Version History

Issue	Date	Author	Approved	Comment
0.1	01/12/2021	A Forster	N/A	Initial Draft
0.2	06/12/2021	A Forster	N/A	Changes based on feedback from meeting on 02/12/2021.
0.3	10/01/2022	A Forster	S Jupe	Introduction Changes.
0.4	10/01/2022	A Forster	S Jupe	Added glossary and minor cosmetic changes.
0.5	24/01/2022	A Forster	N/A	Changes based on comments from Paul Morris and Feedback from 20/01 meeting with Sid/Samuel/Ben/Darren.
0.6	07/02/2022	A Forster	S Jupe	Changes to 6.2 to handle devices which lose communications.
1.0	07/02/2022	A Forster	S Jupe	Formal Issue.

Distribution List

Name	Organisation
Paul Morris	WPD
Samuel Jupe	Nortech
Sid Hoda	Nortech
Ben Brewin	Nortech
Andrew Forster	Nortech

Glossary

Acronym	Meaning
CNN	Convolutional Neural Network
LV	Low Voltage
PQM	Power Quality Monitor
SLD	Single Line Diagram
SN	Smart Navigator
WPD	Western Power Distribution

1 Introduction

Pre-Fix is a WPD NIA project which will develop and demonstrate a Common Disturbance Information Platform (C-DIP), allowing equipment from different vendors to be utilised for pre-fault detection and more accurate fault location. The outputs from this project are expected to deliver game-changing performance benefits for WPD (in terms of reduced customer interruptions and customer minutes lost) in RIIO-ED2.

This document forms the basis for the development of a software module (accessible through iHost) which provides a tool to classify fault (or pre-fault) waveforms found in time series data collected from deployed devices.

The classification module will be used to provide the user with the type of (pre-)fault which most closely resembles the input waveform, based on the fault signature waveforms defined in the IEEE technical report PES-TR73.

This system will facilitate faster identification of the cause of (pre-)faults in the network, as well as any equipment issues or failures which may have contributed to them. This can help to inform decisions about required repairs, and in the case of pre-faults, pre-emptive maintenance of equipment.

2 Workflow

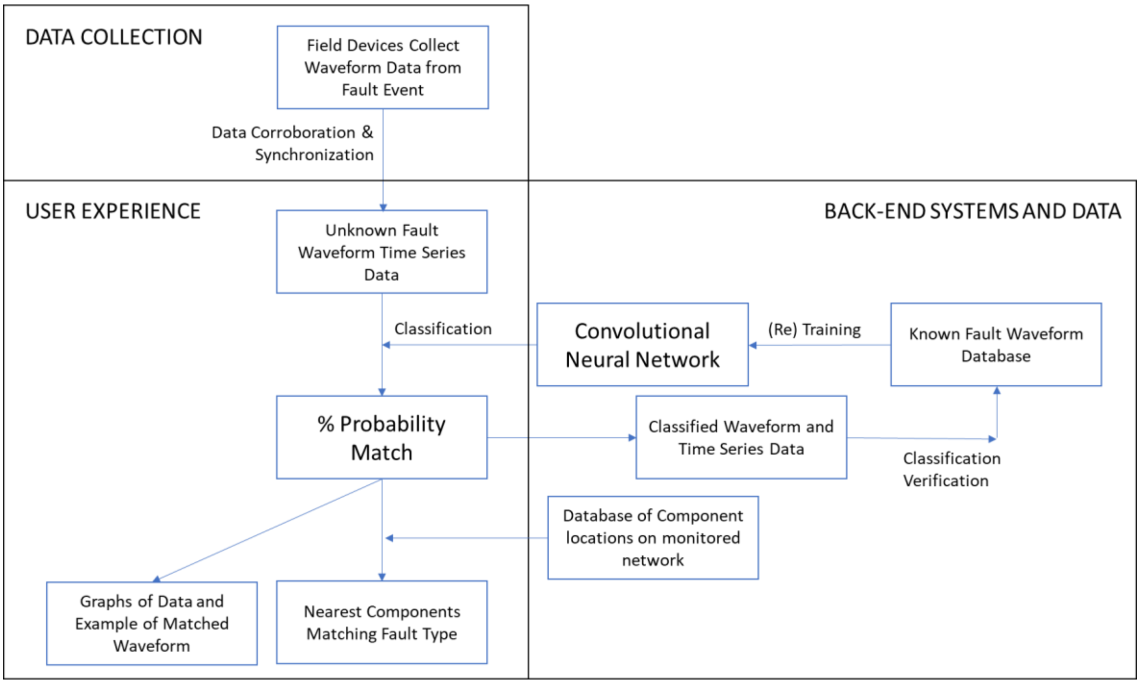

Figure 1 – Workflow diagram

3 Requirements

3.1 Convolutional Neural Network (CNN)

This specification was set up on the basis of using a CNN pre-trained on a large quantity of verified time-series data that show a wide range of common faults. The prototype CNN was created in python using the TensorFlow Keras module. Pre-training the model allows for a faster user experience when it comes to classifying their desired waveform.

