} +5^{th} + 7^{th}) + Transients	33

Chapter 1: Introduction



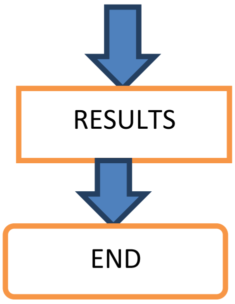


Figure 1: Flowchart of Algorithm

1.1 Overview

In power systems, fault detection and identifying the type of fault both have significant value & with more complex faults occurring the demand for new techniques to identify faults has been significantly boosted. Fault detection aims to recognize different types of faults under the following categories, e.g., Sag, Swell, Harmonics, transients. Fault detection & identification has been proven quite important in applications requiring human-machine interaction, e.g., in-distribution stations, transmission stations, generating stations, consumer centres.

With the advances in pattern recognition and machine learning techniques, many algorithms have been developed for fault detection. For example, Gaussian mixture model, neural network, support vector machine, supervised & unsupervised learning methods and deep neural networks. These approaches obtain satisfactory results but satisfactory is not enough in case of power system applications as one wrong can lead to a big enough loss (e.g. Loss of life) which would be very difficult to repay. We see that most of these algos are conducted on the assumption that the training and testing values are obtained from the same set of data. In practice however, since different faults are collected under different conditions, we have to cope with the cross-detection of faults, where the classifier model trained in one fault is applied to another fault, and often leads to poor recognition performance.

In fault detection in the power system, adaptation of control apparatus to the new environment is very important as there is a lot of variation among different types of faults & in cases of the same fault where the frequency is not fixed as a constant value but given a range of variation. Researchers have proposed many adaptation techniques, e.g., cepstral mean normalization, maximum aposteriori, joint factor analysis, vocal tract length normalization, maximum likelihood linear regression, to boost the system's performance. Over the period of last year deep learning has been applied over large variation of applications e.g. speech & face recognition, image processing, traffic management, chat-bots etc. A very basic characteristic of deep learning neural network is that it can learn features from raw data & does not need much simplification. A common feature of the deep learning neural network is that it can learn high level invariant features of the fault data more efficiently, and they can obtain better recognition performance than traditional algorithms.

All the methods discussed above requires a large number of training data, and do not take into account the influence of the “bias” problem. Transfer learning has gained value in past few years & has started to be seen as an alternative to the above drawback posed by the above algorithm It stores the knowledge obtained from the source data and applies that knowledge to the target data.

Subspace learning or dimensionality reduction, plays an important role in recognition as in this algorithm the original high dimensional features are projected to lower dimensional feature space, where properties of the original data is preserved. Recent years have witnessed widespread interest in sub-space learning techniques. A variety of algos & methods have been used to represent high dimensional data few of them are principal component analysis, linear discriminant analysis, locality preserving projection, locally linear embedding, laplacian eigen-map. Despite the different methods developed over the years, they can be understood as methods of general graph embedding framework.

We can see that in all the above cases that they have pre-assumed that both the training data & the testing data has been drawn from the same data sets or has identical set of values. In our case the major drawback of this is that the values are not at all constant it is variable i.e. it is variable as the frequency of the system is not fixed at a constant value rather it is variable in range of frequencies. To tackle this problem posed we employ the use of subspace-transfer learning algorithm in which the system tries to learn the common invariant features between training & testing data sets. However this cannot be employed or used for our case as it ignores the most important element of power system i.e. frequency as many faults like harmonics are majorly because of change in frequency of the system.

In this paper we propose an algorithm based on subspace-transfer learning but the features to be selected are taken as an additional constraint for our problem. In this way the benefits of subspace-transfer learning is also incorporated without compromising on the features of different faults. In this approach we can learn a projection matrix (Proj) which maps the features of different types of frequencies on to a common subspace while the $l_{2,1}$ norm is applied to Proj for selection of features. A regularization term is employed to maintain the geometric integrity of the data so that the recognition could be more accurate & precise.

All the main observations that we can draw from the above discussion can be summarized as follows:

- To deal with the changes in different data sets (based on frequency majorly) subspace-transfer learning has been modified to incorporate the feature selection process as well. Its main purpose is to incorporate the shortcomings posed by variable frequency in basic subspace-transfer learning.
- A regularization term is employed to maintain the geometric integrity of the data so that the recognition could be more accurate & precise

1.2 Past Works

In this section of the paper we have studied about the past development that have happened in the field of feature selection & subspace-transfer learning. We have not kept our outlook in the survey confined to the power system domain but have gone through various fields to get the oversight of all the developments & learn from shortcomings of the past.

- **Feature Selection:** It is one the most complex & important step to detect any type of fault. In this we try to see which features of the data most significantly denote the characteristics of the data & ignore the other less important features. Based on if the information is available or not feature selection can be divided into 3 categories i.e. supervised, semi-supervised & unsupervised. Of the 3 unsupervised is most difficult but in most practical cases unsupervised is the one which is seen. We focus our study to the unsupervised feature selection. Most of the traditional feature selection algorithms neglect the correlation that exist between different features (different features due to the difference in frequencies). $L_{2,1}$ norm is one the techniques that employ the use of correlations to select the features for different faults. Most of the researches done in the past have either clustered the data or have focused on feature selection only neglecting subspace–transfer learning which is an important next step because all the features obtained after this step must be mapped on to a common subspace for identification. Our approach tries to fill this gap between these 2 processes incorporating the best values of both & integrating them to get precise & accurate results.
- **Subspace Transfer Learning:** The technique which we are incorporating is not a single step algorithm but two algorithms conjucted together i.e. subspace learning & transfer learning. Now as we know that these are 2 different algorithms let's look at them individually first & then we'll see how the merger of the two gives desired results for our problem. In subspace learning we try to find a subspace where the properties of the data is preserved as well as the redundant features of the large amount of data is removed. Some of the subspace learning algorithms are pattern recognition, LDA, Locality preserving algorithm, ISOMAP, local linear embedding, laplacian-eigenmap, etc. Let's now see what is transfer learning, transfer learning is used when knowledge is first obtained from training set of data & then this knowledge so obtained is applied to another set of unknown data. This learning is highly useful in our case as the frequency may change the values of the new set of data but the basic characteristics of any fault can be detected by the use of this algorithm. Now as we saw in above description subspace learning is used to reduce the dimension of data by eliminating redundant features while transfer learning is the next step to detect the type of fault. Subspace-transfer learning helps us to do computation of large volume of data more swiftly and predict the type of fault with accuracy.

Chapter 2: Subspace-Transfer Learning for Power System Disturbance Recognition

Let's now apply all the concepts discussed above & see with the help of mathematical equations how it actually works. In this part of the paper we see all the steps from feature selection to subspace-transfer learning but all in the form of mathematical equations.

Let's start by clearing one difference i.e. all matrix are denoted in capital letters while all vectors are denoted by small letters. Let's understand what we mean by vector, vector are individual faults with different frequency while matrix is combination of multiple frequencies. Here point to note is that fault is constant both for vectors as well as matrix i.e. if vectors are of sag matrix will be of sag only. Mathematical expression is given as follows:-

$$A = a(\mathrm{i,j})$$

The feature matrix for the process is given as

$$X = [Xsource, Xtarget\ ]\ \epsilon\ Rm * n$$

$$Xsource = [x1, x2, x3 \ldots . xns] \epsilon Rm * ns$$

$$Xtarget = [xns + 1, xns + 2, xns + 3 \ldots . xn] \epsilon Rm * nt$$

Now as training & testing data may have different frequency we need to use subspace learning to map the features that were obtained from both individually on to one common subspace. When Xsource & Xtarget are taken onto a common subspace it is defined by following process

$$Y = [Ysource, Ytarget\ ]$$

$$Ysource = [y1, y2, y3 \ldots . yns] \epsilon Rns * c$$

$$Ytarget = [yns + 1, yns + 2, yns + 3 \ldots . yn] \epsilon Rnt * c$$

To convert X to a common subspace the formula that are employed are as follows

$$minimum_{Ysource} \sum_{i,j} U(i,j)||y(i) - y(j)||^2$$

$$(Ysource)transpose\ * B * Ysource\ =\ I$$

In the equation, I denotes the identity matrix i.e. diagonal elements are 1 rest all are 0 where B is given by the equation

$$U = [u(i,j)] \epsilon Rns * ns$$

Where u(i,j) whether the characteristics of both training & testing data are same or not i.e. both are sag, swell or any other type of fault. If they both are same type of fault

$$u(i,j) = 1\ otherwise\ u(i,j) = 0$$

Hence after reducing the dimension of the X matrix by subspace learning we get

$$Ysource = [v1, v2, v3 \ldots, vc] \epsilon Rns * c$$

Now since we have reduced the dimension of the matrix i.e. eliminated the redundant features lets now plot this onto a common subspace so that transfer learning could be applied to the problem at later stage. To find the projection matrix i.e. projection of low dimensional feature matrix on to one common subspace we employ the use of following formula

$$minimum_{Proj}||X^{transpose}\ * P - Y||_F^2$$

Where $||.||_F$ denotes the Frobenius norm of the matrix

Now we come to the transfer learning step of the experiment. Transfer learning as already discussed above is important as the features of the two training & testing data may not be the same but the characteristics or in simpler words the knowledge present in any type of fault is unique & hence this knowledge that is obtained can be applied to multiple set of examples to get the desired outcomes. We employ the use of maximum discrepancy between the training and testing data to get an accurate estimate as to how much of the difference is actually there between the training and testing data. To preserve the properties maximum discrepancy is incorporated into the formula & the updated formula becomes

$$minimum_{Proj}||X^{transpose} * P - Y||_F^2 + \beta * f(p) + \alpha * ||Proj||_{2,1}$$

In the equation β denotes the effect too which maximum discrepancy (denoted as $f(p)$) is incorporated in the process & $||.||_{21}$ denotes the l_{21} norm of the matrix. $f(p)$ & $||Proj||_{21}$ is expressed as follows

$$f(p) = ||[(\frac{1}{ns}) * \sum_{i=1}^{ns} y(i)] - [\left(\frac{1}{nt}\right) * \sum_{j=ns+1}^{n} y(j)] ||^2$$

$$f(p) = trace(Proj^{transpose} * X * M * X^{transpose} * Proj)$$

$$Proj: \left|\left|Proj\right|\right|2,1 = \sum_{i=1}^{m} \left|\left|Proj^i\right|\right|_2$$

Where $Tr(.)$ denotes the trace of the matrix while M denotes maximum discrpancy

With all the above steps we have reduce the dimension as well as made the system to learn to detect different types of faults. As most researchers do we get results by the above steps in this process we ignore a main component i.e. geometric structure of the data which is really important for real world problems. Taking inspiration from manifold learning which says that if 2 points in a particular plane are close to each other then there must exist 2 points which are close to the dimensionally reduced data as well. Let's understand this with the help of the following mathematical equations.

Let there be a set of points $x1; x2; \ldots; xn$ we can construct a nearest neighbour graph for this by the use of following equations

$$W = [w(i,j)] \epsilon Rn * n$$

Where $w(i,j) = 1,$

$= 0$, otherwise

The regularization term to maintain the geometric integrity of the data is given by $g(p)$ as follows

$$g(p) = \left(\frac{1}{2}\right) * \sum_{i,j=1}^{n} \left|\left|z(i) - z(j)\right|\right|^2 * w(i,j)$$

$$f(p) = trace(Proj^{transpose} * X * L * X^{transpose} * Proj)$$

$$L = D - W$$

$$D = diag(d1, d2, .., dn)$$

$$dii = \sum_{j} w(i,j)$$

To optimize the projection matrix according to all the terms discussed above is given by the upcoming equations. This will summarize the procedural aspect of this paper & now we will see the implementation of the theory which we have discussed till now

$$L = ||X^{transpose} * P - [Ysource, Ytarget]||_F^2 + trace(Proj^{transpose} * R * Proj) + \alpha * ||Proj||_{2,1}$$

Where $R = X * (\beta * M + \gamma * L) * X^{transpose}$

$$\frac{\partial L(Ytarget, Proj)}{\partial Proj} = 0$$

$$\boldsymbol{Proj\_optimized = (X * X^{transpose} - R)^{-1} * X * Y}$$

Chapter 3: Deep Learning & Long Short Term Memory (LSTM) Architecture

For a long time the working of human brain has remained a mystery for the researchers. With the improvement in technology the scientists have tried to replicate the human brain (but mostly for specific tasks) as there is still no match for the computing power of the brain & the variety of complex task it performs. One unique feature of the brain is learning & scientists have developed a new set of networks that can in particular are capable of learning a particular task and then performing it with a much faster pace. This leads us into the world of machine learning where we make a machine learn a particular task based on the past values of data & then make predictions for future values before the event has actually occurred. The learning or training can be done in one of these 3 ways:-

1. Supervised learning i.e. with the help of a teacher
2. Unsupervised learning i.e. with experience

Among these 2 the most important one is unsupervised learning. In this type a system is fed with a particular kind of data & the system is expected to find some kind of pattern in the data so that it can predict the future. Deep neural network is a subtype of machine learning in which the data is fed in a systematic manner & the system rather than using a linear function to map the data uses a non-linear function which improves the ability of the system to predict the values more accurately. In our application we have employed the use of a subtype of deep learning neural network i.e. Long Short Term Memory (LSTM). Before we see the results lets understand in brief the working of a LSTM network.

LSTM network is a special type of recurrent neural network (RNN) in which it remembers the values it uses for prediction of the next state while discarding the values it doesn't need for prediction. Let's understand this in a laymen's perspective, to predict any phenomenon we not only need the learning we get from the immediate past set of data but also the learning from a far past may be far more valuable. LSTM as the name suggests Long Short Term Memory i.e. keep the experience of the far past & immediate past to predict the phenomenon. LSTM in terms of mathematical equations can be expressed as follows:-

1. The 1st step is which information to keep & which to throw away

$$Keep/throw = \delta(weight1 * (past\ information/\ new\ information) + bias1)$$

$$Where\ \delta\ is\ a\ sigmoid\ function$$

$$KT = \delta(w1 * (pi/ni) + b1)$$

2. If we have decided to keep the value the next question is which value it will replace & what will be the new values that will be added to past state

Which to update

$$= \delta(weight2 * (past\ information/\ new\ information) + bias2)$$

$$Vector\ of\ info = tanh(weight3 * (past\ information/\ new\ information) + bias3)$$

$$Upd = \delta(w2 * (pi/ni) + b2)$$
$$Vec = \delta(w3 * (pi/ni) + b3)$$

3. Updating the state

$$New\ state = Keep/Throw * old\ state + Which\ to\ update * Vector\ of\ info$$

$$NS = KT * OS + Upd * Vec$$

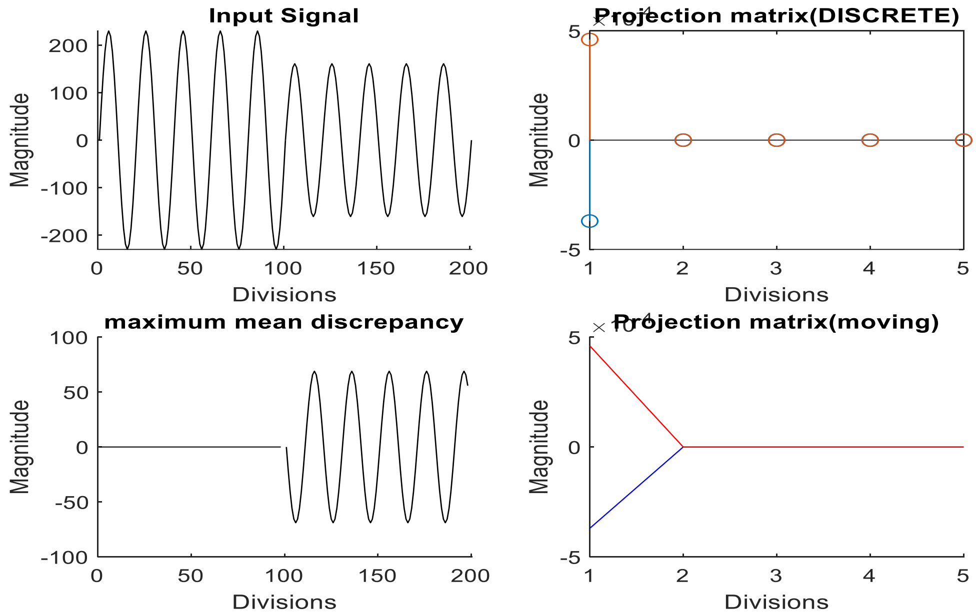
4. Finally we see output equations

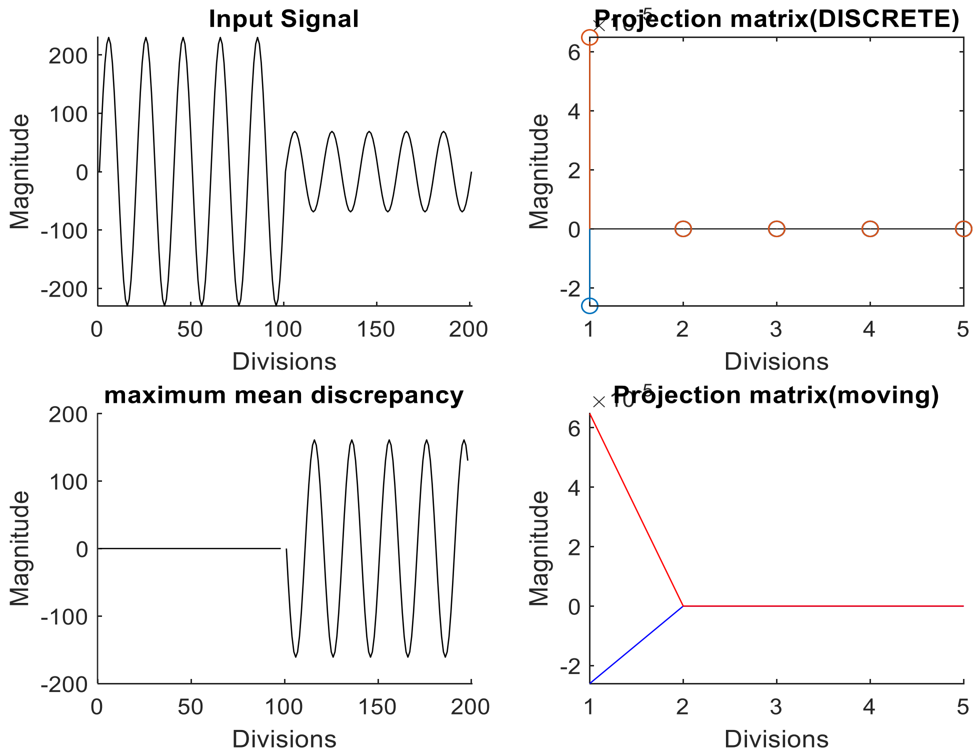
$$\text{Output} = \delta(weight3 * (past\ information/\ new\ information) + bias3)$$
$$Value\ to\ output = Output * tanh(newstate)$$

$$\text{Out} = \delta(w3 * (pi/\ ni) + b3)$$
$$Val = Out * tanh(NS)$$

Chapter 4: Implementation

After all the efforts done till now, now is the time to face the truth i.e. see the implementation of our algorithm to the real world problem & how it performs. In the whole process one thing is clear that we know the training data the main problem is to how to apply it to real time applications with accuracy & precision. The results obtained from subspace-transfer learning are as follows & they speak for themselves.





