UNIVERSIDADE FEDERAL DE JUIZ DE FORA
FACULDADE DE ENGENHARIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Carlos Manuel Viriato Neto

Semi-supervised Deep Rule-Based Approach for the Classification of Wagon Bogie Springs Condition

Juiz de Fora
2022

Carlos Manuel Viriato Neto

Semi-supervised Deep Rule-Based Approach for the Classification of Wagon Bogie Springs Condition

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Juiz de Fora como requisito parcial à obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: Sistemas Eletrônicos

Orientador: Prof. Dr. Eduardo Pestana de Aguiar

Juiz de Fora

2022

Ficha catalográfica elaborada através do Modelo Latex do CDC da UFJF
com os dados fornecidos pelo(a) autor(a)

Viriato Neto, Carlos Manuel.
Semi-supervised Deep Rule-Based Approach for the Classification of Wagon Bogie Springs Condition / Carlos Manuel Viriato Neto. – 2022.
52 f. : il.

Orientador: Eduardo Pestana de Aguiar
Dissertação (Mestrado) – Universidade Federal de Juiz de Fora, Faculdade de Engenharia. Programa de Pós-Graduação em Engenharia Elétrica, 2022.

1. Evolving Fuzzy Systems. 2. Railway applications. 3. Artificial Intelligence . 4. Imaging Processing. I. Aguiar, Eduardo Pestana, orient. II. Título.

FEDERAL UNIVERSITY OF JUIZ DE FORA
RESEARCH AND GRADUATE PROGRAMS OFFICE

Carlos Manuel Viriato Neto

Semi-supervised Deep Rule-Based Approach for the Classification of Wagon Bogie Springs Condition

Dissertation submitted to the Graduate Program in Electrical Engineering
of the Federal University of Juiz de Fora as a partial requirement
for obtaining a Master's degree in Electrical Engineering.
Concentration area: Electronic Systems.

Approved on 08 of June of 2022.

EXAMINING BOARD

Professor/ Dr. Eduardo Pestana de Aguiar – Academic Advisor
Federal University of Juiz de Fora

Professor/ Dr. Gustavo Pessin
Vale Institute of Technology

Professor/ Dr. Helon Vicente Hultmann Ayala
Pontifical Catholic University of Rio de Janeiro

Professor/ Dr. Leonardo Willer de Oliveira
Federal University of Juiz de Fora

Juiz de Fora, 06/08/2022.

Documento assinado eletronicamente por **Eduardo Pestana de Aguiar**, **Professor(a)**, em 09/06/2022, às 19:07, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Gustavo Pessin**, **Usuário Externo**, em 09/06/2022, às 20:40, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **HELON VICENTE HULTMANN AYALA**, **Usuário Externo**, em 09/06/2022, às 23:05, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Leonardo Willer de Oliveira**, **Professor(a)**, em 10/06/2022, às 10:14, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

A autenticidade deste documento pode ser conferida no Portal do SEI-Ufjf (www2.ufjf.br/SEI) através do ícone Conferência de Documentos, informando o código verificador **0788008** e o código CRC **2E42A35C**.

To my parents, Altamiro and Zenaide,
to my sisters, Caroline and Caterine,
and to my fiancée and greatest supporter,
Hisabella Lorena Simões Porto.

AGRADECIMENTOS

I thank God for all he has done for me and all his love.

To my parents, Altamiro and Zenaide, my first educators and inspiration, for giving me precious teachings, for supporting me in my career and for always encouraged the search for knowledge.

To my sisters, Caroline and Caterine, with I have learned much from.

To my love, my fiancée and greatest supporter, Hisabella Lorena Simões Porto.

To all the friends who have made and are still part of my life, especially those who were by my side on this journey.

To my advisor Eduardo Aguiar for guide me in this process, for all the teaching, attention and trust placed in my job.

To the professors of the department of Graduate Program in Electrical Engineering from the Federal University of Juiz de Fora.

To all people who contributed to my progress.

I acknowledge the Federal University of Juiz de Fora for essential support during this work an to the anonymous referees for their valuable comments.

I also thank the MRS Logística S.A. and to all my co-workers for the essential support on this job.

ABSTRACT

The wagons are submitted to stressing cycles with heavy loads, increasing their bogie defects through springs fatigue, then, the capacity to detect critical freight cars conditions enables to guarantee the safety production and high productivity of transportation systems. The image processing and computational intelligence techniques are increasingly participating in the solution for this scenario, especially human interpretable and self-evolve models which can learn new classes actively without human experts' involvement to self-evolve and perform classification on out-of-sample images. In this sense, this dissertation presents a new approach model for the classification of wagon bogie springs condition through images acquired by a wayside equipment. As such, we are discussing the application of a semi-supervised learning approach based on a deep rules-based (DRB) classifier learning approach to achieve a high classification of a bogie, and check if they either have spring problems or not. We use a pre-trained VGG19 deep convolutional neural network to extract the attributes from images to be used as input to the Fuzzy Rule Based (FRB) layer of the semi-supervised DRB (SSDRB) classifier and evaluated with euclidean, cosine, manhattan, minkowski, chebyshev distance measures. The performance is calculated based on the dataset composed of images provided by a Brazilian railway company which covers the two spring condition : normal condition (no elastic reserve problems) and bad condition (with elastic reserve problems). Also, an additive Gaussian noise levels, Cauchy noise and Laplace noise are applied to the images to challenge the proposed model and to represent possible problems on image acquisition. Finally, we discuss the performance analysis of the semi-supervised DRB (SSDRB) classifier and its distinctive characteristics with each distance measure compared with other classifiers. The reported results demonstrate a relevant performance of the SSDRB classifier applied to the questions raised as well the importance of evaluation of distance measure to achieve a high classification.

Keywords: Evolving Fuzzy Systems. Railway applications. Wagons Bogie Springs. Artificial Intelligence. Imaging Processing.

RESUMO

Vagões são submetidos a ciclos de estresse com cargas pesadas, aumentando seus defeitos no truque por fadiga de molas, então, a capacidade de detectar condições críticas dos vagões de carga permite garantir a produção com segurança e alta produtividade dos sistemas de transporte. As técnicas de processamento de imagens e inteligência computacional estão cada vez mais participando da solução para este cenário, principalmente os modelos interpretáveis e auto-evolutivos que podem aprender novas classes ativamente sem o envolvimento de especialistas humanos para auto-evoluir e realizar a classificação em imagens fora da amostra. Nesse sentido, esta dissertação apresenta um novo modelo de abordagem para a classificação do estado das molas dos truques dos vagões através de imagens adquiridas por um equipamento as margens da ferrovia. Como tal, é discutido a aplicação de uma abordagem de aprendizado semi-supervisionado baseada em uma abordagem de aprendizado de classificador baseado em regras profundas (DRB) para obter uma alta classificação do truque e verificar se eles molas com problemas ou não. Uma rede neural convolucional profunda VGG19 pré-treinada é usada para extrair os atributos de imagens a serem usados como entrada para a camada de regras baseadas em fuzzy (FRB) do classificador semi-supervisionado DRB (SSDRB) e avaliados com as métricas de distância euclidiano, cosseno, manhattan, minkowski and chebyshev. O desempenho é calculado com base no conjunto de dados composto por imagens fornecidas por uma empresa ferroviária brasileira que abrange as duas condições de mola: condição normal (sem problemas de reserva elástica) e condição ruim (com problemas de reserva elástica). Além disso, níveis de ruído gaussiano, ruído de Cauchy e ruído de Laplace são aplicados às imagens para desafiar o modelo proposto e representar possíveis problemas na aquisição de imagens. Por fim, discutimos a análise de desempenho do classificador semi-supervisionado DRB (SSDRB) e suas características distintivas a cada medida de distância comparada com outros classificadores. Os resultados relatados demonstram um desempenho relevante do classificador SSDRB aplicado às questões levantadas, bem como a importância da avaliação da medida de distância para alcançar uma classificação alta.

Palavras-chave: Sistemas Fuzzy Evolutivos. Aplicações ferroviárias. Molas de truque do vagão. Inteligência artificial. Processamento de Imagens.

LIST OF ILLUSTRATIONS

LIST OF TABLES

LIST OF ABBREVIATIONS AND ABBREVIATIONS

VGG19	Deep Convolutional Neural Network that is 19 layers deep
DRB	Deep Rule Based
FRB	Fuzzy Rule Based
SSDRB	Semi-Supervised Deep Rule Based
DNN	Deep Neural Network
SVM	Support Vector Machines
KNN	K-Nearest Neighbors
QDA	Quadratic Discriminant Analysis
AWGN	Additive White Gaussian Noise
ANTF	Brazilian National Association of Railway Transporters
TST	Time Stopped Train
MTBF	Meantime Between Failures
PSNR	Peak Signal-to-Noise Ratio
MSE	Mean Squared Error
PCM	Pulse-Code Modulation

LIST OF SYMBOLS

$\sim$	Similarity
$\in$	Belongs to
$\exists$	There Exists
$\approx$	Approximately
$\pm$	Plus-Minus
$\neq$	Not equal
$=$	Equal
$\equiv$	Equivalence
$\longleftarrow$	Assignment
$\#$	Cardinality Operator
$<$	Less than
$>$	Greater than
$\leq$	Less than or Equal to
$\geq$	Greater than or Equal to
$\|\|$	Absolute Value
p	p-value
dB	Decibel
(s)	Second
$'W'$	Win
$'L'$	Loss
$'E'$	Equable
σ	Standard Deviations
$\mathcal{G}$	Sets of Samples

CONTENTS

1 INTRODUCTION

Since the emergence of steam-powered machines, rail transportation has become an effective solution for connecting urban centers, as well as a low-cost alternative to the industries' transactions. Due to these circumstances, the wagons are submitted to stressing cycles with heavy loads, increasing their bogie defects through springs fatigue.

In this context, image processing and computational intelligence techniques are increasingly participating in the solution for this scenario. Since their capacity to detect critical wagon conditions enables to guarantee the safety production and high productivity of this system of these transportation systems.

Gu and Angelov, in [1] introduced deep rule-based (DRB) classifiers [2, 3, 4]. The DRB classifier is a general approach that serves as a strong alternative to current deep neural network (DNN) [2, 3, 4]. It is non-parametric, non-iterative, highly parallelizable and computationally efficient; it achieves very high classification rates, surpassing other methods [4].

Moreover, it further extends the DRB classifier [1] with a self-organising, self-evolving semi-supervised learning strategy by exploiting the idea of "pseudo label" naturally with its prototype-based nature. Starting with a small amount of labelled training images, the semi-supervised DRB (SSDRB) classifier is able to pseudo-label remaining images based on the ensemble properties of the training images using non-parametric measures [5]. As semi-supervised learning is leveraged to obtain labeled cluster samples without a full retraining [6], the SSDRB classifier also can self-organize its system structure and self-update recursively with the pseudo-labelled data and, thus, it supports real-time streaming data processing.

The proposed SSDRB classifier also inherits the advantage of the DRB classifier's transparency as its semi-supervised learning process only concerns the visual similarity between the identified prototypes and the unlabelled samples, which is highly human interpretable compared with the state-of-the-art approaches, SVM [7] and deep learning networks [8].

Furthermore, as mentioned in [1], the proposed SSDRB classifier cannot only perform classification on out-of-sample images but also support recursive online training on a sample-by-sample basis or a chunk-by-chunk basis. Moreover, unlike other semi-supervised approaches, the proposed approach can learn new classes actively without human experts' involvement to self-evolve [9], therefore, it is very important to present SSDRB for the railway sector. However, the prototypes are identified directly from the training data in a non-parametric, self-organizing manner. Accordingly, they are the most representative prototypes representing local maxima in data density and the chosen distance measure affect directly the performance of SSDRB [10].

1.1 OBJECTIVES

This master's thesis discusses the architecture and approach of a semi-supervised deep rule-based (SSDRB) algorithm [2, 3, 4, 1] based on cosine dissimilarity as a distance measure to deal with the classification of wagon bogie springs conditions identifying if they are or are not in bad condition. Furthermore, this work evaluates the performance in terms of euclidean, minkowski, manhattan, chebyshev distance measures on the SSDRB algorithm process to classify the springs conditions.

The performance is calculated based on the dataset composed of images provided by a Brazilian railway company which covers the two spring condition : normal condition (no elastic reserve problems) and bad condition (with elastic reserve problems). Also, an additive Gaussian noise levels, Cauchy noise and Laplace noise are applied to the images to challenge the proposed model and to represent possible problems on image acquisition.

In addition, the reported results are compared with other results from classifiers present in the literature, namely: K-Nearest Neighbors (KNN), Linear SVM, Decision Tree, Randon Forest, Neural Net, Adaboost, Naive Bayes and QDA.

