Metric Learning for Electrical Submersible Pump Fault Diagnosis

Lucas Henrique Sousa Mello
Department of Computer Science
Federal University of Espírito Santo
Vitória, ES, Brazil
lucashsmello@gmail.com

Marcos Pellegrini Ribeiro
CENPES/PDP/TE
Petrobras
Rio de Janeiro, RJ, Brazil
mpellegrini@petrobras.com.br

Thiago Oliveira Santos
Department of Computer Science
Federal University of Espírito Santo
Vitória, ES, Brazil
todsantos@inf.ufes.br

Flávio Miguel Varejão
Department of Computer Science
Federal University of Espírito Santo
Vitória, ES, Brazil
fvarejao@inf.ufes.br

Alexandre Loureiros Rodrigues
Department of Statistics
Federal University of Espírito Santo
Vitória, ES, Brazil
alexandre.rodrigues@ufes.br

***Abstract*—Machine learning classification algorithms are highly dependent of a dataset composed of high-level features. In this paper, a deep learning approach is combined with traditional machine learning classifiers in order to circumvent the need of a specialist for extracting relevant features from one dimensional frequency-domain vibration signals. Our approach relies on a convolutional architecture trained with a triplet loss function for extracting relevant features directly from the raw data. A previously hand-crafted feature set, created by a specialist over the course of many years of research, is compared with the newly extracted feature set. Six conventional classifiers models (K-Nearest Neighbors, Support Vector Machine, Decision Tree, Random Forest, Quadratic Discriminant Analysis and Naive Bayes) are trained in both features set separately and compared in terms of macro F-measure. Results shows statistical evidence towards to the acceptance that the extracted feature set is as good as or better than the hand-crafted feature set, for classification purposes.**

***Index Terms*—Fault diagnosis, electrical submersible pump, classification, metric learning, triplet network**

I. Introduction

Submersible Centrifugal Pumping [1] is an artificial lifting method widely used in oil and gas production and is characterized by the use of a multistage centrifugal pump driven by an electric motor. An Electrical Submersible Pump (ESP) belongs to a class of equipments used in the extraction and exploration of oil and gas subject to severe working conditions. High pressures, high temperatures, high flow rates and the need for continuous operation are critical conditions for any machine. In addition to that, these equipment are deployed under deep water, making any maintenance unfeasible. Failures that require downtime, maintenance, and eventually replacement of these equipment usually lead to significant financial losses due to the very high cost of performing maintenance and especially because of the interruption of production over a long period of time.

One way to reduce the risk of failure in ESP is conducting an analysis in a testing laboratory prior to their deployment. To perform this analysis, a specialist demands multiple accelerometers to be installed along different points of the EPS system so that vibration signal data are collected for long periods of operation (e.g., 72 hours). Subsequently, the specialist uses computational tools to visually analyze the vibration spectrum of these signals. Based on the data collected, the equipment may be considered in proper conditions of operation or not.

While this procedure is quite effective in reducing the risk of failure after deployment of the system, it has a drawback: specialists who are able to effectively perform this analysis are very rare. Typically, the knowledge required to accomplish this task is gained over many years of experience and is not easily taught. Thus, the industry becomes very dependent on the experts capable of performing this test. This is inconvenient because unavailability of the specialist (due to vacation, sickness or retirement) may delay schedules or, even worse, cause less skilled technicians to accept equipment unfeasible for operation. Therefore, it is desirable that such specialized knowledge is incorporated into the company corporate knowledge.

Some previous works have already addressed the fault diagnosis problem of ESP by adopting traditional machine learning process [2]–[4]. In these works, the process for diagnosing systems is composed of the following stages:

1) acquiring raw vibration signals from accelerometers sensors and converting to the frequency domain;
2) using a specialist to label/diagnosis acquired data;
3) extracting custom made features from the spectrum;
4) training a classifier on the custom made features and

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001 and National Council for Scientific and Technological Development – CNPq. This work is partially supported by CNPq (National Council for Scientific and Technological Development) and by CENPES-Petrobras.

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

5) using the trained classifier to diagnose faults in newly observations.