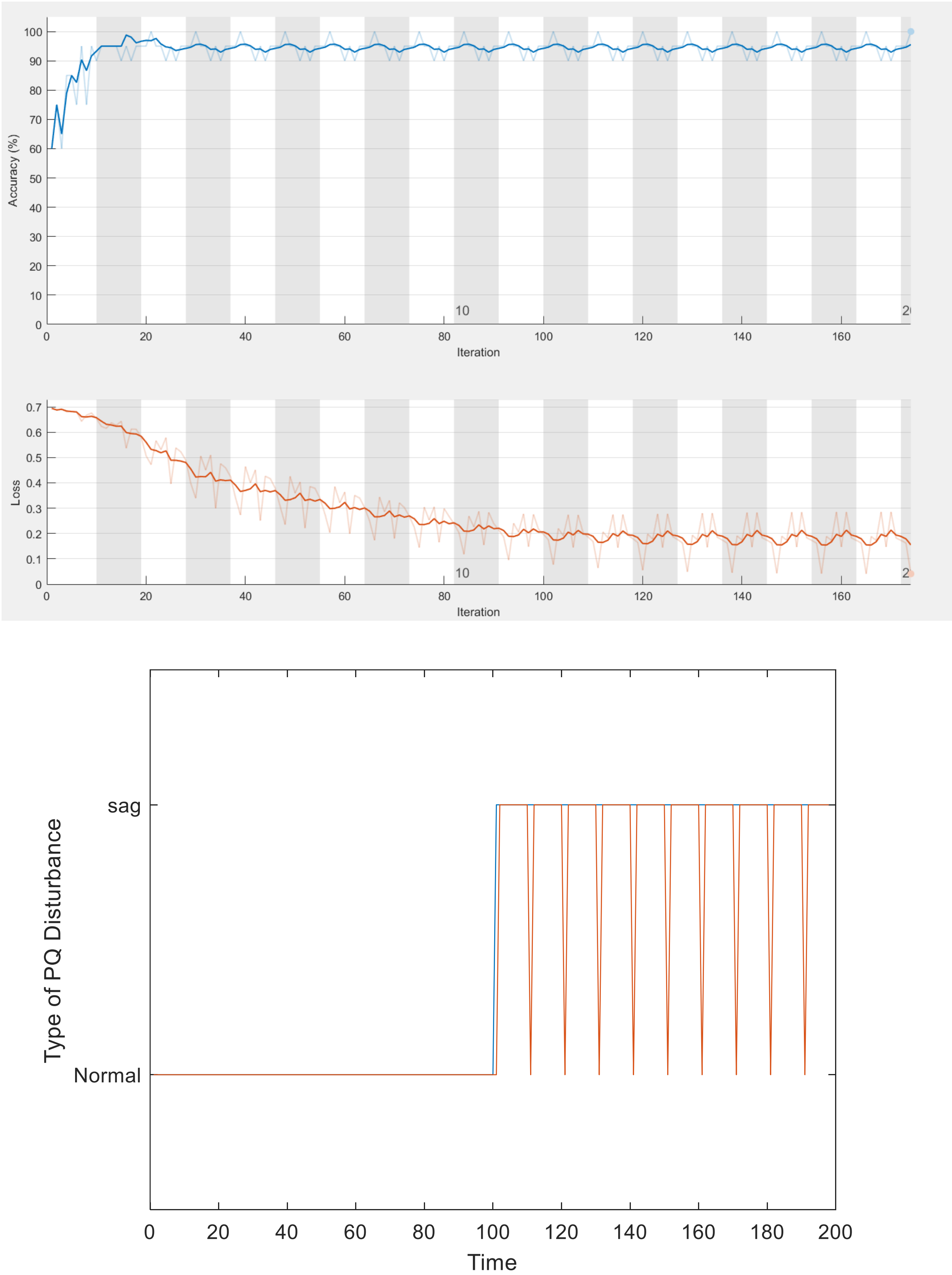
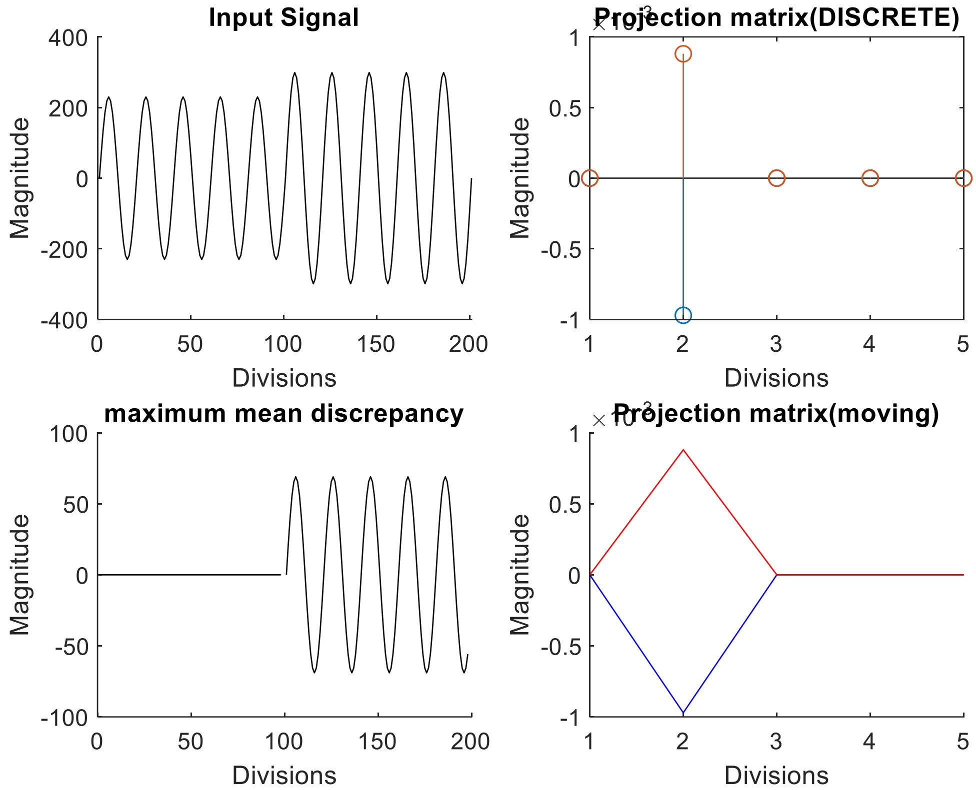
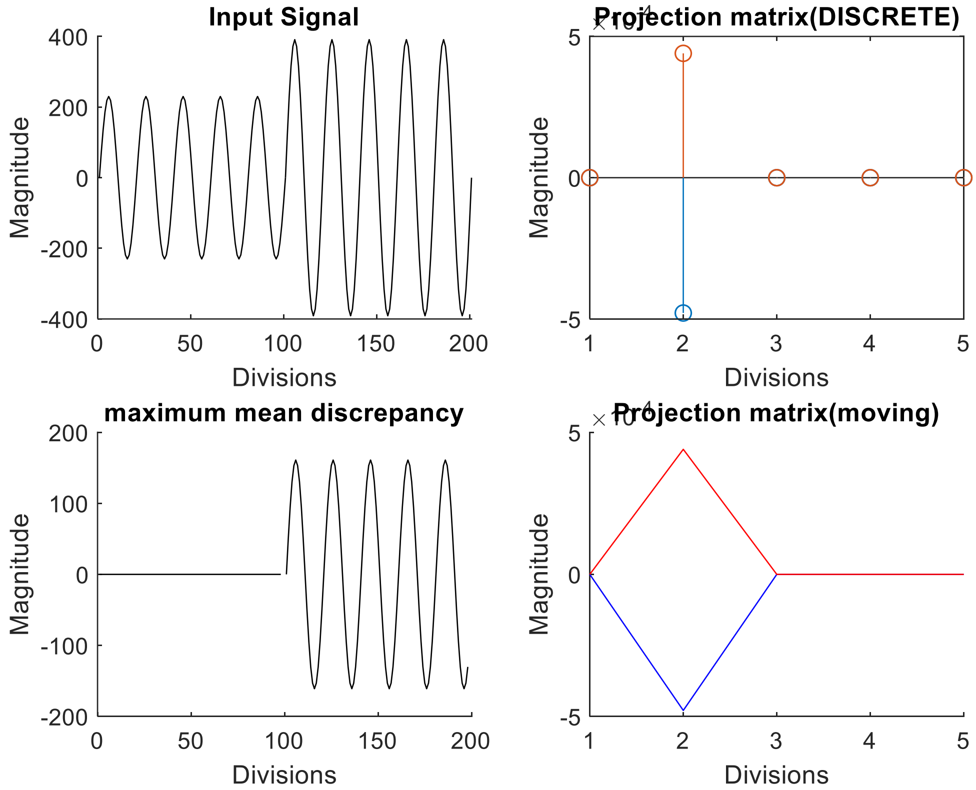
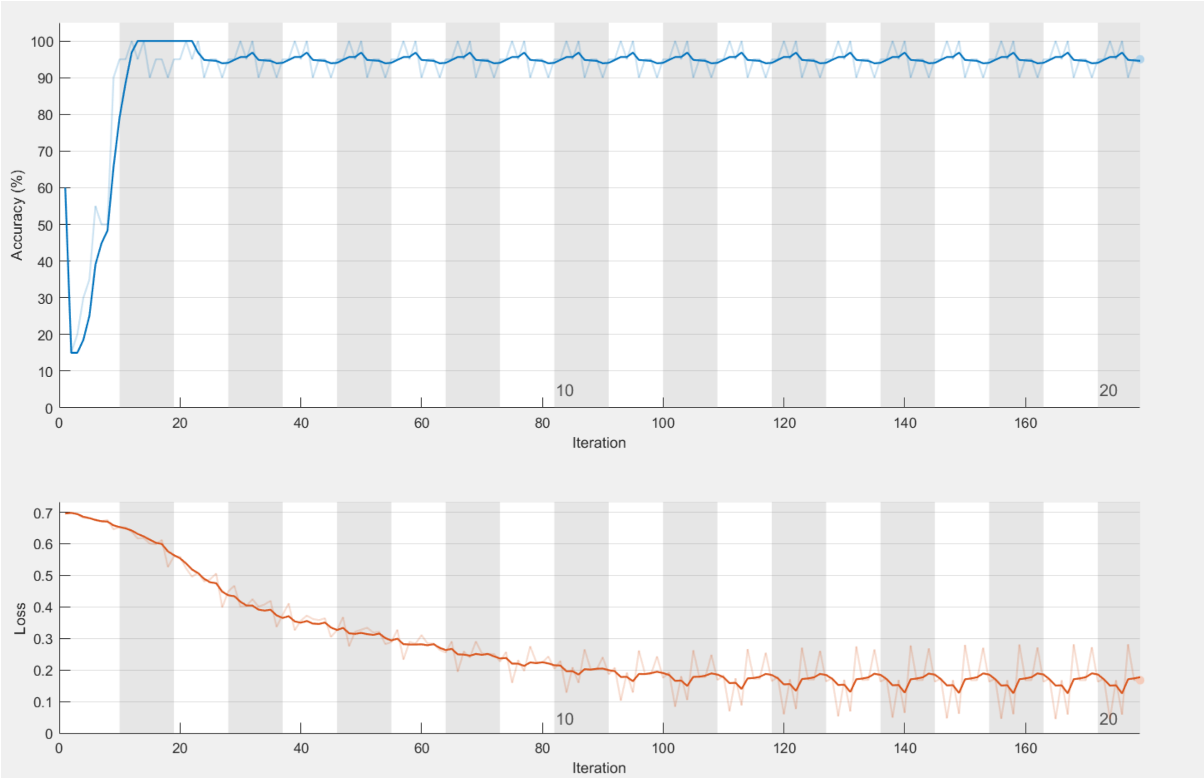


Figure 2: Sag in power system (with different magnitudes) & Training & Testing of LSTM Network


Input Signal
Magnitude
Divisions
Projection matrix(DISCRETE)
Magnitude
Divisions
maximum mean discrepancy
Magnitude
Divisions
Projection matrix(moving)
Magnitude
Divisions


Input Signal
Magnitude
Divisions
Projection matrix(DISCRETE)
Magnitude
Divisions
maximum mean discrepancy
Magnitude
Divisions
Projection matrix(moving)
Magnitude
Divisions



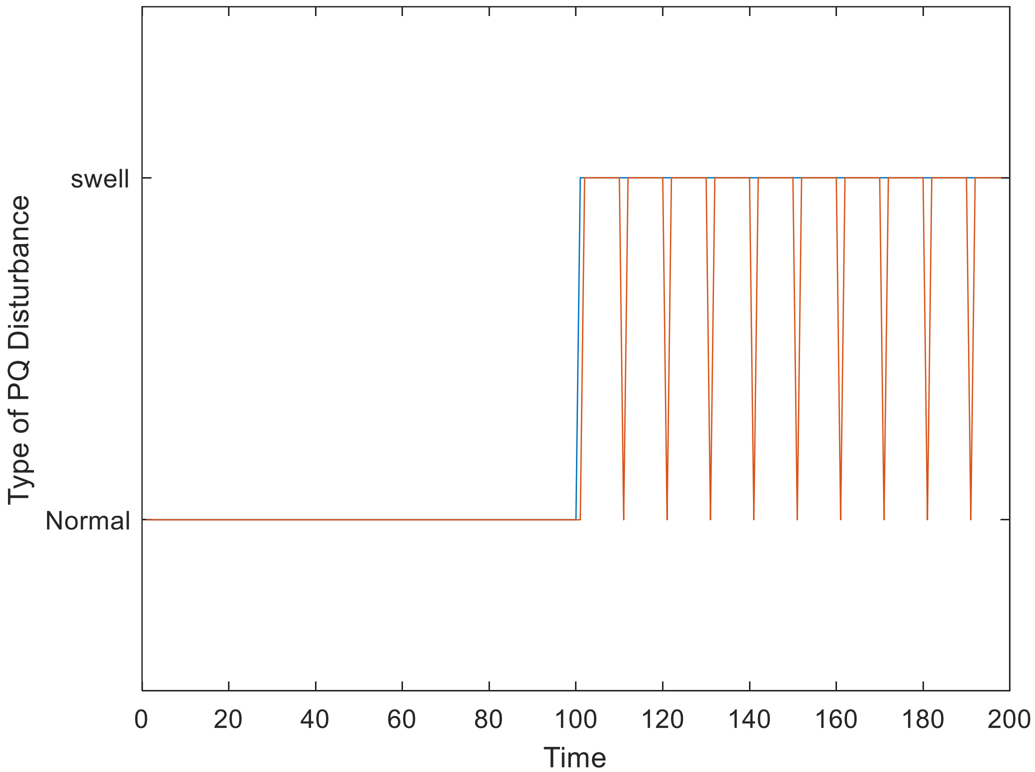
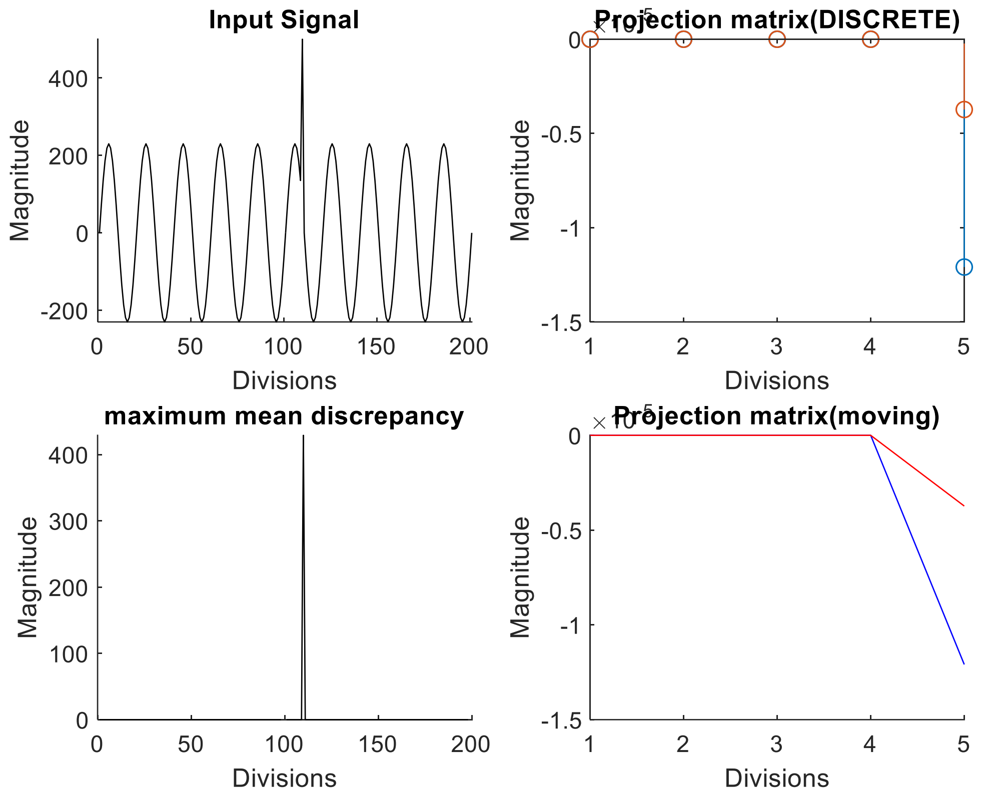
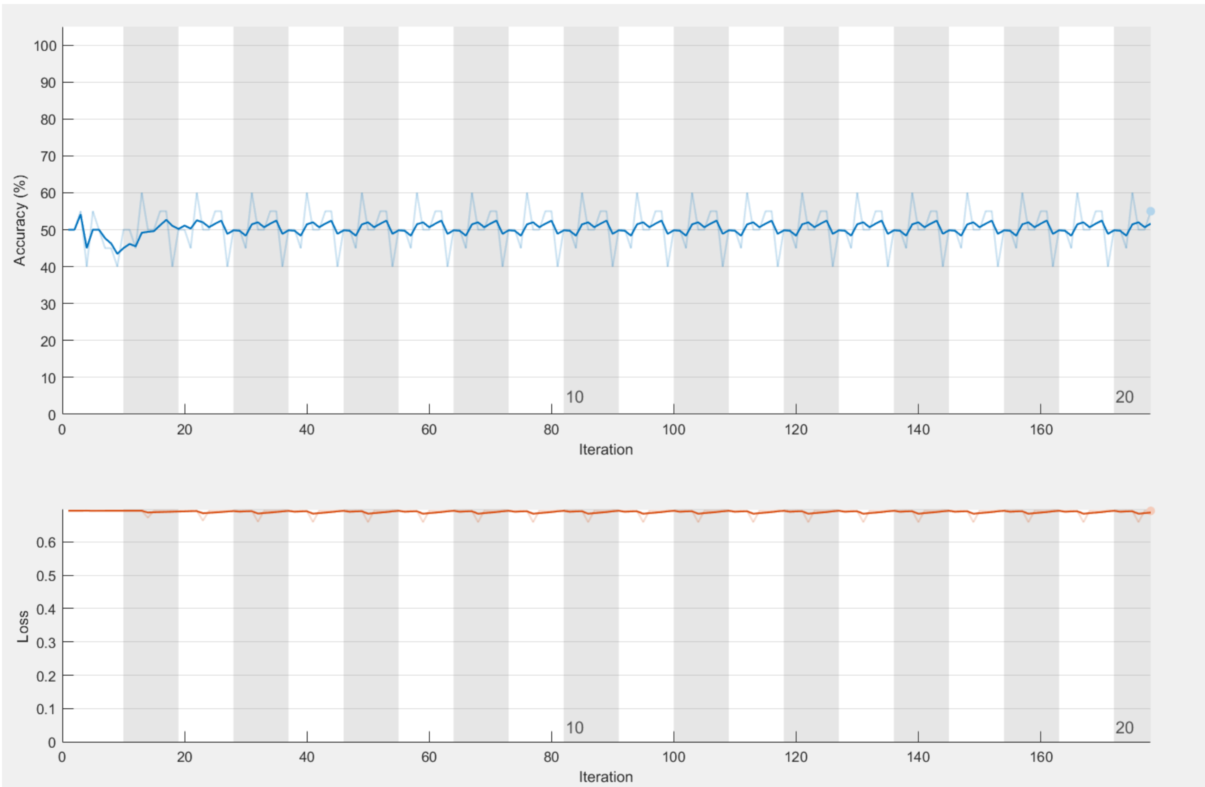