1.2 CONTRIBUTIONS

The main contributions of this work are summarized below:

- The model discussed in this paper has advantages that are not covered by classical methods, such as a learning process that is easy to interpret by a specialist; online or offline training; the capability to classify images outside the sample; capability to deal with uncertainty;
- The study of the wagon bogie springs conditions through SSDRB and the evaluation of distance measures on SSDRB applied to the classification to determine which one best suits the proposed problem has never been addressed before;
- The evaluation of SSDRB results in the classification of the wagon bogie springs conditions in comparison with other classifiers present in the literature (K-Nearest Neighbors (KNN), Linear SVM, Decision Tree, Randon Forest, Neural Net, Adaboost, Naive Bayes and QDA);
- The performance analysis of the SSDRB and other classifiers on classification of the wagon bogie spring conditions which is added to the images a Gaussian noise levels, Cauchy noise and Laplace noise to challenge the models and represent the possible problems on image acquisition;

- We use a pre-trained vgg-verydeep-19 deep convolutional neural network (VGG19) [11] to extract the attributes of the images. By this, the proposed model can learn abstract resources and obtain higher precision;
- We present the performance analysis in terms of the classification accuracy using a dataset with images acquired from railway wayside equipment.

And our major conclusions are:

- The use of VGG19 as a feature extractor is effective for this application;
- The SSDRB classifier with minkowski distance achieved the best average accuracy results on the study compared with other distance measures applied on SSDRB and other classifiers. However, the SSDRB classifier with manhattan distance achieved the best benefit-cost ratio results on the study compared with other distance measures applied on SSDRB e other classifiers;
- Through the obtained results, the SSDRB classifier proved to be an excellent alternative for the classification of wagon bogie springs conditions. It is an interpretable model easily understandable by humans and it can assist in the inspection process along the rail, by reducing inspection times, ensuring greater reliability and availability of the wagons.

1.3 WORK ORGANIZATION

The remainder of this work is organized as follows: Chapter 2 deals with the formulation of the problem. Chapter 3 aims to discuss the concept of SSDRB and distance measures on SSDRB algorithm. Chapter 4 describes the database and the result of the proposed metrics and comparisons with other classifiers present in the literature. The last chapter describes the conclusions, final observations of the work and the presentation of future works.

2 PROBLEM FORMULATION

The bogie suspension springs are a fundamental part of the wagon's damping set which has the function of dissipating the energy caused by some unwanted vertical movements that occur in railway dynamics.

As wagon bogies springs are the main part of suspension and damping, the classification of its condition is a critical development to assist the railway companies in verifying its railway conditions and safety along with granting higher services reliability.

In 2017, 11 wagons of a 33 wagon freight train traveling north of Ely have derailed as presented in the Figure 1. The derailment accident occurred by ineffective damping on the wagon bogies [12]. The line was blocked, affecting passenger services from Peterborough and Cambridge to Stansted Airport and London for 7 days. Rail experts say the cost of the derailment could top £1.0 million [13].

As shown in the Figure 2, a similar accident already occurred 10 years before when a line was closed for six months as result of derailment caused by bogie's suspension problems [14, 12]. The river Ouse had been shut to traffic and Network Rail had to create a 1.3km access route to the site to recover the stricken wagons that were left teetering over the river Ouse [15, 16]. In addition, Network Rail had to rebuild the rail bridge across the River Ouse which cost around £9.0 million [17].

Figura 1 – A crane's lifting the derailed wagons in Cambridgeshire
[18]

Figura 2 – The derailed wagons of freight train on Ely rail bridge
[19]

According to ANTF (Brazilian National Association of Railway Transporters) [20], the volume of goods transported by the railroads increased by 95% in the period from 1997 to 2019.

In 2019, approximately US$ 600 million was invested, allowing for a significant growth in the rolling stock fleet. In 1997, the railroads had 1,154 locomotives and in 2019, they already totaled 3,405 units, representing an increase of 195%. In the same period, the number of wagons went from 43,816 to 115,434, representing an increase of 163%. This trend causes an increase in speed and loads transported, changing the dynamic wheel-rail contact, thus increasing the probability of bogie spring defected.

Defects in the springs occur due to different reasons, for example, as a result of fatigue, due to repetitive passages over the rail components, such as welds, joints, and switches, or due to the impacts of defect wagons bogies springs. If bogie spring defected grow and are delayed, they can lead to high maintenance costs. Therefore, it is essential a rapid and automatic defect detection.

Accordingly, the proposed model can reduce the impact of overhauls on trains operation, since it enables a preventive maintenance routine, which makes it possible for interventions to be carried out only when anomalies are observed. Furthermore, it is an interpretable model easily understandable by humans, making it replicable to other types of wagons and scalable along the rail, thus reducing time spent on inspection.

Based on perspective, it is necessary, especially because it results in order to:

- Prevent accidents caused by unexpected failures;
- Reduces the number of unproductive hours in maintenance;
- Eliminates the manual process of visual inspection;
- Reduces the number of recurrent preventive interventions;
- Reduces the probability of brinelling on a bearing;
- Ensures greater reliability and availability of the wagons;
- Increases the productivity of rail operations, given the reduction in frequency and time of operational maintenance;
- Reduces the TST (Time Stopped Train) index through the increase in the meantime between failures (MTBF).

Taking into account the scenario previously exposed in this Section, this work aims to classify the two main conditions that can occur with the wagon bogie springs: springs with elastic reserve (without defect) and springs without elastic reserve (with bad condition).

As shown in Figure 3, the springs have no elastic reserve problems and It is noted because the springs have space between their turns.

However, in Figure 4 is shown no space between their turn what is a critical defect because they do not have damping capacity anymore.

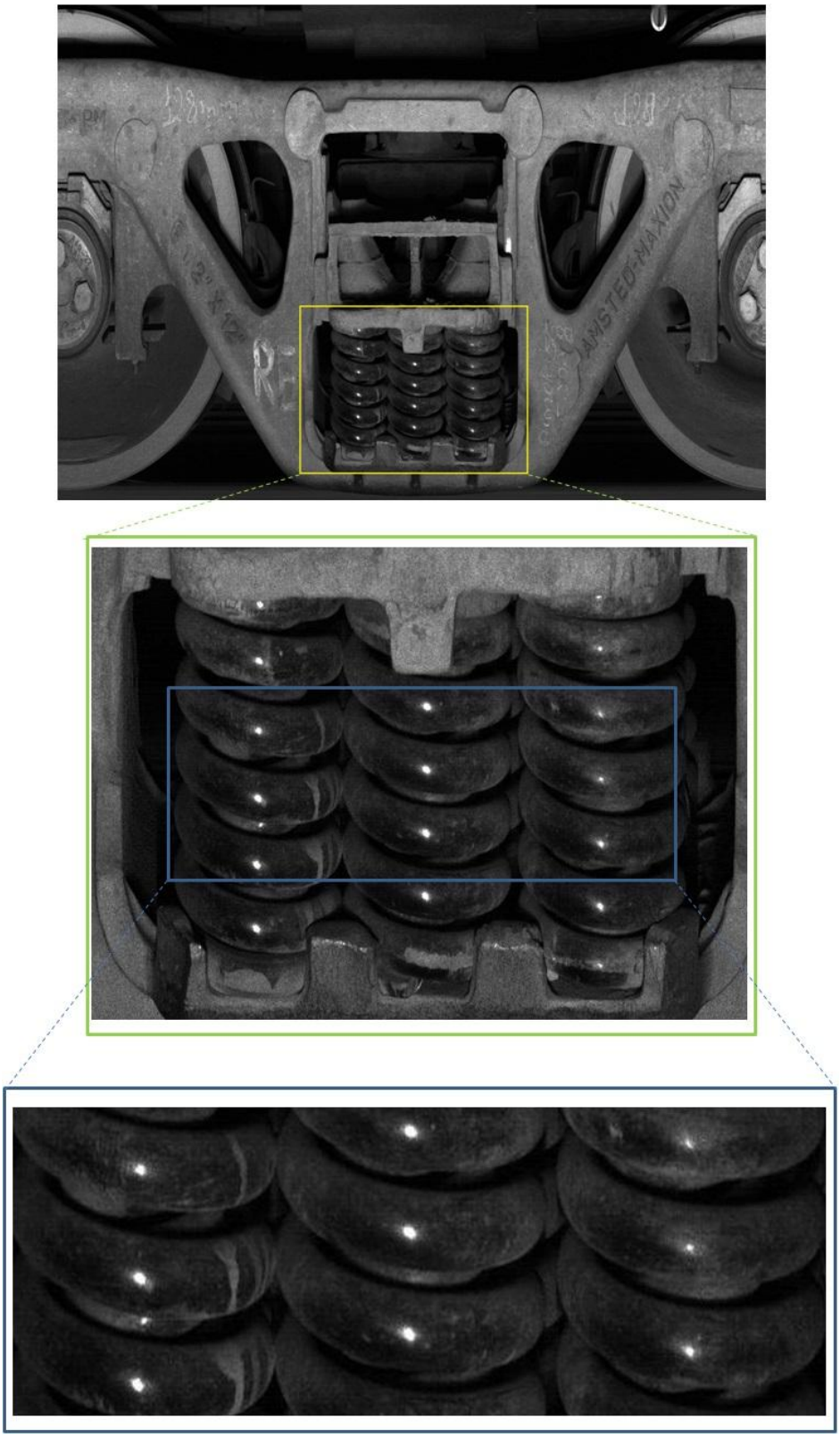


Figura 3 – Bogie springs without defect
[21]

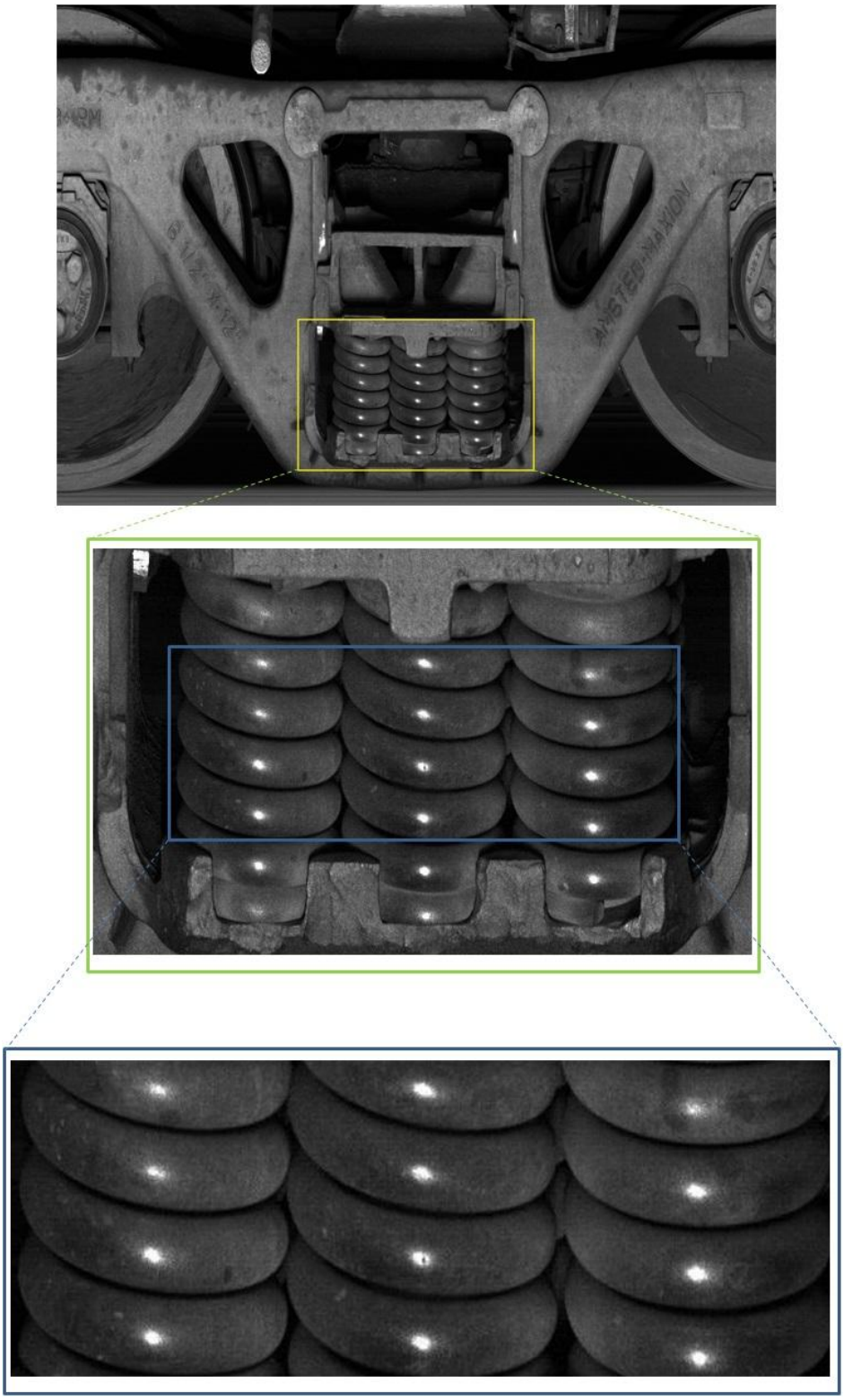


Figura 4 – Bogie springs with bad condition
[21]

3 THE PROPOSAL: SSDRB

DRB architecture is shown in Figure 5. The input **I** consists of a wagons bogies springs image and is subject to a normalization of pixel values up to a range of [0,255] [22]. Subsequently, the image is scaled to 227x227 pixels to increase generalization and reduce computational complexity [22].