As in many machine learning classification problems, the third stage (defining features) is highly dependent of a specialist capable of instructing which features are important and how to extract them, so that it can be processed by machine learning algorithms. In this present work, our main objective is to address this problem by developing an artificial intelligence algorithm capable of learning directly on the raw signal in the frequency domain. To achieve this objective, our methodology combines a convolutional neural network (CNN) trained with a triplet loss learning [5] and standard machine learning algorithm such as Random Forest [6]. This type of network is capable of directly extracting relevant features from the frequency domain data. In addition to a competitive classification performance, this methodology aims at producing a feature space where the euclidean distance approximates the "semantic distance", which is commonly desired when using this type of network [7]. This is useful for detecting the introduction of new fault types or unknown anomalies. One way to detect new fault types is to just compare similarities (euclidean distance) among samples in the new feature space. Moreover, the feature space becomes more visually clear to be interpreted by non-expert analysts.

The remainder of this paper is organized as follows: Section II provides a brief overview of how data is acquired and how previous related works uses the data to propose solutions. In Section III, the fundamentals and the modifications of triplet network learning are presented. Subsequent to the elaboration of our experimental setup and methodology in Section IV, the experimental results and discussions are presented in Section V. Section VI concludes the paper and presents possible future works.

II. Signal Processing for Fault diagnosis systems

An ESP system is composed of electrical motors, pumps and protectors, as showed in Fig. 1. In order to test an ESP, accelerometers sensors are attached in strategic positions of its components, as showed in Fig. 1. These sensors collect vibration signals in the time domain.

In order to be analysed by a specialist, the data is transformed from the time domain to the frequency domain by the Fourier transformation. Although a specialist can usually diagnosis faults by just looking the spectrum of a signal in the frequency domain, a standard machine learning algorithm is not capable of finding the right features when dealing with large input data (about 200000 points for a single signal in the frequency domain). For these reasons, a feature extraction process is usually made in order to elaborate a condensed set of relevant features that standard machine learning techniques can deal with. In this paper, a condensed set of relevant features, hand-crafted and given by a specialist in the field, is used in order to compare with our methodology. This hand-crafted set of features comprises eight real-valued features that jointly create information about the peaks and shape of the spectrum in the range of significant frequencies and are used for identifying the status of the ESP. This set of features is identical to the one presented in [2] and are described as follows. Let F be defined as the rotation frequency in which the BCS is operated. Each feature is defined as:

- **median(3,5)** Median of the amplitudes in the interval (3Hz, 5Hz);
- **median(F-1,F+1)** Median of the amplitudes in the interval (F-1Hz, F+1Hz);
- **a:** Coefficient a of the exponential regression of type $e^{(a \cdot X + b)}$ where X is the array of amplitudes in the interval (5Hz, 19Hz);
- **b:** Coefficient b of the exponential regression of type $e^{(a \cdot X + b)}$ in the interval (5Hz, 19Hz);
- **rotation1x:** Frequency of the highest amplitude in the interval (F-3Hz, F-0.2Hz);
- **peak1x:** Amplitude in **rotation1x**;
- **peak2x:** Amplitude in 2·**rotation1x**;
- **rms(F-1,F+1)** Root mean square of the amplitudes in the interval (F-1, F+1).

Fig. 1. A BCS system with six components and attached 36 sensors. Each square or circle represents a sensor.

III. Feature extraction via triplet network

The aim of a triplet network is to learn an embedded representation of any input object such that objects of the same class are close/similar while objects of distinct classes are distant in the multidimensional space defined by the representation [5]. Despite the name, the triplet network can have an architecture similar to any traditional neural network.

The difference lies on the objective/loss function in the training stage.

Let f be the objective function being learned by the network that maps an input object (a signal) to an embedded representation in the real space, that is,

$$f : \mathbb{R}^m \rightarrow \mathbb{R}^k,$$

where m is the number of input neurons (20000 in this work) and k is the embedded space size (8 in this work). Given three input objects x, x^+, x^-, such that x and x^+ belong to the same class and x^- belongs to any other class, the loss function L being optimized by the triplet network is given by:

$$L(x, x^+, x^-) = \max(0, ||f(x) - f(x^+)|| - ||f(x) - f(x^-)|| + \alpha),$$

where α is the minimum desired margin between objects of distinct classes. In the training stage, three input objects (the triplet) are feed into the neural network separately. Then the loss function using the three objects is computed as well as its gradient. A balanced batch, i.e. a random batch of equally distributed samples of each class, is formed before the triplets are generated and fed into the network.

