



UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE TECNOLOGIA E GEOCIÊNCIAS
DEPARTAMENTO DE ENGENHARIA DE PRODUÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

ANA CLÁUDIA SOUZA VIDAL DE NEGREIROS

**A NOVEL AUTOMATED OIL SPILL DETECTION APPROACH BASED ON THE q -
EXPONENTIAL DISTRIBUTION AND MACHINE LEARNING MODELS**

Recife
2022

ANA CLÁUDIA SOUZA VIDAL DE NEGREIROS

**A NOVEL AUTOMATED OIL SPILL DETECTION APPROACH BASED ON THE q -
EXPONENTIAL DISTRIBUTION AND MACHINE LEARNING MODELS**

Doctoral thesis presented to Universidade Federal de Pernambuco for the doctorate degree attainment as part of the requirements of the Programa de Pós-Graduação em Engenharia de Produção.

Concentration area: Operational Research.

Advisor: Prof. Isis Didier Lins, DSc

Recife
2022

ANA CLÁUDIA SOUZA VIDAL DE NEGREIROS

**A NOVEL AUTOMATED OIL SPILL DETECTION APPROACH BASED ON THE q -
EXPONENTIAL DISTRIBUTION AND MACHINE LEARNING MODELS**

Doctoral thesis presented to Universidade Federal de Pernambuco for the doctorate degree attainment as part of the requirements of the Programa de Pós-Graduação em Engenharia de Produção.

Approved in: 15/07/2022.

BANCA EXAMINADORA

Prof. Isis Didier Lins, DSc (UFPE)

Prof. Márcio José das Chagas Moura, DSc (UFPE)

Prof. Cristiano Alexandre Virgínio Cavalcante, DSc (UFPE)

Prof. Fátima Neusizelma Sombra de Medeiros, DSc (UFC)

Prof. Luís Felipe Ferreira de Mendonça, DSc (UFBA)

ACKNOWLEDGEMENTS

In our lives, we are surrounded by people. Some of them are with us on happy days. Some of them help us on the hard days. And some of them are in both. In my life, I could share the days with essential people. They helped me when I needed it, comforted me when I was lost, gave me important advice, and laughed with me in many happy moments. And I am very grateful to all of them.

First, I thank God for being my light even on the darker days. Second, I thank my mother (Eliane) for everything; she is the most important person in my life. I thank my siblings (Ana Aline e Alex) for being so supportive in many moments. I thank my father for wishing me the best always (Antônio). I thank my Professor (Isis Lins) for teaching me, always opening my eyes to search for important information, and for orientating me since the master's program. I thank Professor Márcio Moura and Professor Caio for co-orientating me. I thank all the CEERMA group (João, Bia, Monalisa, July, Leo, Lucas, Thaís, Joelma, etc.) for all the shared knowledge and laughs and for always helping me when I needed it. I thank all my friends (Jaqueline, Ellen, Elaine, Vanessa, Priscila, Emanuele, Emilia, Erika, Viviane, Márcia) for sharing the happy and sad moments, for the life advice, for the talks.

Thank you!

ABSTRACT

Oil spills are among the most undesirable events in coastal environments because they are substantially harmful, with negative environmental, social, and economic consequences. In general, a risk framework for the event involves prevention, monitoring, detection, and damage mitigation. Regarding detection, rapid oil spill identification is essential for problem mitigation, which generally fosters the use of automated procedures. Usually, automated oil spill detection involves radar images, computer vision, and machine learning techniques to classify these images. In this work, we propose a novel image feature extraction method based on the q-Exponential probability distribution, named q-EFE. Such a probabilistic model is suitable to account for atypical extreme values of the variable of interest, e.g., pixels values, as it can have the power-law behavior. The q-EFE part is combined with machine learning methods to comprise a computer vision methodology to automatically classify images as “with oil spill” or “without oil spill”. Hence, we also propose a new automatic oil spill detection methodology that uses the q-EFE to rapidly identify oil spills in radar images. We used a public dataset composed of 1112 Synthetic Aperture Radar (SAR) images to validate our proposed methodology. Considering the proposed q-Exponential-based feature extraction, the tested Machine Learning methods and Deep Learning models architectures, the Support Vector Machine (SVM) and Extreme Gradient Boosting (XGB) models outperformed deep learning models and Local Binary Pattern (LBP) and Gray Level Co-occurrence Matrix (GLCM) techniques for the biggest dataset size.

Keywords: q-Exponential distribution; feature extraction; machine learning; computer vision; oil spills; risk analysis.

RESUMO

Vazamentos de óleo estão entre os mais indesejáveis eventos que podem ocorrer em ambientes costeiros por causa de seu perigo substancial, com consequências ambientais, sociais e econômicas. Em geral, uma metodologia de risco envolve prevenção, monitoramento, detecção e mitigação dos danos. A respeito da detecção, a rápida identificação de um vazamento de óleo é essencial para a mitigação dos problemas, que geralmente fomenta o uso de procedimentos automáticos. Usualmente, a detecção automática de vazamento de óleo envolve imagens de radar, visão computacional, e técnicas de aprendizado de máquina para classificar as imagens. Neste trabalho, um novo método de extração de características em imagens baseado na distribuição probabilística q-Exponencial, chamado de q-EFE, está sendo proposto. Esse modelo probabilístico é adequado para explicar valores extremos atípicos de variáveis de interesse, e.g., valores de pixels, uma vez que ele tem comportamento de lei de potência (power-law). A parte do q-EFE é combinada com métodos de aprendizado de máquina para compreender uma metodologia de visão computacional para classificar automaticamente imagens como “com vazamento de óleo” ou “sem vazamento de óleo”. Consequentemente, este trabalho propõe uma nova metodologia de detecção automática de vazamento de óleo que usa o q-EFE para identificar rapidamente vazamentos de óleo em imagens de radar. Foi utilizado um conjunto de dados composto por 1112 imagens geradas pelo Synthetic Aperture Radar (SAR) para validar a metodologia proposta. Considerando a extração de características proposta que é baseada na distribuição q-exponencial, os métodos de aprendizado de máquina e as arquiteturas dos modelos de aprendizado profundo testados, os modelos Support Vector Machine e Extreme Gradient Boosting (XGB) superaram os modelos deep learning e as técnicas de Local Binary Pattern (LBP) e Gray Level Co-occurrence Matrix (GLCM) para os maiores tamanhos de conjunto de dados.

Palavras-chave: distribuição q-Exponencial; extração de características; aprendizado de máquina; visão computacional; vazamentos de óleo; análise de risco.

FIGURE LIST

Figure 1 - Classical Computer Vision System (CVS)	16
Figure 2 - Oil's variation, transformation, alteration process after spilling into the sea.....	23
Figure 3 - Systematic scheme	27
Figure 4 - High-level environmental barrier functions to prevent environmental damage	28
Figure 5 - Environmental safety barrier functions to mitigate risk associated with acute oil spill on oil and gas installations.....	28
Figure 6 - Step-by-step of the systematic literature review of oil spill detection.....	37
Figure 7 - Convolutional Neural Networks layers for an oil spill image classification	46
Figure 8 - CNN architectures.....	47
Figure 9 - Generic binary confusion matrix	51
Figure 10 - Methodology for pre-processing and feature extraction using the q-Exponential probabilistic distribution (q-EFE approach)	55
Figure 11 - Feature extraction step using the q-Exponential model.....	58
Figure 12 - SMOTE effect.....	59
Figure 13 - The image of the left side is SAR image, and the image of the right side is the corresponding ground truth mask	61
Figure 14 - Sample of images without oil spills.....	62
Figure 15 - Sample of images with oil spills	62
Figure 16 - Images with oil spills and their respective masks and heatmaps.....	63
Figure 17 - Images without oil spills and their respective masks and heatmaps.....	64
Figure 18 - Outline of the experiments.....	66
Figure 19 - Results overview considering configurations 1 and 2	66
Figure 20 - Average balanced accuracy of 10 rounds for the test set, considering the four dataset sizes, the q-EFE and the GLCM.....	75
Figure 21 - Steps of use of the Oil Spill Detection Web App	77
Figure 22 - This is the first view encountered by the user of the web app for oil spill detection.....	78
Figure 23 - View of the left side of the oil spill web app, where the selects the image size and chooses whether the feature map should or not be displayed.....	78
Figure 24 - Local where the user must upload an image.....	79
Figure 25 - Selection of the machine learning method.....	79
Figure 26 - Button Run to start the process	79

Figure 27 - Feature map generated by the oil spill detection web app.....	80
Figure 28 - Classification result, probability of the result according to pre-trained ML model, and predicted class.....	80

TABLE LIST

Table 1 - Most cited papers about oil spill detection from mid-1990 until 2021, their respective authors, publishers, applied methods, used datasets, and evaluated metric used.....	37
Table 2 - Most recent papers about oil spill detection, their respective authors, journals, applied methods, used datasets, and evaluation metrics used.	38
Table 3 - Description of textural properties.....	42
Table 4 - The two different configurations to extract the image features used in this work...	58
Table 5 - Parameters values for the grid search.....	65
Table 6 - Metric results achieved with q-EFE approach using four dataset sizes to train the classification models, CDF, and configuration 1 and 2. The best value considering each dataset size and each column is highlighted in bold; worst is underlined. The * means that the results were obtained without PCA.....	67
Table 7 - Metric results achieved with q-EFE approach using four dataset sizes to train the classification models, PDF, and configurations 1 and 2. The best value considering each dataset size and each column is highlighted in bold; worst is underlined. The * means that the results were obtained without PCA.....	68
Table 8 - Metric results achieved with q-EFE approach using four dataset sizes to train the classification models, Entropy, and configurations 1 and 2. The best value considering each dataset size and each column is highlighted in bold; worst is underlined. The * means that the results were obtained without PCA.....	69
Table 9 - Metric results achieved with q-EFE approach using four dataset sizes to train the classification models and LBP. The best value considering each dataset size and each column is highlighted in bold; worst is underlined. The * means that the results were obtained without PCA	70
Table 10 - Metric results achieved with q-EFE approach using four dataset sizes to train the classification models and GLCM. The best value considering each dataset size and each column is highlighted in bold; worst is underlined. The * means that the results were obtained without PCA	70
Table 11 - Metric results achieved by the three CNNs using four dataset sizes for training, including using Data Augmentation (DA). The best value of each column is highlighted in bold and the worst is underlined.....	71

Table 12 - Descriptive statistics for the balanced accuracy BAc obtained in 10 rounds, considering the four training set sizes, the ML models, and configuration 1. The best value of each column and different training sizes are highlighted in bold and the worst is underlined for mean, median and, standard deviation.....	72
Table 13 - Descriptive statistics for the balanced accuracy BAc obtained in 10 rounds, considering the four training set sizes, the ML models, and configuration 2. The best value of each column and different training sizes are highlighted in bold and the worst is underlined for mean, median and, standard deviation.....	73
Table 14 - Mean and standard deviation for BAc obtained in 10 rounds MLP, RF, ResNet50 and CNNs, and BAc reached by SVM, LR, and XGB considering the four training set sizes and the LBP and GLCM CV techniques. The best value of each column and different training sizes are highlighted in bold, and the worst is underlined for mean and standard deviation.	74
Table 15 - Required time to extract features and train the best models presented in this work	76

CONTENT

1	INTRODUCTION	13
1.1	MOTIVATION	17
1.2	OBJECTIVES	18
1.2.1	General Objective	18
1.2.2	Specific Objectives	18
1.3	THESIS OUTLINE	19
2	THEORETICAL BACGOROUND AND LITERATURE	21
2.1	PHYSICAL AND CHEMICAL PROPERTIES OF CRUDE OIL	21
2.2	OIL SPILLS CONSEQUENCES	23
2.3	RISK MANAGEMENT FRAMEWORK	26
2.4	REMOTE SENSING	32
2.4.1	Use of Radars	32
2.4.2	Use of Infrared	34
2.5	OIL SPILL SYSTEMATIC REVIEW	34
2.5.1	Manual Detection	35
2.5.2	Semiautomatic and Automatic Detection	35
2.6	IMAGE PROCESSING TECHNIQUES	40
2.6.1	Grey Level Co-occurrence Matrix	40
2.6.2	Local Binary Patterns	41
2.7	MACHINE LEARNING METHODS	43
2.7.1	Multilayer Perceptron	43
2.7.2	Random Forest	43
2.7.3	Support Vector Machine	44
2.7.4	Logistic Regression	44
2.7.5	Extreme Gradient Boosting	45
2.7.6	Convolutional Neural Network	45

2.7.7	ResNet50	47
2.8	PRINCIPAL COMPONENT ANALYSIS	48
2.9	DATA AUGMENTATION	49
2.9.1	Synthetic Minority Over-sampling Technique	50
2.10	EVALUATION METRICS	50
2.11	q-EXPONENTIAL PROBABILITY DISTRIBUTION	52
3	THE q-EXPONENTIAL COMPUTER VISION APPROACH	54
3.1	PROPOSED METHODOLOGY	54
3.2	q-EXPONENTIAL DISTRIBUTION AS IMAGE FEATURE	56
3.3	FEATURE EXTRACTION CONFIGURATIONS	58
4	DATASET	60
4.1	VISUAL ANALYSIS OF THE OIL SPILL DATASET	63
5	EXPERIMENT'S RESULTS	65
5.1	Q-EFE + ML RESULTS WITH CONFIGURATIONS 1 AND 2	66
5.1.1	CDF	66
5.1.2	PDF	67
5.1.3	Entropy	69
5.2	LBP + ML AND GLCM + ML RESULTS	69
5.3	CNN AND RESNET50 RESULTS WITH ORIGINAL IMAGES	71
5.4	COMPARISON OF RESULTS	71
6	WEB APP FOR OIL SPILL DETECTION	77
7	CONCLUSION	81
7.1	WORK LIMITATIONS	82
7.2	FUTURE WORKS	83
	REFERENCES	84
	APPENDICE A – TEST RESULTS	103

1 INTRODUCTION

Oil spills are a specific type of catastrophic environmental disaster provoked by human activities. They are a form of pollution described as the release of a liquid petroleum hydrocarbon into the environment, especially marine areas [1], e.g., originated in refineries, oil platforms, oil tankers that have an accident or “clean” their tanks in the ocean, operative discharge from ships [2], [3]. Indeed, considering the pollution by liquid petroleum, the illicit outflow of ballast and tank cleaning oily residues from oil tankers and ships are the main causes contributing to the contamination of seas and oceans [2] [4]. Oil spills are among the environmental accidents with the most significant economic, ecological and social impacts [5]–[10]. They can be very costly to companies, from fines to the government to the very waste of spilled oil. And as an ecological concern, such disasters can lead to severe consequences that can affect much of the natural marine environment [6], [11], and even human health [12], [13]. Thus, to attenuate the environmental disaster, control oil dispersion, and ensure human lives are not in danger, soon detection of the oil spill and immediate warning from authorities become crucial [14].