The VGG19 is used for resource extraction due to its simpler structure and better performance. The extracted data are processed by the fuzzy rule-based (FRB) layer, which constitutes a massively parallel set of nebulous irrigations of the AnYa type 0-order fuzzy [23], which is the basis of the DRB classifier. Then, there is no need to define ad hoc association functions on FRB layer [10].

Finally, the decision-making classifies the images based on the degree of similarity with the prototypes generated in the training stage.

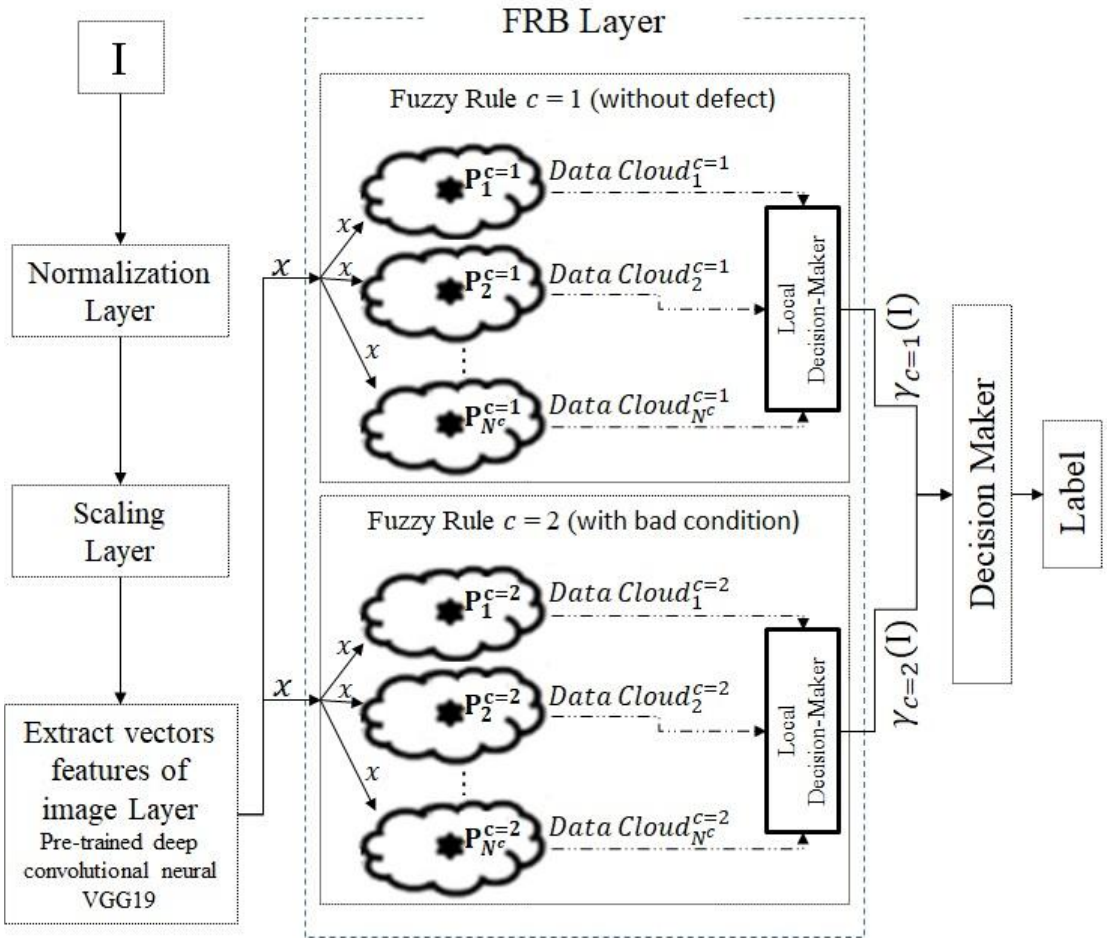


Figura 5 – Architecture of the DRB classifier to wagon bogie springs condition [1]

The use of a pre-trained deep convolutional neural networks to extract global vectors of image features to train generic classifiers is an alternative widely used since it allows the classifier to learn more abstract and discriminative attributes of high level

obtaining greater precision [22, 24]. In this work, the VGG19 is used to vectors extracted from the wagon bogie spring images that have a dimension of [1,4096] [24].

A previously mentioned FRB layer is the learning mechanism of the DRB classifier. The FRB subsystems are independent of each other and can be changed without influencing the others. Furthermore, each FRB subsystem contains a set of massively parallel fuzzy rules, formulated around generalized prototypes P or learned from the corresponding class segments. As they all have the same consequence can be combined through the logical connector "OR", as follows:

$$\textbf{IF } (I \sim P_1^c)\ OR\ (I \sim P_2^c)\ OR ...\ OR\ (I \sim P_{N^c}^c)\ \textbf{THEN}\ (CLASS\ c) \tag{3.1}$$

In Eq. 3.1 "$\sim$"denotes similarity; $c = 1, 2, ..., C$; N^c is the number of prototypes of the c^{th} class.

During the training process, the prototypes are identified from the data density in the feature space and from these prototypes the corresponding fuzzy rules are generated $X_1^c\,(\,X_1^c = X_{1,1}, X_{1,2}, ..., X_{1,M})$ [25] where for this paper it was adopted $M = 4096$ and the following distance measures was evaluated on the similarity rule of the classifier.

3.1 DISTANCES MEASURES

The corresponding fuzzy rules generated from the data density in the feature space for each prototype are used for identifying the underlying patterns and, thus, it is the key to the similarity rule of the classifier [9]. Because of the very high dimensionality of the feature vectors, Gu and Angelov, in [1] use cosine dissimilarity as the distance measure [1, 25, 26].

However, it is well known that different distance types have different abilities in disclosing the ensemble properties and mutual distribution of the data, and the differences are even more significant in higher dimensional data spaces [27, 28]. Choosing the most suitable distance type for a specific problem is of great importance for a meaningful classifier result [29]. Then, each following distance measures is used to evaluate the performance of the SSDRB to classify the springs conditions.

3.1.1 **Cosine**

In problems whose dimensions are large, it is very common to use the cosine distance. This distance measures the cosine of the angle between two nonzero vectors.

The cosine distance between $\mathbf{x}_i$ e $\mathbf{x}_j$ is formulated as:

$$d_{cos}(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{2 - 2\cos\theta_{\mathbf{x}_i,\mathbf{x}_j}} \tag{3.2}$$

where $\theta_x^{i,j}$ represents the angle between $\mathbf{x}_i$ and $\mathbf{x}_j$ in a Euclidean space. With this, it is possible to represent the inner product of the two vectors as:

$$\cos\theta_{\mathbf{x}_i,\mathbf{x}_j} = \frac{(\mathbf{x}_i, \mathbf{x}_j)}{\|\mathbf{x}_i\| \, \|\mathbf{x}_j\|} = \frac{\sum_{i=1}^{M} \mathbf{x}_i x_j}{\|\mathbf{x}_i\| \, \|\mathbf{x}_j\|} \tag{3.3}$$

but the norm of x_i is:

$$\|\mathbf{x}_i\| = \sqrt{(\mathbf{x}_i, \mathbf{x}_j)} = \sqrt{\sum_{i=1}^{M} x_i^2} \tag{3.4}$$

After some manipulation, it is possible to arrive at the cosine dissimilarity as the distance measure given as [30]:

$$d_{cos}(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{2 - 2\cos\theta_{\mathbf{x}_i,\mathbf{x}_j}} = \sqrt{2 - 2\frac{\sum_{i=1}^{M} x_i x_j}{\|\mathbf{x}_i\| \, \|\mathbf{x}_j\|}} = \left\| \frac{\mathbf{x}_i}{\|\mathbf{x}_i\|} - \frac{\mathbf{x}_j}{\|\mathbf{x}_j\|} \right\| \tag{3.5}$$

3.1.2 **Euclidean**

The Euclidean distance, named after the Greek mathematician Euclid (about 325 BC to 265 BC), between two data samples, $\mathbf{x}_i$, $\mathbf{x}_j \in \{\mathbf{x}\}_C$, is calculated based on the following equation [31]:

$$d_{euc}(\mathbf{x}_i, \mathbf{x}_j) = ||\mathbf{x}_i - \mathbf{x}_j|| = \sqrt{\sum_{l=1}^{M} (x_{i,l} - x_{j,l})^2} \tag{3.6}$$

The Euclidean measure is the most used measure due to its simplicity and computational efficiency despite attributing equal weight and importance to each of the M dimensions.

3.1.3 **Minkowski**

The Minkowski distance, named after the German mathematician Hermann Minkowski (1864-1909), is used when it is necessary to identify and, in fact, ignore irrelevant features and when there are a large number of anomalous clusters. This distance is a normalized vector space, which can generalize the Euclidean distance, the Manhattan distance and the Chebychev distance, and is given by:

$$d_{min}(\mathbf{x}_i, \mathbf{x}_j) = \left(\sum_{l=1}^{M} (|x_{i,l} - x_{j,l}|)^h \right)^{\frac{1}{h}} \tag{3.7}$$

For $h = 1$ we have the Manhattan distance, $h = 2$ the Euclidean distance and for $h \to \infty$ the Chebychev distance. For other values of h there are no specific names.

Therefore, for the present work, in order to differentiate the results of euclidean, manhattan and chebychev distances, $h = 1.5$ was chosen to compare the use of Minkowski distance to the others.

3.1.4 **Manhattan**

The Manhattan distance (also known as *City Block*) is included in the same family of distance functions as a special Minkowski case where $h = 1$. The expression to calculate the distance between $\mathbf{x}_i$ e $\mathbf{x}_j$ is:

$$d_{man}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{l=1}^{M} |x_{i,l} - x_{j,l}| \tag{3.8}$$

where the module || denotes absolute values. We can see by analyzing the above equation that the distance is the sum of the absolute differences of their Cartesian coordinates.

3.1.5 **Chebychev**

The Chebychev distance, named after the Russian mathematician Pafnuty Chebyshev (1821-1894), It is also known as chessboard. It is also a special Minkowski case for $h \to \infty$. The expression to calculate the distance between $\mathbf{x}_i$ e $\mathbf{x}_j$ is:

$$d_{che}(\mathbf{x}_i, \mathbf{x}_j) = \max_{l} |x_{i,l} - x_{j,l}| \tag{3.9}$$

where the module || denotes absolute values. We can see by analyzing the above equation that the distance is defined as the greatest of difference between two vectors along any coordinate dimension.

3.2 DRB TRAINING PROCESS

The DRB training process is mentioned in [4] in which, the DRB classifier identifies prototypes of the segments of the observed images of each class in an autonomous and non-parametric way and forms clouds of data around the prototypes of similar segments of the same class.

In this way, the C rules of massively parallel diffuse parallel C order of the type AnYa in total are formed (learned) through the training processes independently, based on the identified prototypes.

The detailed training process for the FRB subsystems is described in [2] and [32], and the main procedure of the training process is summarized in pseudocode form, as shown in the Algorithm 1.