In general, there are $O(n^3)$ triplets to been chosen for training the network, therefore a method for selecting triplets is crucial. Inspired by [8], in this paper we present a method for increasingly selecting difficult triplets. The training process is described as follows. Only triplets considered not "easy", i.e, those with a positive loss using an initial margin $\alpha = 1.0$, are considered for training the network. For this purpose, a initial learning rate of 10^{-3} is used. The training is done in 560 epochs, where after a pre-defined number of epochs the learning rate is reduced by 10% and α is reduced by 25%. In the end, the learning rate will be approximately $2 \cdot 10^{-4}$ and α will be 0.013. Reducing margin with time makes triplets with loss close to the margin be discarded. Subjectively speaking, the main idea of reducing margin with time lies on the premise that some triplets have objects that are too hard to be separated by a "large" margin. Preliminary experiments confirmed that this strategy had better performance than increasingly focusing on hard or semi-hard triplets.

The output of the triplet network can be seen as a new feature vector, extracted from the input, with higher value of complexity. In this paper, this point of view is adopted and the new feature space is used for training traditional machine learning algorithms, such as the K Nearest Neighbors (KNN).

The architecture and parameters of our CNN network are shown in Table I. Note that an output with the same number of hand-crafted features is used so that both features set are more easily comparable. A stride of one and zero padding is used in the convolutional layers. The Leaky ReLU function with a negative slope of 0.05 was used as an activation function.

For a fair comparison among deep learning approaches, a convolutional network (ConvNet), with similar architecture as the proposed triplet network, was trained using standard training procedure. The only difference in the architecture lies on a additional output layer with one neuron for each class. The training process uses the cross entropy of the prediction and desired target classes as a loss function. The training lasts for a maximum of 560 epochs and uses back-propagation with a initial learning rate of 10^{-3}. After a pre-defined number of epochs, the learning rate is reduced by 10%. Additionally after training the ConvNet, similarly to the triplet network, the internal network (all layers but the output) is used for extracting new features where machine learning algorithms, such as KNN, can be trained with. For the sake of simplicity, the feature space extracted by the triplet network will be called from now on of triplet-space whereas the feature space extracted by the traditional ConvNet will be called deep-space.

TABLE I
THE ARCHITECTURE USED FOR THE TRIPLET NETWORK

Layer	Type	Feature Maps	Filter Size	Dropout
0	Signal	6100	-	-
1	Conv	6096 x 16	5	0.2
2	Max Pool	1524 x 16	4	-
3	Conv	1520 x 32	5	0.2
4	Max Pool	380 x 32	4	-
5	Conv	376 x 64	5	0.2
6	Max Pool	94 x 64	4	-
7	FC	192	-	0
8	Output	8	-	-

The classification performance of ConvNet trained in the frequency domain, as well as six machine learning algorithms trained in the deep-space, are compared with the performance on algorithms trained in the triplet-space in Section V.

IV. Experimental Methodology

The main purpose of this work is to collect statistical evidence towards the acceptance/rejection of our hypothesis about the proposed methods. This should be done in an objective and quantitative way, therefore a statistical hypothesis test is performed. The null hypothesis states that, given a metric for evaluating classification performance, the average performance of a classifier algorithm using the hand-crafted feature space is equal to the average performance of a second classifier algorithm using an automatically extracted feature space given by a neural network model. In this paper, two cases of this hypothesis are analysed. In the first one, a single classifier model is trained and compared in two different features spaces while in the second one, the best classifier models trained for each feature space are compared with each other. The second is when both C_1 and C_2 are the best algorithms found for their respective feature space. Both cases are important. The latter seems to be the most important in practice since it is obviously desirable to use the best method to solve a problem, assuming it can be determined. However, the former is relevant for our better understanding of the methodology developed in this paper.

In the next subsections, we define the dataset to be used, the sampling method, the set of classifiers algorithms used and how the statistics are estimated.

A. Dataset

A dataset comprising 5617 observations of vibration signals acquired by accelerometers strategically attached on compo-

nents of a ESP system was created. The signals were collected separately when the ESP was under various possible operations conditions. Each signal is an one-dimensional real vector acquired from a single sensor operating under a single specific operation condition. Typically, a single vibration signal in the time domain is composed of 400000 data points collected at sampling rate of 4096 data points per second. After Fourier transform is applied, the result is mirrored at 0Hz, therefore only half of the resulting spectrum is considered to avoid redundancy. Each vibration signal is considered independently of each other. All of the 5617 vibration signals are classified as having strong evidences of a fault in the ESP system or not by an human expert. This paper consider only three types of motor pump faults: shaft misalignment, pump blade unbalance, and mechanical rubbing. Additionally, a faulty sensor generating abnormal vibration behaviour is considered as a faulty pattern, although the abnormal behaviour is not necessarily related to the equipment. An expert performed a visual inspection of the vibration spectrum of each sensor attributing one of the five considered categories: normal, faulty sensor, unbalance, misalignment or rubbing. Table II shows the class distribution of collected dataset.