Outputs of the CNN should give the user the closest match between their input data and a known fault type, along with a % certainty score. The user should also have the option to view their input data and the match waveform as graphs, to allow for manual verification before continuing. One of the potential classifications will need to be "no fault present", to allow for the system to ignore non-faulting/non-defect-based events.

The CNN will need to be able to be retrained in order to add further fault/pre-fault categories and training data in future. The CNN should be retrained on a monthly basis during the course of the Pre-Fix project.

3.2 Known Fault Waveform Database

The training data for the CNN needs to have a large quantity of time-series data for each kind of fault to allow for accurate model training. This database could be periodically updated with extra fault data that had previously been categorised as a certain type of fault (either manually or by the CNN).

3.3 Dynamic Time Warping and Data Normalisation

The time series data that is input into the model will not necessarily adhere to the length that the CNN is expecting. Data will need to be standardised and normalised before being categorised for the best chance of correct categorisation. To standardise the length of the data, a system such as dynamic time warping could be employed. Data can be normalised by bounding it between -1 and 1.

3.4 Closest Matching Components (Nice-to-have)

One of the future steps to improve the system for the user would be to allow the user to input a location for the fault (in the same format as WPD locations are currently stored, or in a format suitable for iHost SLD mapping), and have the system pick out the closest components that match the fault type. This would require access to a database of WPD network components with locations, such as a mirror of CROWN component data.

4 Display Requirements

4.1 Output Mock-up

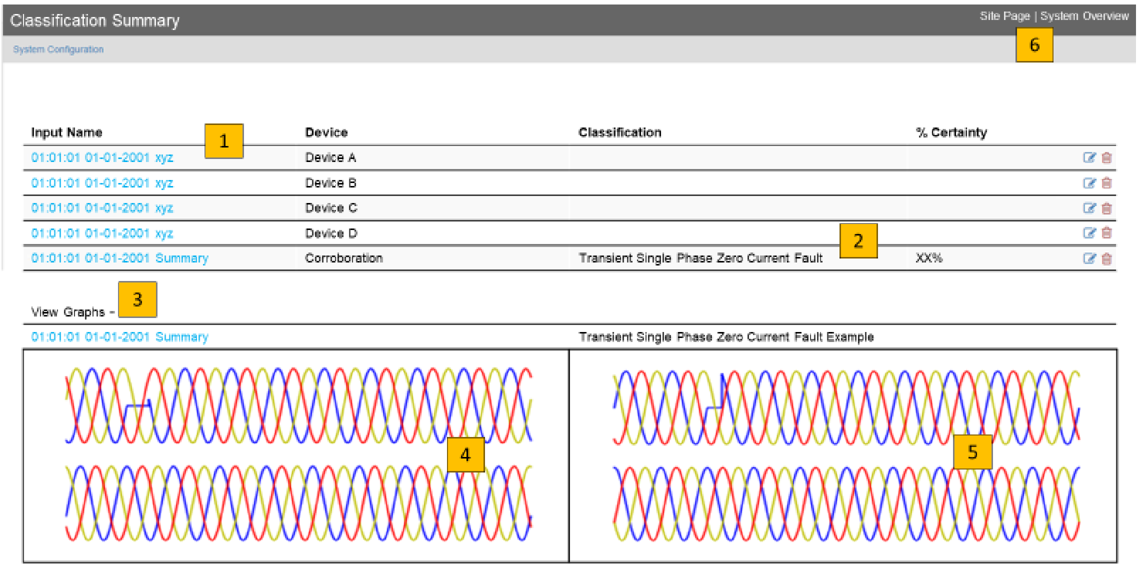

Figure 2 – Classification system output diagram

1) Table with the inputs and devices that captured the fault.
2) Classification of corroborated fault waveform (the data with the highest priority out of the collected data based on device) and % certainty.
3) Option to view graphs (expanded in diagram)
4) Graph of the corroborated waveform data for current and voltage.
5) Example graph of the fault classification.
6) Links back to the feeder page in Pre-Fix dashboard or iHost system overview.

5 User Data Input Requirements

5.1 Data Format

Initial Prototypes of the classification system have been trained to accept 10 cycles of time-series data, with 3600 timestamps per input dataset.

For the data collected for use in the Pre-Fix project, the Nyquist theorem should be applied, meaning that the sample rate of the waveform data should be at least double that of the waveform frequency component in order to ensure that the collected waveform data is a faithful reconstruction of the witnessed waveforms.

The number of cycles reported and timestamps per cycle will need to be consistent, meaning that data collected from field devices may need to be transformed (e.g., using Dynamic Time Warping) in order to fit this requirement and be usable by the classification system. The number of cycles reported will need to be low enough that the devices collecting waveform data consistently provide enough data to be able to use, but high enough that the entire fault waveform is consistently captured.

The system will need to be able to flag unusable data where necessary, such as in the case of file corruption or the waveform data cutting out early. The system will also need to be able to flag data that may be partially usable, such as when a single voltage phase data is incomplete and attempt to classify this data.

6 System Integration Requirements

6.1 Distance-to-Fault Integration Requirements

In future, the system will need to be able to corroborate the results from the distance-to-fault system with both the classifications it makes and a dataset of components. This will allow for an estimate of the most likely components to have caused the fault observed within the search zone provided by the distance-to-fault system.