Figure 3: Swell in power system(with different magnitudes) & Training & Testing of LSTM Network


Input Signal
Magnitude
Divisions
Projection matrix(DISCRETE)
maximum mean discrepancy
Projection matrix(moving)


Accuracy (%)
Iteration
Loss

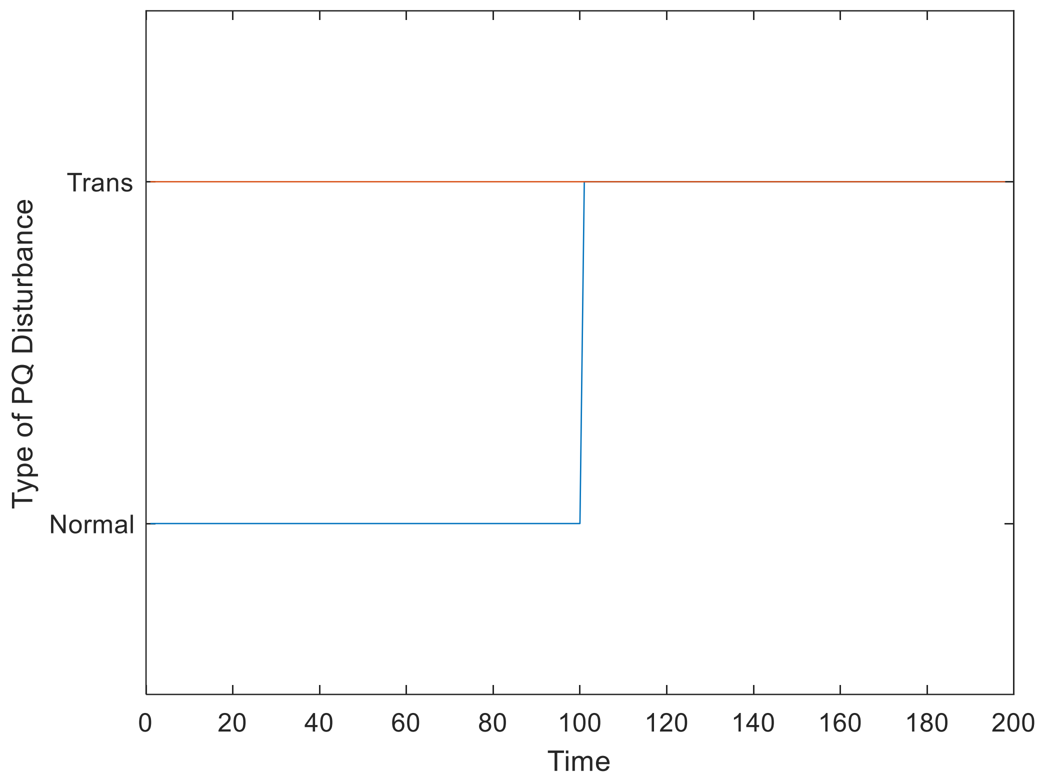
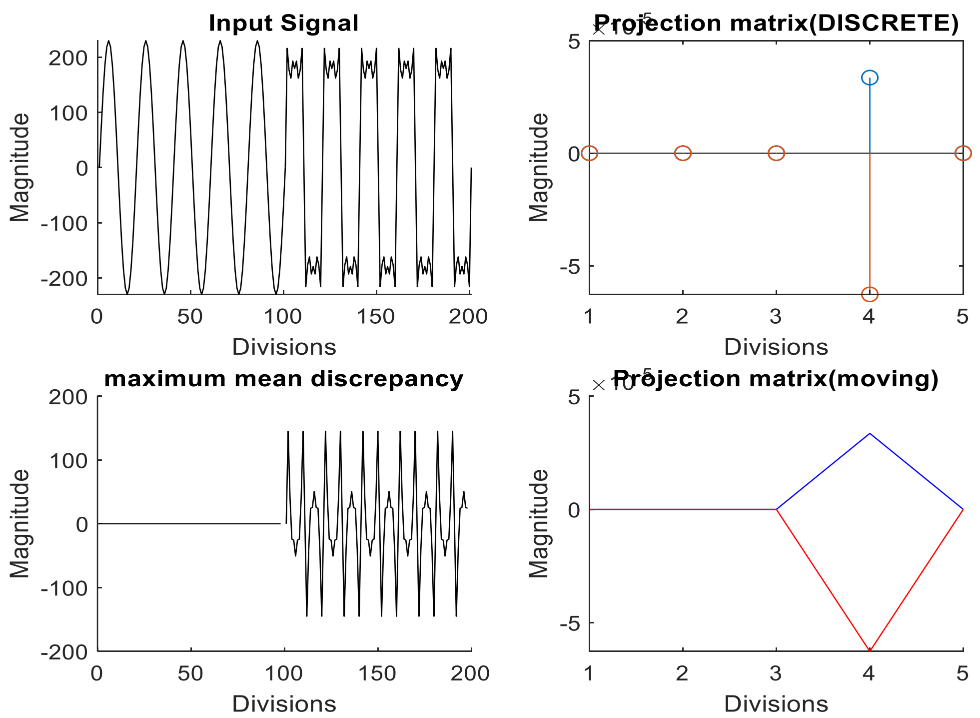
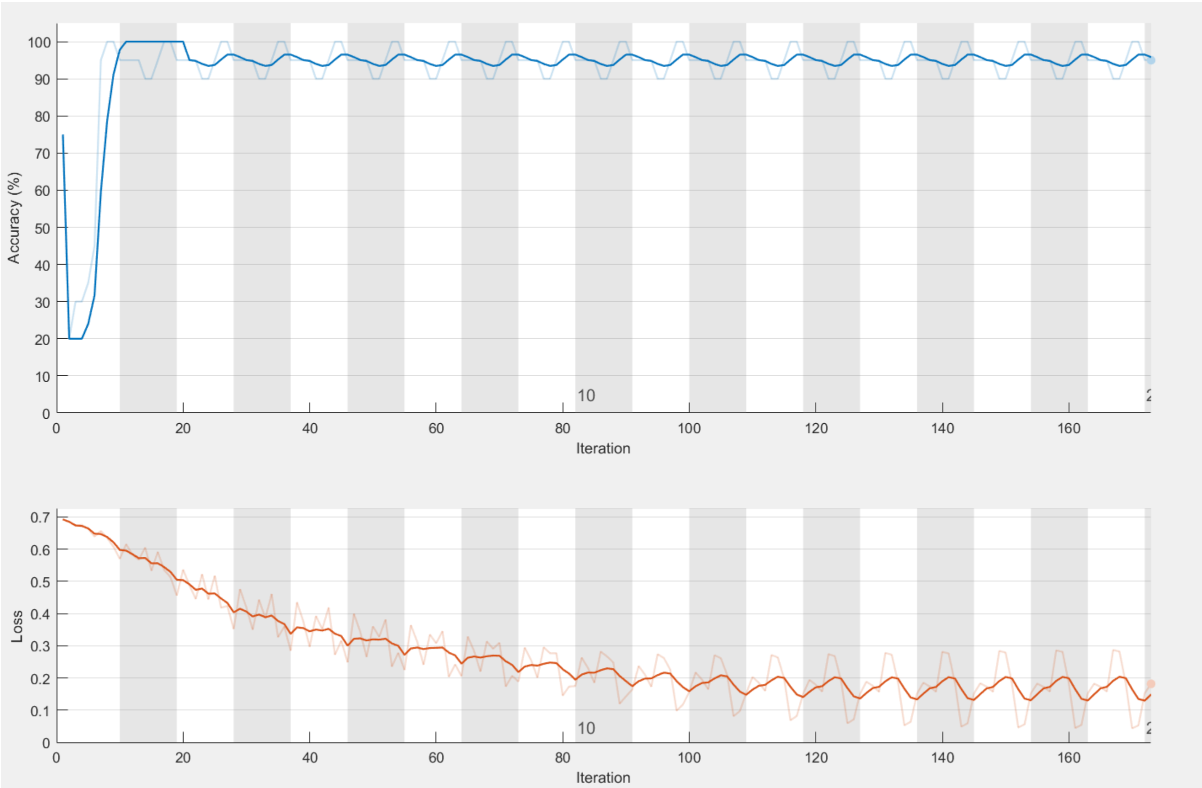


Figure 4: Transients in power system & Training & Testing of LSTM Network





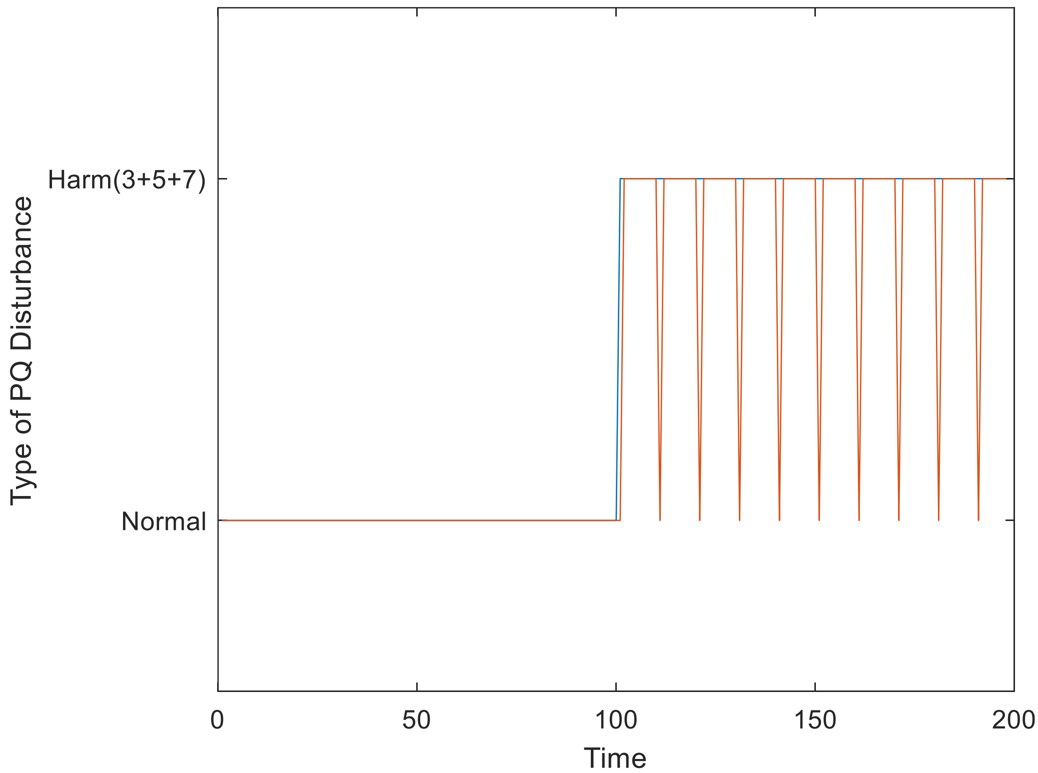
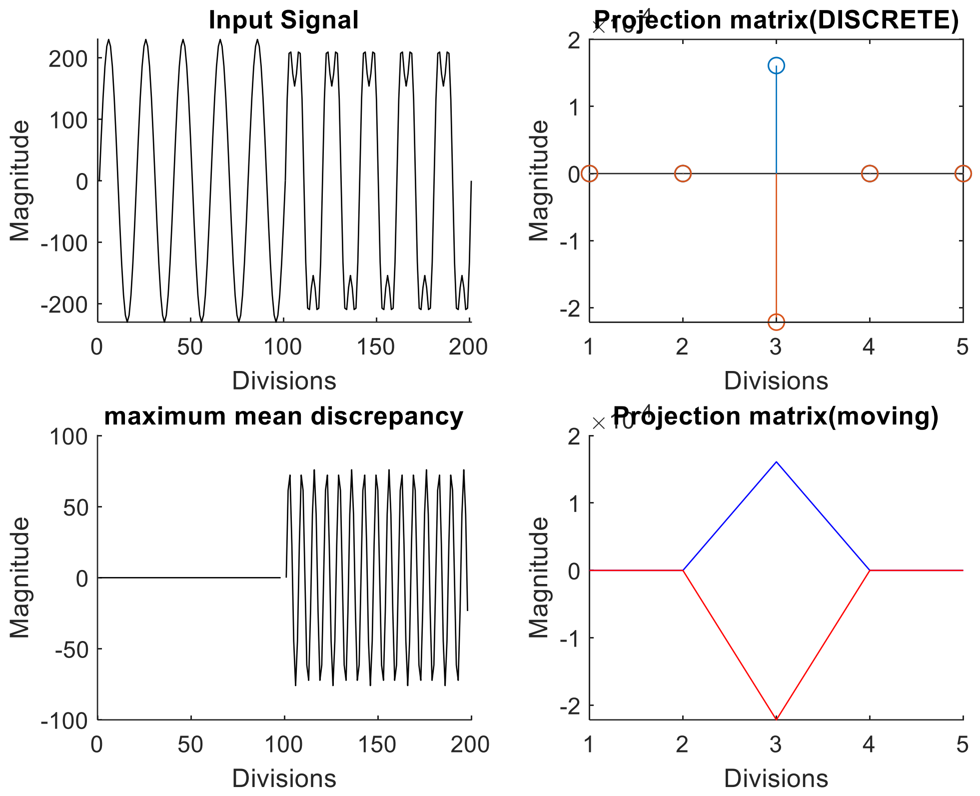
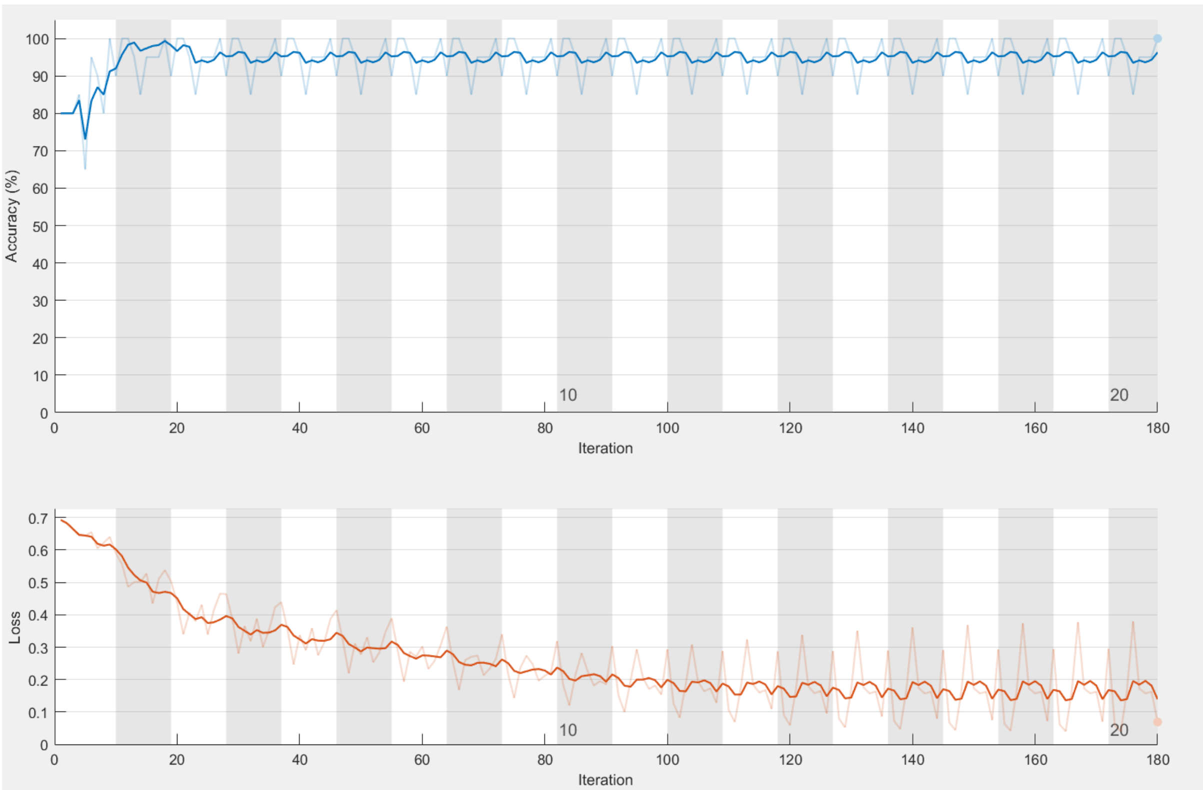


Figure 5: Harmonics in Power system (3^{rd} + 5^{th} +7^{th}) & Training & Testing of LSTM Network


Input Signal
Magnitude
Divisions
Projection matrix(DISCRETE)
$\times 10^4$
maximum mean discrepancy
Projection matrix(moving)