TABLE II
CLASS DISTRIBUTION OF 5617 COLLECTED VIBRATION SIGNALS

Class name	A priori distribution [%]
Normal	80
Rubbing	4.86
Faulty sensor	5.25
Misalignment	0.93
Unbalance	8.96

B. Classification

As showed in Table II, there are 5 possible classes to be predicted for a single vibration signal. Although some of them can theoretically occur simultaneously, there was no observation where a vibration signal had more than one fault. Therefore, this paper consider the problem of detecting faults in ESP systems based on the analysis of vibration signals as a multi-class problem of 5 classes.

To solve this multi-class classification problem, six classification algorithms commonly used in the literature were chosen: K-Nearest Neighbors (KNN) [9], Support Vector Machine (SVM) [10], Decision Trees (DT) [11], Random Forest (RF) [6], Quadratic Discriminant Analysis (QDA) [12] and Naive Bayes (NB) [12]. In order to improve classification performance, the training process of these algorithms have a preliminary stage to tune their particular hyper-parameters. The tuning is done by grid-search, that is, testing each possible combination of hyper-parameter values. The hyper-parameters values used to tune this classifiers model are shown in Table III.

The tunning is also done for the hyper-parameters of the ConvNet and the TripletNet. For both the ConvNet and the TripletNet, the tunning is done individually for each classifier, therefore each classifier model uses the triplet-space that it suits best. Table IV shows the hyper-parameters used for ConvNet and TripletNet. A total of 27 configurations of hyper-parameters were evaluated for each network.

TABLE III
RANGE OF VALUES TESTED FOR TUNING THE HYPER-PARAMETERS OF CHOSEN CLASSIFIERS MODELS.

Method	Hyper-parameter	Values
NB	None	-
KNN	Number of neighbors	{1, 3, 5, 7, 9, 11, 13, 15}
SVM	γ	{2, 8}
	C	$\{2^5, 2^7, 2^{13}, 2^{15}\}$
DT	maximum number of leaf nodes	{1, 2, 3 ,4, 5}
	maximum tree height	{3, 6, 9, 12, 15}
RF	number of features	{1, 2, 3 ,4, 5}
	number of trees	{100,1000}
QDA	Covariance regularization	{0, 10^{-5}, 10^{-6}}

TABLE IV
RANGE OF VALUES TESTED FOR TUNING THE HYPER-PARAMETERS OF THE CONVOLUTIONAL NETWORK AND TRIPLET NETWORK. THE PARAMETER "STEP SIZE" INDICATES THE NUMBER OF EPOCHS IN WHICH THE LEARNING RATE WILL BE PERIODICALLY REDUCED.

Method	Hyper-parameter	Values
Convolutional Network	Batch size	{128, 256, 512}
	Learning rate	{10^{-4}, 10^{-3}, 10^{-2}}
	Step size	{20, 30, 40}
Triplet Network	Class samples per batch	{4, 8, 16}
	Learning rate	{10^{-4}, 10^{-3}, 10^{-2}}
	Step size	{20, 30, 40}

C. The Evaluation Framework

The performance criterion used in this work is the macro-averaged F-measure [13] as defined in equation 1. This motivation comes from the simultaneous consideration, in one single value, of precision and recall, derived from the confusion matrix. Macro-averaging is chosen since it treats all classes equally, which is desired in this problem due to its imbalanced dataset, while micro-averaging favors classes with more examples. Consider a multi-class classification problem with c classes. For each class j, the individual true positives, false positives and false negatives are defined as tp_j, fp_j and fn_j, respectively. The macro-averaged precision and macro-averaged recall are defined as

$$\text{Precision}_M = \frac{1}{c}\sum_{j=1}^{c}\frac{\text{tp}_j}{\text{tp}_j + \text{fp}_j}$$

and

$$\text{Recall}_M = \frac{1}{c}\sum_{j=1}^{c}\frac{\text{tp}_j}{\text{tp}_j + \text{fn}_j}.$$

The macro-averaged F-measure is defined as the harmonic mean of precision and recall:

$$F_M = \frac{2 \cdot \text{Precision}_M \cdot \text{Recall}_M}{\text{Precision}_M + \text{Recall}_M}. \quad (1)$$

In order to ensure a fair comparison among classifiers, a 10-fold stratified cross-validation is used. For tuning hyper-parameters, each configuration of parameters is evaluated by training a new classifier with this configuration in eight of the training folds and then evaluating its performance in terms of f-measure in the remaining fold (the 9th fold present in the training dataset). The best model trained is then tested on the unseen data of the 10th fold and only this result is used for computing the averaged F-measure.