Oil spills have increased considerably in the last decades [10] with the occurrence of the following well-known accidents [15]: the Amoco Cadiz in France in 1978, the Exxon Valdez in Alaska in 1989, the “GulfWar” in Kuwait in 1991, the Aegean Sea in Galicia, Spain in 1992, the Erica in France in 1999, the Prestige in Spain and France in 2002, and the British Petroleum platform Deepwater Horizon in the Gulf of Mexico in 2010. More recently, in 2019, there was a huge oil spill on the Brazilian coast [14], [16]–[18], which caused numerous negative impacts on the environment, tourism business, fishermen, among others. These accidents evidence the following three points: i) investment in prevention is the most central measure to avoid this type of disaster; ii) detection of the leakage is the second most important measure to prevent more significant damages, and iii) risk treatment is crucial to ensure that relevant and adequate safety barriers are in place to prevent accidents and to ensure mitigation in case an accident occurs; besides, oil and gas activities are associated with a major accident potential [19].

Thus, to prevent such accidents, there are safety barriers. In this context, there are two important concepts concerning safety barriers [20]. The first is a barrier function (BF) planned to prevent, control, or mitigate accidents or undesired events. The second is a barrier system (BS) designed and implemented to perform one or more barrier functions and risk influencing factors (RIFs).

RIFs (e.g., maintenance) influence barrier performance. The process of the safety barriers is composed of preventative and consequence reduction barriers. This work involves techniques that can be used in the second group of barriers, once the intention is to detect as soon as possible the oil spill and mitigate the disastrous consequences. This is an important task because a large time gap between the oil spill incident and the start of the cleaning procedure generally emphasizes the negative oil spill impacts [21]. Thus, once the oil spill is detected, response techniques can be applied to mitigate the consequences as soon as possible [22]

The oil spill detection in images can be made specially in two ways, manual/visual detection and automated detection [19]. In the manual detection of oil spills, most of the process is made by humans. Contextual information is important in this process, such as oil rigs' and pipelines' location and wind and speed direction [3]. A trained operator goes through the entire image using an image viewer that can calculate some spot characteristics to find some possible oil spills and discriminate between oil spills and look-alikes. The look-alikes are petroleum-free false targets [23]; they are dark spots caused by atmospheric fronts [24], such as currents and eddies [7], [15]. Then, the operator evaluates some features provided by the image viewer, such as the contrast level to the surroundings, homogeneity of the surroundings, wind speed, nearby oil rigs and ships, natural slicks nearby, and edge and shape characteristics of the spot [25]. This process is time-consuming and labor-intensive, given a large number of images that must be analyzed in a short period for effective oil spill monitoring [26]. In addition, the success of manual detection is highly dependent on the knowledge and experience of operators, and the results are subjective. Besides, according to Jiao et al. [27], as manual detection cannot rapidly detect oil spills, enterprises' operating costs remain high. In contrast, their detection methods hardly prevent oil pollution. However, even with an effective automatic detection process, manual detection is still valuable in the final stage process in an optimized way, as a validation procedure in the support for decision-making.

In turn, automated detection uses systems capable of detecting patterns in the images and performing the identification. Nevertheless, a common problem in detecting oil spills is that they are easily confused with the called look-alikes [7], [15]. These look-alikes may result in misidentification [28], [29]. Another problem is because the oil spill images are often very dark because they are acquired by remote sensing, which uses radars composed of sensors mounted at a distance (in general in satellites and aircraft) from the interesting object (the oil spill, in this study). It is important to use methods capable of detecting oil spills even when the images are

dark because, in general, the oil spills images have problems that difficulties the classification task. Besides, according to Kerf et al [30], during the nighttime, the odds of detecting an oil spill are lowered significantly, since not every part of the water is illuminated. However, an efficient automatic oil spill detection system is, in general, faster, cheaper, and with greater reliability compared to a manual system [26].

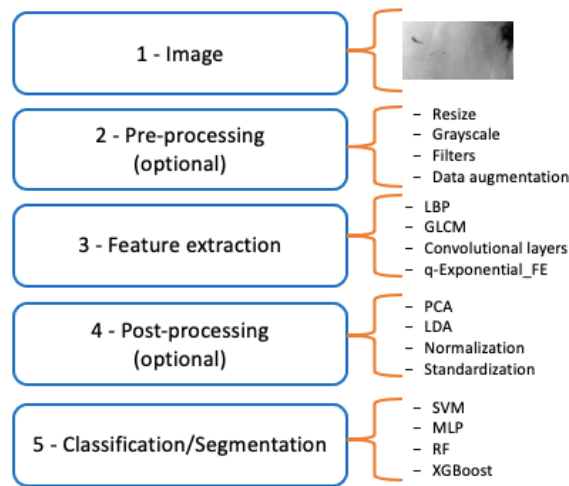
On this basis, many approaches to detect oil spills try to apply efficient methods capable of extracting good features from images to handle the referred problems that may arise. For example, methods as Local Binary Pattern (LBP) [31], Gray Level Co-occurrence Matrix (GLCM) [32], Local Tetra Patterns (LTrP) [33] are well-known feature extraction methods, and they can be applied in many types of researches of different areas. The idea is that such methods can extract from data (images) small important features that will be input into the classification models. Automatic image feature extraction techniques that compose computer vision-based approaches (CV-B) have become more and more common because of their efficiency and practical applications [34]–[40]. In addition, there are some problems where traditional CV-B techniques, which use the context of the image as a whole [41], are a better solution compared to deep learning-based (DL-B) methods [42]–[45]. For example, identifying whether multiple objects in an image are the same or different targets [46].

In this work, we propose a novel feature extractor based on the q-Exponential probabilistic model (q-EFE) to extract features from oil spill images. This new technique uses related q-Exponential functions to extract complex features of Synthetic Aperture Radar (SAR) oil spills images, combining computer vision and statistical techniques to improve the extraction process power. Besides, we apply the ResNet50 model, three other deep learning-based (DL-B) models and two classical CV techniques (LBP and GLCM) to perform result comparisons. Such approaches are being applied in this work motivated by the classification power generally provided by computer vision-based (CV-B) and DL-B methods. The use of CV-B techniques become more and more common because of their efficient feature extraction process, and, it has attracted growing attention owing to their useful applications [34]–[37]. In addition, there are some problems where traditional CV-B techniques with global features are a better solution compared to DL-B methods [42], such as identifying whether multiple objects in an image are the same or different, this limitation was demonstrated by Marcus [46]. However, it is undeniable that methods based on deep learning emerged as a solid and effective kind of computational technique. This is because, according to O’Mahony et al. [42], DL-B methods

such as Convolutional Neural Networks (CNNs) usually improve prediction results using big data and abundant computing resources and have pushed the boundaries of what was possible. This fact is such a huge technology advance for the entire humanity, once such techniques have been applied in the most diverse fields in order to solve problems and/or improve human tasks.

The methodology proposed in this research, which is based on the q-EFE, englobes parameter estimation, feature extraction, and machine learning (ML) methods (including a deep learning model - CNN). The purpose is to encounter a interesting new option to extract complex and important features from SAR oil spill images. We expect that by using these extracted features as input data into ML models, such methods can classify the oil spill images with a satisfactory balanced accuracy rate. Figure 1 presents a general Computer Vision System (CVS) to perform tasks like classification or segmentation. In stage 4, methods such as Principal Components Analysis (PCA) are a post-processing step because the feature extraction is our reference, the focal point in this work. The structure of the methodology proposed in this work follows such steps.

Figure 1 - Classical Computer Vision System (CVS)



Source: The author (2022).

The remainder of this work are organized as follows: Chapter 2 brings a literature review about oil spills characteristics, remote sensing, classification methods, q-Exponential distribution, recent works, and others. The idea of this literature review chapter is to present to the reader the theoretical basement to construct this work. Chapter 3 describes the novel q-EFE methodology to extract features from SAR imagery and to classify these images into one of the

two classes: images with oil spill, or image without oil spill. Chapter 4 presents the applied dataset to validate our proposed methodologies and visual analysis. Chapter 5 exhibits all the experiments results obtained with the proposed approaches and the classical CV technique applied. Finally, chapter 6 brings the conclusion and outlooks.

1.1 MOTIVATION

As time goes on, humanity understands more and more about the importance of care for the place where they live. People had experimented with discoveries and revolutions, such as industrial revolutions that brought technological, socioeconomic, and cultural changes worldwide. Such modifications happened combined with many other aspects. For instance, the world started producing and consuming more and more. In this context, the petroleum industry kept pace with this rhythm with a rapidly growing demand globally for petroleum products [51]. Nowadays, we are still extremely dependent on products manufactured from crude oil, such as gasoline, kerosene, diesel fuel, lubricants, and others.

However, it is common to encounter petroleum in the deep sea, and in the last years, there was an expansion of offshore petroleum extraction [47]. This fact also means that sea pollution by petroleum (crude oil) also became most common at the time we are. In general, such pollution is caused by oil spills, which is a real and dangerous problem that must be combated hard. This phenomenon causes damages that can be seen since the marine biota [48] until the human health [49].

Thereby, considering the oil spills phenomena, studies have been carried out in order to prevent it and/or to minimize its damages. The prevention of this type of environmental disaster is performed, generally, by the called barriers functions [20]. Nevertheless, when these barriers fail it is extremely important to mitigate the damage caused by such undesirable event. Thus, the first step required is to know that it happened, and, considering the sea constitutes the largest single environment on earth, with a vast expanse and remoteness area, the identification of this kind of occurrence is definitely not an easy task.

Therefore, the motivation to perform this research can be justified by the importance of the early oil spills detection for the planet, once undesirable events verified in specific globe spots might, and generally do, bring negative impacts for the entire humanity. More specifically, we can enumerate environmental, economic, and social aspects directly affected by the results

obtained with this research. Such issues can also be seen as motivation to do this work. The environmental aspects are probably the most important ones for conducting this research once the faster it is the oil spill detection less environmental negative impacts will exist. From an economic view, we can think of the costs required to clean up the contaminated areas and the business losses when the oil spill occurs very close to or into coastal environments. Finally, about the social issues, oil spills generally affect the greenhouse and interfere in human health. More directly, some studies conclude that the oil spill clean-up workers have a high probability of developing specific diseases such as chronic rhinitis and cardiac conditions.

Besides, considering an academic motivation, in this work we are proposing a new approach to extract features from images based on the q-Exponential probabilistic distribution with promising results in the context of oil spill detection.

1.2 OBJECTIVES

1.2.1 General Objective

The general objective of this work is to develop a new feature extraction approach based on the q-Exponential probabilistic distribution (q-EFE) to be coupled with machine learning techniques to detect oil spills automatically in Synthetic Aperture Radar images.

1.2.2 Specific Objectives

To achieve the general goal previously presented, we defined some specific objectives exposed in the following:

- To perform a literature review about petroleum (crude oil), barriers functions, remote sensing, use of radars in oil spill detection, oil spill detection methods, machine learning methods, deep learning method, computer vision techniques, data augmentation (DA) techniques, q-Exponential distribution, and its related functions.
- To propose a new computer vision-based technique to extract features from oil spill images using the q-Exponential model (q-EFE).

- To perform tests in the feature extraction step using some q-Exponential related functions.
- To generate new oil spill data using DA.
- To create archives of features extracted from the oil spills images using the proposed CV-based approach.
- To perform visual analysis about the heatmaps generated using the q-EFE proposed approach.
- To apply CV approach using a feature extractor (q-EFE, GLCM, or LBP) and an ML model (MLP, RF, SVM, LR, XGB, CNN) or DL models alone (CNN, ResNet50) to classify images for performance comparison.
- To identify limitations and current gaps of this thesis to be addressed in future research.

1.3 THESIS OUTLINE

The content present in each of the following Chapters of this thesis is briefly described below:

- **Chapter 2** presents the theoretical background and literature review to locate this work as an element of safety barriers in a risk-based framework and to understand the concepts, models, and metrics used in the thesis. We present the characteristics of crude oil (chemical and physical properties), the consequences of oil spills, key concepts about the remote sensing field, the ways to prevent this type of dangerous event (safety barriers); the q-Exponential distribution and its related functions, the machine learning classification methods, the applied classical CV techniques for texture analysis, data augmentation, metric evaluations.

- **Chapter 3** describes the proposed q-EFE approach to extract features from SAR images step by step. Besides, we explain why this specific probabilistic model is chosen in this feature extraction context. Also, we expatiate about the three related q-Exponential functions (cumulative distribution function-CDF; probability distribution function-PDF, and entropy).

- **Chapter 4** brings the information about the applied oil spill dataset used in this study. We perform a visual analysis of this SAR imagery, with and without oil spills, and present related aspects of these images. Also, we perform other visual analyses on the images, and obtain their respective masks and heatmaps generated using the proposed q-EFE.

- **Chapter 5** shows all the experiment results performed throughout the realization of this study. Besides, we present comparisons between all the applied approaches to detect oil spills automatically in images.
- **Chapter 6** brings a technological product for the proposed methodology: a web app for oil spill detection.
- **Chapter 7** presents the thesis conclusion and outlook of this work. We show the main results and give future perspectives in this knowledge area.

2 THEORETICAL BACKGROUND AND LITERATURE REVIEW

This chapter presents the theoretical background of this research to explain the important concepts involved.

2.1 PHYSICAL AND CHEMICAL PROPERTIES OF CRUDE OIL

Petroleum is defined by geochemists as any subsurface material that, when produced, yields crude oil and/or gas. In general, the thermal degradation of kerogen during burial is responsible for forming petroleum, although biogenic or bacterially formed gases should also be included [50]. The basic components of petroleum are hydrocarbons and different organic and inorganic compounds containing nitrogen, sulfur, and oxygen, besides metals such as iron, vanadium, and nickel. Oil is the product acquired by a distillation process of petroleum [2] and the liquid fraction at standard temperature and pressure, while the remainder is gas. Each type of petroleum has its distillation curve, which specifies its chemical identity. After the physical distillation process, different hydrocarbon groups are produced within different temperature ranges, called fractions or cuts [51].