6.2 Device Integration Requirements

The system will need to be able to corroborate fault waveform data from multiple nearby devices in the field in order to determine if they are seeing the same fault. To this end, the system will need to be able to account for clock drift and poor signal by designing for latency and having a single point of truth for time to compare against (such as a GPS time value). LV monitors will need to be configured to call in upon an alarm being triggered to ensure that their data is received by the system within the latency period. The ability to handle situations which interrupt data corroboration such as loss of comms or inaccurate timestamps is described in the C-DIP technical specification (see Appendix 2).

There will also need to be checks for the proximity of devices when deciding whether two devices recording a fault at similar times are seeing the same fault, or whether two faults have occurred at similar times in different sections of the network. This means that installed devices will need to have latitude/longitude information assigned to them.

The system will also require the ability to corroborate input data with nearby circuit breaker tripping and auto-recloser operation.

Once the waveform data from field devices has been received, there will need to be a hierarchy of data priority based on device:

- PQMs installed at primary substations will take priority. If there is fault waveform data available from a PQM, then the classification system will use that data as the input. If multiple of these PQMs have seen the fault, it will be classed as an upstream problem.
- If PQM data for the fault is not available, NX-44 waveform data will take priority, and will be used in the classification system.
- If neither PQM nor NX-44 waveform data is available, waveform data from SNs will be used in the classification system.
- If no waveform data is available, the system will return an error to this effect.

This hierarchy will need to consider data rejection when choosing the data that represents the fault best (e.g., if the data from a PQM is corrupted or otherwise unusable, the system will move on to the next highest in the hierarchy, an NX-44).

The system will also need to be able to display if data has been flagged as partially usable, as this may suggest to the user that investigating the data lower in the hierarchy could be valuable for validating the fault classification given.

7 System Performance Requirements

7.1 CNN Output Requirements

The CNN must be able to return a classification for the input data, as well as a percentage confidence value for this classification. The system should be able to classify waveforms to a high degree of accuracy and with little loss.

During testing, the system needs to be able to accurately identify 90% of the test data. This testing will be fully described in the Methods for Performance Testing documentation provided as an appendix to this document, but the basic procedures will include:

- Providing a known fault waveform to see if the system correctly identifies it.
- Providing a waveform with multiple faults to see if the system correctly identifies multiple faults.
- Providing a waveform with an unknown fault to see if the system classifies it as a different fault type.
- Providing a waveform with no fault to see if the system correctly classifies it as displaying no fault waveform features.

7.2 Output Dashboard Requirements

The output dashboard must display to the user the CNN outputs, alongside the option to show both the input data and an example of a waveform that it has been classified as. Future improvements could allow the user to find the closest components in the network that fit the location and classification of the fault.

7.3 Computational Performance Requirements

The user facing part of the tool must be able to provide results within 10 minutes of the fault occurring. This should ensure that the fault classification and potential component information is available to patrol teams once they begin their patrols.

To this end, the (re)training of the CNN should be done periodically behind the scenes, providing a saved model that the dashboard can load when a user wished to classify a waveform.

8 Error Handling Requirements

8.1 CNN Error Handling Requirements

The CNN system should be able to provide error codes when an error occurs, such as when the input data is unusable to the system.

8.2 Device Communications Dropout/Restoration Handling

When integrated with devices in the field, it is important that the system can continue to corroborate and synchronise data even when one or more devices has lost communications. When the device(s) restore communications, there should be a check to see if the waveform data for faults occurring during the dropout is available, and if so, prompt the user that they may need to rerun the classification of that fault event.

9 Limitations of Waveform Classification

9.1 Temporary and Permanent Faults

If there are two faults, both of the same variety (e.g., zero current fault), then there can be some overlap between faults that are classified as temporary or permanent. This can lead to some temporary faults being classified as permanent and vice-versa.

9.2 Phase to Phase Differences

Some fault types that are similar but not identical (e.g., zero current on one phase and zero current on all phases) can have enough overlap that one is classified as the other. In these scenarios, it would require some user discretion on the nature of the data shown.

9.3 Extreme Noise

The classification system currently has some capacity to filter out noise in input waveforms. However, if noise is made too extreme, it can lead to the system classifying the input waveform differently than it would do otherwise.

In initial trials, the classification system should be able to accurately classify fault waveforms with a signal-to-noise ratio of 20dB or higher. This would allow for faults which are characterised by extreme noise (such as arcing faults) to be differentiated from a noisy fault of a different variety. This boundary may be able to be changed based on trial data collection and performance.

9.4 Lack of Available Data

The classification system will require a robust database of known fault waveforms in order to correctly classify incoming data. If the database used to train the system is too limited, there are likely to be accuracy issues.

This can be partially mitigated in the future with the retraining of the system using new (pre-)fault waveforms collected. There is currently a decent dataset of post-fault data to use during training, and pre-fault data is currently being collected.

Appendices

Appendix 1	Methods for Performance Testing
Appendix 2	Common Disturbance Information Platform (C-DIP) Technical Specification (D_003960)