On all features of all spaces, a data standardization is applied, i.e, subtracting the feature mean and then dividing by its standard deviation. It is important to note that the data standardization parameters, mean and standard deviation, are estimated only using the training dataset. Data standardization is not applied on the triplet-space as preliminary experiments showed a significant decay in the classifiers performance.

D. Statistical Analysis

As stated before, the purpose of this paper is to compare several pairs of classifiers and conclude whether there is statistical difference in each one. One of the assumptions of the well-known paired t-test is the independence among samples. This is not the case for the experimental framework used in this paper, since there is an intersection of training datasets among folds. Therefore, the corrected t-test proposed for this situation by [14] was used instead.

V. Experimental Results

The experiments were conducted in the python programming language using PyTorch [15] as an artificial neural network framework and Scikit learn [16] as a machine learning framework and as an experimental testing platform. Performances of traditional machine learning classifiers on both features spaces are presented individually and compared.

The aim of the experiments is to determine if the new features extracted by the triplet network is suitable for classification purposes. A new feature space is defined suitable if it is as "good" as the hand-crafted feature space for classification algorithms, i.e, if machine learning classifiers achieve equal or better performance on the new feature space when compared to the hand-crafted space.

The boxplot in terms of F-measure on the 10-fold cross validation is presented in Fig. 2. The highest median of each space are given by Random Forest on hand-crafted feature space, with a value of 0.829, KNN on the deep-space, with a value of 0.814 and Quadratic Discriminant Analysis on triplet-space, with a value of 0.858. Important to observe the large difference between performance of classifiers SVM, NB and QDA in the feature spaces. These models performed relatively poorly when trained in the hand-crafted features. Meanwhile, there was no model trained in the triplet-space that performed poorly in this same way. The ConvNet achieved a median of 0.78 and an average of 0.71. The major problem regarding to ConvNet is its variation. For instance, a minimum f-measure of 0.45 and a maximum of 0.87 was given by the ConvNet. Clearly, the combination of conventional machine learning algorithms with neural networks outperformed the ConvNet, with the exception of NB trained in the deep-space. To improve ConvNet performance in this dataset, one should investigate the reason for such behavior.

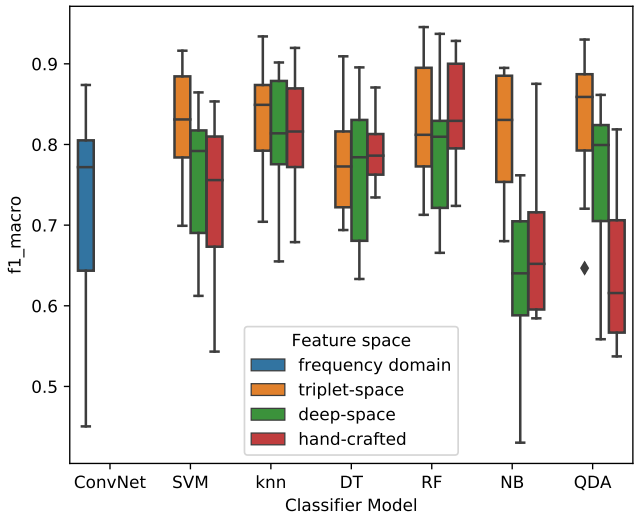

Fig. 2. Boxplot of resulting macro F-measures. With the exception of the traditional convolutional network (ConvNet) which was trained directly on the frequency domain space, each triple indicates a classifier model being trained in the triplet-space (left), the deep-space (middle) and the hand-crafted feature space (right).

The average F-measure estimated from the 10-fold cross validation is presented in Table V. One may see that methods QDA and KNN performed well in both extracted spaces while performed relatively poorly on hand-crafted space. It seems that both triplet-space and deep-space are optimized for this classifiers models, which is confirmed by observing Fig. 5 where examples of a class tends to lie close to the centroid of that class. One may also note the best model trained in the triplet-space (KNN) has a very similar performance to the best model trained in the hand-crafted features (RF), with a slightly higher median. Moreover, the classifier models NB, QDA, SVM, KNN were better when trained in the triplet-space.