Accuracy (%)
Iteration
Loss
10
20

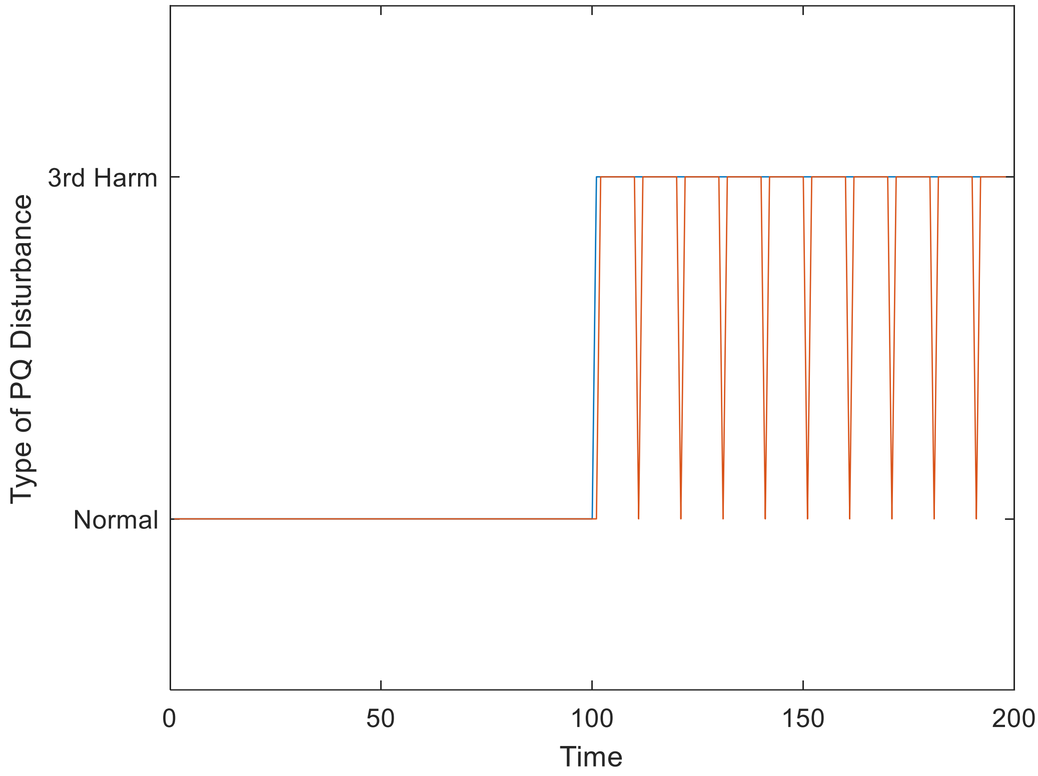
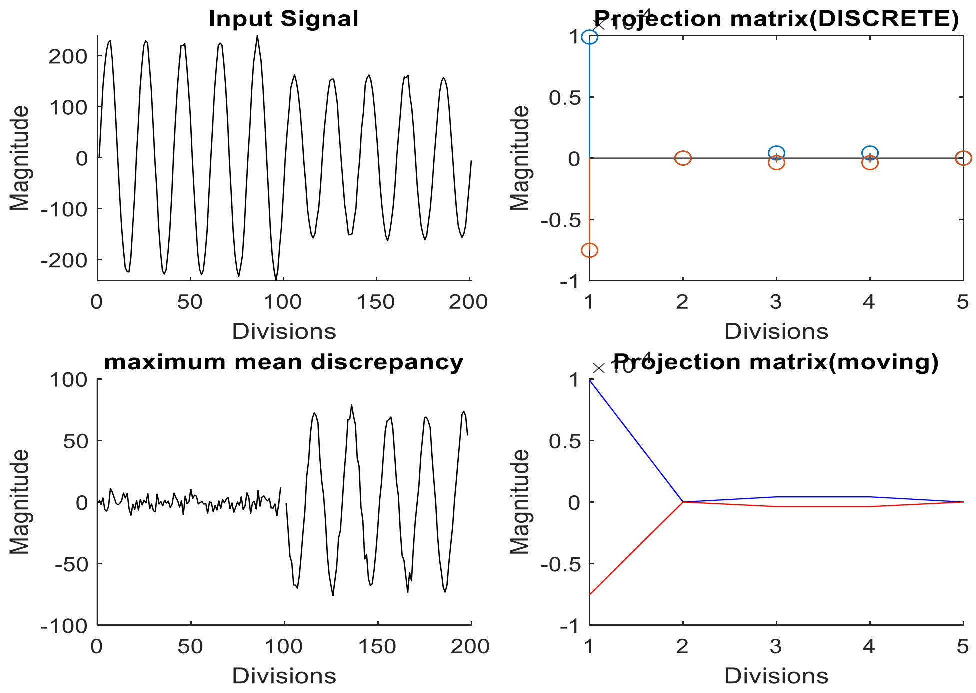
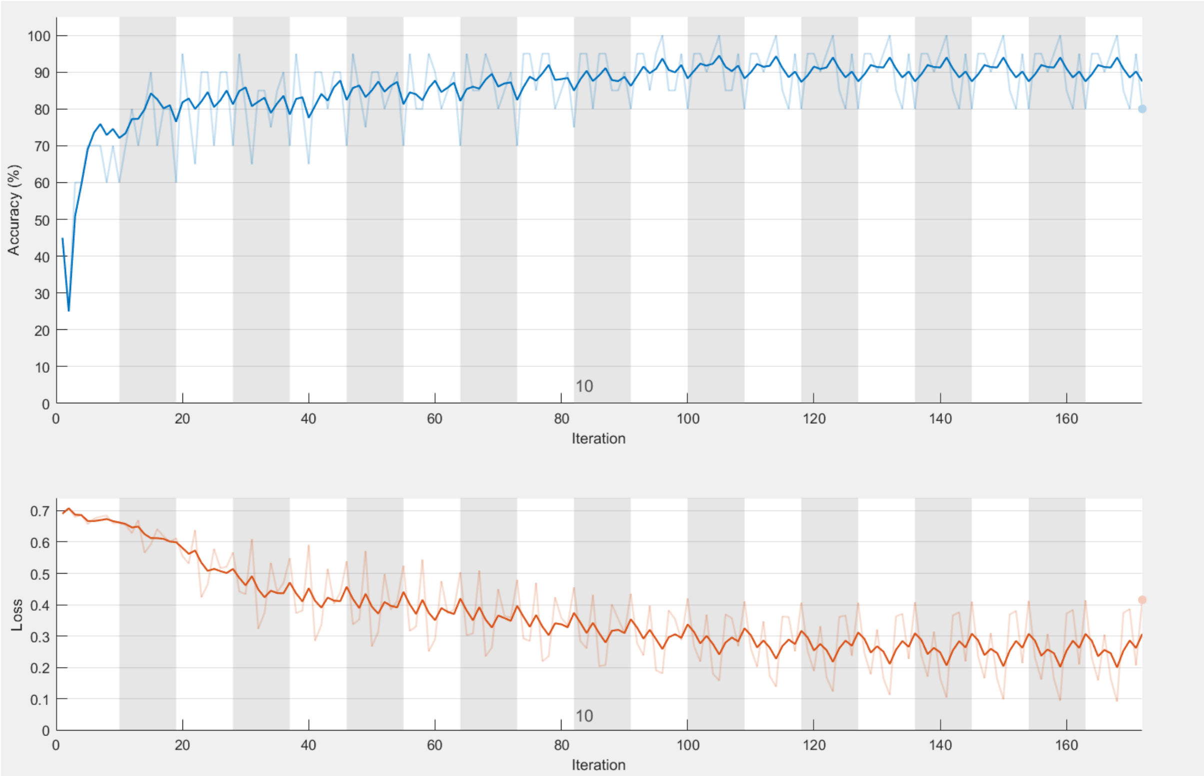


Figure 6: 3rd Harmonic in Power system & Training & Testing of LSTM Network





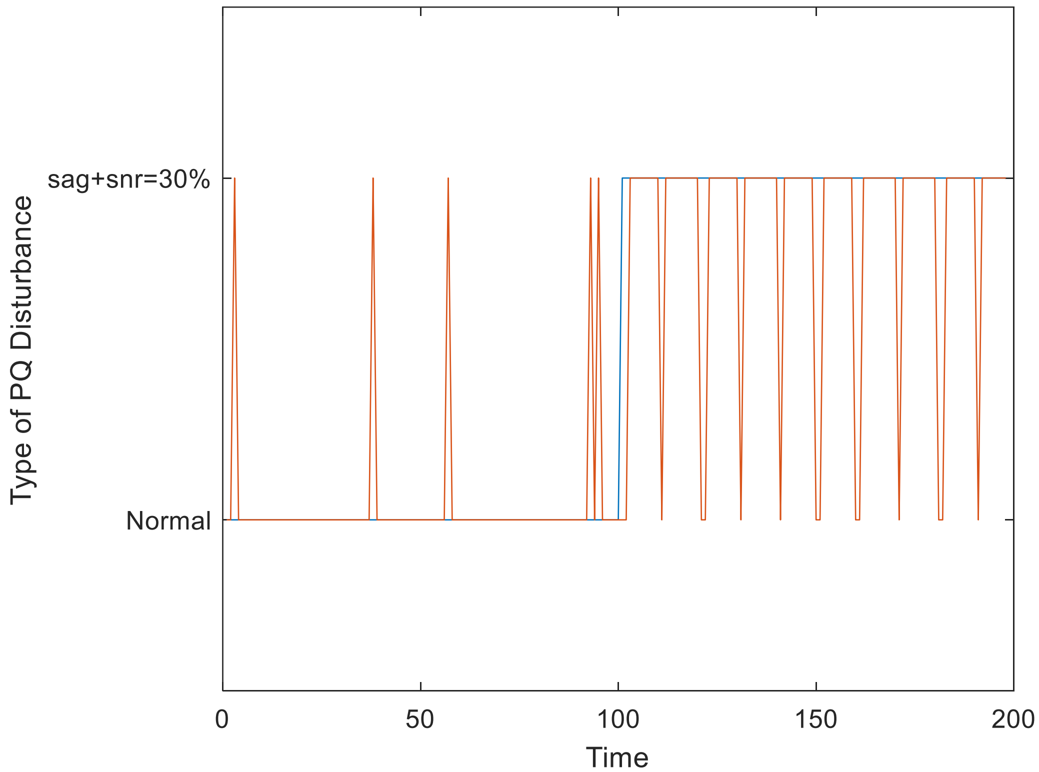
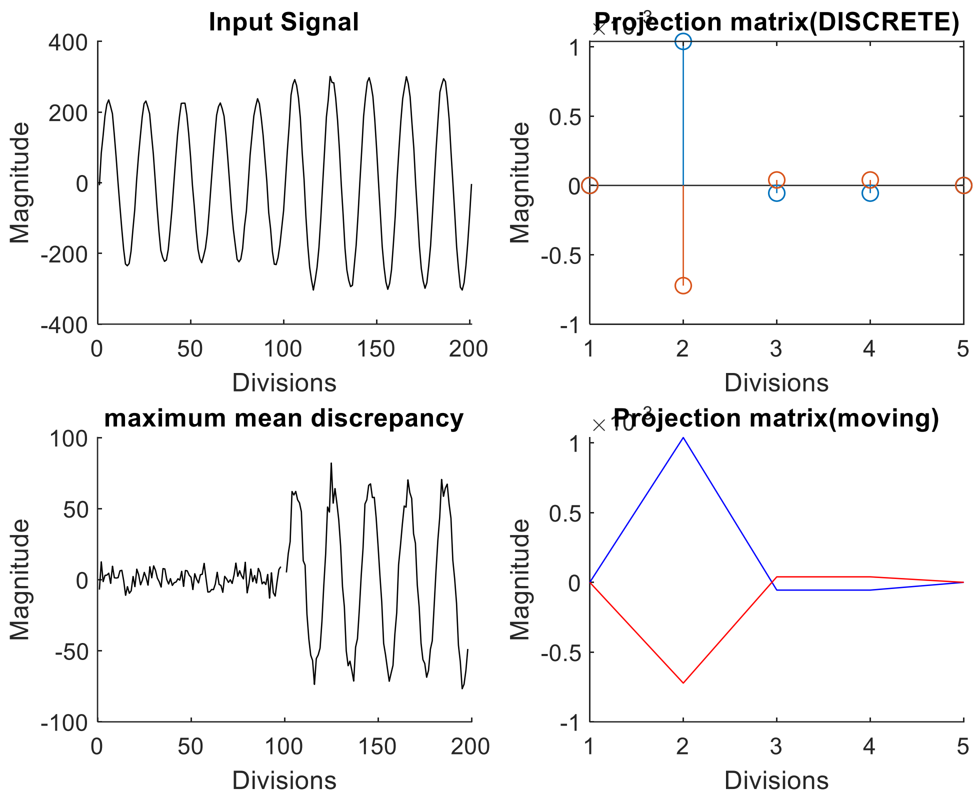
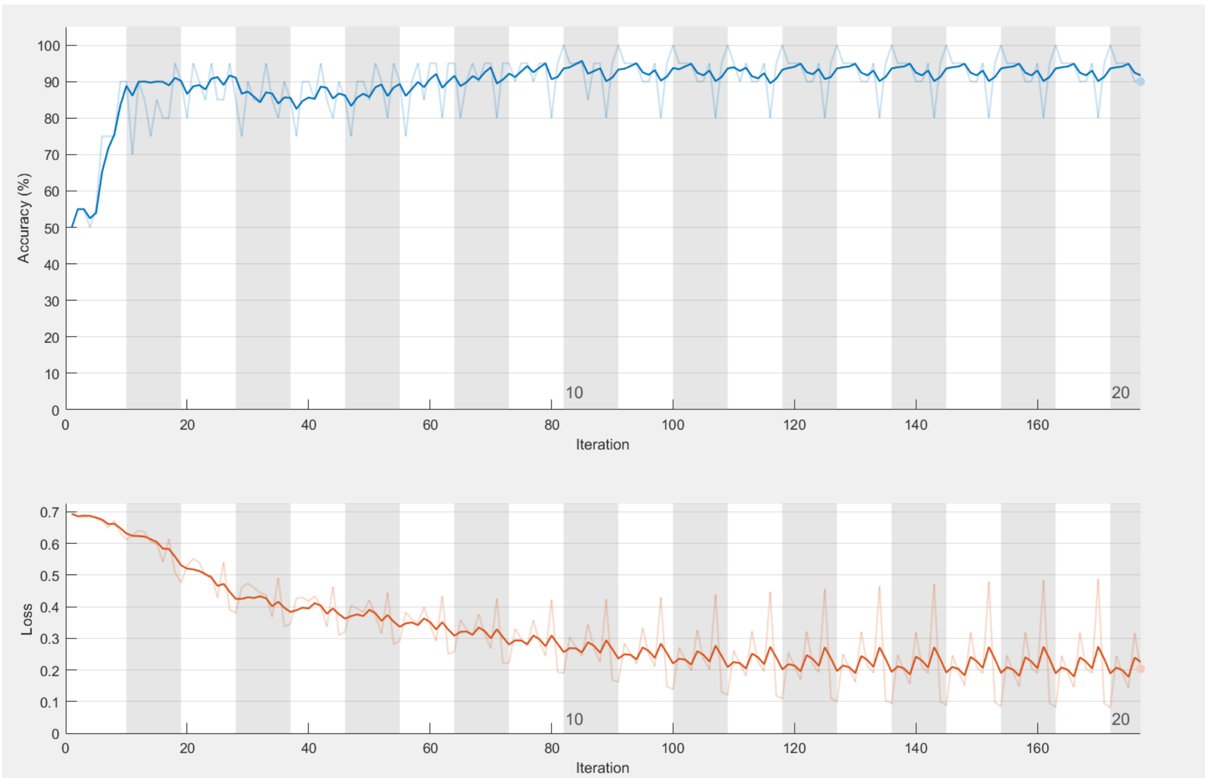


Figure 7: Sag with signal to noise ratio = 30% & Training & Testing of LSTM Network


Input Signal
Projection matrix(DISCRETE)
maximum mean discrepancy
Projection matrix(moving)
Magnitude
Divisions


Accuracy (%)
Iteration
Loss
Iteration

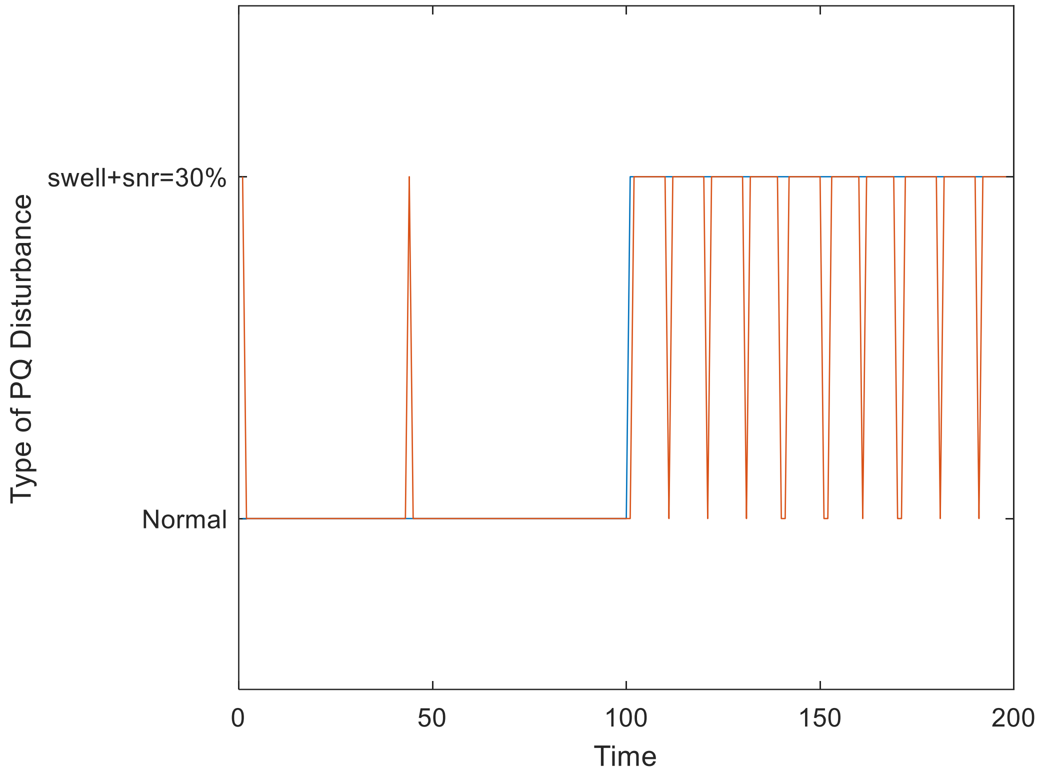
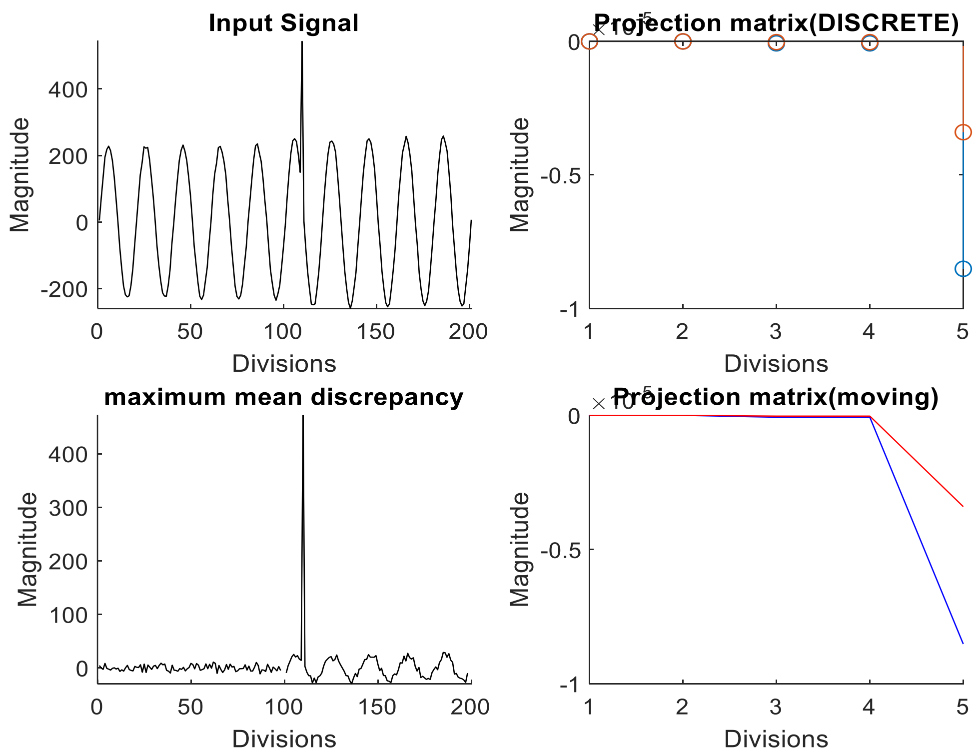