The oil has important physical characteristics (e.g., density, surface tension, pour point, viscosity, water solubility) that affect the spreading potential [50], [52]. The process of oil scattering occurs rapidly on water surfaces and forms a thin layer called an oil slick. The oil slick then thins to finally become a sheen (i.e., a very thin layer). In addition, in the marine environment, the spilled oil immediately experiences a variety of weathering processes, such as dissolution, emulsification, microbial degradation, photo-oxidation, adsorption to suspended matter, deposition on the seafloor, and evaporation. Also, the way how the spilled oil will damage the environment depends on the level of each of such weathering processes [53]. According to [54], evaporation is often the dominant process in the early stages of weathering, and biodegradation plays a dominant role in the later stages. Besides, the complex oil chemical properties are influenced mostly by hydrocarbons, which are the dominant oil chemical component [55]. Furthermore, other oil components are oxygen, nitrogen, trace metals, and sulfur [56], [57]. Also, [58] explain that oils can be divided into unsaturated hydrocarbons and saturated, refined products, aromatic hydrocarbons, resins, and asphaltenes. Besides, the quantity of the oil, thus of chemical components, to which organisms are exposed is crucial in determining how populations respond to spilled oil [59]. For instance, a study of Garcia et al.

[60] concluded that, according to petroleum biomarkers tools, the deposition of hydrocarbons is different for each zone along the terrestrial-aquatic gradient.

In the marine environment, a problem associated with the oil spill is that many components of oxygenation, such as acids, aromatic components, and alcohols, cause water pollution for a long time [61]. Additionally, contamination of soils (either marine or ground) with petroleum products leads to changes in soil compressibility. It also stipulates alteration of the grain-size distribution of sandy-clayey soils [62].

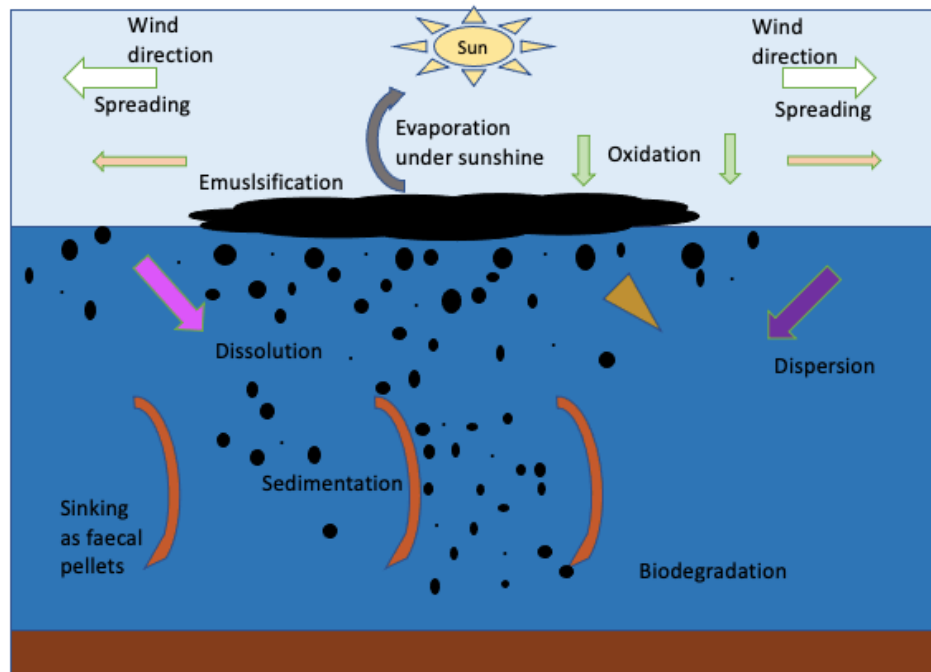
In this context, Figure 1 shows the oil's variation, transformation, and alteration process after spilling into the sea. The oil spreading depends on some factors, such as the wind direction and air temperature. For instance, it is possible the sinking as faecal pellets, which is material from zooplankton [63], sedimentation, and biodegradation of oil.

The process in Figure 1 characterizes the spreading and the persistence of the film after an oil spill occurs, and it relies on the [64]:

- evaporation, depending on the volatility of the components and air temperature;
- biodegradation, due to the presence of microorganisms that can decompose some compounds depending on temperature and oxygen level;
- dispersion, caused by turbulence on the sea surface, which is highly dependent on the type of oil and weather conditions;
- emulsification and dissolution as a consequence of a physical and chemical mixing with water;
- oxidation, a very slow degrading process based on sunlight; and

sedimentation, allowed by suspended solid particles.

Figure 2 - Oil's variation, transformation, alteration process after spilling into the sea



Source: Adapted from [55].

The oil's physical and chemical characteristics explain why this product contaminates the marine environment when an oil spill occurs. Also, the oil spreading potentiality is mostly determined by natural aspects, such as wind direction, sunlight, temperature, etc.

2.2 OIL SPILL CONSEQUENCES

This subsection is dedicated to presenting a literature review about the impacts caused by oil spill occurrences. Impacts caused by this kind of event can be seen in the environment, economy, and human society.

The understanding about important consequences that resulted from oil spill events starts with the recognition that this is a challenging task once oil spill negative impacts are conditional upon the particular geographic, ecological, societal, and temporal contexts in which the disaster occurs [59]. Also, the quantity of oil spilled and spillage rate is also key determinants of the severity of the consequences [59]. However, some negative impacts are common for almost every oil spill occurrence.

In this context, avoiding bigger negative oil spill impacts is a task that also helps to reach the Global Challenges. According to The Global Challenges Foundation [65], there are three interconnected global risks: Climate change, other large-scale environmental degradation, and weapons of mass destruction. Thus, the framework developed and proposed in this work can auxiliary in the two first global risks, once the oil spill generally affects the air atmospheric; hence, it has effects on the greenhouse, which has huge action on climate change. Besides, this research also influences the second global risk, once the oil spill event can promote big environmental degradation. A recent study by Irakulis-Loitxate et al. [66] affirms that if the companies prevent oil and gas spills, it is possible to reduce methane emissions, which would help the world fight against climate change. Thereby, there are many interests in containing the consequences caused by oil spill accidents. In addition, 1 liter of oil can contaminate up to 25 liters of water, taking to death of many marine creatures [66]. Also, Gupta et al. [67] ratify that, for example, a typical mining operation produces 140000 liters of oil-contaminated water per day.

According to Ribeiro et al. [14], generally, oil spills contaminate coral reefs, fishes, reptiles, and mammals, and, because of this, the environmental impacts on marine fauna caused by this type of disaster are often immeasurable. Besides, Dalton & Jin [68] ratifies that the pollution caused by crude oil can affect coastal and marine resources in numerous ways. Indeed, Capuzzo [69] explain that exposure to oil and other hydrocarbons can affect the growth, feeding, development, and reproduction of living organisms. Thus, oil can directly affect the survival of sea creatures, such as fish, sea birds, and marine mammals. Also, it indirectly affect these organisms by reducing the availability of prey [53], oil can still affect heat insulation and buoyancy by penetrating the plumage of sea birds or fur marine mammals [70]. Besides, toxicity pathways in different species are myriad and some examples may include DNA damage, oil's ingestion, impacts to immune functioning, accumulation of contaminants in tissues, cardiac dysfunction, mass mortality of eggs and larvae, e.g., in fish, loss of buoyancy and insulation for birds [49], [71], [72].

Dalton & Jin [68] explains that when a large spill of crude oil or petroleum products happens in a biologically sensitive marine area, environmental damages can amount to billions of dollars. However, it is important to understand that the smaller and more frequent oil spills are also very harmful to the environment. In this context, according to Sandrini-Neto et al. [73], in general, frequent low-dosage oil spills are more deleterious than infrequent high-dosage ones.

Also, from an economic point of view, in the oil spills, in general, there are costs involving clean-up of affected areas, state and federal law spending, and compensation for losses to fisherman and businesses also affected by this type of disaster. In 1989, there was a huge oil spill in Alaska, the Exxon Valdez, Exxon paid \$2.2. billion for clean-up, \$1 billion to settle state and federal lawsuits, and \$300 million for lost wages to 11,000 fishermen and business firms [68], [74]. According to Cohen [75], the cost to the fisheries of south-central Alaska was estimated to be \$108.1 million, the bigger component being a \$65.4 million reduction in the pink salmon fishery in the first year following the accident. However, the impact of an oil spill on aquaculture and commercial fisheries would depend, for example, upon aspects including the volume of the oil spill and location relative to fishing/cultivation areas; the types of species harvested in the region, e.g., whether species are sedentary or mobile; tides, currents, and wave action that disperse the oil; and government decisions relating to fishing bans and compensation schemes [59].

Also, the negative economic impacts can be seen from another perspective, for instance, from the perspective of the fisherman tourism business. These fishermen and companies can also be largely impacted, once some communities that depend on fishing and tourism are directly affected as beaches have become banned for swimming or purely less attractive to tourists due to oil contamination [76], [77]. For example, in August of 2019, crude oil had appeared on Brazil's beaches, mainly in the Northeast region, causing numerous economic problems [78], [79]. In total, 1009 affected locations across 130 municipalities located in all nine northeast states and two southeast states were identified only five months after this tragic event [14], which is, according to IBAMA [80], considered the most massive Brazilian environmental disaster. Also, the impact of this oil spills occurred in Brazil is already being assessed with regard to marine food webs in the affected areas, because crude oil tends to undergo fragmentation, driving to the accumulation of microparticles at different trophic levels, such as in eggs and larval stages [78]. A study made by Ribeiro et al. [14] showed that, considering the economic impacts provoked by this specific oil spill event, the most affected areas were the most dependent on tourism and fishing. Besides, the environmental and socioeconomic impacts caused by the oil contamination in Brazil are incalculable [16].

Besides, according to Ribeiro et al. [14], the economic impacts of this type of event can be worse if they occur in regions with poorly diversified economies and, consequently, more

dependent on tourism, such as the Brazilian Northeast. Indeed, studies had demonstrated that domestic tourism contributes to decreasing regional inequalities in Brazil [14], [81].

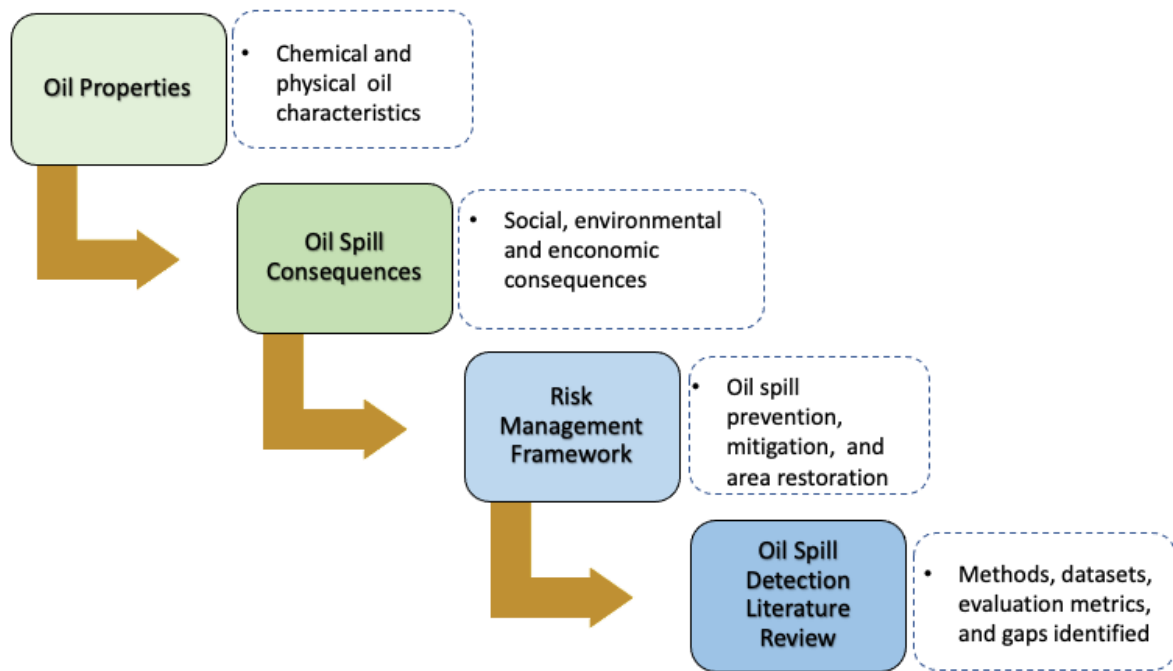
However, different from most people's understanding, the impacts provoked by oil spill occurrences are not only associated with marine environmental effects, and economic expenses, but also to human health. In this context, relatively little is understood about the injurious human health consequences of toxic exposures from oil spill occurrences [13]. However, recent articles have been carried out in order to explain possible direct and indirect consequences for human health.

On this basis, a study carried out by Weblar and Lord [12] showed that humans' health can be affected by oil spills in two main ways: stressors can directly harm humans, e.g., health impacts from breathing oil vapors, and oil can affect ecological processes that cause direct harm, e.g., health impacts from eating seafood with bioaccumulated oil toxins. Also, D'Andrea et al. [13] fulfilled a study that proved that a majority of the oil spill cleanup workers of the 2010 Deepwater Horizon Gulf disaster developed chronic rhinosinusitis and reactive airway dysfunction syndrome. Also, the cardiac abnormalities that were seen in the initial health assessment persisted even 7 years after the disaster in most of the oil spill cleanup workers, and, the oil spill exposed workers reported prolonged or worsening illness symptoms that were present even 7 years after their initial exposure. Besides, still considering the 2010 Deepwater Horizon Gulf oil spill, during the first 18 months after this disaster, residents of Southeastern Louisiana reported increased symptoms of depression, anxiety, and posttraumatic stress [82]–[84].

2.3 RISK MANAGEMENT FRAMEWORK

Considering the whole systematic view and literature review that are being presented in this thesis, as part of the research, Figure 3 illustrates this scheme.