Algoritmo 1 Algorithm: Training process of the FRB subsystem
Distance measure: Cosine

While new feature vector x_k of the c^{th} class is available
i. $x_k \longleftarrow \frac{x_k}{\|x_k\|}$ ($\|x_k\|$ is the norm of x_k);
ii. **If** $(k = 1)$ **Then**
  1. $\mu_c \longleftarrow x_k$ (μ_c is the global mean)
  2. $N_c \longleftarrow 1; p_{c,1} \longleftarrow x_k; S_{c,1} \longleftarrow 1; r_{c,1} \longleftarrow r_0$;
(N_c is the number of prototypes; $p_{c,1}$ is the first prototype; $S_{c,1}$ is the corresponding support; $r_{c,1}$ is the radius, $r_0 = \sqrt{2 - 2\cos(\frac{\pi}{6})}$)
iii. **Else**
  1.Update global mean; $\mu_c \longleftarrow (k-1)\frac{\mu_c}{k} + \frac{x_k}{k}$;
  2.Calculate the density of x_k:
    $D(x_k) = \frac{1}{1+\frac{\|x_k - \mu_c\|^2}{1-\|\mu_c\|^2}}$
  3. Update the densities of prototypes $(j = 1, 2, ..., N_c)$;
    $D(x_k) = \frac{1}{1+\frac{\left\|p_{c,j} - \mu_c\right\|^2}{1-\|\mu_c\|^2}}$
  4. **If** $(min(D(p_{c,j})) \leq D(x_k) \leq max(D(p_{c,j})))$ **Then**;
    -Find nearest prototype: $p_{n,c} = argmin(\|x_k - p_{c,j}\|)$;
    -**If** $(\|x_k - p_{c,n}\| > r_{c,n})$ **Then**;
      * $N_c \longleftarrow N_c + 1; p_{c,N_c} \longleftarrow x_k; S_{c,N_c} \longleftarrow 1; r_{c,N_c} \longleftarrow r_0$;
    Else
      * $p_{c,n} \longleftarrow S_{c,n}\frac{p_{c,n}}{(S_{c,n}+1)+\frac{x_k}{S_{c,n}+1}}$
      * $S_{c,n} \longleftarrow S_{c,n} + 1; r_{c,n}^2 \longleftarrow \frac{1}{2}(r_{c,n}^2 + (1 - \|p_{c,n}\|^2))$;
    End If
  5.**Else**
    $N_c \longleftarrow N_c + 1; p_{c,N_c} \longleftarrow x_k; S_{c,N_c} \longleftarrow 1; r_{c,N_c} \longleftarrow r_0$;
  6. **End If**
iv. **End If**
v. Generate/update the AnYa type fuzzy rule;
End While

3.3 DRB TEST PROCESS

Once the training process is completed, the classification of new images can be performed using the identified FRB. As shown in [1], during the validation process, each test image receives a confidence score from the fuzzy rules identified in the training stage [4]:

$$\gamma^c(I) = max(exp(-d^2(x, p_i^c))) \tag{3.10}$$

As a result, for each testing image, a vector of 1 x C dimensional scores of confidence of the nearest prototypes (one per spring's condition class) is generated by the same measure distance chosen in 3.2 and detail in 3.1.

$$\gamma(I) = [\gamma^1(I), ..., \gamma^C(I)] \tag{3.11}$$

The label of this testing image is decided by using the "winner-takes-all" principle [4]:

$$Label = argmax(\gamma^c(I)) \tag{3.12}$$

3.4 SSDRB

In the SSDRB classifier, after the training step performed by the DRB with the labeled images, the model has the ability to learn from the unlabeled images. For a set of images $\{U\}$ with U unlabeled images, a confidence vector $\gamma(U_i) = (i = 1, 2, ..., U)$ is extracted from each U image using Eq. 3.10.

The images that satisfy the Eq. 3.13 condition will be used to update the meta-parameters.

$$IF(\gamma^{*max}(U_i) > \phi \cdot \gamma^{**max}(U_i)) \tag{3.13}$$

Where $\gamma^{*max}(U_i)$ denotes the highest score of confidence; $\gamma^{**max}(U_i)$ denotes the second highest score; $\phi(\phi > 1)$ is a free parameter.

4 EXPERIMENTAL RESULTS

Based on the theory addressed in the previous topic, tests were carried out to evaluate the classification model proposed in this work.

4.1 DATA BASE

The database used in this application is made from wagons bogies of Brazilian railway company MRS Logística S.A. [33]. Such database is composed of images that are captured by a railway wayside equipment fixed to the MRS railroad in order to capture wagons bogies images each time the train passes through the site.

The images are taken from both wagon sides when it is crossing through the equipment and the examples of the images contained in the database are shown in the Figures 3 and 4. In addition, it is important to mention that the wagons are empty when they pass through the site, i.e., it means the springs expected standard condition is that they are not compressed and the springs should have space between their turns.

The types of bogie that can be used on freight wagons may vary according to the type of wagon. Nonetheless, for each type of bogie the springs type and its geometric distribution are particular to each bogie type. Therefore, for the present work we used bogie images that correspond to the most representative portion of the main transport flow of MRS Logística S.A.

The experiment was conducted by collecting 250 images evenly distributed over different periods of the day which 125 presented wagon bogie with springs without elastic reserve problems and 125 images in which correspond to bogie with elastic reserve problems.

In addition, all datasets were presented to the classifier 200 times using the Monte Carlo holdout method [34], therefore, the data were randomly divided into training and testing sets at each round. Moreover, in each interaction, the dataset was distributed for the test phase in an equal and random way, with the premise of 70 % of the database for training and 30 % for testing.

Furthermore, the tests performed with the SSDRB classifier were done in offline mode and considering $\phi = 1.2$ and all simulations were performed in the MATLAB 2017 environment running with an Intel Core i7-3537U CPU at 2.00GHz with 12GB DDR4 2700MHZ and in operational system Windows 10 64-bit.

For the representation of possible occurrences of defects in the acquired images and in order to challenge the proposed models, we added three different intensities of Additive white Gaussian noise (AWGN) in all dataset, PSNR = 20 dB, PSNR = 6 dB and PSNR = 3 dB as shown in Figure 6 and 7. The peak signal to noise ratio (PSNR) [35] is the most used parameter to measure a corrupted image quality when compared

to the original one [36]. The three different intensities adopted values for the PSNR is enough to report problems that can occur due some factors, such as dirt in the lens of the equipment responsible for acquiring the images [37, 38, 39, 40]

Such parameter is the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. It is computationally lightweight and is usually expressed in terms of the logarithmic decibel scale. There is an inverse relationship between PSNR and MSE. So, the higher the value of PSNR indicates the best image quality. Considering a noise-free $m\,x\,n$ monochrome image I and its noisy approximation K, we can calculate the PSNR from a corrupted image as follows:

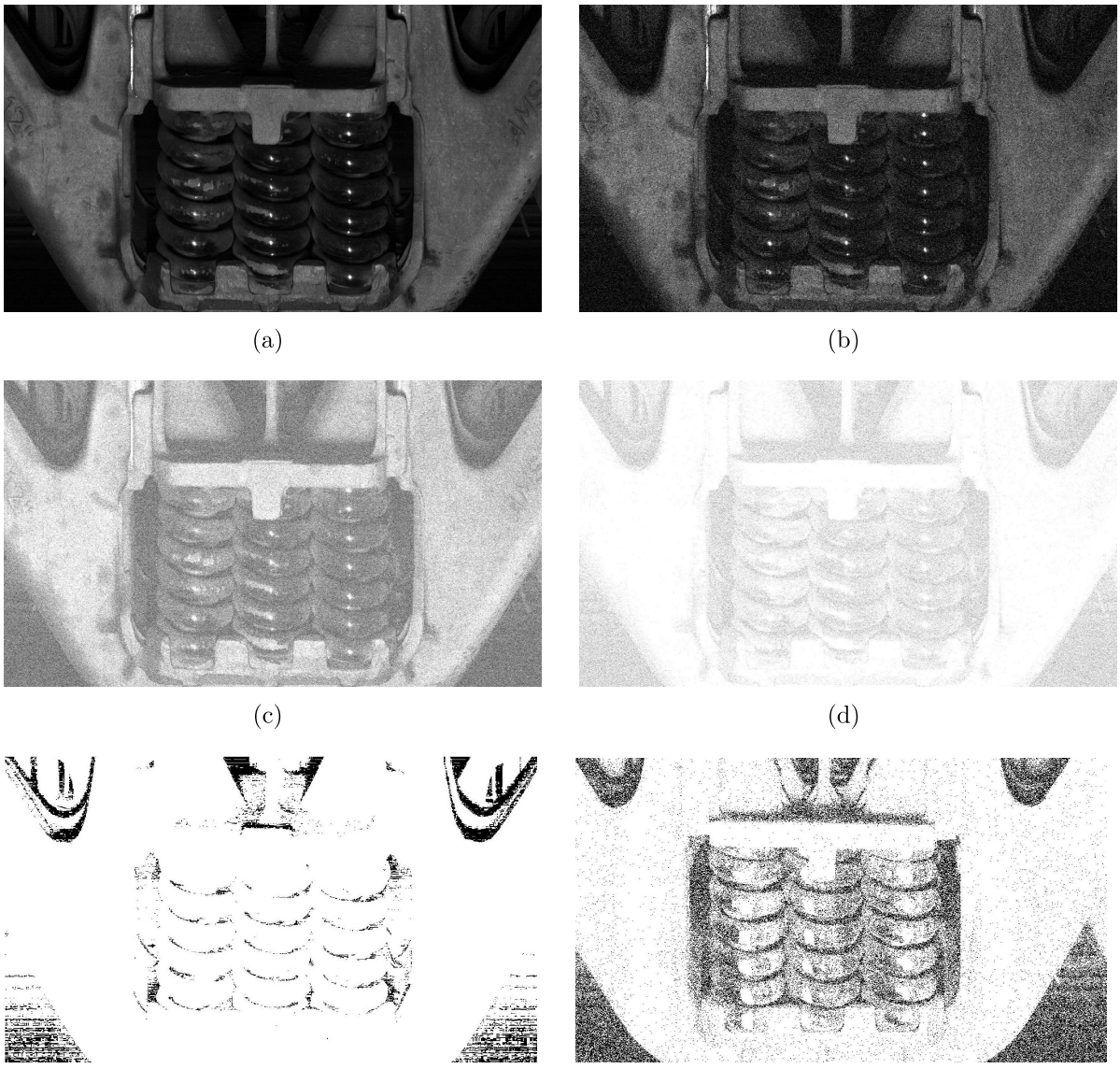
$$PSNR = 10\log_{10}(\frac{255^2}{MSE}) \tag{4.1}$$

where

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2. \tag{4.2}$$

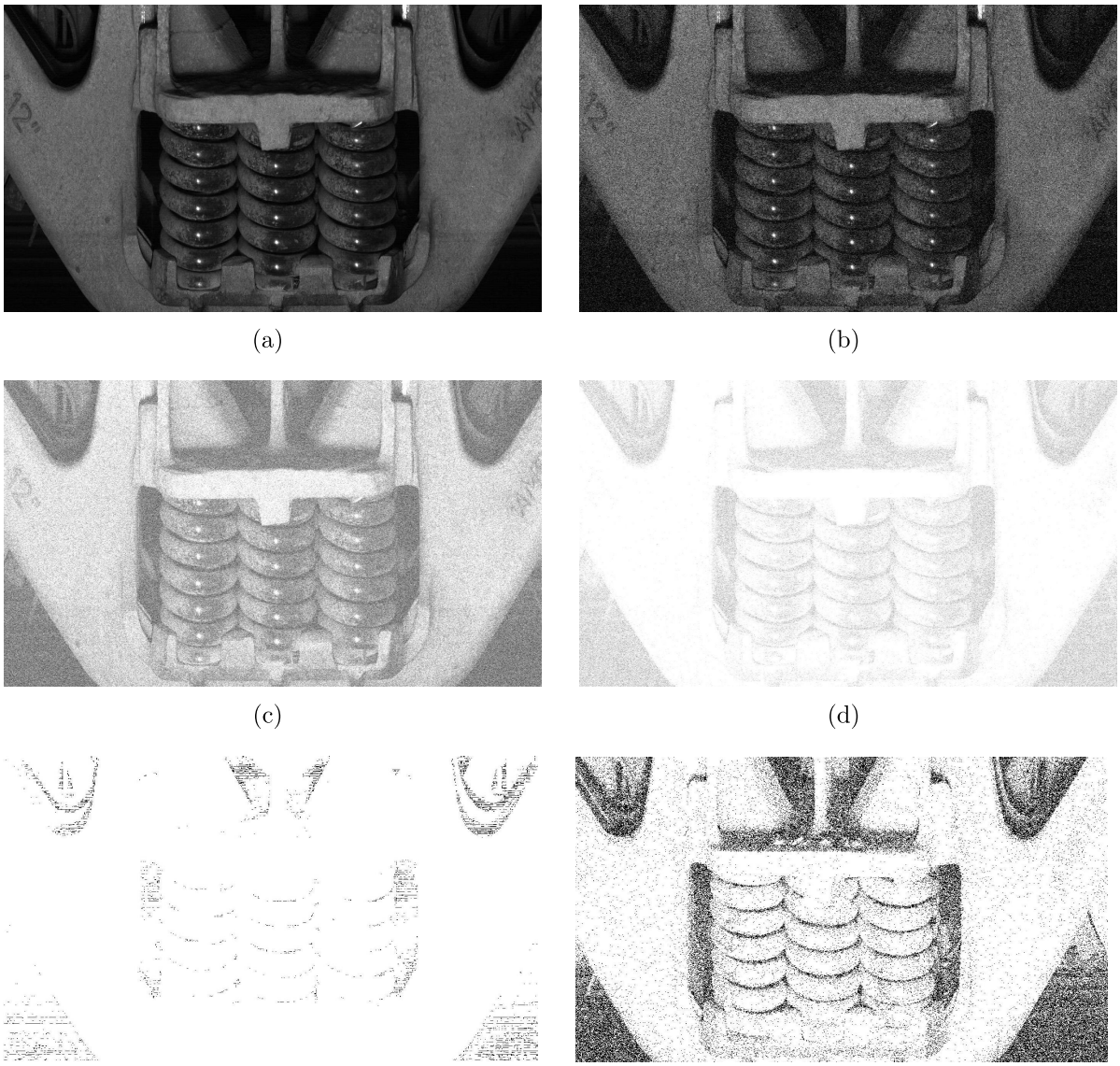
In Eq.4.1, the elements of the matrix are represented by using linear pulse-code modulation (PCM) [41] with B bits per sample, where the maximum is $2^B - 1$. Therefore, the value 255^2 denotes the maximum possible pixel value of the image, due to the fact of the pixels in this work are represented using 8 bits per element.