Figure 8: Swell with signal to noise ratio = 30% & Training & Testing of LSTM Network



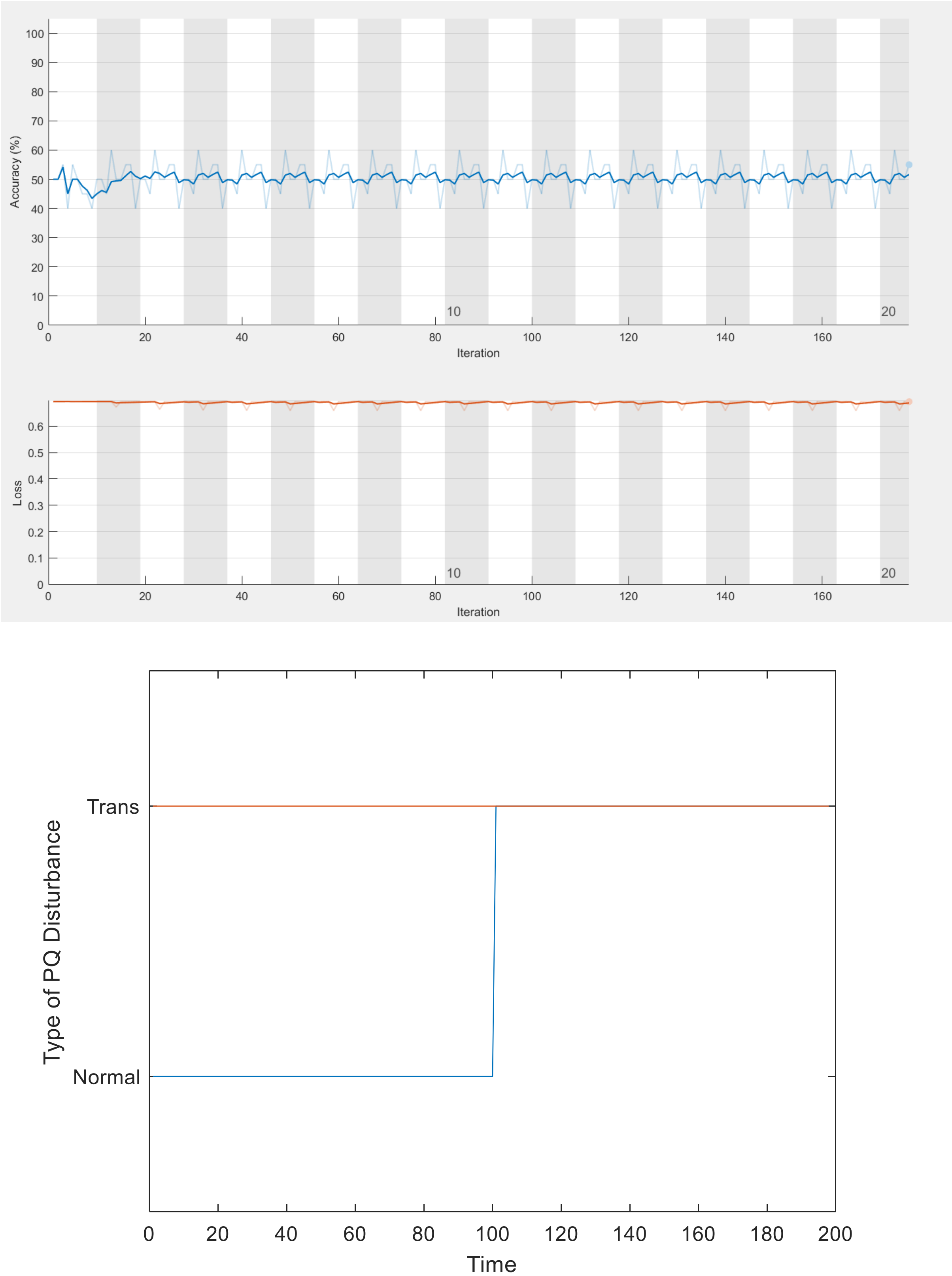
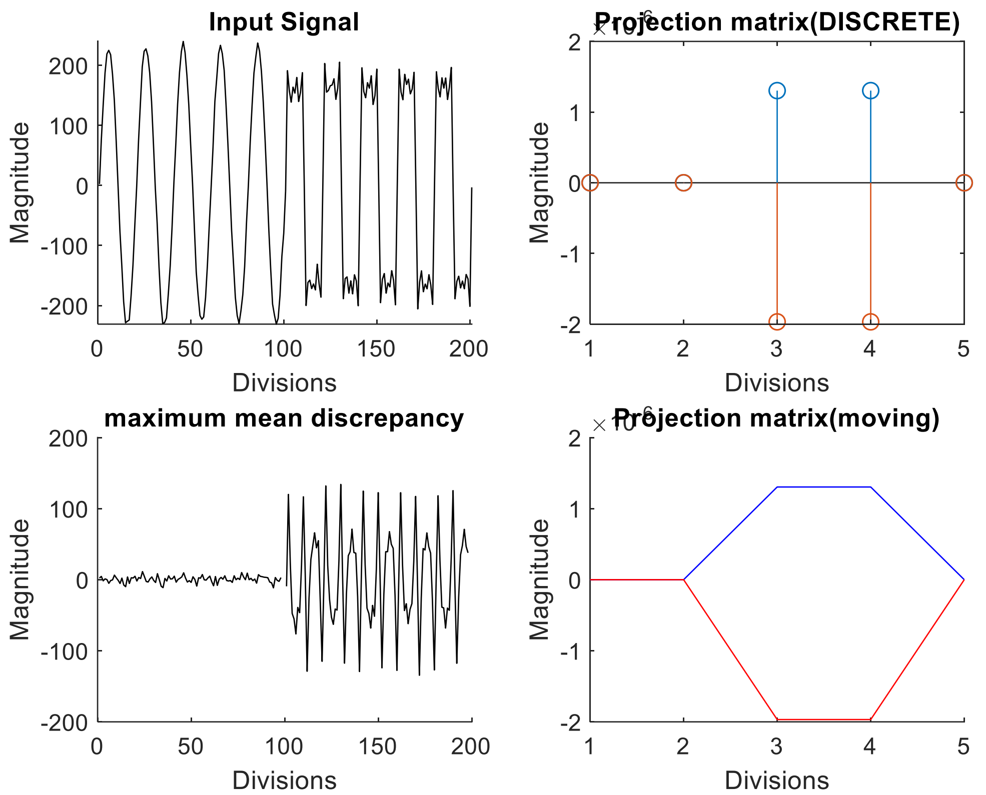
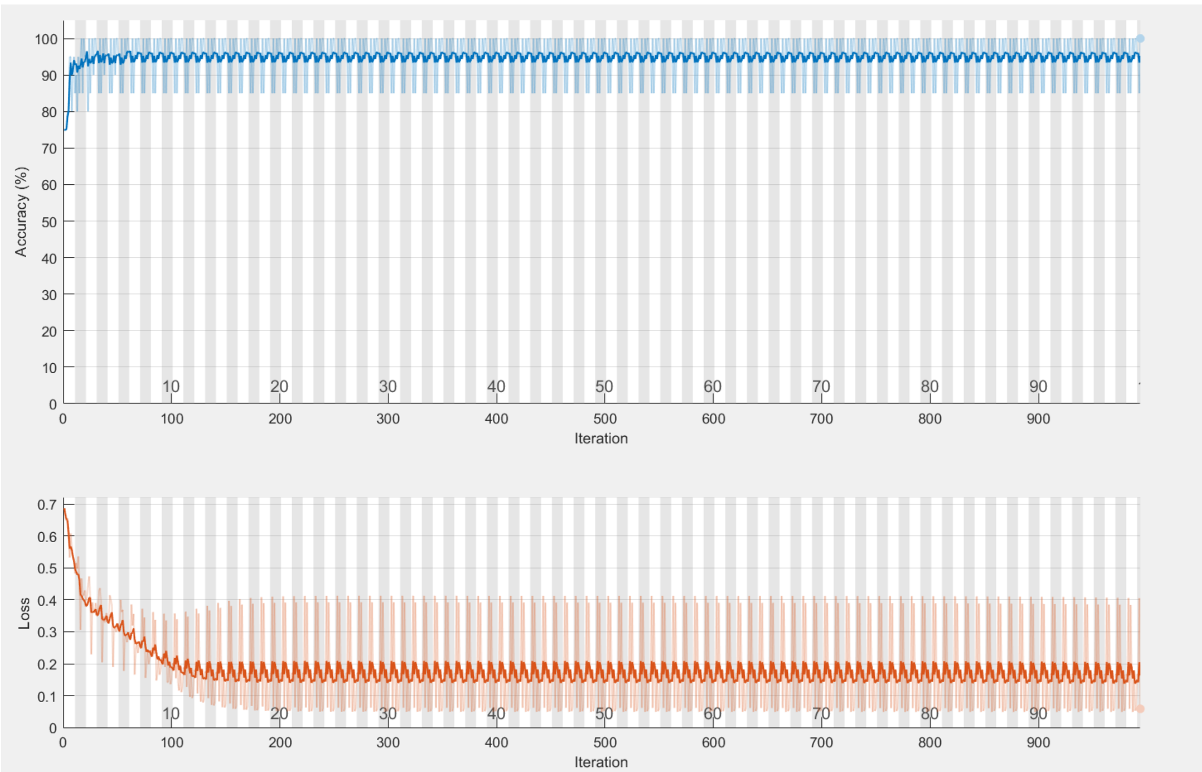


Figure 9: Transients with signal to noise ratio = 10% & Training & Testing of LSTM Network


Input Signal
Projection matrix(DISCRETE)
maximum mean discrepancy
Projection matrix(moving)
Magnitude
Divisions
×10⁶


Accuracy (%)
Loss
Iteration

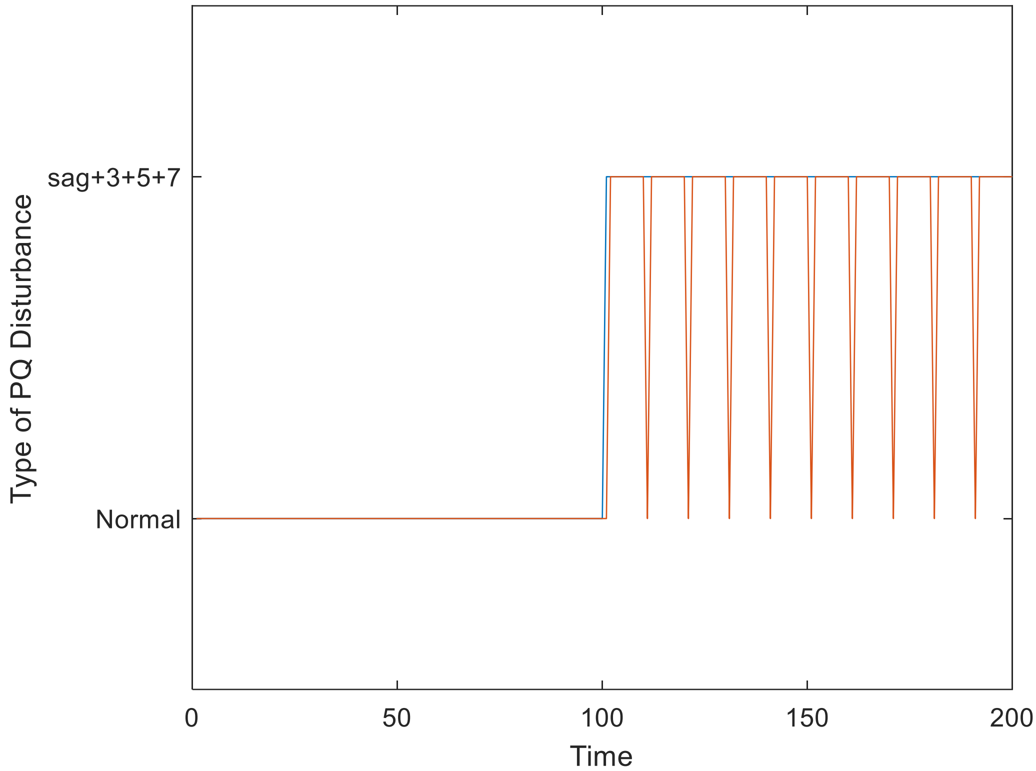
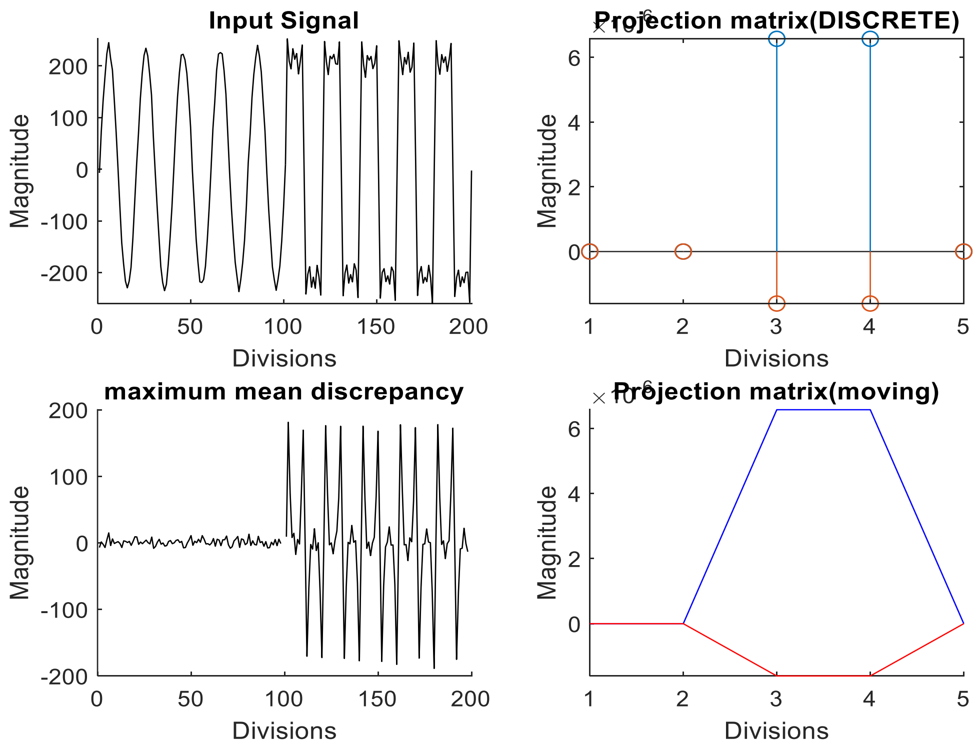
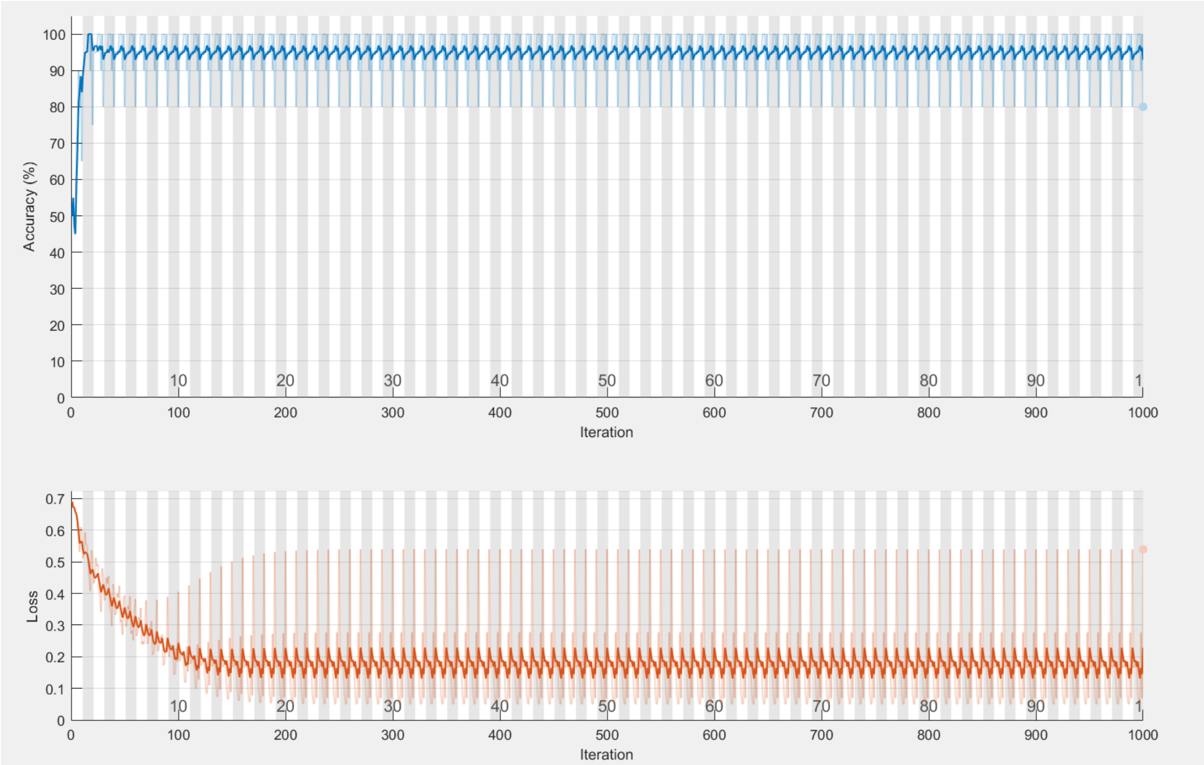