Hypothesis tests were conducted with respect to the triplet-space, deep-space and hand-crafted space. The p-values are presented in two tables: Table VI, which compares average performance of a classifier model trained in different features spaces, and Table VII, which compares average performance of the best models of each space. A significance level of 5% is adopted. It should be noticed that no model trained on any extracted feature space has been shown to have a significant lower average F-measure than the same model but trained on the hand-crafted features. Moreover, note the existence of a significant statistical difference on NB, QDA and SVM. The triplet-space is statistically better than the deep-space when using QDA, NB, or SVM as classifier model, as shown in

TABLE V
AVERAGE F-MEASURE FOR ALL CONSIDERED CLASSIFIER MODELS.

Classifier Model	Feature space	F-measure
ConvNet	frequency domain	0.71478
DT	deep-space	0.76760
	hand-crafted	0.79285
	triplet-space	0.77906
NB	deep-space	0.62356
	hand-crafted	0.67328
	triplet-space	0.81089
QDA	deep-space	0.76185
	hand-crafted	0.63896
	triplet-space	0.82854
RF	deep-space	0.79280
	hand-crafted	0.83599
	triplet-space	0.82764
SVM	deep-space	0.76218
	hand-crafted	0.73495
	triplet-space	0.82529
KNN	deep-space	0.80759
	hand-crafted	0.81650
	triplet-space	0.82944

TABLE VI
P-VALUES OF 18 CONDUCTED HYPOTHESIS TESTS. THE FIRST COLUMN SHOWS FOR EACH CLASSIFIER MODEL THE P-VALUE WHEN THE AVERAGE PERFORMANCE ON THE HAND-CRAFTED SPACE AND THE TRIPLET-SPACE ARE COMPARED. IN BOLD ARE P-VALUES LESS THAN THE ADOPTED SIGNIFICANCE LEVEL (5%).

	Hand vs Triplet	Hand vs Deep	Deep vs Triplet
QDA	**0.0024**	**0.0267**	**0.0156**
NB	**0.0200**	0.2347	**0.0266**
SVM	**0.0493**	0.1954	**0.0452**
KNN	0.5705	0.6940	0.3316
DT	0.5785	0.4244	0.6965
RF	0.7403	0.1242	0.1635

Table VI. This is mainly because the triplet network indirectly optimizes the feature space for these classifiers, as discussed before. High p-values presented in Table VII means that the best method for each feature space are competitive with each other.

TABLE VII
P-VALUES OF THREE CONDUCTED HYPOTHESIS TESTS FOR THE BEST MODEL OF EACH SPACE. THE STAR (*) INDICATES MODELS TRAINED IN THE TRIPLET FEATURE SPACE WHILE THE PLUS SIGN (+) INDICATES MODELS TRAINED IN THE DEEP-SPACE.

Classifier models	p-value
RF vs KNN^{+}	0.2853
KNN^{+} vs KNN*	0.3316
RF vs KNN*	0.7991

For better understanding the concept and results of the new extracted features spaces, a triplet network and a ConvNet were trained on 80% of random samples of the dataset and their feature spaces are visually presented in Fig. 5 and Fig. 4. The hand-crafted feature space is also presented in Fig. 3. In all figures, only the remaining 20% samples (test samples) were considered. The figures comprise a scatter plot for each pair of newly extracted features. The estimated univariate distributions, one for each class, are plotted in the matrix diagonal. Only the three most relevant feature out of eight are presented in each figure. The complete figures with all features displayed are available in *https://github.com/Lucashsmello/TripletNet-on-ESP/wiki/Supplementary-material*.

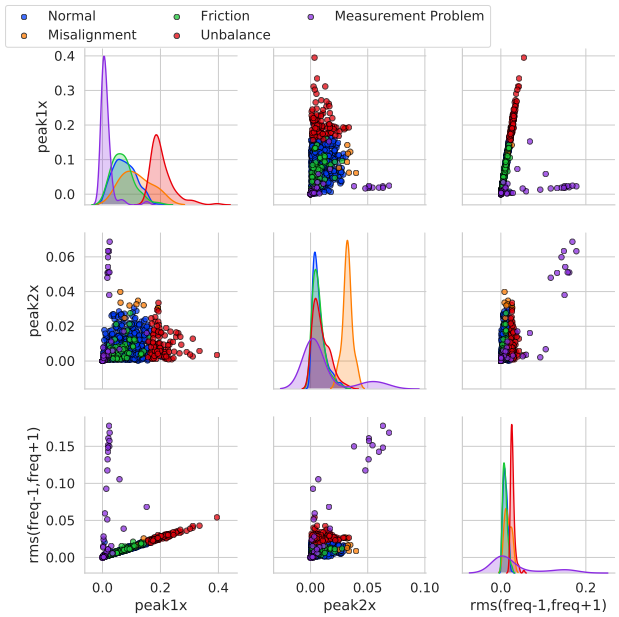

Fig. 3. Scatter plot of the testing dataset for each pair of hand-crafted features. Each color represents a class. Estimated univariate distributions, one for each class, are plotted in the diagonal.