In practical applications, the measurement noise can be heavy-tailed non-Gaussian noise [42, 43, 44]. Therefore, as propose of white Gaussian noise applied in the original dataset, Cauchy noise [45] and the Laplace noise [46] were added in all dataset in order to evaluate the classifiers and in addition to represent possible of non-Gaussian variations problems on image acquisition. The Cauchy noise where applied with location parameter $x_0 = 0$ and scale parameter $\gamma = 1$. The Laplace noise was added with location parameter $\mu = 0$ and scale parameter $b = 35$. The corrupted images result of which noise is shown in Figure 6 and 7.



(a) (b) (c) (d) (e) (f)

Figura 6 – Demonstrative image considering Bogie springs without defect. a) Original Image b) Image corrupted with AWGN - PSNR = 20 dB, c) Image corrupted with AWGN - PSNR = 6 dB, d) Image corrupted with AWGN - PSNR = 3 dB, e) Image corrupted with Cauchy noise and f) Image corrupted with Laplace noise.



(a) (b)

(c) (d)

(e) (f)

Figura 7 – Demonstrative image considering Bogie springs with bad condition. a) Original Image b) Image corrupted with AWGN - PSNR = 20 dB, c) Image corrupted with AWGN - PSNR = 6 dB, d) Image corrupted with AWGN - PSNR = 3 dB, e) Image corrupted with Cauchy noise and f) Image corrupted with Laplace noise.

4.2 PERFORMANCE ANALYSIS

To evaluate the performance of the SSDRB classifier with cosine, euclidean, minkowski, manhattan, chebyshev distance measures, the results were compared with the following classifiers published in the literature: KNN [47], Linear SVM [48], Decision Tree [49], Random Forest [50], Artificial Neural Network [51], AdaBoost [52], Naive Bayes [53] and QDA [54].

Each classifier published in the literature was implemented through the Scikit-learn package and their default parameters as presented below [55]:

- KNN

  n_neighbors=5, *, weights='uniform', algorithm='auto', leaf_size=30, p=2, metric='minkowski', metric_params=None and n_jobs=None

- Linear SVM

  penalty='l2', loss='squared_hinge', *, dual=True, tol=0.0001, C=1.0, multi_class='ovr', fit_intercept=True, intercept_scaling=1, class_weight=None, verbose=0, random_state=None and max_iter=1000

- Decision Tree

  *, criterion='gini', splitter='best', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features=None, random_state=None, max_leaf_nodes=None, min_impurity_decrease=0.0, class_weight=None and ccp_alpha=0.0

- Random Forest

  n_estimators=100, *, criterion='gini', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=None, min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None, random_state=None, verbose=0, warm_start=False, class_weight=None, ccp_alpha=0.0 and max_samples=None

- Artificial Neural Network

  hidden_layer_sizes=(100,), activation='relu', *, solver='adam', alpha=0.0001, batch_size='auto', learning_rate='constant', learning_rate_init=0.001, power_t=0.5, max_iter=200, shuffle=True, random_state=None, tol=0.0001, verbose=False, warm_start=False, momentum=0.9, nesterovs_momentum=True, early_stopping=False, validation_fraction=0.1, beta_1=0.9, beta_2=0.999, epsilon=1e-08, n_iter_no_change=10 and max_fun=15000

- AdaBoost

  base_estimator=None, *, n_estimators=50, learning_rate=1.0, algorithm='SAMME.R' and random_state=None

- Naive Bayes

  *, alpha=1.0, fit_prior=True, class_prior=None and min_categories=None

- QDA

  *, priors=None, reg_param=0.0, store_covariance=False and tol=0.0001

The numerical results presented in Table 1, 2, 3, 4, 5 and 6 show the overall performance results for the Original dataset, images with white Gaussian noise (AWGN) PSNR of 20 dB, 6 dB and 3, images with Cauchy noise and images with Laplace noise, respectively. Four metrics were used to measure performance during the test phases: Accuracy, Mean Square Error (MSE), Cohen's kappa coefficient [56] and F-score [57].

Tabela 1 – Performance metrics for overall results - Original Dataset

Method	Test Accuracy (%)	Test MSE (%)	Test Kappa (%)	Test F-Score (%)	Training Time (s)	Test Time (s)
SSDRB (Cosine)	96.15 (± 1.81)	3.85 (± 1.81)	92.30 (± 3.62)	96.15 (± 1.77)	13.3 (± 2.8)	21.4 (± 3.2)
SSDRB (Euclidean)	96.80 (± 1.54)	3.20 (± 1.54)	93.61 (± 3.08)	96.74 (± 1.58)	21.7 (± 1.9)	39.7 (± 2.4)
SSDRB (Minkowski)	**97.26 (± 1.44)**	2.74 (± 1.44)	94.53 (± 2.87)	97.19 (± 1.51)	72.5 (± 14.3)	418.2 (± 71.7)
SSDRB (Manhattan)	97.15 (± 1.63)	1.63 (± 1.63)	94.30 (± 3.26)	97.06 (± 1.71)	32.2 (± 5.1)	127.5 (± 12.6)
SSDRB (Chebyshev)	95.93 (± 2.11)	4.07 (± 2.11)	91.85 (± 4.22)	95.91 (± 2.06)	8.4 (± 3.7)	10.1 (± 5.7)
KNN	95.15 (± 1.81)	4.85 (± 1.81)	90.30 (± 3.61)	95.14 (± 1.81)	2.8 (± 5.9)	27.5 (± 6.9)
Linear SVM	95.59 (± 1.96)	4.41 (± 1.96)	91.17 (± 3.92)	95.58 (± 1.96)	35.6 (± 7.4)	18.2 (± 6.0)
Decision Tree	90.10 (± 2.70)	9.90 (± 2.70)	80.20 (± 5.39)	90.08 (± 2.71)	227.3 (± 31.4)	2.8 (± 5.9)
Randon Forest	93.02 (± 2.49)	6.98 (± 2.49)	86.03 (± 4.98)	93.00 (± 2.50)	20.4 (± 7.4)	5.5 (± 7.4)
Neural Net	93.53 (± 4.24)	6.47 (± 4.24)	87.05 (± 8.49)	93.45 (± 4.62)	988.5 (± 383.4)	8.1 (± 7.7)
Adaboost	94.52 (± 2.22)	5.48 (± 2.22)	89.03 (± 4.44)	94.51 (± 2.22)	452.4 (± 121.5)	155.3 (± 9.2)
Naive Bayes	94.48 (± 2.01)	5.52 (± 2.01)	88.96 (± 4.02)	94.46 (± 2.02)	17.8 (± 5.4)	23.2 (± 7.8)
QDA	66.44 (± 7.41)	33.56 (± 7.41)	32.88 (± 14.82)	66.20 (± 7.42)	187.3 (± 12.5)	27.8 (± 4.6)

Tabela 2 – Performance metrics for overall results - White Gaussian Noise (AWGN) - PSNR of 20 dB

Method	Test Accuracy (%)	Test MSE (%)	Test Kappa (%)	Test F-Score (%)	Training Time (s)	Test Time (s)
SSDRB (Cosine)	95.53 (± 2.34)	4.47 (± 2.34)	91.05 (± 4.69)	95.54 (± 2.26)	16.0 (± 3.6)	24.1 (± 4.5)
SSDRB (Euclidean)	96.56 (± 1.65)	3.44 (± 1.65)	93.12 (± 3.31)	96.51 (± 1.68)	21.1 (± 2.2)	39.1 (± 2.9)
SSDRB (Minkowski)	**96.91 (± 1.58)**	3.09 (± 1.58)	93.81 (± 3.17)	96.84 (± 1.64)	78.0 (± 15.6)	446.9 (± 74.6)
SSDRB (Manhattan)	96.88 (± 1.70)	3.12 (± 1.70)	93.76 (± 3.40)	96.81 (± 1.75)	28.7 (± 2.0)	118.9 (± 5.25)
SSDRB (Chebyshev)	95.28 (± 2.43)	2.43 (± 4.72)	90.55 (± 4.85)	95.33 (± 2.31)	8.6 (± 3.60)	10.1 (± 2.8)
KNN	95.19 (± 1.75)	4.81 (± 1.75)	90.38 (± 3.49)	95.18 (± 1.76)	2.5 (± 5.6)	28.1 (± 6.7)
Linear SVM	95.22 (± 1.81)	4.78 (± 1.81)	90.45 (± 3.62)	95.22 (± 1.81)	34.8 (± 7.1)	17.8 (± 5.5)
Decision Tree	90.16 (± 2.93)	9.84 (± 2.93)	80.32 (± 5.86)	90.14 (± 2.94)	225.2 (± 26.0)	3.2 (± 6.3)
Randon Forest	92.50 (± 2.38)	7.50 (± 2.38)	85.00 (± 4.76)	92.49 (± 2.39)	20.0 (± 7.2)	6.0 (± 7.6)
Neural Net	93.65 (± 4.22)	6.35 (± 4.22)	87.29 (± 8.43)	93.55 (± 4.86)	966.7 (± 331.0)	8.5 (± 7.7)
Adaboost	95.18 (± 1.95)	4.82 (± 1.95)	90.37 (± 3.90)	95.18 (± 1.95)	451.7 (± 194.1)	151.6 (± 17.6)
Naive Bayes	94.45 (± 1.91)	5.55 (± 1.91)	88.90 (± 3.82)	94.44 (± 1.92)	16.6 (± 3.7)	21.1 (± 7.6)
QDA	64.7 (± 7.08)	35.30 (± 7.08)	29.39 (± 14.15)	64.48 (± 7.14)	178.2 (± 10.8)	27.4 (± 5.3)

Tabela 3 – Performance metrics for overall results - White Gaussian Noise (AWGN) - PSNR of 6 dB

Method	Test Accuracy (%)	Test MSE (%)	Test Kappa (%)	Test F-Score (%)	Training Time (s)	Test Time (s)
SSDRB (Cosine)	96.22 (± 1.95)	3.78 (± 1.95)	92.43 (± 3.91)	96.18 (± 1.95)	15.3 (± 2.8)	23.8 (± 4.2)
SSDRB (Euclidean)	96.56 (± 1.58)	3.44 (± 1.58)	93.12 (± 3.17)	96.52 (± 1.59)	23.4 (± 3.4)	42.0 (± 3.8)
SSDRB (Minkowski)	**96.92 (± 1.53)**	3.08 (± 1.53)	93.84 (± 3.06)	96.85 (± 1.58)	75.5 (± 15.1)	434.2 (± 75.8)
SSDRB (Manhattan)	96.84 (± 1.57)	3.16 (± 1.57)	93.68 (± 3.14)	96.77 (± 1.63)	36.2 (± 10.6)	142.5 (± 25.2)
SSDRB (Chebyshev)	95.75 (± 2.12)	4.25 (± 2.12)	91.50 (± 4.24)	95.75 (± 2.09)	8.7 (± 1.3)	10.3 (± 1.2)
KNN	94.64 (± 1.98)	5.36 (± 1.98)	89.28 (± 3.96)	94.63 (± 2.00)	2.2 (± 5.3)	29.0 (± 6.7)
Linear SVM	94.53 (± 2.10)	5.47 (± 2.10)	89.07 (± 4.19)	94.53 (± 2.10)	36.9 (± 7.8)	18.6 (± 6.1)
Decision Tree	88.72 (± 3.02)	11.28 (± 3.02)	77.43 (± 6.05)	88.70 (± 3.04)	233.1 (± 29.5)	3.5 (± 6.5)
Randon Forest	91.07 (± 2.60)	8.93 (± 2.60)	82.13 (± 5.19)	91.04 (± 2.61)	20.1 (± 7.0)	5.88 (± 7.5)
Neural Net	93.18 (± 4.16)	6.82 (± 4.16)	86.36 (± 8.33)	93.11 (± 4.37)	940.7 (± 293.5)	7.7 (± 7.8)
Adaboost	94.09 (± 2.19)	5.91 (± 2.19)	88.18 (± 4.37)	94.08 (± 2.19)	456.5 (± 266.9)	155.2 (± 11.2)
Naive Bayes	90.96 (± 2.42)	9.04 (± 2.42)	81.91 (± 4.84)	90.94 (± 2.43)	18.1 (± 5.7)	22.4 (± 7.8)
QDA	64.20 (± 7.39)	35.80 (± 7.39)	28.39 (± 14.78)	63.94 (± 7.48)	179.3 (± 14.3)	26.3 (± 7.2)