Figure 10: Harmonics + Sag with signal to noise ratio = 35% & Training & Testing of LSTM Network





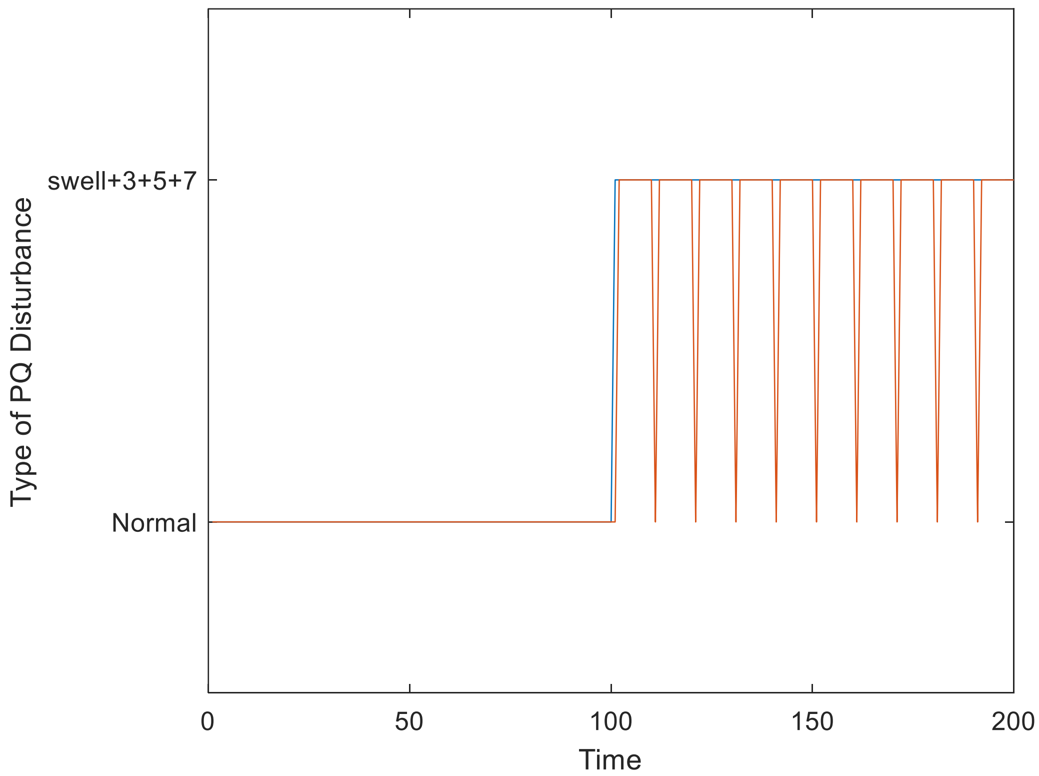
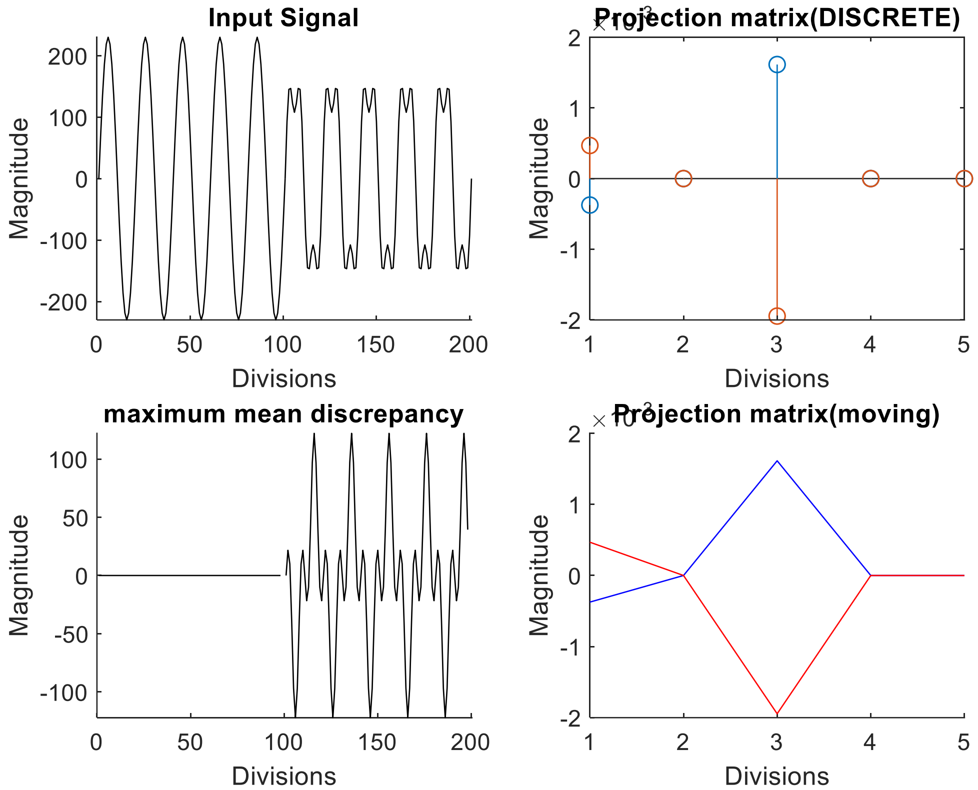
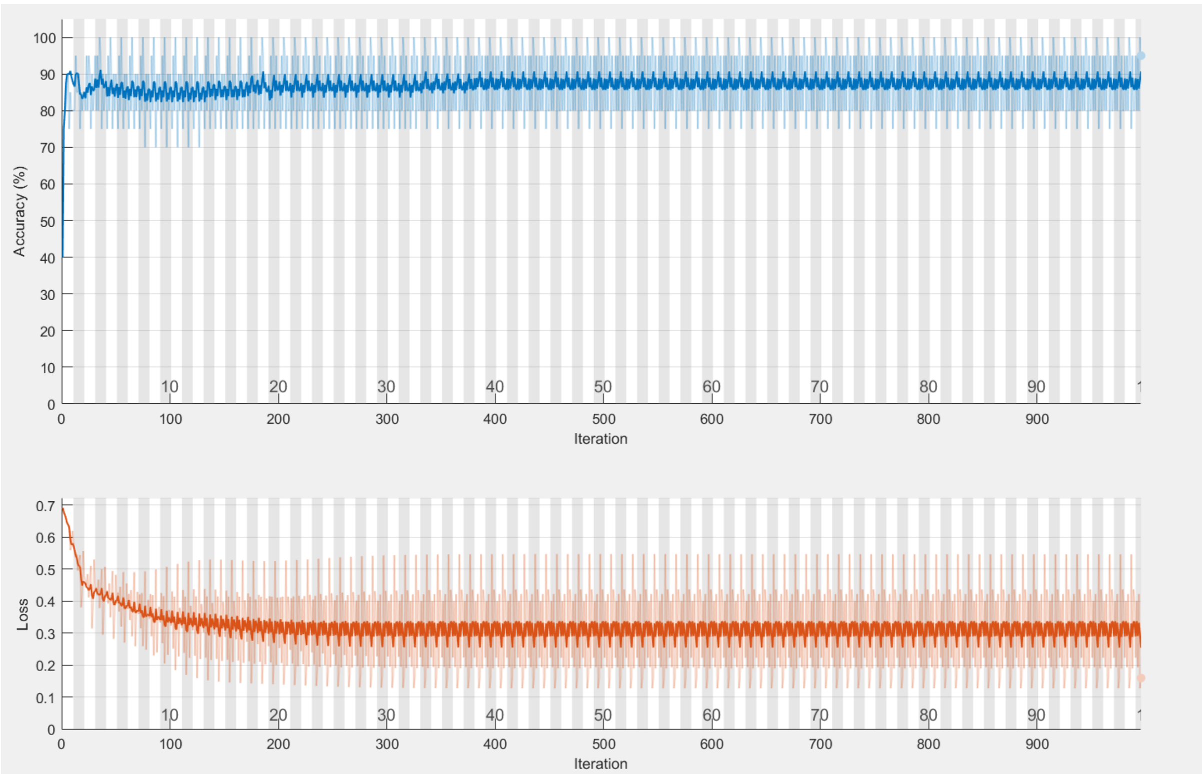


Figure 11: Harmonics + Swell with signal to noise ratio = 38% & Training & Testing of LSTM Network


Input Signal
Magnitude
Divisions
Projection matrix(DISCRETE)
Magnitude
Divisions
maximum mean discrepancy
Magnitude
Divisions
Projection matrix(moving)
Magnitude
Divisions


Accuracy (%)
Iteration
Loss
Iteration

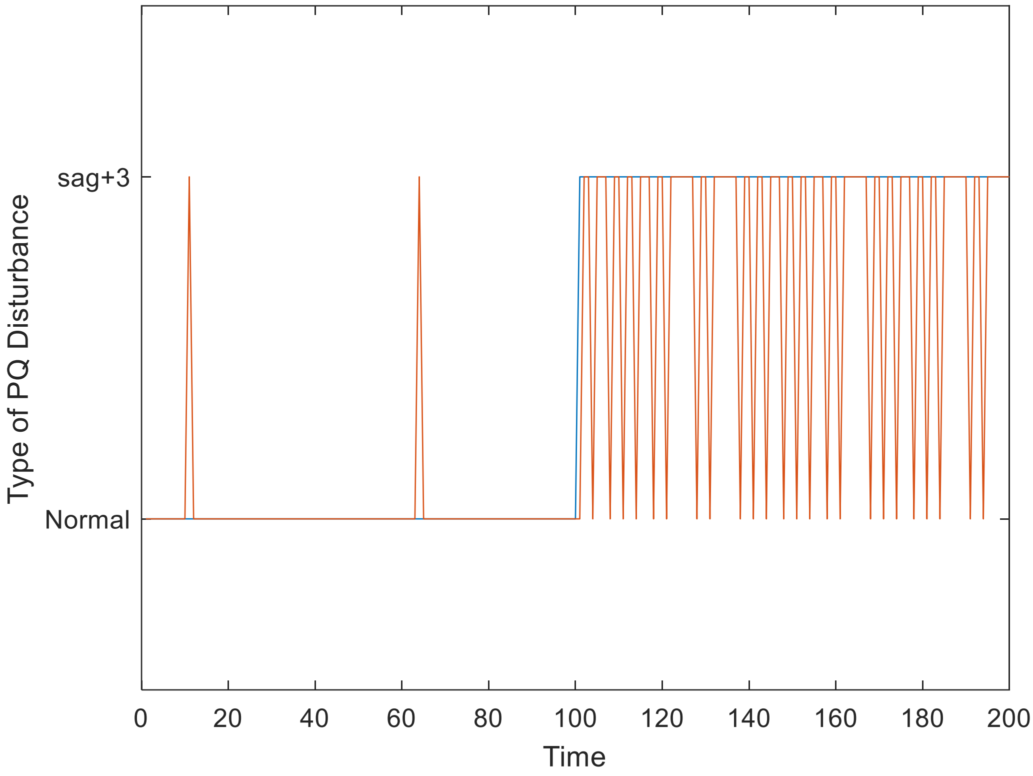
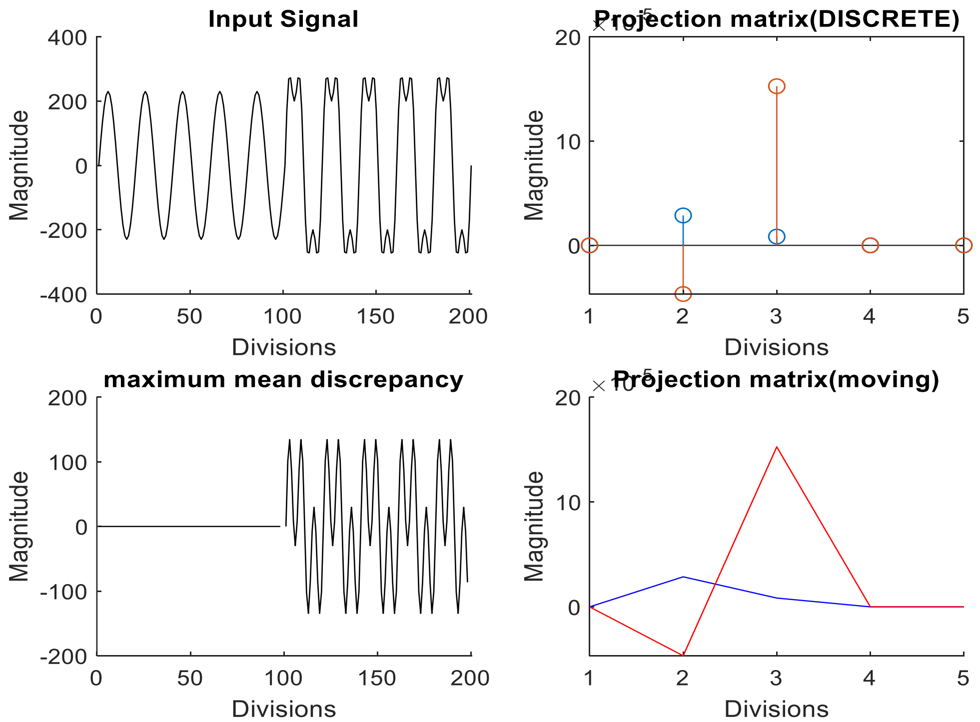


Figure 12:3rd Harmonic+ Sag & Training & Testing of LSTM Network



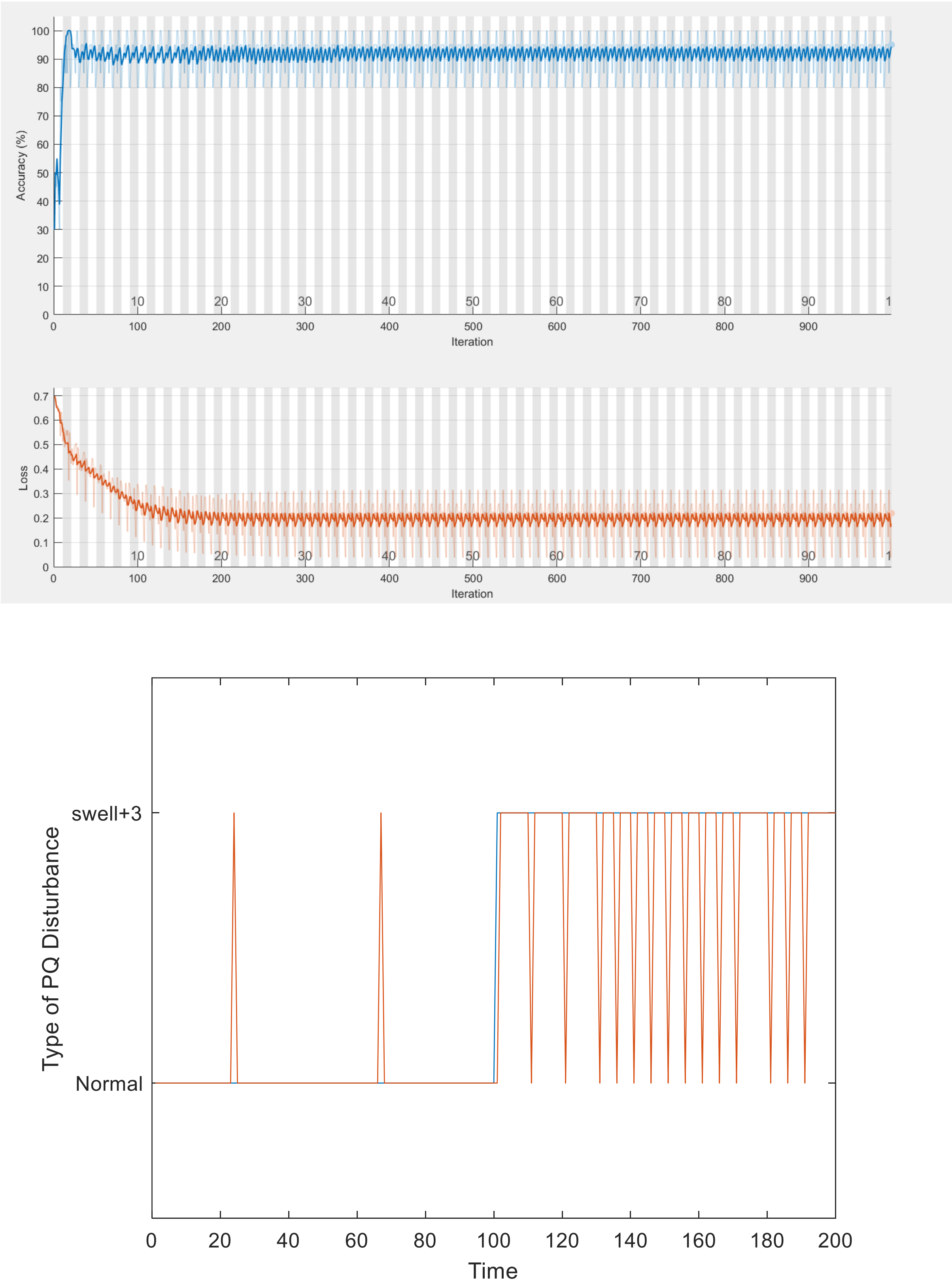
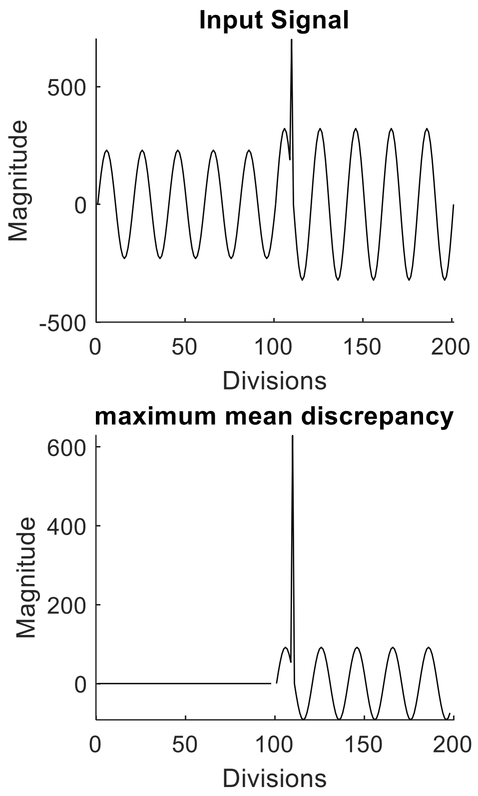
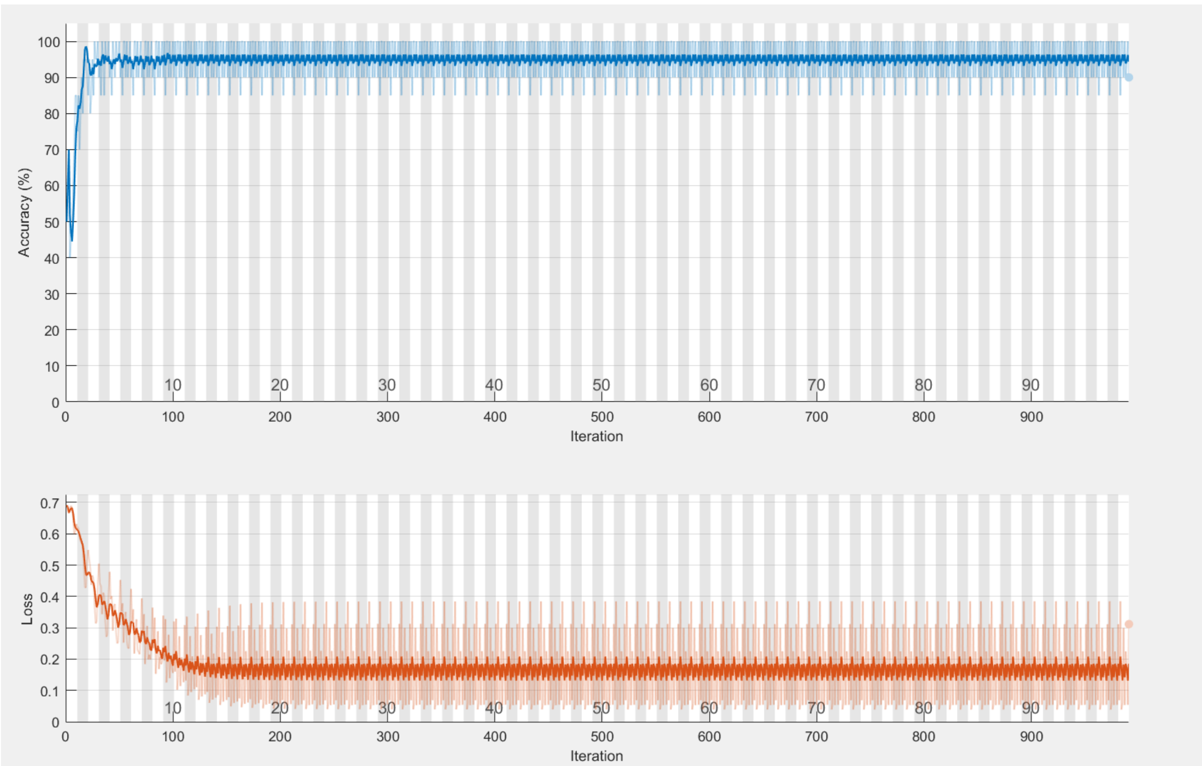