Figure 3 - Systematic scheme



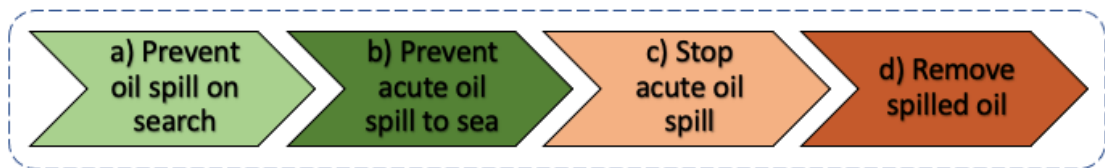
Source: The author (2022).

Many government regulations around the world serve as preventive measures for oil spills in an offshore environment. Besides the prevention task, many of these regulations involve the requirement of a contingency plan to handle the oil spills occurrences. Such a plan must include, for example, the production rate, evaluations of the area, potential recovery amounts of oil and gas, recovery methods, monitoring procedures, costs, technical proposals, and environmental factors [85]. In an overview from Sklet[20], safety barriers are always regarded as non-physical or physical approaches aiming to protect assets from damage. First of all, the term safety barrier originates from road transportation. In this context, they refer to central reservations of roads to prevent collisions or those concrete obstacles on bridges and road edges to avoid driving out of the way [86]. In the oil spill context, the definition is similar once the intent is to prevent catastrophic events.

In the oil & gas and process industries, safety barriers can be shut down valves in pipelines, blowout preventers, or evacuation ways [87]. In situations where the oil was already spread onto the water surface, barriers work on trying to restrain the spread of oil and facilitate its recovery at a later stage [88]. Besides, the division of barrier functions and sub-functions should continue until the barrier function ‘materializes’ and ‘becomes’ a barrier element [89]. In addition, Roed & Bjerga [19] explain that barrier functions can be organized at several levels where the top-level barrier is to ‘prevent environmental damage’. Figure 4 shows the high-level

environmental barrier functions where the left-hand side of an initiating event focuses on preventive barriers (a) and (b), and the right-hand side focuses on consequence reduction barriers (c) and (d). This figure is similar to the traditional thinking about barriers in a bow-tie diagram [19]. In this context, we will deal with this issue as a general risk management framework to avoid or restrain oil spills occurrences.

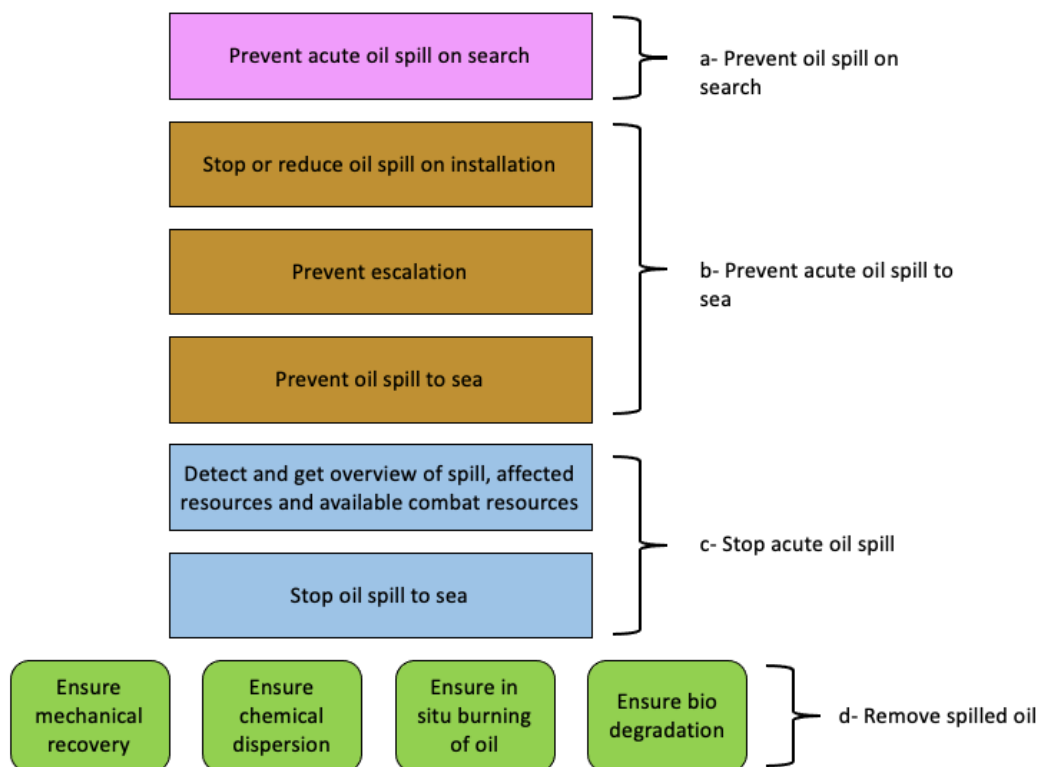
Figure 4 - High-level environmental barrier functions to prevent environmental damage



Source: Adapted from [19].

Regarding the step “a” presented in Figure 4, the first top-level barrier function in Figure 5, that presents the barrier function of Figure 3 subdivided at a lower level, is the same presented in the left hand in Figure 4. This barrier involves many preventive factors, such as maintenance, training, and updating of the procedures and manuals.

Figure 5 - Environmental safety barrier functions to mitigate risk associated with acute oil spill on oil and gas installations



Source: Adapted from [19].

In “Prevent acute oil spill on the installation”, the leakage can be avoided in scenarios such as riser and pipeline leaks, discharge from storage tanks, and leaks during loading and offloading of oil [90]. For scenarios, there are combinations of barrier sub-functions and barrier systems that must be chosen according to the context. In this step and in the next one, rapid oil spill detection is extremely important. If the leakage is detected before it spills into oceans or seas, it is possible to prevent greater negative environmental effects. In this stage, it is common the use sensors [91] to detect possible leakages on settlement.

If the “Prevent acute oil spill on installation” (step “b” in 3) barrier function fails, and the oil spill occurs, the following barrier function (“Stop or reduce the oil spill on the installation – step “c”) can involve some mitigation devices such as shut-off valves, drain systems, and bunding. However, “Stop or reduce acute oil spill on the installation” can be split into the sub-functions: “Ensure detection of the leak” and “Stop or reduce leak from the vessel”. Once again, “Ensure detection of the leak” can be further divided into “Ensure visual detection” or “Ensure manual detection” [3] and “Ensure automatic detection” [26].

In this context, “Ensure visual detection” is made, for example, by operators as organizational barrier elements and descriptions of check routines as operational barrier elements. The “ensure automatic detection” is made by a technical sensor and alarm system, and also operational barrier elements, such as alarm panels, and organizational barrier elements, such as someone observing them. This detection enables the use of machine learning and computer vision techniques. According to Roed & Bjerga [19], there may be situations where weather conditions such as snow or ice may block the drain systems or make it more difficult to observe an ongoing leakage by visual detection, for example, due to darkness. Thus, in general, automatic detection has advantages compared to manual detection; it is cheaper, more reliable, and more accurate [26]. Section 5 brings a more comprehensive review of manual and automatic detection and methods used to detect oil spills automatically.

If the measures to prevent the oil spill fail, it is necessary to choose the best solution to clean up the contaminated area. According to [92], oil spill clean-up has been a questioned issue due to the impossible cleaning of all the discharged and dumped oil into the seawater. There are some options available to clean up the affected areas, called recovery methods. They are physical techniques, chemical techniques, in-situ burning (or thermal) techniques, and bioremediation (or biological) techniques [55]. However, the type of selected methods must be

based on the oil spill characteristics, such as type and quantity, on the conditions of weather and environment [52].

Physical methods to control and prevent oil spills are used [55]. They work as a barrier to avoid oil spreading, but oil's physical and chemical aspects are not changing. The main physical barriers used in this context are [93]–[95]:

- Booms: it is a common type of equipment used to prevent oil spills and slicks from spreading [96]. They act as physical barriers that enclose floating oil and prevent it from spreading [93]. Besides, booms are also used to concentrate oil and maintain an adequate thickness or to deviate oil from biologically sensitive areas so that skimmers can be used or other clean-up techniques [93].
- Skimmers: skimmers are used with booms to recover oil spills from the surface of seawater but oil's physical and chemical properties are maintained hence recovered oil can be reprocessed and reused, this is the step after the use of booms to limit the effective area of spilled [97]. The type and thickness of the oil spill and weather conditions generally determine the success of skimming. In general, skimmers are effective in calm waters and subject to clogging by floating debris [93].
- Adsorbent materials: this type of material facilitates the conversion of a liquid to a semi-solid phase and is considered at the final clean-up step after using skimmers with a high capacity for absorbing the oil and repelling water [98]. These adsorbents can be natural organic, natural inorganic, or synthetic materials.

Chemical methods are used in combination with physical ones, and it treats oil spills by remediating the marine environment due to the capabilities of changing the oil spill's physical and chemical properties [55]. According to Tewari and Sirvaiya [99], chemical methods are among the best remediation techniques available for both onshore and offshore situations. However, chemical methods are not environmentally amicable due to the use of disagreeable chemical agents [100]. The main chemical products used to treat oil spills are:

- Dispersants: they are called surfactants, which are surface-active agents [93], and, might be used in larger areas [55]. The main objective of dispersants application is to weather the oil slicks into small droplets, which submerge into the depth of the water and become quickly diluted and easily degraded [93]. Dispersants are applied by spraying the water with the chemical from vessels or aircraft [55].

Solidifiers: they are high molecular weight polymers that have porous matrices and large oleophilic surface areas and are introduced to chemically bond the solidifier with the oil and transform the physical and chemical properties of the oil during an oil spill [101], [102]. Also, solidifiers were considered as dry granular materials, which functioned by reacting with oil compounds aiming at changing liquid oil into a solid state and they were able to be removed easily [103] and is used to convert oil spill into solid or semi-solid materials, it is contained in booms, pillows, pads, among others [104].

The thermal or in-situ burning method is a simple and quick technique for oil spill remediation that can proceed with minimal specialized equipment and with higher rates of oil removal efficiency [55], [85]. Also, it involves the burning of oil, and it is a more effective oil spill response in the open water with calm winds and is best applicable for refined oil products that will burn quickly without causing any danger to marine life [99]. Also, the major constraints in the use of this method are fear of secondary fires, destruction of aquatic lives and vegetation near the site, which can be affected by burning, long-term alteration in aquatic animals and plants, and risk to human health due to the gases emitted from thermal combustion. Thermal remediation methods are usually affected by speed, water temperature, wind direction, wave amplitude, slick thickness, oil type, and amount of weathering and emulsification that have occurred.

The bioremediation method is a reasonably cheap and straightforward remediation technique. Also, this is a natural process in which microorganisms like bacteria degrade, break up, and metabolize any complex compounds and chemical substances into their food to reestablish and overcome the environmental quality [99], [105]. Therefore, the bioremediation technique has been accepted globally as an in-situ therapy for polluted sites [106].

In this setting, safety barriers (risk management framework) are designed to prevent oil spill accidents. Such barriers can be physical or non-physical assets. The first stage of a barrier system is to prevent the oil spill into the installation. In this regard, an occurred oil spill study that compiles information about where and how such barriers failed and occasioned spilled oil is also missing in the literature. Knowing the failure reason, the system can be improved. Also, studies that show avoided accidents because of the use of structured risk management would be very important for clarifying the real power of such a risk framework, and it could influence the propagation of its use of it.

2.4 REMOTE SENSING

According to Fussell & Rundquist [107] remote sensing was focused on collecting information about the Earth's environment. In addition, its phenomenon has been used to collect information about other planets also and even about the nature of space itself [107]. In other words, remote sensing consists of a group of techniques used to obtain important information about objects of interest in specific surfaces that use distant or remote sensors.

Considering the oil spill detection task, satellite and airborne remote sensing techniques have been extensively used to identify, characterize, monitor, and compute estimates of the thickness of oil spills. These techniques include thermal, visible, and infrared multispectral, microwave, hyperspectral, and laser fluorosensors. Oils in seas and oceans exhibit different characteristics in various wavelengths across the spectrum. For example, in general, there are differences between an oil spillage event captured by Sentinel multispectral and SAR sensors in two distinct places. Each remote sensing technique has its specific advantages and drawbacks [108].

In this context, active microwave sensors are frequently used remote sensing systems for oil spill identification and monitoring due to their broad coverage and capabilities in collecting day-and-night data under all weather conditions. The main types of radar imaging used in detecting and monitoring oil spills are SAR and side-looking airborne radar (SLAR) systems. SAR and SLAR receive/transmit backscattered radio waves, and the reflection of target-surface properties are recorded to produce two-dimensional images of the scene. Both systems operate based on the same synthetic aperture principle and share the same side-looking imaging geometry [108].

2.4.1 Use of Radars

As introduced by this work, the early detection of oil spills can substantially minimize its degree of impact on the environment. That being said, an image analysis system, mostly compounded by satellite and radar images, is a valuable tool for identifying the oil pools in their early stages.

Sea waves backscatter microwave (radar) signals, producing an illuminated image called “sea clutter” and the oil on the sea attenuates capillary waves and the sea clutter from radar imagery [94]. Oil is then shown as a “dark” spot on the sea [109]. Considering SAR imagery, the signals acquired by this type of radar are two-dimensional, but it is possible to create a three-

dimensional model using signal processing methods. Any object or surface that comes into contact with the beam scatters the signal in all directions which is why we use the term “scatterer” to describe them [110].

The image resolution depends on the antenna size or antenna aperture. The larger the antenna aperture is, the higher the image resolution, which may affect classification tasks. In SAR equipment, the movement of the platform at high speed acts as another form of antenna aperture [111]. Also, it has three modes of acquisition: stripmap, spotlight, and scanSAR modes [110].

Chaturvedi et al. [112] affirm that SAR is perceived as the most important remote sensing apparatus for ocean oil spill examination, recording, documentation, and propagation. These SAR sensors are commonly used for monitoring and early identification of oil slicks due to their capability for operating efficiently regardless of the weather and illumination conditions [11]. According to Solberg [113], the sensor is considered a good option due to its effectiveness in all-weather and diversifying illumination conditions and its robustness to cloud occlusions. Despite reported problems with the SAR imagery [114], they are essential for oil spill remote sensing [94].