The amplitudes in the harmonics (features *peak1x* and *peak2x*) are good features to discriminate a reasonable amount of misalignment, unbalance and faulty sensor, as can be observed from Fig. 3. High values of *peak1x* usually indicates an unbalance problem, while very low values indicates a faulty sensor. A High value in feature *rms(freq-1,freq+1)* also indicates a faulty sensor, however less than one third of examples with faulty sensors can be diagnosed in this way. With the exception of these features, no other feature individually gives a reasonable amount of discriminative power, unless combined with more than two features.

In the triplet-space, some features separate a single specific class, for instance, feature F6 separates most of samples with faulty sensor class, feature F5 separates samples of unbalance and feature F4 separates samples of misalignment. In the deep-space, only the faulty sensor samples are easily discriminated by visually looking the pair of features.

The most notable difference between deep-space and triplet-space on Fig. 5 and Fig. 4 is the variance in which samples with faulty sensor class are distributed in the space. The variance of samples with faulty sensor in the deep-space is much higher than that of samples of other classes and seems to have no boundary for this class. In the other-hand, the triplet-space seems to have a notable boundary for each one of the classes and samples of the same class are much closer. It seems

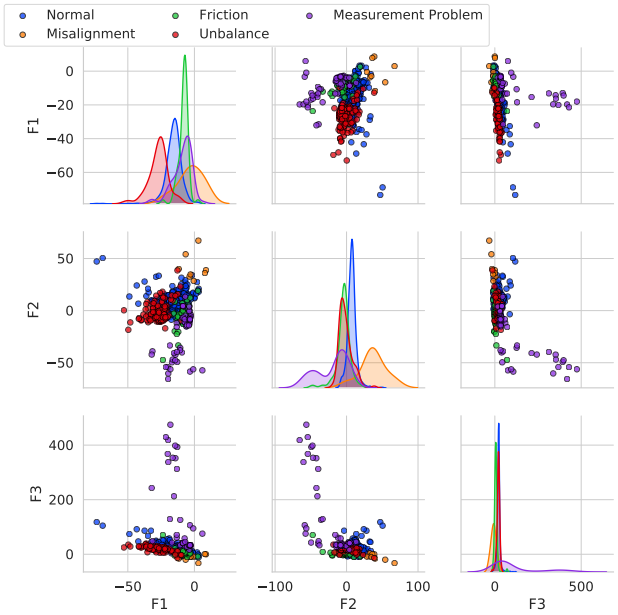

Fig. 4. Pair scatter plot of the testing dataset for three features from the deep-space. Each color represents a class. Estimated univariate distributions, one for each class, are plotted in the diagonal. Chosen extracted features are named as F1, F2, F3 in no specific order.

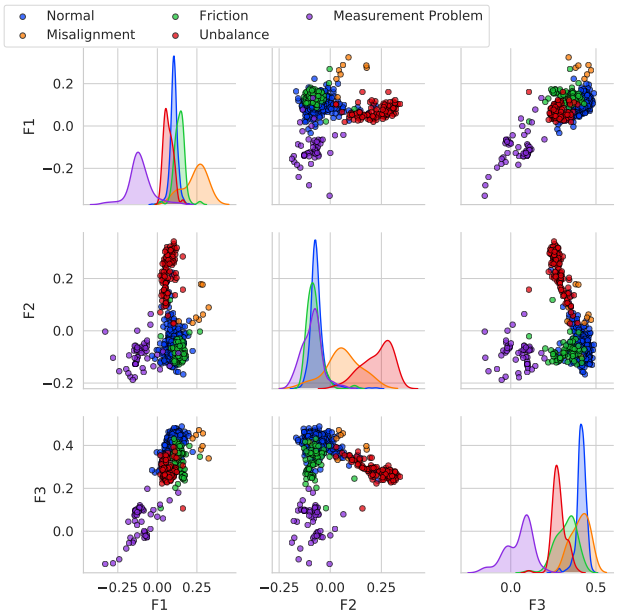

Fig. 5. Scatter plot of the testing dataset for each pair of features from the triplet-space. Each color represents a class. Estimated univariate distributions, one for each class, are plotted in the diagonal. Chosen extracted features are named as F1, F2, F3 in no specific order.