Figure 13: 3rd Harmonic + Swell & Training & Testing of LSTM Network



Input Signal
Magnitude
Divisions
×10-6 Proj matrix(DISC)
Magnitude
Divisions
maximum mean discrepancy
Magnitude
Divisions
×10-6 Proj matrix
Magnitude
Divisions



Accuracy (%)
Iteration
Loss
Iteration

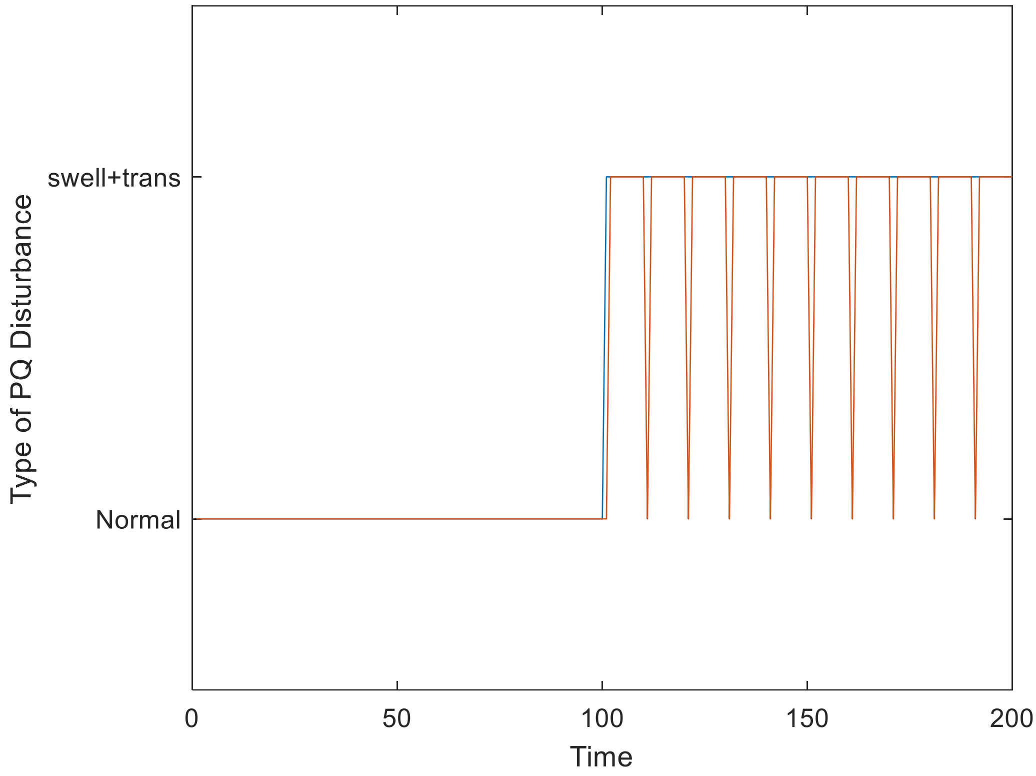
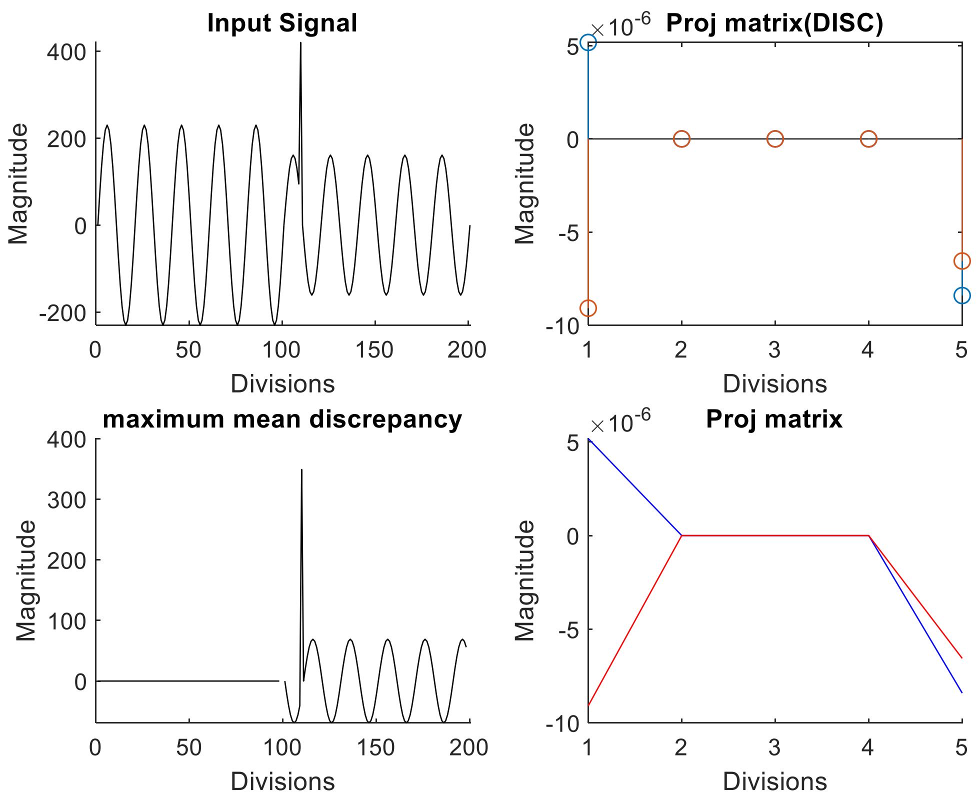


Figure 14: Transients + Swell & Training & Testing of LSTM Network



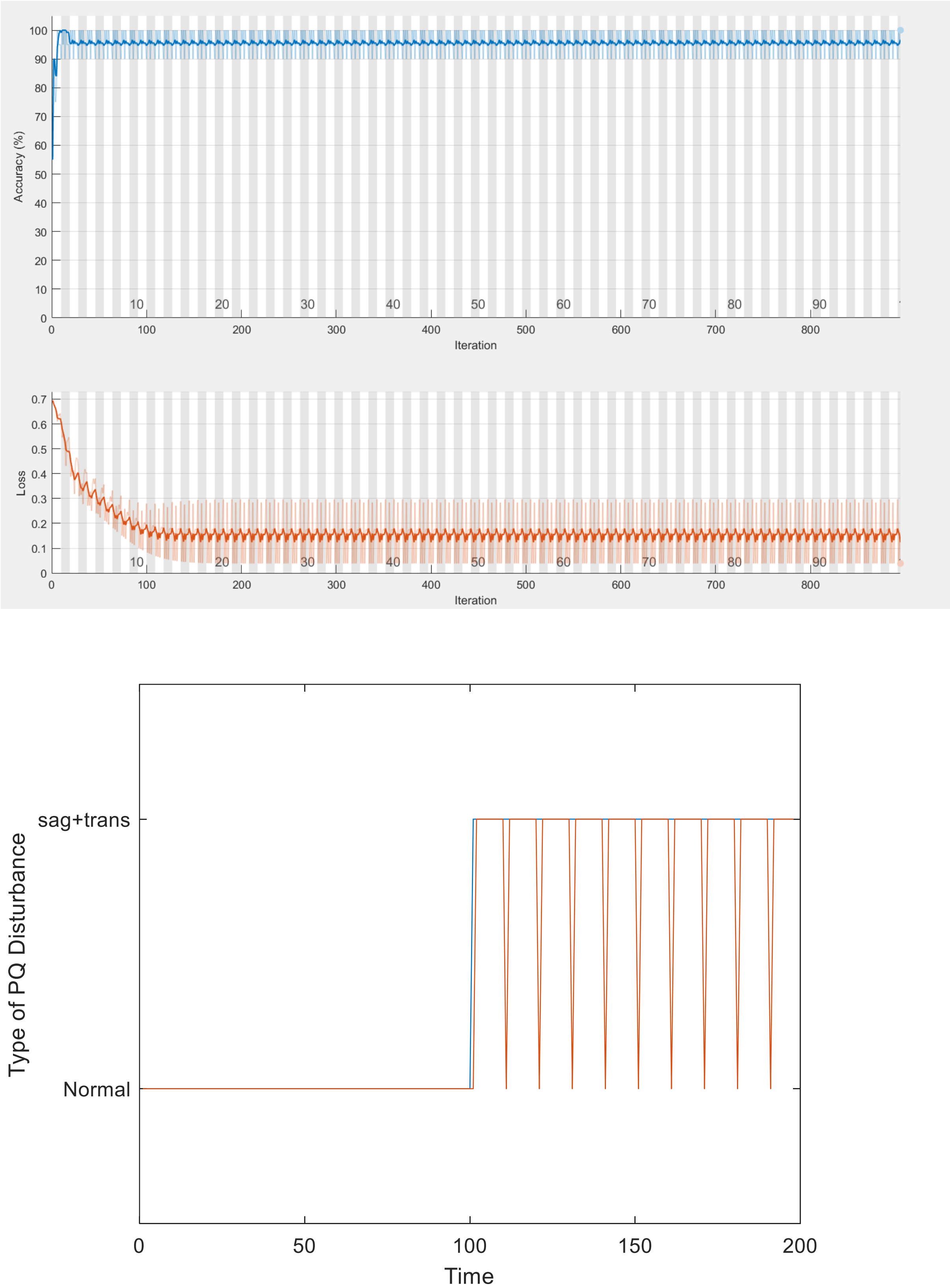
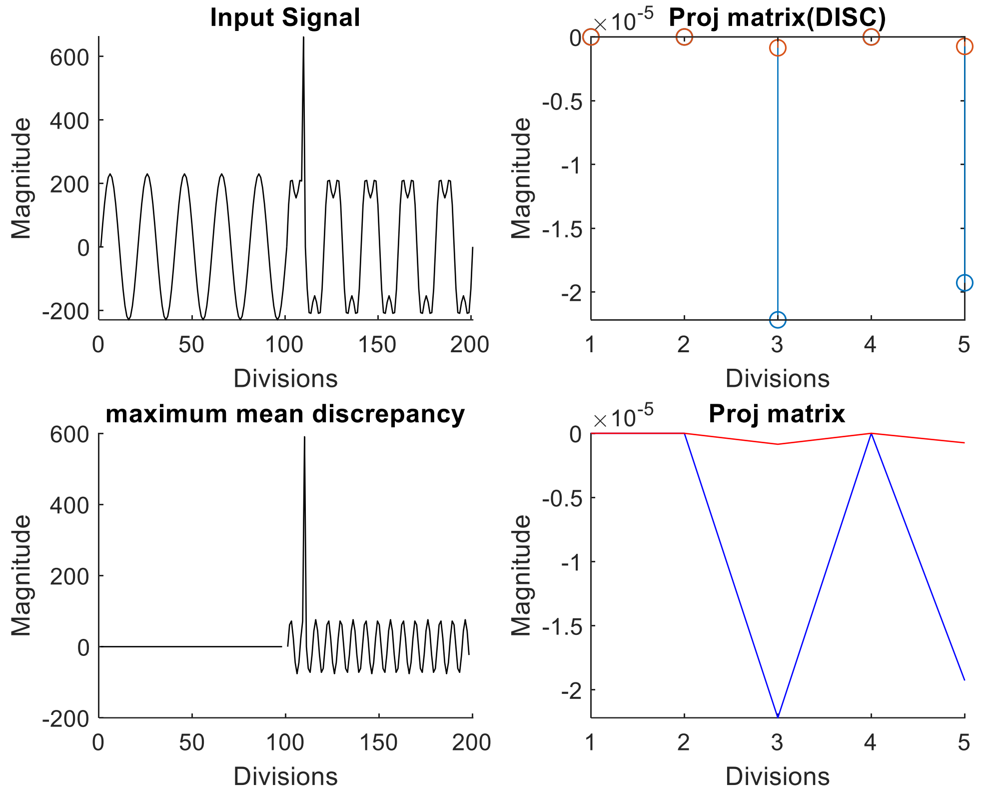
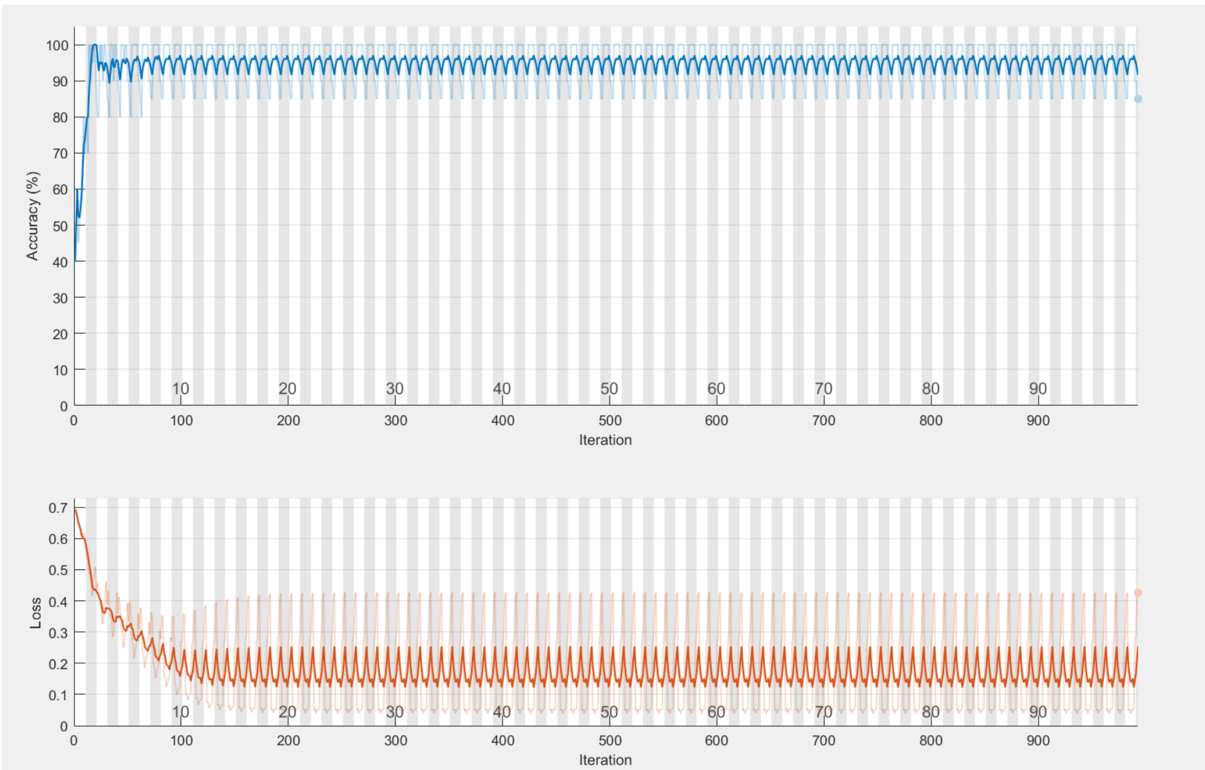


Figure 15: Transients + Sag & Training & Testing of LSTM Network


Input Signal
Proj matrix(DISC)
maximum mean discrepancy
Proj matrix
Magnitude
Divisions