Many authors have used SAR images in many types of scientific research involving statistical, machine learning, and computer vision techniques over the years [115]–[118]. SAR are active microwave sensors that capture two-dimensional images and are extensively used for dark formation detection in the marine environment, as their recording is independent of weather and clouds [119]. Also, the image brightness reflects the surface's microwave backscattering properties [3]. They can be used in many contexts and for many different objectives; they have properties that can be used, among other applications, to discriminate types of land use and to develop specialized filters for speckle noise reduction [120]. Besides, this kind of image is the most effective way to monitor marine oil spills because the difference in backscattering capabilities between the surface of the sea and the smooth surface of an oil spill area causes their performance on the SAR image to differ [121].

In addition, to provide other information, signal polarizations using vertical (V) and horizontal (H) electromagnetic wave propagation can be used [122]. Collins et al. [123] comment that there is a new form of polarimetry, the compact SAR, in which the system utilizes a mixed

polarization where the transmitter polarization is either circular or orientated at 45 degrees and the receivers are aligned horizontally and vertically.

2.4.2 Use of Infrared

Infrared (IR) is a kind of electromagnetic wave that has a frequency lower than that of red light and is not part of the spectrum visible to the human eye. As many other important technologies' advances, the origin of a modern IR detector technology also refers to World War II [124]. Until today it is important in many detection tasks of different knowledge areas.

According to Pisano et al. [125], oil spill remote sensing techniques are based on visible, microwave, infrared, and radar sensors. Fingas & Brown [101] explain that the oil emissivity of infrared is greater than the water, the oil will after heating emit infrared radiation. However, Despite infrared cameras offering some potential as oil spill sensors, it has several limitations [94]. For instance, in infrared (IR) images, thick oil appears hot, intermediate thicknesses of oil appear cool and thin oil or sheens are not detected [101]. In addition, most infrared sensing of oil spills takes place in the thermal infrared at wavelengths of 8–14 μm , and specific studies in the thermal infrared show that there is no spectral structure in this region [126].

2.5 OIL SPILL DETECTION SYSTEMATIC REVIEW

In this subsection we present three approaches (manual, semiautomatic, and automatic) used to detect oils spills, we also present a systematic review of oil spill detection.

The detection of oil spills happens in phases “b” and “c” in the scheme shown in Figure 4. The rapid detection of oil spill occurrence, such as other oil and gas accidents [127], is essential, requiring methods capable of detecting it even when the images are dark or with a lot of noise. According to Kerf et al.,[30], the odds of detecting an oil spill are lowered significantly during the night since not every part of the water is illuminated. Besides, a considerable time gap between the oil spill incident and the start of the cleaning procedure generally emphasizes the negative oil spill impacts.

On this basis, according to Vasconcelos et al. [128], the rise in the number of sensor systems and the consequent increase in the availability of images to be used in the detection of spills is one of the reasons that explain the increasing trend in the number of publications about oil spills that occurred especially in the last 20-30 years. Thus, the greater quantity of available images

allows the development of novel models to rapidly detect oil spills. In addition, the authors mention a higher frequency of oil spills in the seas in the last decades, alarming the oil spill consequences.

Considering the increase in publications, we present a literature review of the oil spill detection manually and automatically in this section. However, in the automatic detection, we analyzed, considering the period from 1994 until 2021, the most cited works (i.e., articles with ten or more citations, as detailed below) and identified some important aspects of the proposed automatic detection methods, such as used technique, metrics considered, achieved results, database. We used the SCOPUS database to perform our analyzes, such as [128]. However, we considered a different time window and filters and performed other analyses.

2.5.1 Manual detection

In manual detection of oil spills, most of the process is made by humans. Therefore, contextual information such as external information about the location of oil rigs and pipelines and wind and speed direction is important [3]. A trained operator goes through the entire image using an image viewer to calculate some spot characteristics to find some possible oil spills and discriminate between the oil spills and look-alikes. Then, the operator evaluates specific features provided by the image viewer, such as the contrast level to the surroundings, homogeneity of the surroundings, wind speed, nearby oil rigs and ships, natural slicks nearby, and edge and shape characteristics of the spot [25].

Such evaluation is performed to assign the presence of possible oil spills with high, medium, or low confidence levels. This process is time-consuming, and it is also labor-intensive given the large number of images that must be analyzed quickly for effective oil spill monitoring [26]. In addition, the success of manual detection is highly dependent on the knowledge and experience of operators, and the results are subjective. Besides, according to Jiao et al. [27], as manual detection cannot rapidly identify oil spills, enterprises operating costs remain high while their detection methods hardly prevent oil pollution.

2.5.2 Semiautomatic and fully automatic detection

Manual oil spill detection has been replaced by automatic solutions. The automatization of it might follow the same steps as the manual detection because, considering SAR imagery (the most common type of image used for oil spill detection [29], [121], [121]), oil spills appear as

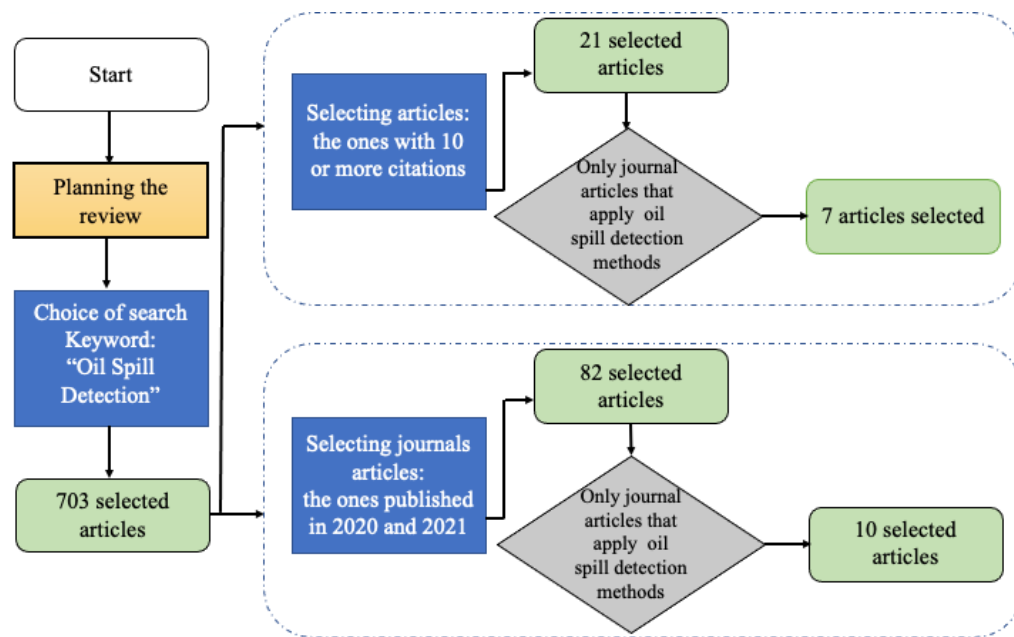
dark spots in these images. One problem is that similar dark spots may arise from a range of unrelated meteorological and oceanographic phenomena, the called look-alikes, this fact results in misidentification [28], [29]. However, differently from the manual process, in automated oil spill detection systems these step processes are performed faster, cheaper, and with greater reliable [26].

In this context, because of the increased number of publications in the last decades [128], we present in Table 1 the papers more cited from 1994 until 2021. We found 703 articles in SCOPUS base in this period, we used “oil spill detection” to perform this search. Besides, Figure 6 brings a graphic representation of the step-by-step of this systematic review performed to construct Tables 1 and 2 (most cited papers and most recent papers). However, such a systematic review is also summarized below:

- First, we used the term “oil spill detection” to select, using VOSviewer software tool, all the articles encountered in SCOPUS database published from 1994 until 2021;
- Then, we used two filters to perform the analysis:
 - First filter: we selected, the papers that had 10 or more citations, we reached 21 papers;
 - Then, we selected from this amount only the papers published in journals and the ones that applied some oil spill detection method, manual or automatic method, we reached 7 papers, they are presented in Table 1;
 - Second filter: we selected the most recent ones, from the original 703 papers, we only considered the ones published in 2020 and 2021, we reached 82 papers;
 - Then, again, we selected only the papers published in journals, and the ones that applied some oil spill detection method, we reached 10 papers, they are presented in Table 2.

In this regard, the idea was to perform an analysis of the methods used to detect the oil spill phenomenon, the datasets used, and the metrics achieved over time. In Table 1, all works apply some method based on ML or some feature extractor before applying a classifier. The most recent article in this list was published in 2007, and the most used metric to evaluate the approaches is the overall accuracy. Besides, some works do not present the number of images used and the paper of Karathanassi et al. [129] used a very small dataset (12 images). Also, there were no papers with manual approaches to detecting oil spills.

Figure 6 - Step-by-step of the systematic literature review of oil spill detection



Source: The author (2022).

On the other hand, Table 2 presents a list of recent papers about oil spill detection published in journals last in 2020 and 2021. In general, accuracy is chosen as an evaluation metric. Two papers, [130] and [131], used the same dataset, but they do not use the same evaluation metric, which means that the results cannot be compared. Also, from the ten papers presented in Table 2, six of them were published in the Remote Sensing journal.

Table 1 - Most cited papers about oil spill detection from mid-1990 until 2021, their respective authors, publishers, applied methods, used datasets, and evaluated metric used

	Authors and Year	Title	Publisher	Methods	Dataset	Metric
1	Fabio Del Frate, Andrea Petrocchi, Juerg Lichtenegger, and Gianna Calabresi, 2000 [130].	Neural networks for Oil Spill Detection using ERS-SAR Data	IEEE Transactions on Geoscience and Remote Sensing	Multilayer Perceptron (MLP)	Dataset composed by 600 images have been taken in the years 1997 and 1998 over various areas of the Mediterranean Sea.	The overall accuracy of 86% in oil spill detection.
2	B. Fiscella , A. Giancaspro , F. Nirchio , P. Pavese & P. Trivero, 2000 [131].	Oil spill detection using marine SAR images	International Journal on Remote Sensing	Feature extraction based on dark spots detected on images and two classification procedures: one based on a Mahalanobis classifier, and the second on compound probability classifier	Dataset composed by 123 images, being 80 with oil spills and 43 with look-alike.	The overall accuracy about 80% in oil spills detection.
3	Karathanassi, Topouzelis, pavlakis, and rokos, 2006 [129]	An object-oriented methodology to detect oil spills	International Journal on Remote Sensing	Object Oriented Image Classification: the first step extracts homogeneous image objects in any chosen resolution that are subsequently classified by means of fuzzy logic.	Dataset composed by 12 images.	The overall accuracy of 98.84% in oil spills detection.

4	Miroslav kubat, robert c. Holte, stan matwin, 1998 [132]	Machine Learning for the Detection of Oil Spills in Satellite Radar Images	Machine Learning	Feature engineering to define useful features and C4.5 and the 1-nearest-neighbor (1-NN) to classify the images.	Dataset composed by 937 images, being 41 with oil spill and 896 with look-alike.	The $g - mean^*$ of 81.1% for the C4.5 and 67.2% for the 1-NN.
5	Maged Marghany, 2001 [133]	RADARSAT automatic algorithms for detecting coastal oil spill pollution	International Journal of Applied Earth Observation and Geoinformation	Texture analysis using gray level co-occurrence matrices. Lee algorithm was used to determine the linearity of oil movements, and the Gamma algorithm was used to determine oil spill spreading.	The RADARSAT data were acquired on 26 October 1997.	Evaluation classification metrics were not provided.
6	F. Nirchio, m. Sorgente, a. Giancaspro, w. Biamino, e. Parisato, r. Ravera, and p. Trivero, 2005 [64]	Automatic detection of oil spills from SAR images	International Journal on Remote Sensing	A method is used to compute some features from interested images. Then a Logistic Regression is performed to distinguish between oil spills and look-alikes.	Dataset composed by 390 images of ERS-1 and -2 satellites, 390 in total 237 oil spills and 153 lookalikes.	The overall accuracy about 90% in oil spill detection.
7	K. Topouzelis, V. Karathanassi, P. Pavlakis, D. Rokos., 2007 [119]	Detection and discrimination between oil spills and look-alike phenomena through neural networks	Journal of Photogrammetry and Remote Sensing	Filters and data normalization as pre-processing. Then, a combination of two Multilayer Perceptrons were used for dark formation detection and for oil spill and look-alike discrimination using high resolution SAR images.	Dataset composed by 159 images, 90 with look-alikes and 69 with oil spills. 50% of the dataset was used to test the model.	The balanced accuracy about 88% in oil spill detection.

Source: The author (2022).

In this regard, we identified in this research that most of the articles that propose some oil spill detection methods focused on automatic detection. In the most cited papers about oil spill detection, there was a limited amount of data available, accuracy was the evaluation metric used for almost every article, and machine learning methods were the most applied. In recent papers there is a greater quantity of data available, accuracy is still very used to evaluate the models but less than before, and the methods are more complex. However, it is not common for papers to use the same dataset as we found only two papers that used the same one but consider different evaluation metrics which does not allow direct comparisons. Thus, performing more benchmarking is important in this field to make methods evaluation easier for future practice applications. Also, SAR imagery is the most used type of image, other type of images could be more tested in the future, such as images generated by unmanned aerial vehicles (UAVs).

Table 2 - Most recent papers about oil spill detection, their respective authors, journals, applied methods, used datasets, and evaluation metrics used.