Tabela 4 – Performance metrics for overall results - White Gaussian Noise (AWGN) - PSNR of 3 dB

Method	Test Accuracy (%)	Test MSE (%)	Test Kappa (%)	Test F-Score (%)	Training Time (s)	Test Time (s)
SSDRB (Cosine)	96.53 (± 2.22)	3.47 (± 2.22)	93.05 (± 4.45)	96.53 (± 2.16)	12.3 (± 1.3)	19.2 (± 1.56)
SSDRB (Euclidean)	96.53 (± 1.67)	3.47 (± 1.67)	93.05 (± 3.35)	96.48 (± 1.71)	21.7 (± 2.8)	40.2 (± 3.1)
SSDRB (Minkowski)	96.86 (± 1.66)	3.14 (± 1.166)	93.72 (± 3.32)	96.80 (± 1.71)	85.9 (± 13.1)	478.2 (± 65.2)
SSDRB (Manhattan)	96.68 (± 1.60)	3.32 (± 1.60)	93.35 (± 3.19)	96.63 (± 1.64)	41.1 (± 2.5)	150.3 (± 5.2)
SSDRB (Chebyshev)	96.49 (± 2.23)	3.51 (± 2.23)	92.97 (± 4.47)	96.47 (± 2.20)	10.4 (± 2.9)	11.9 (± 2.8)
KNN	94.93 (± 2.07)	5.07 (± 2.07)	89.87 (± 4.14)	94.92 (± 2.08)	2.3 (± 5.4)	28.0 (± 6.3)
Linear SVM	**97.10 (± 1.47)**	2.90 (± 1.47)	94.20 (± 2.93)	97.10 (± 1.47)	33.2 (± 5.8)	16.2 (± 3.7)
Decision Tree	92.29 (± 2.78)	7.71 (± 2.78)	84.59 (± 5.57)	92.28 (± 2.79)	255.5 (± 24.7)	2.8 (± 5.9)
Randon Forest	92.80 (± 2.35)	7.20 (± 2.35)	85.61 (± 4.70)	92.78 (± 2.37)	19.78 (± 7.1)	6.4 (± 7.6)
Neural Net	94.63 (± 4.82)	5.38 (± 4.82)	89.25 (± 9.65)	94.51 (± 5.66)	927.3 (± 381.3)	9.0 (± 7.6)
Adaboost	95.87 (± 1.99)	4.13 (± 1.99)	91.74 (± 3.97)	95.86 (± 1.99)	450.9 (± 136.7)	155.4 (± 9.4)
Naive Bayes	93.86 (± 2.13)	6.14 (± 2.13)	87.73 (± 4.26)	93.83 (± 2.16)	17.8 (± 5.4)	20.1 (± 7.0)
QDA	67.10 (± 6.39)	32.90 (± 6.39)	34.20 (± 12.78)	66.86 (± 6.41)	179.4 (± 10.8)	26.9 (± 5.3)

Tabela 5 – Performance metrics for overall results - Cauchy noise

Method	Test Accuracy (%)	Test MSE (%)	Test Kappa (%)	Test F-Score (%)	Training Time (s)	Test Time (s)
SSDRB (Cosine)	95.75 (± 2.19)	4.25 (± 2.19)	91.50 (± 4.39)	95.72 (± 2.19)	14.2 (± 1.9)	22.4 (± 2.5)
SSDRB (Euclidean)	96.05 (± 1.88)	3.95 (± 1.88)	92.11 (± 3.77)	96.00 (± 1.91)	20.4 (± 5.7)	37.4 (± 3.1)
SSDRB (Minkowski)	**96.99 (± 1.69)**	3.01 (± 1.69)	93.99 (± 3.39)	96.89 (± 1.80)	66.5 (± 7.1)	389.2 (± 28.8)
SSDRB (Manhattan)	96.41 (± 1.82)	3.59 (± 1.82)	92.82 (± 3.63)	96.36 (± 1.85)	29.2 (± 6.9)	118.8 (± 8.3)
SSDRB (Chebyshev)	95.51 (± 2.24)	4.49 (± 2.24)	91.01 (± 4.48)	95.50 (± 2.17)	8.1 (± 1.2)	9.5 (± 1.4)
KNN	94.45 (± 2.32)	5.54 (± 2.32)	88.91 (± 4.65)	94.44 (± 2.33)	1.7 (± 4.9)	29.5 (± 6.4)
Linear SVM	95.16 (± 2.17)	4.83 (± 2.17)	90.33 (± 4.35)	95.16 (± 2.18)	37.8 (± 7.8)	19.4 (± 7.2)
Decision Tree	90.88 (± 3.21)	9.11 (± 3.21)	81.77 (± 6.43)	90.87 (± 3.22)	222.3 (± 30.4)	3.1 (± 6.2)
Randon Forest	92.38 (± 2.45)	7.61 (± 2.45)	84.77 (± 4.90)	92.36 (± 2.46)	21.33 (± 7.7)	6.2 (± 7.7)
Neural Net	92.72 (± 4.53)	7.27 (± 4.53)	85.45 (± 9.06)	92.63 (± 4.79)	937.3 (± 383.9)	7.9 (± 7.8)
Adaboost	94.29 (± 2.23)	5.71 (± 2.23)	88.57 (± 4.47)	94.27 (± 2.45)	453.8 (± 155.8)	154.1 (± 11.2)
Naive Bayes	93.28 (± 2.23)	6.71 (± 2.23)	86.57 (± 4.46)	93.27 (± 2.24)	17.3 (± 4.8)	21.2 (± 7.5)
QDA	65.09 (± 7.43)	34.90 (± 7.43)	30.19 (± 14.87)	64.86 (± 7.44)	177.3 (± 11.2)	24.6 (± 5.9)

Tabela 6 – Performance metrics for overall results - Laplace noise

Method	Test Accuracy (%)	Test MSE (%)	Test Kappa (%)	Test F-Score (%)	Training Time (s)	Test Time (s)
SSDRB (Cosine)	95.36 (± 2.40)	4.64 (± 2.40)	90.73 (± 4.80)	95.40 (± 2.28)	13.1 (± 1.4)	21.9 (± 1.6)
SSDRB (Euclidean)	96.11 (± 1.83)	3.89 (± 1.83)	92.23 (± 3.65)	96.07 (± 1.83)	20.6 (± 1.9)	38.9 (± 2.4)
SSDRB (Minkowski)	**96.13 (± 1.67)**	3.87 (± 1.67)	92.26 (± 3.34)	96.07 (± 1.70)	62.8 (± 4.1)	366.4 (± 16.6)
SSDRB (Manhattan)	**96.13 (± 1.76)**	3.84 (± 1.76)	92.32 (± 3.52)	96.11 (± 1.80)	30.4 (± 18.3)	122.7 (± 35.4)
SSDRB (Chebyshev)	95.34 (± 2.25)	4.66 (± 2.25)	90.68 (± 4.49)	95.35 (± 2.17)	8.1 (± 1.4)	9.7 (± 1.3)
KNN	94.19 (± 2.07)	5.81 (± 2.07)	88.38 (± 4.13)	94.17 (± 2.08)	2.3 (± 5.5)	28.5 (± 6.5)
Linear SVM	93.95 (± 2.06)	6.04 (± 2.06)	87.91 (± 4.13)	94.17 (± 2.08)	36.6 (± 7.9)	18.3 (± 5.9)
Decision Tree	90.45 (± 3.48)	9.54 (± 3.48)	80.90 (± 6.97)	90.43 (± 3.49)	227.4 (± 28.5)	2.7 (± 5.9)
Randon Forest	91.63 (± 2.65)	8.36 (± 2.65)	83.27 (± 5.31)	91.61 (± 2.67)	19.5 (± 6.9)	7.0 (± 7.7)
Neural Net	93.19 (± 3.65)	6.80 (± 3.65)	86.38 (± 7.30)	93.13 (± 3.84)	826.1 (± 197.5)	6.9 (± 7.7)
Adaboost	94.30 (± 2.14)	5.69 (± 2.14)	88.61 (± 4.27)	94.29 (± 2.14)	454.2 (± 173.6)	152.7 (± 10.1)
Naive Bayes	93.50 (± 2.08)	6.49 (± 2.08)	87.01 (± 4.14)	93.48 (± 2.09)	16.7 (± 3.9)	21.8 (± 7.6)
QDA	62.29 (± 6.71)	37.70 (± 6.72)	24.58 (± 13.44)	62.05 (± 6.77)	180.6 (± 14.2)	26.6 (± 4.9)

4.3 STATISTICAL ANALYSIS

The accuracy metric results in Table 1, 2, 3, 4, 5, and 6 was evaluate by Shapiro–Wilk test of normality where the data are normally distributed and the null hypothesis is not rejected when the p-value greater than 0.05 [58]. As present in the Table 7 the results show the data are normally distributed for overall accuracy metric results, then the statistical analysis was performed as in [59].

Therefore the two-sample t-test was applied for the test accuracy metric in order to evaluate the results in Table 1, 2, 3, 4, 5, and 6. Considering two sets of samples $\mathcal{G}_1$ and $\mathcal{G}_2$, the two-sample t-test allows us to infer assumptions from two independent data samples and to verify their statistical validity. This statistical test is expressed as:

$$\tau = \frac{\overline{\mathcal{G}_1} - \overline{\mathcal{G}_2}}{\sqrt{\frac{\sigma^2_{\mathcal{G}_1}}{L_{\mathcal{G}_1}} + \frac{\sigma^2_{\mathcal{G}_2}}{L_{\mathcal{G}_2}}}} \tag{4.3}$$

where $\overline{\mathcal{G}_1}$, $\overline{\mathcal{G}_2}$, $\sigma^2_{\mathcal{G}_1}$ and $\sigma^2_{\mathcal{G}_2}$ are the means and standard deviations values of the samples belonging to $\mathcal{G}_1$ and $\mathcal{G}_2$, respectively. Also, $L_{\mathcal{G}_1} = \#\{\mathcal{G}_1\}$ and $L_{\mathcal{G}_2} = \#\{\mathcal{G}_2\}$, in which $\#$ denotes the Cardinality operator. The degree of freedom is defined as $L_{\mathcal{G}_1} + L_{\mathcal{G}_1} - 2$. In addition to the determination of τ, it becomes important to infer the hypothesis, which are given by:

$$\begin{cases} \mathcal{H}_0 : \overline{\mathcal{G}_1} = \overline{\mathcal{G}_2} \\ \mathcal{H}_1 : \overline{\mathcal{G}_1} \neq \overline{\mathcal{G}_2} \end{cases} \tag{4.4}$$

Given a significance level α, usually around 0.05, the p-value is calculated from τ and represents the lowest value of α to reject the null hypothesis ($\mathcal{H}_0$). Thus, values of the p-value below α means that the null hypothesis is not true [60].

The sets of samples $\mathcal{G}_1$ and $\mathcal{G}_2$ of Eq. (4.3) are the test accuracy metric obtained for the 200 times. SSDRB classifier with minkowski distance presents best SSDRB average accuracy results in which scenario listed in table 1, 2, 3, 4, 5 and 6. Then, $\mathcal{G}_1$ refers to SSDRB classifier with minkowski distance and $\mathcal{G}_2$ can denote which other SSDRB with others distance measures and others classifier methods, as listed in Table 8, 9, 10 and 11. The degree of freedom presented in these statistical tests is 64, which is relatively high, so there is no need to verify the normality of error distributions [60].

Table 8 presents the results of the two-sample t-test performed for the test accuracy metric in Table 1 and the results of the t-test for Table 2, 3 and 4 are presented in Table 9. In addition, the Table 10 and Table 11 presents the results of the two-sample t-test performed for the test accuracy metric in Table 5 and 6, respectively. The rejection of the null hypothesis are indicated by the letters ‘W’ and ‘L’ representing respectively the wins and the losses of the method tested. Meanwhile the acceptance of the null hypothesis is described by ‘E’ which means equality of the tested methods.