that samples of each class in the triplet-space tends to follow a multivariate normal distribution which, if true, can make the detection of new unknown classes much easier, because it is possible to calculate the likelihood that a sample belongs to a class. If a new sample have a low likelihood for all classes, we expect this sample to probably belongs to a new unknown fault type. This is due to the way the triplet loss is designed. Consequently, the deep-space cannot be used in the same way as the triplet-space for detecting new type or class of samples for, at least, the present dataset.

VI. Conclusions

In this paper, a methodology based on triplet neural networks is developed for automatically extracting and discovering relevant features for detecting and diagnosing faults in an electrical submersible pump system. Empirical evidence shows equal effectiveness of automatically extracted features and specialist extracted features. The hypothesis tests conducted indicate there is no classifier model trained on the hand-crafted feature spaced that performed better than the respective classifier model trained on the new feature space. In addition to that, the hypothesis test indicated that three classifiers models had their average f-measure significantly higher when trained in the new feature space. Moreover, evidences show better performance when using neural networks as features extractors for conventional machine learning algorithms. Therefore, we believe that our methodology for extracting relevant features can be used to circumvent the need of a specialist investing time in the process of designing features to be extracted.

The methodology proposed here provides a solution for the problem of finding a feature space where the euclidean distance approximates the "semantic distance" in the signal processing field. In addition, the new feature space could be useful for detecting the introduction of a new class/fault type. This can be done by comparing similarities, using the euclidean distance, among samples in the new feature space. If a new example falls far from the centroids of each class, then its is probably a new example of a distinct class. The reliability of such a method should be investigated in future works.

Other future work may investigate the meaning and understanding of the new extracted features. This may lead to the discovery and understanding of new features that specialists were not aware of, consequently increasing the human knowledge about the problem domain. Lastly, the combination of the new feature space within the hand-crafted feature space in order to improve classification performance should be evaluated.

References

[1] G. Takacs, *Electrical submersible pumps manual: design, operations, and maintenance*. Gulf professional publishing, 2017.

[2] T. Oliveira-Santos, T. W. Rauber, F. M. Varejão, L. Martinuzzo, W. Oliveira, M. P. Ribeiro, and A. Rodrigues, "Submersible motor pump fault diagnosis system: A comparative study of classification methods," in *2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI)*, Nov 2016, pp. 415–422.

[3] T. W. Rauber, T. Oliveira-Santos, F. de Assis Boldt, A. Rodrigues, F. M. Varejão, and M. P. Ribeiro, "Kernel and random extreme learning machine applied to submersible motor pump fault diagnosis," in *2017 International Joint Conference on Neural Networks, IJCNN 2017, Anchorage, AK, USA*, 2017, pp. 3347–3354.

[4] T. Oliveira-Santos, A. Rodrigues, V. Rocha, T. Rauber, F. Varejão, and M. Ribeiro, "Combining classifiers with decision templates for automatic fault diagnosis of electrical submersible pumps," *Integrated Computer-Aided Engineering*, vol. 25, pp. 1–16, 2018.

[5] E. Hoffer and N. Ailon, "Deep metric learning using triplet network." in *ICLR (Workshop)*, Y. Bengio and Y. LeCun, Eds., 2015.
[6] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
[7] M. Kaya and H. Bilge, "Deep metric learning: A survey," *Symmetry*, vol. 11, p. 1066, 08 2019.
[8] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering." in *CVPR*. IEEE Computer Society, 2015, pp. 815–823.
[9] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transaction on Information Theory*, vol. 13, no. 1, p. 21–27, Sep. 2006.
[10] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, p. 273–297, Sep. 1995.
[11] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. Monterey, CA: Wadsworth and Brooks, 1984.
[12] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Springer series in statistics New York, 2001, vol. 1, no. 10.
[13] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing & Management*, vol. 45, no. 4, pp. 427 – 437, 2009.
[14] C. Nadeau and Y. Bengio, "Inference for the generalization error," *Mach. Learn.*, vol. 52, no. 3, p. 239–281, Sep. 2003.
[15] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.
[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine Learning in Python ," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.