	Authors and Year	Title	Journal	Methods	Dataset	Metrics
1	Fan Y., Rui X., Zhang G., Yu T., Xu	Feature merged network for oil spill detection using sar images	Remote Sensing	Firstly, a threshold segmentation algorithm is used to process the	Oil spill detection dataset from the	The intersection-over-union

	X., Poslad S., 2021. [121]			original data to obtain the high frequency the information of the original image in the frequency domain. Then the convolutional neural network is used to extract features from the global features, and finally providing decision-making for the segmentation model.	European Space Agency (ESA). Dataset composed by 1110 images.	(IoU) of 49.95% in oil spill detection.
2	Almulih A., Alharithi F., Bourouis S., Alroobaea R., Pawar Y., Bouguila N., 2021 [134]	Oil spill detection in sar images using online extended variational learning of dirichlet process mixtures of gamma distributions	Remote Sensing	Authors developed a variational learning approach based on the infinite Gamma mixture model.	Oil spill detection dataset from the European Space Agency (ESA). Dataset composed by 1110 images. Sentinel-1 wave mode (TenGeoP-SARwv).	Accuracy of 97.96% and FPR of 0.02 for the first dataset. Accuracy of 94.53% and FPR of 0.05 for the second dataset.
3	Wandi WANG, Hui SHENG, Yanlong CHEN, Shanwei LIU, Jijun MAO, Zhe ZENG, Jianhua WAN. 2021. [135]	A fast, edge-preserving, distance-regularized model with bilateral filtering for oil spill segmentation of SAR images	Journal of Oceanology and Limnology	Authors proposed an edge-preserving framework (segmentation) based on the distance-regularized level set evolution (DRLSE).	Two set of remotely sensed Polarimetric Radarsat-2 data were used.	Overall segmentation accuracy of 97.83%.
4	Li Y., Lyu X., Frery A.C., Ren P, 2021. [136].	Oil spill detection with multiscale conditional adversarial networks with small-data training	Remote Sensing	Authors developed a multiscale conditional adversarial network (MCAN) that is able to adversarially learn an oil spill detection model with a limited amount of training data.	Synthetic Aperture Radar images acquired by ERS-1, ERS-2, and Envisat-1. Authors obtained all of the oil spill observation images used in the experiments from the NOWPAP database.	Accuracy of 97.6% Precision of 63.2% Recall of 66.8% F1-Score of 50.6%.
5	Conceição M.R.A., Mendonça L.F.F., Lentini C.A.D., Lima A.T.C., Lopes J.M., Vasconcelos R.N., Gouveia M.B., Porsani M.J., 2021. [137]	Sar oil spill detection system through random forest classifiers	Remote Sensing	Authors proposed a framework to detect oil spills based on machine learning, by making use of random forest models to predict image contents.	Images generated by satellites with a C-band synthetic aperture radar (SAR) operating in the wide range and TOPS mode.	Accuracy about 93% for two classes (oil spill and sea surface).
6	Li G., Li Y., Hou Y., Wang X., Wang L., 2021. [138]	Marine oil slick detection using improved polarimetric feature parameters based on polarimetric synthetic aperture radar data	Remote Sensing	Authors proposed an improved polarimetric feature combination based on difference in scattering mechanism and Random Forest classifier.	The RADARSAT-2 image in Case 1. The RADARSAT-2 image in Case 2. The RADARSAT-2 image in Case 3.	Accuracy about 91%.
7	Wang B., Shao Q., Song D., Li Z., Tang Y., Yang C., Wang M. 2021. [139]	A spectral-spatial features integrated network for hyperspectral detection of marine oil spill	Remote Sensing	Authors proposed a spectral-spatial features integrated network (SSFIN) for	Data obtained from the flight mission on 24 July 2010 for	Precision, Recall, and F1-Score of 99.09%.

				marine oil spill detection from hyperspectral images. Specifically, 1-D CNN and 2-D CNN models have been employed for the extraction of the spectral and spatial features, respectively.	Dalian offshore oil spill monitoring are the airborne hyperspectral data acquired by the spectral imager sensor—AISA Eagle (made in Finland).	
8	Mohammadi M., Sharifi A., Hosseingholizadeh M., Tariq A. 2021. [140]	Detection of Oil Pollution Using SAR and Optical Remote Sensing Imagery: A Case Study of the Persian Gulf	Journal of the Indian Society of Remote Sensing	Authors proposed a object-based image analysis (OBIA) method using optical data to detect oil spills.	Sentinel-1 SAR data processed with the Sentinel Application Platform (SNAP) Sentinel-2 data processed with e-Cognition software that utilized the scale	Accuracies of 92% and 79%.
9	Shamsudeen Temitope Yekeen, Abdul-Lateef Balogun, Khamaruzaman B. Wan Yusof. 2020. [141]	A novel deep learning instance segmentation model for automated marine oil spill detection	Journal of Photogrammetry and Remote Sensing	Authors developed a novel deep learning oil spill detection model using computer vision instance segmentation Mask-Region-based Convolutional Neural Network (Mask R-CNN) model.	Sentinel SAR data with a frequency of 5.405 GHz on a Band-C covering a larger area of 250 km with pixel resolution of 10 X 10 m.	96.6% of overall accuracy for oil spill classification.
10	Mohammed Ozgis, Jorg Kaduk, Claire Jarvis, Polyanna da Conceição Bispo, Heiko Balzter. 2020. [142]	Detection of oil pollution impacts on vegetation using multifrequency SAR, multispectral images with fuzzy forest and random forest methods	Environmental Pollution	Authors compared the Fuzzy Forest (FF) and Random Forest (RF) methods in detecting and mapping oil-impacted vegetation from a post spill multispectral sentinel 2 image and multifrequency C and X Band Sentinel – 1, COSMO Skymed and TanDEM-X images.	Six different datasets.	Different reached metrics for the different datasets and algorithms.

Source: The author (2022).

2.6 IMAGE PROCESSING TECHNIQUES

2.6.1 Grey Level Co-occurrence Matrix

The Grey Level Co-occurrence Matrix (GLCM) and its derived features are tools for image classification that were initially proposed by Haralick et al. [32]. It is a statistical method that computes the frequency of pixel pairs having the same gray levels in an image and applies additional knowledge obtained using spatial pixel relations [143]. Also, according to Eichkitz et al. [144] the GLCM can be explained as a measure of how often different combinations of pixel brightness values occur in an image.

GLCM is calculated on the basis of two parameters: relative distance (d) in between the pair of pixels, and relative orientation (φ) among those pixels. For a 2D image, the immediate neighboring pixels can be in four different directions (0° , 45° , 90° , and 135°). The following equation is used to compute a 2D GLCM:

$$M(i, j) = \sum_{x=1}^X \sum_{y=1}^Y \begin{cases} 1, & G(x, y) = i \text{ AND } G(x + \Delta x, y + \Delta y) = j \\ 0, & G(x, y) \neq i \text{ OR } G(x + \Delta x, y + \Delta y) \neq j \end{cases},$$

where i and j vary from 1 to N_g (number of grey levels).

The correlation between the reference pixel and the neighboring pixels is calculated to determine the textural features of the image [143]. In this work, we applied the GLCM combined with six Haralick features (i.e., contrast, correlation, energy, homogeneity, dissimilarity, and ASM). Contrast, correlation, energy, and homogeneity are presented in Table 3, and the ASM and dissimilarity are presented below:

$$ASM = \sqrt{\sum_{i,j} p(i, j)^2},$$

$$Dissimilarity = \sum_{i,j} p(i, j)^2 |i, j|.$$

To run the GLCM experiments we set (2, 0) as the hyperparameters, which is a distance of 2 and an angle of 0, other values were tested, such as (1, 0), (1, 180), (2, 180), but the results were inferior.

2.6.2 Local Binary Patterns

In order to compare the results achieved by the proposed q-EFE approach, in this work, we will also apply some traditional CV techniques. We will use CV methods to classify texture, which is generally considered as a human perception [145]. However, there are some CV techniques developed aiming classifying different texture images. In this context, we believe that images with oil spills have a different texture from images without oil spills, this is the motivation to use this type of technique.

In this work, we will use the traditional Local Binary Pattern (LBP), originally proposed by Ojala et al. [146], and four LBP variants techniques that take into consideration central intensity, the neighboring intensity, and the radial difference in the grayscale image, such LBP variations were proposed by Liu et al. [147]. The traditional LBP and its variants were applied

many times for texture classification and analysis, image description, face recognition, signal processing, etc. [148]–[152]. The classical LBP is denoted by

$$LBP_{p,r} = \sum_{n=0}^{p-1} s(x_{r,n} - x_{0,0})2^n, \quad s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (1)$$

The LBP variants were proposed by Liu et al. [147] in order to overcome some drawbacks and limitations of the classical LBP, such as

- (i) sensibility to image rotation;
- (ii) small spatial support;
- (iii) loss local textural information; and
- (iv) high sensitivity to noise.

The LBP variants and additional explanation can be encountered in Liu et al. [147]. For each of the four LBP image we compute four features (i.e., contrast, correlation, energy, and homogeneity) such features were extracted from [153]. Also, we extracted more four features from the LBP image (i.e. thresh out, entropy, local variance and standard deviation) as proposed by [154]. The outcome of this process is a feature vector composed of twenty-four features. Table 3 provides a detailed description of the referred features.

Table 3 - Description of textural properties

Feature	Description	Formulation
Contrast	Intensity contrast between a pixel and its neighbor over the whole image.	$\sum_{i,j} i - j ^2 p(i, j)$
Correlation	Correlation between a pixel and its neighbor over the whole image.	$\sum_{i,j} \frac{(i - \mu_i)(j - \mu_j)p(i, j)}{\sigma_i \sigma_j}$
Energy	Sum of squared elements in the GLCM.	$\sum_{i,j} p(i, j)^2$
Homogeneity	The closeness between the distribution of elements in GLCM and its diagonal.	$\sum_{i,j} \frac{p(i, j)}{1 + i - j }$
Thresh Out	The relative proportion of the cracks to non-crack objects.	The threshold value for the Sobel edge detection method
Entropy	A statistical measure of randomness.	$-\sum_p q \log_2 q$
Local variance	The average local standard deviation of 3×3 neighborhood around each pixel in the image.	<i>stdfilt</i> function in MATLAB divided by the number of pixels in the image
Standard Deviation	The standard deviation of all values.	<i>std2</i> function in MATLAB

Source: The author (2022).

2.7 MACHINE LEARNING METHODS

This subsection presents the machine learning classification methods applied in this work in order to validate the proposed methodology.

2.7.1 Multilayer Perceptron

Multilayer Perceptron (MLP) is a well-known machine learning method that is a class of feedforward artificial neural networks. According to Ramchoun et al. [155], the MLP is the most utilized model in neural network applications using the back-propagation training algorithm. It consists of a system of simple interconnected neurons or nodes. Each of these neurons has a model in the network that includes a nonlinear activation function – this function is the sum of the inputs to nodes modified by a simple nonlinear transfer. The transfer function can be linear or nonlinear. If it is linear, then the multilayer perceptron would only be able to model linear functions. One of the most transfer functions used in this kind of method is the logistic one due to its easily computed derivative. This model can perform static mapping between an input space and an output [156]. The MLP can have one or more hidden layers and, finally, an output layer. MLPs are reported as being fully connected if each node is connected to every node in the next and previous layers [157].

2.7.2 Random Forest

Proposed by Breiman [158], the Random Forest (RF) is an ML algorithm that combines tree predictors in which each depends on the value of a random vector, sampled independently from the same distribution of all other modeled trees. Many works have been applied and/or studied the RF [159]–[161]. The convergence of the tree and the strength of each of them is important because the algorithm generalization depends on that. The RF is an ensemble technique that generates multiple and independent decision trees from a vector of random parameters.

Thus, each input vector gives to its corresponding tree an output owing to classification or regression. For classification problems, the task performed in this work, the RF is composed of trees depending on a random vector θ , in which, the tree predictor, $h(x, \theta)$, is a numerical value opposed to the class labels. The output is the classes, and the training set is independent of the distribution of the random vector. It can be characterized as an ensemble of B decisions

trees $\{T_{1(X)}, \dots, T_{B(X)}\}$, where $X = \{x_1, \dots, x_B\}$ is a B -dimensional vector. Its predictor is formed by calculating the average over all trees $h(x, \theta_B)$.

2.7.3 Support Vector Machine

Support Vector Machine (SVM) [162], [163] works as a supervised and unsupervised learning model widely used to recognize patterns and perform regression. It is based on statistical learning theory and, as its training involves a convex and quadratic optimization problem, the Karush-Kuhn-Tucker (KKT) conditions are necessary and sufficient to guarantee the global optimum. After the learning phase, an estimated classifier or regression function is obtained and can be used to predict a class, or a function value related to unseen input data.

In this work, the SVM is used for classification. Given a training set of N data points $\{y_k, x_k\}_{k=1}^N$, where $x_k \in R$ and corresponding binary class labels $y_k \in \{-1, +1\}$, the support vector method approach aims at constructing a classifier of the form [164]: $y(x) = \text{sign}[\sum_{k=1}^N \alpha_k y_k \psi(x, x_k) + b]$, where α_k are positive real constants and b is a real constant; $\psi(.,.)$ is the kernel function, which has the task of fit different models and obtain different predicted values. In other words, the choice of the kernel is based on the data shape, for example, if the data is linearly separable, the kernel linear is the best choice, it is also computationally faster. Several choices for the kernel function are possible, such as linear, polynomial and radial function basis [165].

2.7.4 Logistic Regression

Logistic Regression (LR) models, in general, are used to fit data obtained under experimental conditions [166]. In a multiple logistic regression model, the hypothesis of interest may be the effect of a specific covariate in the presence of other covariates, besides there is more than one covariate in the model [167]. Under certain assumptions, the linear regression is appropriate in cases where the response variable is quantitative (e.g. serum cholesterol level) [168]. However, for a binary (two classes, e.g., “Oil spill detected” and “Oil spill not detected”) or multiclass response (more than two classes) adjustment based on the linear model is not appropriate and the LR must be used.

In this work, when a response variable y is dichotomous, the observed outcome for a given subject j is either $y_j = 0$ if the oil spill was not detected or $y_j = 1$ if the oil spill was detected.

The LR model presumes that the probability of to detect the oil spill in the images (y) has a sigmoidal relationship with X , the vector of independent variables (the image pixels and feature vectors in this work).

2.7.5 Extreme Gradient Boosting

Extreme Gradient Boosting (XGB) [169] is a decision-tree ensemble-based ML model that uses Gradient Boosting (GB) as its framework. This method is considered an optimized and scalable ML system for tree boosting; it performs parallel processing, handling missing values, and avoiding overfitting and bias. XGB is widely applied for problems involving unstructured data, such as images and text, yet it is also used to solve regression and classification problems, such as store sales prediction, high energy physics event classification, web text classification, customer behavior prediction, and others [170]. Among other advantages, Dhaliwal et al. [171] explain that the XGB has a parallel processing, that is uses all the cores of the machine it is running on. Even using minimal resources, this model generates billions of examples using distributed or parallel computation and algorithmic optimization operations. Therefore, it is highly effective in dealing with issues such as classification of data and high-level pre-processing of data. In fact, due its highly scalability, which is an important factor that helps the success of XGB, it can to model different types of problems – and its adaptability to its algorithm (e.g. credit approval, predicting house prices, online predictions) - and algorithmic optimizations. It applies the principle of boosting weak learners using the gradient descent method.