Tabela 7 – p-value from Shapiro–Wilk test for the accuracy metric for overall results

Method	Original Dataset	AWGN - PSNR 20 dB	AWGN - PSNR 6 dB	AWGN - PSNR 3 dB	Cauchy noise	Laplace noise
SSDRB (Cosine)	0.29	0.61	0.55	0.64	0.68	0.24
SSDRB (Euclidean)	0.50	0.56	0.48	0.52	0.12	0.42
SSDRB (Minkowski)	0.31	0.14	0.28	0.65	0.18	0.17
SSDRB (Manhattan)	0.31	0.17	0.23	0.63	0.16	0.11
SSDRB (Chebyshev)	0.74	0.46	0.59	0.38	0.17	0.29
KNN	0.48	0.65	0.38	0.54	0.58	0.48
Linear SVM	0.30	0.59	0.37	0.61	0.49	0.36
Decision Tree	0.22	0.25	0.49	0.43	0.11	0.35
Randon Forest	0.24	0.54	0.56	0.6	0.09	0.09
Neural Net	0.29	0.59	0.42	0.72	0.48	0.27
Adaboost	0.44	0.38	0.7	0.34	0.48	0.32
Naive Bayes	0.63	0.52	0.54	0.52	0.56	0.16
QDA	0.60	0.29	0.61	0.76	0.48	0.56

Tabela 8 – Statistical analyses performed by two-sample t-test for the test accuracy metric for overall results - Original Dataset

Dataset	$\mathcal{G}_1$	$\mathcal{G}_2$	p-value	Lower boundary	Upper boundary	$\mathcal{H}_0$
Original Dataset	SSDRB (Minkowski)	SSDRB (Cosine)	0.00	0.79	1.44	W
	SSDRB (Minkowski)	SSDRB (Euclidean)	0.00	0.17	0.75	W
	SSDRB (Minkowski)	SSDRB (Manhattan)	0.46	-0.19	0.42	E
	SSDRB (Minkowski)	SSDRB (Chebyshev)	0.00	0.98	1.69	W
	SSDRB (Minkowski)	KNN	0.00	1.79	2.43	W
	SSDRB (Minkowski)	LinearSVM	0.00	1.34	2.01	W
	SSDRB (Minkowski)	DecisionTree	0.00	6.41	7.28	W
	SSDRB (Minkowski)	RandomForest	0.00	3.96	4.76	W
	SSDRB (Minkowski)	NeuralNet	0.00	2.90	3.96	W
	SSDRB (Minkowski)	AdaBoost	0.00	2.38	3.10	W
	SSDRB (Minkowski)	NaiveBayes	0.00	2.44	3.13	W
	SSDRB (Minkowski)	QDA	0.00	29.77	31.88	W

Tabela 9 – Statistical analyses performed by two-sample t-test for the test accuracy metric for overall results - White Gaussian Noise (AWGN)

Dataset	$\mathcal{G}_1$	$\mathcal{G}_2$	p-value	Lower boundary	Upper boundary	$\mathcal{H}_0$
AWGN - PSNR = 20 dB	SSDRB (Minkowski)	SSDRB (Cosine)	0.00	0.99	1.77	W
	SSDRB (Minkowski)	SSDRB (Euclidean)	0.03	0.03	0.66	W
	SSDRB (Minkowski)	SSDRB (Manhattan)	0.87	-0.30	0.35	E
	SSDRB (Minkowski)	SSDRB (Chebyshev)	0.00	1.23	2.03	W
	SSDRB (Minkowski)	KNN	0.00	1.39	2.04	W
	SSDRB (Minkowski)	LinearSVM	0.00	1.35	2.02	W
	SSDRB (Minkowski)	DecisionTree	0.00	6.39	7.35	W
	SSDRB (Minkowski)	RandomForest	0.00	3.86	4.64	W
	SSDRB (Minkowski)	NeuralNet	0.00	2.52	3.56	W
	SSDRB (Minkowski)	AdaBoost	0.00	1.38	2.09	W
	SSDRB (Minkowski)	NaiveBayes	0.00	2.11	2.80	W
	SSDRB (Minkowski)	QDA	0.00	31.20	33.33	W
AWGN - PSNR = 6 dB	SSDRB (Minkowski)	SSDRB (Cosine)	0.00	0.36	1.05	W
	SSDRB (Minkowski)	SSDRB (Euclidean)	0.02	0.05	0.66	W
	SSDRB (Minkowski)	SSDRB (Manhattan)	0.60	-0.22	0.39	E
	SSDRB (Minkowski)	SSDRB (Chebyshev)	0.00	0.81	1.53	W
	SSDRB (Minkowski)	KNN	0.00	1.93	2.63	W
	SSDRB (Minkowski)	LinearSVM	0.00	2.03	2.75	W
	SSDRB (Minkowski)	DecisionTree	0.00	7.79	8.71	W
	SSDRB (Minkowski)	RandomForest	0.00	5.66	6.51	W
	SSDRB (Minkowski)	NeuralNet	0.00	3.54	5.00	W
	SSDRB (Minkowski)	AdaBoost	0.00	2.44	3.18	W
	SSDRB (Minkowski)	NaiveBayes	0.00	5.66	6.36	W
	SSDRB (Minkowski)	QDA	0.00	31.67	33.78	W
AWGN- PSNR = 3 dB	SSDRB (Minkowski)	SSDRB (Cosine)	0.09	-0.05	0.72	E
	SSDRB (Minkowski)	SSDRB (Euclidean)	0.04	0.00	0.66	W
	SSDRB (Minkowski)	SSDRB (Manhattan)	0.26	-0.14	0.50	E
	SSDRB (Minkowski)	SSDRB (Chebyshev)	0.06	-0.02	0.76	E
	SSDRB (Minkowski)	KNN	0.00	1.55	2.29	W
	SSDRB (Minkowski)	LinearSVM	0.13	-0.55	0.07	E
	SSDRB (Minkowski)	DecisionTree	0.00	4.16	5.09	W
	SSDRB (Minkowski)	RandomForest	0.00	3.52	4.36	W
	SSDRB (Minkowski)	NeuralNet	0.00	1.20	2.48	W
	SSDRB (Minkowski)	AdaBoost	0.00	0.58	1.30	W
	SSDRB (Minkowski)	NaiveBayes	0.00	2.62	3.37	W
	SSDRB (Minkowski)	QDA	0.00	28.84	30.68	W

Tabela 10 – Statistical analyses performed by two-sample t-test for the test accuracy metric for overall results - Cauchy noise

Dataset	$\mathcal{G}_1$	$\mathcal{G}_2$	p-value	Lower boundary	Upper boundary	$\mathcal{H}_0$
Cauchy noise	SSDRB (Minkowski)	SSDRB (Cosine)	0.00	0.50	1.78	W
	SSDRB (Minkowski)	SSDRB (Euclidean)	0.24	-0.13	0.51	E
	SSDRB (Minkowski)	SSDRB (Manhattan)	0.35	-0.48	0.17	E
	SSDRB (Minkowski)	SSDRB (Chebyshev)	0.00	0.69	1.44	W
	SSDRB (Minkowski)	KNN	0.00	2.14	2.94	W
	SSDRB (Minkowski)	LinearSVM	0.00	1.44	2.21	W
	SSDRB (Minkowski)	DecisionTree	0.00	5.60	6.61	W
	SSDRB (Minkowski)	RandomForest	0.00	4.19	5.02	W
	SSDRB (Minkowski)	NeuralNet	0.00	3.59	4.94	W
	SSDRB (Minkowski)	AdaBoost	0.00	2.31	3.1	W
	SSDRB (Minkowski)	NaiveBayes	0.00	3.31	4.1	W
	SSDRB (Minkowski)	QDA	0.00	30.83	32.96	W

Tabela 11 – Statistical analyses performed by two-sample t-test for the test accuracy metric for overall results - Laplace noise

Dataset	$\mathcal{G}_1$	$\mathcal{G}_2$	p-value	Lower boundary	Upper boundary	$\mathcal{H}_0$
Laplace noise	SSDRB (Minkowski)	SSDRB (Cosine)	0.00	0.36	1.17	W
	SSDRB (Minkowski)	SSDRB (Euclidean)	0.94	-0.33	0.36	E
	SSDRB (Minkowski)	SSDRB (Manhattan)	0.98	-0.34	0.33	E
	SSDRB (Minkowski)	SSDRB (Chebyshev)	0.00	0.40	1.18	W
	SSDRB (Minkowski)	KNN	0.00	1.57	2.31	W
	SSDRB (Minkowski)	LinearSVM	0.00	1.80	2.54	W
	SSDRB (Minkowski)	DecisionTree	0.00	5.14	6.22	W
	SSDRB (Minkowski)	RandomForest	0.00	4.06	4.93	W
	SSDRB (Minkowski)	NeuralNet	0.00	2.38	3.50	W
	SSDRB (Minkowski)	AdaBoost	0.00	1.45	2.20	W
	SSDRB (Minkowski)	NaiveBayes	0.00	2.25	2.99	W
	SSDRB (Minkowski)	QDA	0.00	32.87	34.80	W

4.4 DISCUSSION OF RESULTS

As shown in Table 1, the SSDRB classifier with the minkowski distance metric presents the best average accuracy result. Nevertheless, it also presents the lowest standard deviation for the accuracy results and, in addition, the best result for the kappa test and F-score. However, the training time for the SSDRB classifier with the minkowski distance metric is the longest among the SSDRB classifiers with each distance metric while the testing time is the longest among all classifiers.

Nevertheless, it can be noted that these particularities of results are repeated for Table 2, 3, 5, 6 in which the images were corrupted with PSNR $20dB$, PSNR $6dB$, Laplace and Cauchy, respectively. That is, the SSDRB classifier with the minkowski distance metric presents the best result of average accuracy, kappa test and F-score, but with a high computational cost given the high training time and test time.

As shown in Table 4, despite maintaining an average accuracy of the SSDBR classifier above 96%, the Linear SVM classifier presented a better average accuracy result, lower standard deviation for the accuracy results and better result for the kappa test and F -score. In addition, the training and testing time with the presented Linear SVM classifier are significantly lower when compared to SSDRB with the minkowski distance metric, which is the SSDRB metric with the best average accuracy for corrupted images with PSNR $3dB$.

Furthermore, the results presented in Tables 1, 2, 3, 4, 5 and 6 show that the SSDRB classifier with the manhattan distance metric can be an excellent alternative given that the average accuracy is close to the results obtained with the minkowski metric and statistically equal as presented in Tables 8, 9, 10 and 11. In addition, as observed in all tables of results, the SSDRB classifier with the manhattan distance metric presents significantly lower training time and test time when compared to the SSDRB with the minkowski distance metric.

5 CONCLUSIONS

The work discusses the application of image processing techniques and computational intelligence by introducing the SSDRB classifier to the analysis of its results in the classification of wagon bogie springs. VGG19 was used for preprocessing the images, which proved to be very effective in extracting attributes from images due to its simple structure and better performance.

Compared to the classical classifiers, the SSDRB classifier showed higher accuracy in tests. The SSDRB with minkowski distance results turned out to be a great alternative since they presented the best result for the original dataset. Despite it was not presented the best result numerically for images with PSNR of $3dB$ where Linear SVM results showed higher accuracy, the results are statistically equal and the results of SSDRB has shown the maintenance of mean accuracy greater than 96% even to non-Gaussian noises.

Futhermore, in Tables 8, 9, 10 and 11 it is showed equality of the tested methods SSDRB minkowski distance and SSDRB manhattan distance. Then, the manhattan distance results ended up being a good choice as it features a shorter training time and shorter testing time compared to the SSDRB with minkowski distance.

In addition, the model presented in this paper has other advantages that are not covered by classical methods, such as a learning process that is easy to interpret by a specialist, online or offline training, the ability to classify images outside the sample and ability to deal with uncertainty.

Detecting and classifying the condition of bogie springs can be done in the present context, generating a significant reduction in cost and time. It is therefore worth mentioning that the use of intelligent systems can support decision-making processes, increasing the flexibility and efficiency of the process.

Intelligent systems can assist in decision-making processes, bringing more agility and efficiency to the process. In this respect, the SSDRB classifier is an attractive alternative to quickly and efficiently diagnosing and classifying the condition of the wagon bogie springs, reducing costs and time spent on inspection.

The model discussed in this paper is limited in the definition of meta-parameters such as the ϕ of the semi-supervisor DRB and h of minkowski distance. Therefore, as future work it is intended to optimally define the meta-parameters using the effective method reported in literature like presented in [61]. In addition, we intend to improve the process by researching more efficient techniques for pre-processing image classification and to explore the implementation of dataset augmentation in order to increase the database and evaluate the results of the different SSDRB distance metrics.

REFERENCES

1 X. Gu, P. Angelov, Semi-supervised deep rule-based approach for image classification, Applied Soft Computing 68 (2018) 53–68. doi:10.1016/j.asoc.2018.03.032.

2 P. Angelov, X. Gu, Mice: Multi-layer multi-model images classifier ensemble, in: 2017 3rd IEEE International Conference on Cybernetics (CYBCONF), 2017, pp. 1–8. doi:10.1109/CYBConf.2017.7985788.