Accuracy (%)
Iteration
Loss
Iteration

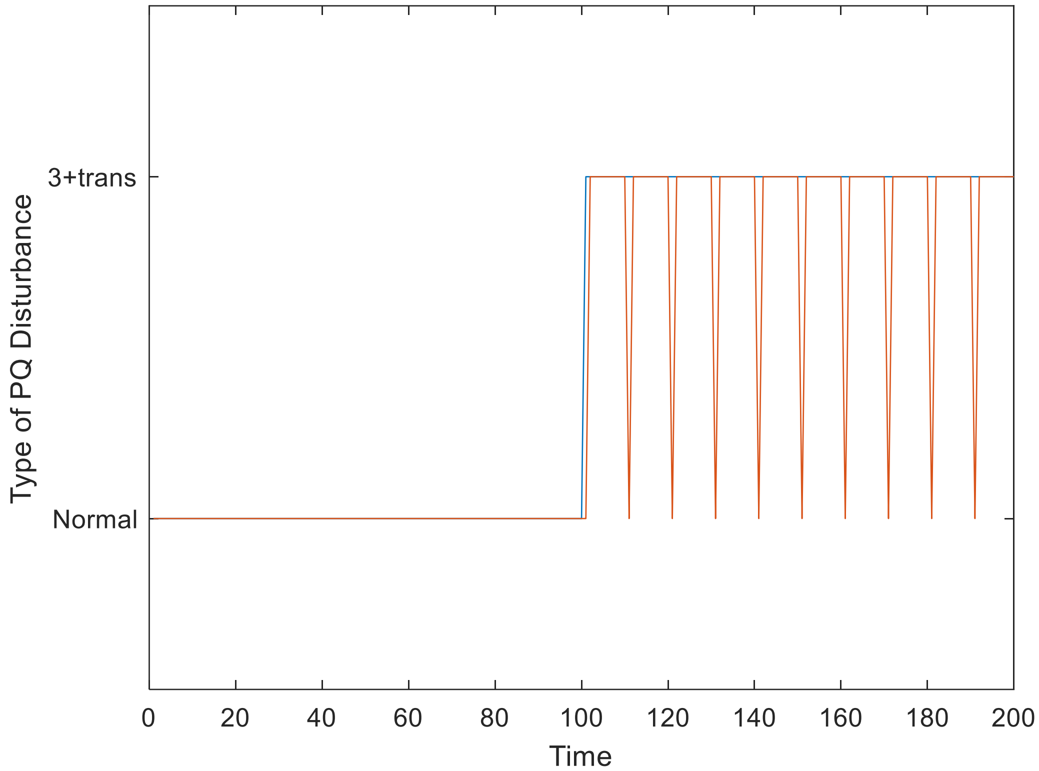
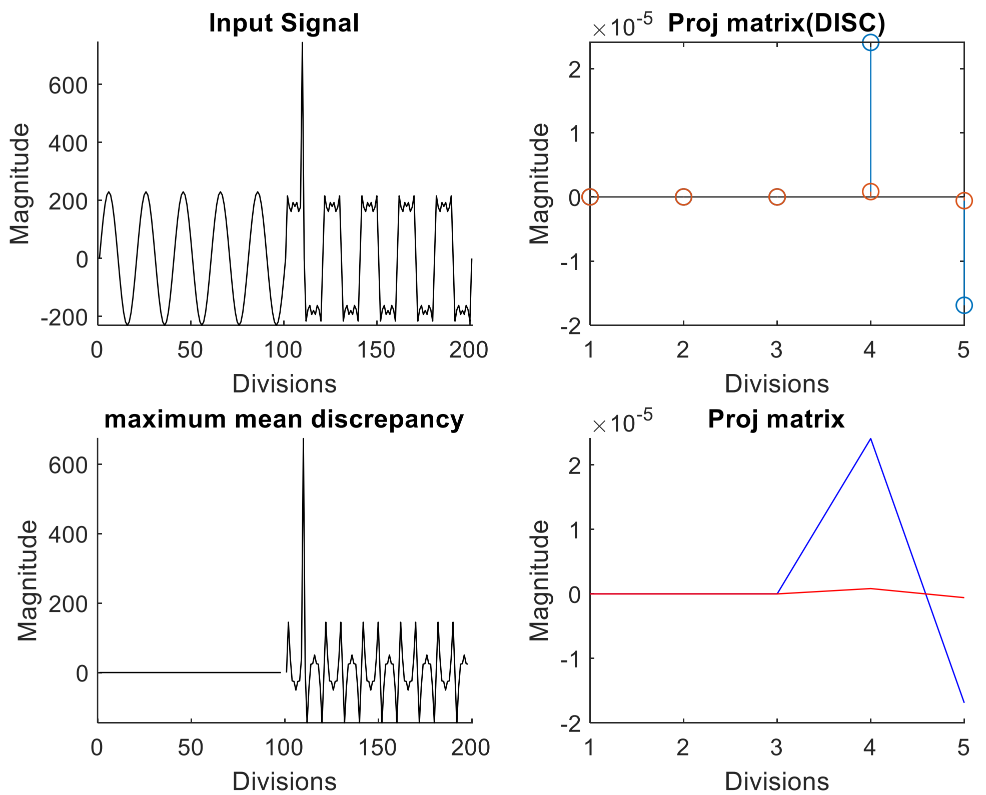
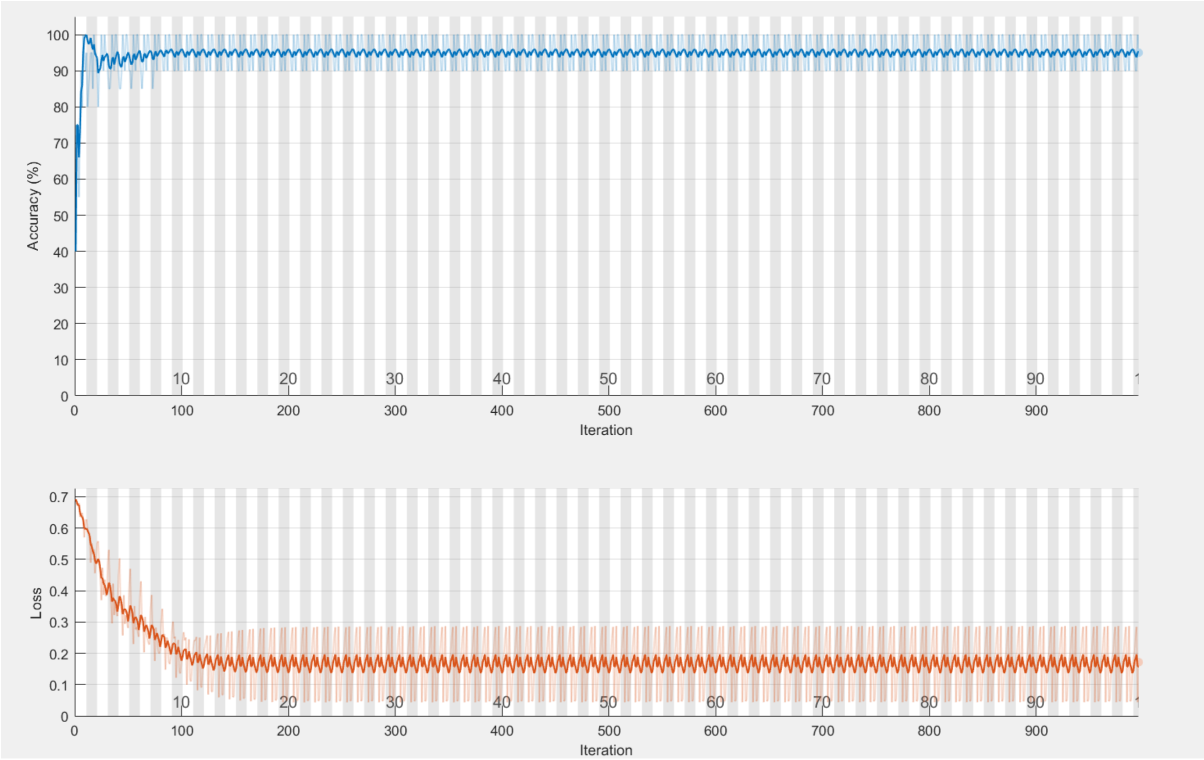


Figure 16: 3rd Harmonic + Transients & Training & Testing of LSTM Network





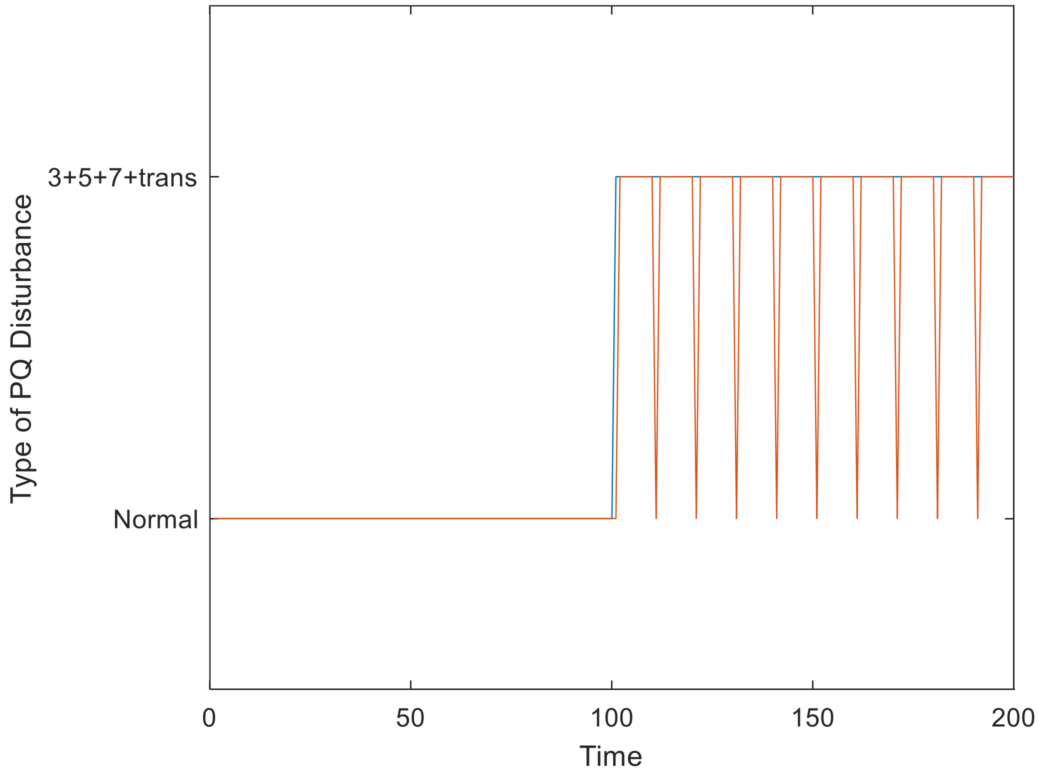


Figure 17: Harmonics + Transients & Training & Testing of LSTM Network

- After seeing the above results let's look at them in the form of words i.e. how to differentiate between 2 identical faults that is if sag or swell of different magnitude will occur how we will be able to differentiate among them. Here we will also see how different type of faults have different type of outputs. In case of sag, if the magnitude of sag is more than the value of projection matrix will be comparatively lower as compared to the sag which is closer to the ideal waveform that we should get. The same is valid for swell also. In case of harmonics & transients we get a unique waveform which tells us about the fault. When there is intermixing of faults especially sag/swell + harmonics the projection shows us about the sag amount & in this process harmonics is ignored but this is taken care for suitably by the maximum mean discrepancy matrix which shows us the exact harmonics present.

- **Impact of noise on fault detection:** As we know nothing in this world is ideal & so the voltage & current are also not free from noise. In this part we study how noise effects detection of various faults. The experiment is performed for both simulated as well as industrial data & we can high level of accuracy in detection in both the types. In simulated form we have distorted the wave till the time detection is not achievable this situation is very rare to observe in real time data's. The data's are distorted to extent of as high as 30 % Signal to noise ratio. Comprehensive evaluation of our algorithm shows the efficacy of the process to detect various faults.

Type of disturbance	**SNR**	**Accuracy in detection**
Sag	100%	99%
Sag	33%	90%
Swell	100%	99%
Swell	35%	95%
Transients	100%	99%
Harmonics	100%	99%
Sag		88%
Swell		82%

Type of Fault	Accuracy in Detection
Sag + 3^{rd} Harmonic (SNR=35%)	87%
Sag + 5^{th} Harmonic (SNR=30%)	94.5%
Sag +3^{rd} Harmonic + 5^{th} Harmonic (SNR=30%)	95%
Swell + 3^{rd} Harmonic (SNR=25%)	92%
Swell + 5^{th} Harmonic (SNR=28%)	85%
Swell + 3^{rd} Harmonic + 5^{th} Harmonic (SNR=33%)	95%
Sag + 3^{rd} Harmonic + Transients (SNR=26%)	95%
Sag + 5^{th} Harmonic + Transients (SNR=27%)	95.5%
Sag + 3^{rd} Harmonic + 5^{th} Harmonic + Transients (SNR=30%)	95%
Swell + 3^{rd} Harmonic + Transients (SNR=34%)	91%
Swell + 5^{th} Harmonic + Transients (SNR=29%)	85%
Swell + 3^{rd} Harmonic + 5^{th} Harmonic + Transients (SNR=26%)	95%
AVERAGE ACCURACY	**92.08%**

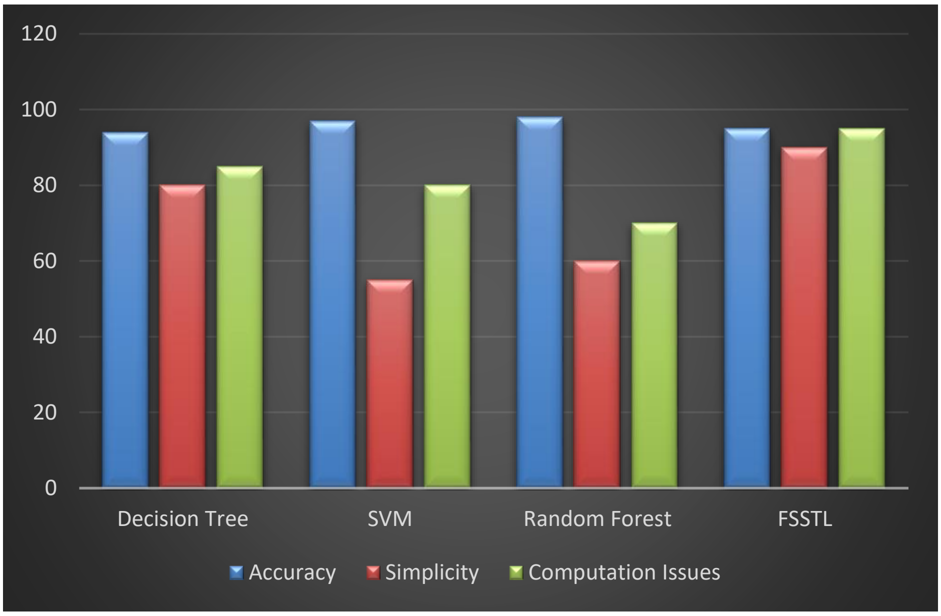
Chapter 5:- Comparison with other latest techniques

In all of the above discussion we have studied about the whole process as to how the fault is detected and then seen its actual interpretation on simulated & real time data. In this part of the document we would like to compare the algorithm with past works which are used for detection of the power system faults & what drawbacks it has posed for us to dive into research in this particular domain.

- Most 3rd world countries are still not employing the use of any computation technique & is involved with human presence at the particular station all the time. The fault is detected by a circuit breaker and relay combination which merely triggers to cut the faulty circuit from the rest of the system & does not tell us anything about the particular type of fault which has occurred.
- With invention of electronics and more so faster computation all the process done by a human mind can now be thought to be done by computers. Many computational techniques have been developed by various scholars & is algos being applied to current system e.g. Decision tree, support vector machine, random forest, etc. The main problem with all this is that the operator must be a highly educated person to understand the intricacies of the program while as we have discussed that most 3rd world countries are having mostly only secondary level of education.
- Some more algorithms have also started to gain momentum a few of them being:-
    1. Artificial Neural Network(ANN)
    2. Fuzzy logic
    3. ANN + Fuzzy logic
    4. Genetic Algorithm
    5. State observer model

Let's understand in brief about the above mentioned models. First, the method which have gained lot of attention due to the higher computational powers of the computer & a way to mimic the brain to solve particular set of problems i.e. ANN. ANN has come a long way from solving basic level problems to higher levels of problems. Second, it is fuzzy logic which tries to capture the events to as close an accuracy as possible as it is having an upper-hand over binary logic systems. Third, genetic algorithm which is working on principle of survival of the fittest given by Darwin which is basically inspired from the evolution of human genes. Our method could be taken as an advanced version of the ANN+ Fuzzy logic with improved performance than a basic neural network system. These all methods are inspired in one or the other way from human body.

- For our program to make a considerable change we need to look at these countries as most population of the world resides in them. Comparison with other latest techniques with supporting numbers is given in following bar diagram. The diagram for itself speaks that maybe our technique is lagging in some parts but when it is a matter of simplicity to understand & computational issues our technique is much better than other techniques. By this paper we have tried to make a change for the betterment of human life as a whole.


120
100
80
60
40
20
0
Decision Tree
SVM
Random Forest
FSSTL
Accuracy
Simplicity
Computation Issues

Chapter 6:- Conclusion: What is the way Forward?

In this paper we have given an approach i.e. Feature Selection based transfer subspace learning to detect the faults more efficiently and quickly. Using the approach we have classified various types of faults based on differences we saw in the values of projection and maximum mean discrepancy matrix. On comparing the result obtained from this approach compared to previous applied methods we found out some advantages over them; the biggest and foremost being the ability of our approach to detect variations in the magnitude of faults with accuracy.

In the age of machine learning & Artificial Intelligence we have started to look towards a future where majority of the processes will be automated. This paper is a step in direction of automation for electrical systems. This paper is a good example of unsupervised learning in power system. In the future we plan to go forward & make this process of detection more AI based which will take away the need of any kind of learning beforehand to be done before applying the algorithm to the actual system.

REFERENCES

[1] S. Baloch and M. S. Muhammad, "An Intelligent Data Mining-Based Fault Detection and Classification Strategy for Microgrid," in IEEE Access, vol. 9, pp. 22470-22479, 2021, doi: 10.1109/ACCESS.2021.3056534.

[2] A. Dadashzade, F. Aminifar and M. Davarpanah, "Unbalanced Source Detection in Power Distribution Networks by Negative Sequence Apparent Powers," in IEEE Transactions on Power Delivery, vol. 36, no. 1, pp. 481-483, Feb. 2021, doi: 10.1109/TPWRD.2020.3029437.

[3] J. Liu, H. Song, H. Sun and H. Zhao, "High-Precision Identification of Power Quality Disturbances Under Strong Noise Environment Based on FastICA and Random Forest," in IEEE Transactions on Industrial Informatics, vol. 17, no. 1, pp. 377-387, Jan. 2021, doi: 10.1109/TII.2020.2966223.

[4] Y. Liu, T. Jin, M. A. Mohamed and Q. Wang, "A Novel Three-Step Classification Approach Based on Time-Dependent Spectral Features for Complex Power Quality Disturbances," in IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1-14, 2021, Art no. 3000814, doi: 10.1109/TIM.2021.3050187.

[5] Y. Xu, Y. Gao, Z. Li and M. Lu, "Detection and classification of power quality disturbances in distribution networks based on VMD and DFA," in CSEE Journal of Power and Energy Systems, vol. 6, no. 1, pp. 122-130, March 2020, doi: 10.17775/CSEEJPES.2018.01340.

[6] R. Gong and T. Ruan, "A New Convolutional Network Structure for Power Quality Disturbance Identification and Classification in Micro-Grids," in IEEE Access, vol. 8, pp. 88801-88814, 2020, doi: 10.1109/ACCESS.2020.2993202.

[7] A. Barona, F. Ferrandis, J. Olarte and J. L. Iribarren, "New power quality solutions especially designed for industrial applications," 2007 9th International Conference on Electrical Power Quality and Utilisation, 2007, pp. 1-7, doi: 10.1109/EPQU.2007.4424076.

[8] Upendar, J., Gupta, C.P., Singh, G.K., Ramakrishna, G., PSO and ANN-based fault classification for protective relaying. Iet Generation Transmission & Distribution 4, (2010) 1197-1212.

[9] Eisa, A.A.A., Ramar, K. Accurate one-end fault location for overhead transmission lines in interconnected power systems. International Journal of Electrical Power & Energy Systems 32, (2010) 383-389