2.7.6 Convolutional Neural Network

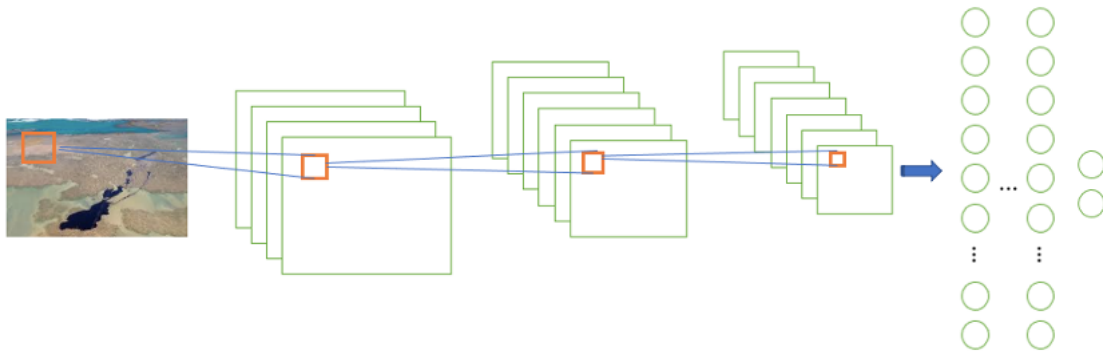
Humans have an enviable learning process. Indeed, most deep learning algorithms were created with the objective to imitate the way the human brain uses distinct parts of information and makes the associative processes that lead to learning [172]. In Deep Artificial Neural Networks (DANN) each node in the net is called “neuron” because its properties are similar to actual neurons in the human nervous system. Biological neurons have the known dendrites that act as input channels, a body that processes and uses information, and an axon that acts like a connector between the output to another neuron. On the other side, artificial neurons have a number of input channels and an output that can be used as an input to another neuron.

The Convolutional Neural Network (CNN) can be explained just as in the previous paragraph, and its original goal is to imitate the way the human brain recognizes images. Thus, for the

training process, images are inputted to feed the model. However, today, CNNs have been also used with other types of input (e.g., feature vectors and text). The convolution operation in CNNs performs operations in small windows of an image, the size of this window is set by the programmer. In the human recognition process, to cover the whole image, there is overlap between different regions, while in CNNs there is also an overlap measure that can be tuned. This process is similar, for example, when feature vectors are used instead of images.

The core layers of CNN are the convolutional layers, which creates the called filter bank, that in turn creates a feature map of the signal analyzed (e.g. image) connecting with the previous layer with a set of weights. In a convolutional setting, each input neuron is not connected to each output neuron in the next layer, but is divided into locally connected segments instead, besides, there is a fully connected layer, which is responsible for the classification or regression task, just after the convolutional operations made by the first convolutional layers [173].

Figure 7 - Convolutional Neural Networks layers for an oil spill image classification



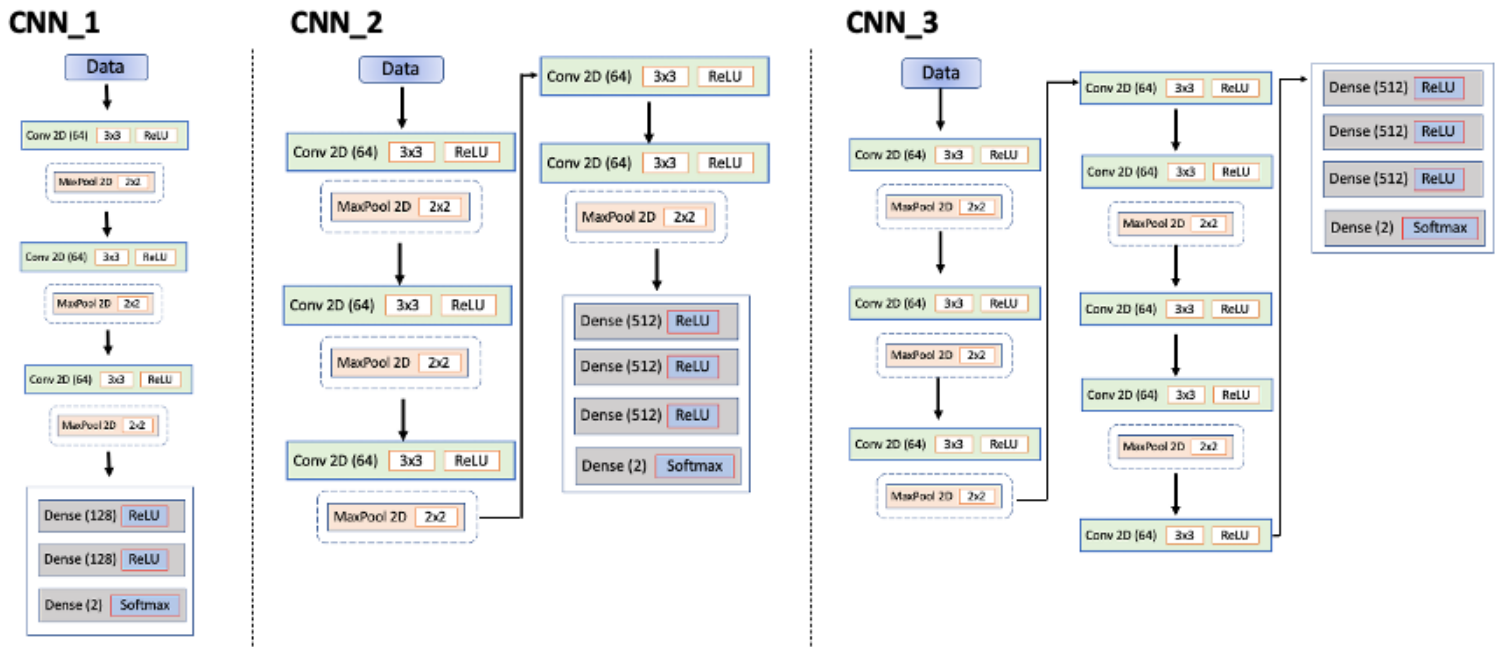
Source: Adapted from Lecun et al. [174].

In this context, we applied three the architectures of CNNs in this work (Figure 8). Before presenting a simple explanation of the architectures of CNN proposed in this research, it is worth mentioning that we tested some other architectures, including using a library for hiperparameter optimization in Python, named “hparams” [175]. However, there was no improvement in the results. CNN_1 is the simplest CNN applied in this work. It has only three convolutional layers, three max pooling layers, and three last dense layers. The max pooling layers are important because they aggregate activations of spatial locations to produce a fixed-size vector in several state-of-the-art CNNs [176]. In the convolutional layers there are 64 neurons in each, in the dense layers, two has 128 neurons, and the last one has 2 neurons,

because we are working in binary classification problem. CNN_2 is more complex than the first model, it has five convolutional layers, three max pooling layers, and four last dense layers. In the convolutional layers there are 64 neurons in each, in the dense layers, three has 512 neurons, and the last one has 2 neurons. Finally, CNN_3 is more complex than the first and second models, it has eight convolutional layers, five max pooling layers, and four last dense layers. In the convolutional layers there are 64 neurons in each, in the dense layers, three has 512 neurons, and the last one has 2 neurons.

These deep models were trained using TensorFlow [177] and Keras [178] libraries in Python running on a free Tesla K80 GPU provided by Google Colaboratory. The image size was 256x256, the number of epochs for training was 100 with an early stop if the validation loss (i.e., binary cross-entropy) did not improve in ten consecutive epochs, and when the training was stopped the best weights were restored. We considered a learning rate of 0.001 and the “adam” optimizer. Other parameters were tested, such as different learning rates and optimizers, but the results were inferior. The batch size (number of samples per gradient update) was equal to 32, which is the Keras default.

Figure 8 - CNN architectures



Source: The author (2022).

2.7.7 ResNet50

ResNet50 is a deep learning model based on deep residual learning [179]. This model was proposed in 2016 and is considered a successful deep learning model in image recognition with high power in complex classification tasks [180], mainly because of its deep feature extraction process. Resnet50 was mainly inspired by the philosophy of VGG [181], which was proposed in 2015. According to He et al. [179], the convolutional layers mostly have 3×3 filters and follow two simple design rules: (i) for the same output feature map size, the layers have the same number of filters; and (ii) if the feature map size is halved, the number of filters is doubled to preserve the time complexity per layer. It is possible to see the graphic representation and obtain more details in He et al., [179]. They also performed downsampling directly by convolutional layers with a stride of 2. The network ends with a global average pooling layer and a 1000-way fully-connected layer with softmax. ResNet50 has been used in many knowledge areas, such as health [182]–[184], agriculture [185], cognitive computing [186], satellite image classification [187], and failure detection [188]. In this work, the ResNet 50 ran in the same machine of CNN models presented in Figure 8, we used the pre-trained weights of imagenet.

2.8 PRINCIPAL COMPONENT ANALYSIS

Principal Component Analysis (PCA) uses orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables [189].e. Its objective is to extract the important information from the dataset, to represent it as a set of new orthogonal variables called principal components, and to display the pattern of similarity of the observations and of the variables as points in maps [190]. In this work we used the PCA as a dimensionality reducer, Bro and Smilde [191] explains that it is quite common to use PCA as a preprocessing step in order to get a nicely compact representation of a dataset. In practice, the dataset can be expressed in terms of the few (R) principal components instead of the original many (j) variables.

Mathematically, PCA depends upon the eigen-decomposition of positive semi-definite matrices and upon the singular value decomposition (SVD) of rectangular matrices. The PCA components are obtained from the SVD of the data table X . Specifically, with $X = P\Delta Q^T$, where P is the $I \times L$ matrix of left singular vectors, Δ is the diagonal matrix of singular values, and Q is the $J \times L$ matrix of right singular vectors, the $I \times L$ is also denoted matrix of factors scores, and also denoted by F , which is obtained by:

$$F = P\Delta.$$

The matrix Q gives the coefficients of the linear combinations used to compute the factors scores. This matrix can also be interpreted as a projection matrix because multiplying X by Q gives the values of the projections of the observations on the principal components. Besides, as an alternative to the PCA for dimensionality reduction, it is possible to use, for example, Linear Discriminant Analysis (LDA) and Auto-encoders. The LDA tool is developed to transform the features into a lower-dimensional space, maximizing the ratio of the between-class variance to the within-class variance, thus guaranteeing maximum class separability [192]. Also, the Auto-encoder is a way to transform representation. According to Wang et al. [193], to achieve desired dimensionality reduction effect, it is necessary to restrict the number of hidden layer nodes to less than the number of original input nodes, thus, we can get a compressed representation of the input.

2.9 DATA AUGMENTATION

Data augmentation (DA) is an effective method for improving the accuracy of modern image classifiers [194], indeed it is a technique also used to augmentation other type of data, such as audio used in speech recognition [195]. In addition, the well-known Deep neural networks are powerful machine learning systems that tend to work well when trained on massive amounts of data [194]. In this context, DA is an effective technique to increase both the amount and diversity of data by randomly “augmenting” it [196], [197].

In a generic manner, DA refers to any technique that artificially inflates the original training set with label preserving transformations and can be represented as the mapping [198]:

$$\psi: \mathcal{S} \mapsto \mathcal{T} \tag{2}$$

Where, \mathcal{S} is the original training set and \mathcal{T} is the augmented set of \mathcal{S} . The artificially inflated training set is thus represented as

$$\mathcal{S}' = \mathcal{S} \cup \mathcal{T} \tag{3}$$

In addition, \mathcal{S}' contains the original training set and the respective transformations defined by ψ . The term label preserving transformations refers to the fact that if data x is an element of

class y then $\psi(x)$ is also an element of class y .). In this work, DA process was performed in Python, using the Keras library [178], performing rotation, zoom, flip, width shift range, height shift range, and shear range.

2.9.1 Synthetic Minority Over-sampling Technique

Synthetic Minority Over-sampling Technique (SMOTE) is a method proposed by Chawla et al. [199] to handle the imbalanced class problem, which occurs when one of the target classes has a small number of instances compared to the other class [200]. Its used for unidimensional data. Using imbalanced data sets in training can cause problems in the generalization ability of the machine learning model. Some parameters must be set to perform the SMOTE technique, such as the ‘k_neighbors’, which is the number of nearest neighbors used to construct synthetic samples, and the ‘random_state’, which is the seed used by the random number generator. In this work, we used the default value for ‘k_neighbors’ (5), and ‘random_state’ was randomly chosen as 42.

In some situations, applying a technique to handle the imbalanced dataset is a step very important, which could significantly improve the classification performance. Basically, there are two simplest techniques to solve this problem, a random over-sampling technique and a random under-sampling technique [201]. In fact, both methods are sampling the dataset until the classes are approximately equally represented, resulting in a balanced dataset. However, the random over-sampling technique may create the decision regions smaller and more specific and it can lead to the overfitting problem [202]. On other hand, a random under-sampling technique encounters the problem of diminishing some important information from a dataset. For handling these problems, the SMOTE was proposed, it generates synthetic examples in a less application-specific manner, by operating in “feature space” rather than “data space”. The minority class is over-sampled by taking each minority class sample and introducing synthetic examples along the line segments joining any/all of the k minority class nearest neighbors. Neighbors from the k nearest neighbors are randomly chosen depending upon the amount of over-sampling required [199].

2.10 EVALUATION METRICS

The performance of the classifiers is commonly evaluated by a binary confusion matrix [200] as illustrated in Figure 9 below. In this work we have a binary classification problem, however the matrix is analogous for the multiclass classification.

Figure 9 - Generic binary confusion matrix

		Predicted	
		Negative	Positive
Actual	Negative	$C_{0,0}$	$C_{0,1}$
	Positive	$C_{1,0}$	$C_{1,1}$

Source: The author (2022).

Where:

$C_{0,0}$: represent the number of true negatives (TN);

$C_{0,1}$: represent the number of false negatives (FN).