3 P. Angelov, X. Gu, A cascade of deep learning fuzzy rule-based image classifier and svm, in: 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2017, pp. 746–751. doi:10.1109/SMC.2017.8122697.

4 X. Gu, P. P. Angelov, C. Zhang, P. M. Atkinson, A massively parallel deep rule-based ensemble classifier for remote sensing scenes, IEEE Geoscience and Remote Sensing Letters 15 (3) (2018) 345–349. doi:10.1109/LGRS.2017.2787421.

5 P. P. Angelov, X. Gu, J. C. Príncipe, A generalized methodology for data analysis, IEEE Transactions on Cybernetics 48 (10) (2018) 2981–2993. doi:10.1109/TCYB.2017.2753880.

6 Y. Li, M. Zhang, C. Chen, A deep-learning intelligent system incorporating data augmentation for short-term voltage stability assessment of power systems, Applied Energy 308 (2022) 118347. doi:10.1016/j.apenergy.2021.118347.

7 N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines and Other Kernel-based Learning Methods, Cambridge University Press, 2000. doi:10.1017/CBO9780511801389.

8 Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–44. doi:10.1038/nature14539.

9 P. Angelov, Autonomous Learning Systems, Wiley Online Library, 2013.

10 J. P. P.P. Angelov, X. Gu, A generalized methodology for data analysis, IEEE Trans. Cybern. (2017). doi:10.1109/TCYB.2017.2753880.

11 M. Mateen, J. Wen, D. Nasrullah, S. Song, Z. Huang, Fundus image classification using vgg-19 architecture with pca and svd, Symmetry 11 (2018) 1–2. doi:10.3390/sym11010001.

12 Commuters face train delays for days (August 2017).
URL https://www.itv.com/news/anglia/2017-08-15/commuters-face-train-delays-for-days

13 Cost of freight train derailment could top £1 million (August 2017).
URL http://tiny.cc/52xouz

14 Derailed freight train near ely causes chaos in the east (August 2017).
URL https://www.bbc.com/news/uk-england-cambridgeshire-40935930

15 Removal of derailed train resumes (July 2007).
URL http://news.bbc.co.uk/2/hi/uk_news/england/cambridgeshire/6283186.stm

16 Freight wagons lifted from ely rail bridge (July 2007).
URL https://www.networkrailmediacentre.co.uk/news/freight-wagons-lifted-from-ely-rail-bridge

17 Construction of new ely rail bridge begins (October 2007).
URL https://www.networkrailmediacentre.co.uk/news/construction-of-new-ely-rail-bridge-begins

18 Crane moves in to remove derailed ely freight train (August 2017).
URL https://www.bbc.com/news/uk-england-cambridgeshire-40950072

19 Wagon lift on ely rail bridge begins (July 2007).
URL https://www.networkrailmediacentre.co.uk/resources/wagon-lift1

20 O setor ferroviário de carga brasileiro (2019).
URL https://www.antf.org.br/informacoes-gerais/

21 MRS Logística S.A., https://www.mrs.com.br, MRS logística S.A. wagon bogies images database, Database is not public available (2020).

22 K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556 (2014).

23 P. Angelov, R. Yager, A new type of simplified fuzzy rule-based system, International Journal of General Systems 41 (2) (2012) 163–185. doi:10.1080/03081079.2011.634807.

24 G.-S. Xia, J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, L. Zhang, X. Lu, Aid: A benchmark data set for performance evaluation of aerial scene classification, IEEE Transactions on Geoscience and Remote Sensing 55 (7) (2017) 3965–3981. doi:10.1109/TGRS.2017.2685945.

25 X. Gu, P. P. Angelov, D. Kangin, J. C. Príncipe, A new type of distance metric and its use for clustering, Evol. Syst. 8 (3) (2017) 167–177. doi:10.1007/s12530-017-9195-7.

26 M. e. a. SENOUSSAOUI, Efficient iterative mean shift based cosine dissimilarity for multi-recording speaker clustering, IEEE International Conference on Acoustics, Speech and Signal Processing (2013) 7712–7715doi:10.1109/ICASSP.2013.6639164.

27 A. K. D. A. AGGARWAL, Charu C.; HINNEBURG, On the surprising behavior of distance metrics in high dimensional space., International conference on database theory (2001) 420–434doi:10.1007/3-540-44503-X_27.

28 K. e. a. BEYER, When is "nearest neighbor" meaningful?, International conference on database theory (1999) 217–235doi:10.1007/3-540-49257-7_15.

29 X. Gu, P. Angelov, Z. Zhao, A distance-type-insensitive clustering approach, Applied Soft Computing 77 (2019) 622–634. doi:https://doi.org/10.1016/j.asoc.2019.01.028.

30 X. e. a. GU, Self-organised direction aware data partitioning algorithm, Information Sciences 423 (2018) 80–95. doi:10.1016/j.ins.2017.09.025.

31 X. ANGELOV, Plamen P.; GU, Empirical Approach to Machine Learning, Springer, Cham, Switzerland, 2019. doi:10.1007/978-3-030-02384-3.

32 P. P. Angelov, X. Gu, Deep rule-based classifier with human-level performance and characteristics, Inf. Sci. 463-464 (2018) 196–213. doi:10.1016/j.ins.2018.06.048.

33 MRS Logística S.A., Available in: https://www.mrs.com.br. (2021).

34 Q.-S. Xu, Y.-Z. Liang, Monte carlo cross validation, Chemometrics and Intelligent Laboratory Systems 56 (1) (2001) 1–11. doi:https://doi.org/10.1016/S0169-7439(00)00122-2.
URL https://www.sciencedirect.com/science/article/pii/S0169743900001222

35 V. Pullano, A. Vanelli-Coralli, G. E. Corazza, PSNR evaluation and alignment recovery for mobile satellite video broadcasting, in: 6th Advanced Satellite Multimedia Systems Conference and 12th Signal Processing for Space Communications Workshop, ASMS/SPSC 2012, Vigo, Spain, September 5-7, 2012, 2012, pp. 176–181. doi:10.1109/ASMS-SPSC.2012.6333072.

36 T. Huang, W. Dong, X. Xie, G. Shi, X. Bai, Mixed noise removal via laplacian scale mixture modeling and nonlocal low-rank approximation, IEEE Transactions on Image Processing PP (2017) 1–1. doi:10.1109/TIP.2017.2676466.

37 R. Prasad, P. Kishore, Performance of active contour models in train rolling stock part segmentation on high-speed video data, Cogent Engineering 4 (01 2017). doi:10.1080/23311916.2017.1279367.

38 E. Pestana de Aguiar, T. Fernandes, F. Nogueira, D. Silveira, M. Vellasco, M. Ribeiro, A new model to distinguish railhead defects based on set-membership type-2 fuzzy logic system, International Journal of Fuzzy Systems 23 (10 2020). doi:10.1007/s40815-020-00945-3.

39 K. Kaja, R. Prasad, M. Prasad, Computer vision assistant for train rolling stock examination using level set models, ARPN Journal of Engineering and Applied Sciences 13 (2018) 8607–8624.

40 R. Deshpande, L. Ragha, S. Sharma, Video quality assessment through psnr estimation for different compression standards, Indonesian Journal of Electrical Engineering and Computer Science 11 (2018) 918–924. doi:10.11591/ijeecs.v11.i3.pp918-924.

41 R. R. S. Tomar, K. K. Jain, Lossless image compression using differential pulse code modulation and its purpose, 2015. doi:10.21742/IJSBT.2015.3.1.02.

42 L. Yan, C. Di, Q. J. Wu, Y. Xia, S. Liu, Distributed fusion estimation for multisensor systems with non-gaussian but heavy-tailed noises, ISA Transactions 101 (2020) 160–169. doi:10.1016/j.isatra.2020.02.004.

43 M. Ding, T.-Z. Huang, S. Wang, J.-J. Mei, X.-L. Zhao, Total variation with overlapping group sparsity for deblurring images under cauchy noise, Applied Mathematics and Computation 341 (2019) 128–147. doi:10.1016/j.amc.2018.08.014.

44 K. Shi, G. Dong, Z. Guo, Cauchy noise removal by nonlinear diffusion equations, Computers Mathematics with Applications 80 (9) (2020) 2090–2103. doi:10.1016/j.camwa.2020.08.027.

45 J.-H. Yang, X.-L. Zhao, J.-J. Mei, S. Wang, T.-H. Ma, T.-Z. Huang, Total variation and high-order total variation adaptive model for restoring blurred images with cauchy noise, Computers Mathematics with Applications 77 (5) (2019) 1255–1272. doi:10.1016/j.camwa.2018.11.003.

46 C. Boncelet, Chapter 7 - image noise models.

47 S. Tan, An effective refinement strategy for KNN text classifier, Expert Syst. Appl. 30 (2) (2006) 290–298. doi:10.1016/j.eswa.2005.07.019.

48 C. Hsieh, K. Chang, C. Lin, S. S. Keerthi, S. Sundararajan, A dual coordinate descent method for large-scale linear SVM, in: Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008, 2008, pp. 408–415. doi:10.1145/1390156.1390208.

49 M. Friedl, C. Brodley, Decision tree classification of land cover from remotely sensed data, Remote Sensing of Environment 61 (3) (1997) 399–409. doi:10.1016/S0034-4257(97)00049-7.

50 M. Pal, Random forest classifier for remote sensing classification, International Journal of Remote Sensing 26 (1) (2005) 217–222. doi:10.1080/01431160412331269698.

51 S. Dreiseitl, L. Ohno-Machado, Logistic regression and artificial neural network classification models: a methodology review, Journal of Biomedical Informatics 35 (5) (2002) 352–359. doi:10.1016/S1532-0464(03)00034-0.

52 T.-K. An, M.-H. Kim, A new diverse adaboost classifier, in: 2010 International Conference on Artificial Intelligence and Computational Intelligence, Vol. 1, 2010, pp. 359–363. doi:10.1109/AICI.2010.82.

53 I. Rish, An empirical study of the naive bayes classifier, in: IJCAI 2001 workshop on empirical methods in artificial intelligence, Vol. 3, IBM New York, 2001, pp. 41–46. doi:10.1.1.330.2788.

54 D. Menaka, L. P. Suresh, S. S. P. Kumar, Land cover classification of multispectral satellite images using qda classifier, in: 2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2014, pp. 1383–1386. doi:10.1109/ICCICCT.2014.6993178.

55 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, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

56 S. V. Stehman, Estimating the kappa coefficient and its variance under stratified random sampling, Photogrammetric Engineering and Remote Sensing 62 (1996) 401–407. doi:10.1.1.461.9979.

57 M. Sokolova, N. Japkowicz, S. Szpakowicz, Beyond accuracy, f-score and roc: A family of discriminant measures for performance evaluation, Vol. 4304, 2006, pp. 1015–1021. doi:10.1007/11941439_114.

58 J. A. V. Alva, E. G. Estrada, A generalization of shapiro–wilk's test for multivariate normality, Communications in Statistics - Theory and Methods 38 (11) (2009) 1870–1883. doi:10.1080/03610920802474465.

59 R. Amaral, M. Ribeiro, E. Pestana de Aguiar, Type-1 and singleton fuzzy logic system trained by a fast scaled conjugate gradient methods for dealing with binary classification problems, Neurocomputing 355 (2019) 57–70. doi:10.1016/j.neucom.2019.05.002.

60 D. Moore, The Basic Practice of Statistics, Vol. 38, 2009. doi:10.1080/00401706.1996.10484558.

61 Y. Li, R. Wang, Z. Yang, Optimal scheduling of isolated microgrids using automated reinforcement learning-based multi-period forecasting, IEEE Transactions on Sustainable Energy 13 (1) (2022) 159–169. doi:10.1109/TSTE.2021.3105529.

APPENDIX A – PUBLICATIONS

The list of conference papers published during the master period are as follows:

- **C. M. Viriato Neto**, L. Garcia, E. P. de Aguiar., Deep rule-based approach for the classification of wagon bogie springs condition, in: Anais do 15 Congresso Brasileiro de Inteligência Computacional, SBIC, Joinville, SC, 2021, pp. 1–6. doi: http://dx.doi.org/10.21528/CBIC2021-48.

The list of papers submitted to journals during the master period are as follows

- **C. M. Viriato Neto**, E. P. de Aguiar, Semi-supervised Deep Rule-Based Approach for the Classification of Wagon Bogie Springs Condition, Evolving Systems (*Accepted for publication*) .
- **C. M. Viriato Neto**, E. P. de Aguiar, Semi-supervised Deep Rule-Based Approach for the Classification of Wagon Bogie Springs Condition: New Perspectives Based on Distance Measures, Applied Intelligence (*Submitted on April 19, 2022*).