$C_{1,0}$: represent the number of false positives (FP);

$C_{1,1}$: represent the number of true positives (TP);

We used five performance measures on classification to evaluate the results of the proposed methodology in this work: accuracy (Ac), precision (Pr), recall (Re) or sensitivity, specificity (Sp), and balanced accuracy (BAC). The BAC is a classification performance metric devised to account for imbalanced classes. These metrics can be computed by the following Equations 4 - 8:

$$Ac = \frac{(TN+TP)}{(TN+FN+TP+FP)} \quad (4)$$

$$Pr = \frac{TP}{TP+FP} \quad (5)$$

$$Re = \frac{TP}{TP+FN} \quad (6)$$

$$Sp = \frac{TN}{TN+FP} \quad (7)$$

$$BAc = \frac{\left[\frac{TP}{TP+FN} + \frac{TN}{TN+FP} \right]}{2} \quad (8)$$

The accuracy computes the number of correct predictions divided by all the predictions (correct and wrong ones). This metric does not consider the imbalance of the classes. The precision is interested in evaluating the examples predicted correctly as positive (“image with oil spills”) divided by all the positive predictions. A low precision rate indicates that the model lets pass many positive examples, or many examples can be predicted as positive when it is not. The recall is a rate interested in the positive examples, the correctly predicted ones, and the wrongly predicted as “negative”. A small recall indicates that the model cannot accurately predict the images with oil spill, which could be very harmful because measures will not be taken in a timely manner. The specificity is the opposite of the recall; it is interested in negative examples. And, the balanced accuracy is a weighted mean between recall and specificity. It penalizes the majority class and improves the value of the correct predictions of the minority class.

2.11 q-EXPONENTIAL PROBABILITY DISTRIBUTION

The q-Exponential related functions will be used to try to extract important features from the oil spill images. The key idea is that these functions can be as sensitive as possible to classify correctly the images with and without oil spills.

The q-Exponential distribution has the following probability density function (PDF)

$$f(t) = \frac{2-q}{\eta} \exp_q \left(-\frac{t}{\eta} \right) = \frac{2-q}{\eta} \left[1 - \frac{(1-q)t}{\eta} \right]^{\frac{1}{1-q}}, \quad (9)$$

Where, $t > 0$, $\eta > 0$ is the scale parameter and $q < 2$ determines the PDF shape and is known as entropic index. When $q < 1$, Equation (9) has a limited support with an upper bound that depends on η and q , see the Equation (10):

$$t \in \begin{cases} (0, \infty), & 1 < q < 2 \\ \left(0, \frac{\eta}{1-q} \right), & q < 1 \end{cases} \quad (10)$$

The q-Exponential has the following cumulative distribution function (CDF):

$$F(t) = 1 - \left[\exp_q \left(-\frac{t}{\eta} \right) \right]^{2-q} = 1 - \left[1 - \frac{(1-q)t}{\eta} \right]^{\frac{2-q}{1-q}}, t \geq 0. \quad (11)$$

Based on maximum likelihood method, which has some important properties (such as asymptotic unbiasedness, strong consistency and efficiency, for a given sample $t = (t_1, \dots, t_i, \dots, t_n)$ of pixel values, the q-Exponential likelihood function is given by

$$L(t|q, \eta) = \prod_{i=1}^n \frac{2-q}{\eta} \exp_q \left(-\frac{t_i}{\eta} \right) = \prod_{i=1}^n \frac{2-q}{\eta} \left[1 - \frac{(1-q)t_i}{\eta} \right]^{\frac{1}{1-q}}. \quad (12)$$

The corresponding log-likelihood function is, as can be seen in [203]

$$l(t|q, \eta) = n \ln \ln \left(\frac{2-q}{\eta} \right) + \frac{1}{1-q} \sum_{i=1}^n \ln \left[1 - \frac{(1-q)t_i}{\eta} \right]. \quad (13)$$

When $q < 1$ we have q-Exponential estimation problems using the maximum likelihood method as explained by Negreiros et al. [203]. Beyond presenting the q-Exponential problem when $q < 1$, the authors also proposed a corrected q-Exponential log-likelihood function to be used in this case. This correction was made applying the Firth method [204]. The general equation for the corrected log-likelihood is presented in Equation (14)

$$l^*(t|q, \eta) = l(t|q, \eta) + \frac{1}{2} \ln \ln |K|, \quad (14)$$

where K refers to the determinant of the Fisher information matrix, and the penalization term $|K|^{\frac{1}{2}}$ is the Jeffreys invariant prior [205].

Tsallis entropy is a generalization of the Boltzmann-Gibbs-Shannon (BGS) entropy [206], and it is defined as [47]

$$S_q = \frac{1 - (\sum_{i=1}^w p_i^q)}{q-1}, \quad (15)$$

where $w \in N$ is the total number of possible configurations, p_i is the associated probabilities and $\sum_{i=1}^w p_i = 1$. The q-Exponential entropy is given by

$$S_{q-Exp} = \frac{\left[1 - \frac{(1-q)t}{\eta} \right]^{\frac{2-q}{1-q}}}{q-1}. \quad (16)$$

3 THE q-EXPONENTIAL COMPUTER VISION APPROACH

The entire methodology will be presented in this Section in some steps explained in detail below. Steps 2 and 4 of the CVS (Figure 1) are optional, even though such steps are, in general, applied to increase the system power. Our proposed approach is inserted in step 3 (Figure 10), the feature extraction. This process can be performed by other methods, such as LBP, GLCM, and CNN, in its convolutional layers.

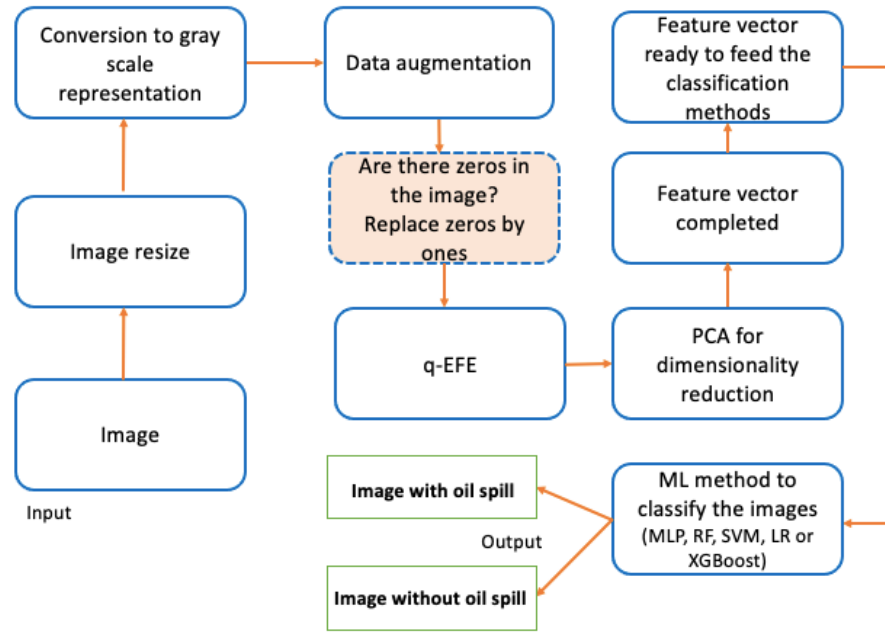
3.1 PROPOSED METHODOLOGY

A flowchart of the whole proposed methodology has a graphical representation shown in Figure 10. Thus, our proposed q-EFE approach is detailed explained in the following steps:

- First, the image is loaded in a chosen size ($p \times q$). Then, it is converted to a grayscale representation. This grayscale conversion can be made using any method available. In our case, we used “np.dot” available in Python.
- Then, the zero-valued pixels of the image are replaced by one once the feature extraction process uses the q-Exponential log-likelihood maximization to estimate the distribution parameters, and one of the model restrictions is related to the positivity of values (Equation 10). We chose one to replace zero because it is the closest integer to the latter in grayscale (0, 1, 2, ..., 255). Indeed, it is impossible to see the difference between these two tons of grey with an unaided eye.
- Then, the choice of using DA or not is considered. If the dataset is too small or very imbalanced, the DA maybe improve the results.
- The feature extraction (Figure 11) starts on the resized grayscale with no null pixels in the image. Features are small newsworthy, descriptive, or informative patches in images [42]. In this step, we take $n \times n$ image patches (i.e., n^2 pixels), we maximize the q-Exponential log-likelihood function, using some numeric maximization method and compute the interested q-Exponential function ϕ_i (e.g., PDF, CDF, entropy). In this work, we used Nelder-Mead [207] maximization method. We repeat this process for the entire image considering a stride size (Δ). The output of this process is feature vectors with sizes that depend on the image’s dimension, the patches size, and the stride.
- The feature vector often presents a high dimension. For example, for $p = q = 64$, $\Delta = 1$ and $n = 4$, the output feature vector has dimension 3721. Equations 17 and 18 bring

the formulas to compute the feature vector size. Hence, for dimensionality reduction, the feature vector undergoes PCA for obtaining the first k principal components. After this, the reduced feature vector is ready to feed the machine learning classifier. Also, another dimensionality reduction can be used instead of PCA, such as LDA and Auto-encoders.

Figure 10 - Methodology for pre-processing and feature extraction using the q-Exponential probabilistic distribution (q-EFE approach)



Source: The author (2022).

In Figure 10, the step of converting the image to a grayscale representation is performed because we are working with only one channel of color, as the classical computer vision methods (e.g., LBP and GLCM). The proposed q-EFE is a method that can be considered as a feature extractor based on probabilistic distribution, the q-Exponential. In this context, there are other probabilistic models used to extract features from images, such as \mathcal{G}_A^0 [208], [209] and, q-sigmoid function [210].

To compute the size of the feature vector (M), the following formulas are given [211]:

- If $\Delta = n$ and $p = q$ (without overlapping),

$$M = (p/\Delta)^2, \quad (17)$$
- If $\Delta < n$ and $p = q$ (with overlapping),

$$M = ((p - (n - 1))/\Delta)^2. \quad (18)$$

Considering the feature extraction process (Figure 11), this process of acquiring features is very important in a CVS. According to O'Mahony et al. [17], features are small newsworthy, descriptive, or informative patches in images. Thus, our idea is to extract features descriptive enough to detect oil spills in images. Thereby, this feature extraction process starts with the q-Exponential parameter estimation by the maximum likelihood method [132]. This step is very important for the proposed approach because these extracted features will define the classification model's power. In our experiments, we take image patches 4 x 4 (16-pixel values in total) to estimate the q-Exponential parameters by the Nelder-Mead optimization method [133], which proved to be a successful numeric method to find the maximum of the q-Exponential log-likelihood function as can be seen in Sales Filho et al. [134]. Also, the q-Exponential distribution has two parameters – q (shape) and η (scale) – and, when $q < 1$ we use the corrected q-Exponential log-likelihood demonstrated by Negreiros et al. [86], which is also maximized by the Nelder-Mead. In practice, less than 3% of the cases had $q < 1$, according to the experiments. In such cases, the q-Exponential presents the power-law behavior and we have evidence of rare events, with more heavy distribution tails. Also, in such situations, the results obtained in [86] are used in practice to maximize the q-Exponential log-likelihood, estimate the parameters and compute the related function values.

3.2 q- EXPONENTIAL DISTRIBUTION AS IMAGE FEATURE EXTRACTOR

The q-Exponential is a probabilistic distribution very used in reliability field because of its ability in modeling the three phases of the bathtub curve [203], [212]. However, it is also used in other fields, such as finances and physics [213], [214]. Nevertheless, there are no studies in which q-Exponential has been used to extract features from images before this work. This model has the characteristic of modeling very well rare events, presenting power-law behavior when the shape parameter is between 1 and 2 ($1 < q < 2$). Besides, recent studies provide findings that proves that the original q-Exponential can be better than other reliability models (e.g. Weibull distribution) to fit failure data when the system is in the improvement phase (failure rate decreasing and power-law behavior) [212], and when the system is in the wear-out phase (last phase of the bathtub curve), the corrected q-Exponential [203] is better than other reliability distribution (e.g Weibull, q-Weibull and Modified Extended Weibull). Thus, as in general the oil spills appear as small dark spots in images, they can be considered as rare events.

Considering the q-Exponential power to fit rare events, this is the reason to choose this specific model to develop this methodology.

In this regard, we tested three functions (CDF, PDF, and entropy) to extract complex and important features from the oil spill images. In other words, we intend to extract features descriptive enough to classify these images correctly. The CDF is a function that computes the probability of the random variable (T) takes a value less than or equal to a real value (t); the q-Exponential CDF is presented in Equation 4. Thus, considering that the oil spill in images is generally dark spots, and, in grayscale, dark pixels are close to 0, we expect that pixel windows that contain oil spills will have high CDFs, and pixel windows that don't contain oil spills will have lower CDFs. To compute the CDF value of each pixel window, we take the mean patch value. We also tested the median of the image patches, but the mean resulted in better results. The minimum and maximum values should not be used as summary statistics because they would lead to CDF closer to 0 and 1, respectively. There are no references of works that used the q-Exponential CDF computed per each image patch, and used for classification purposes the way we did in this work. However, other works also used probability distributions to extract features from images. For instance, Rodrigues et al. [210] proposed the q-sigmoid function derived from non-extensive Tsallis statistics and used it for feature extraction in tasks of regions enhancement in ultrasound images. On the other hand, Marques et al. [208] applied the \mathcal{G}_A^0 distribution to perform segmentation in SAR imagery to characterize image regions.

In this context, the PDF methods is applied in many situations, such as modeling chemically reacting turbulent flows [215]. However, as the q-Exponential CDF, no studies used the q-Exponential PDF to extract features from images. We take the estimates parameters to compute the PDF value of each pixel patch. Then, the output is a feature vector composed of PDF values. The idea is that these values can extract some important information to aid in the oil spill classification task. If these values can identify some complex pattern, we believe that such vectors can be used for oil spill image detection.

Using entropy to extract image features and classify different image textures has been widely used by researchers of different fields and for distinct types of image textures [136]–[138]. Also, according to Amigó et al. [206], entropy appears in many contexts (statistical mechanics, information theory, thermodynamics, etc) to measure different properties (complexity, randomness, uncertainty, etc). Specifically, we use the q-Exponential entropy to imagery

feature extraction. Given the properties of the q-Exponential distribution discussed in Section 2 and the application of entropy in CV context, the q-Exponential entropy may provide good results.

3.3 FEATURE EXTRACTION CONFIGURATIONS

In this work, we will test two specific configurations to extract the features, and it is displayed in **Erro! Fonte de referência não encontrada..** In configuration 1 we will have feature vectors of 3721 elements, and in the second configuration, we will have feature vectors of 4096 elements.

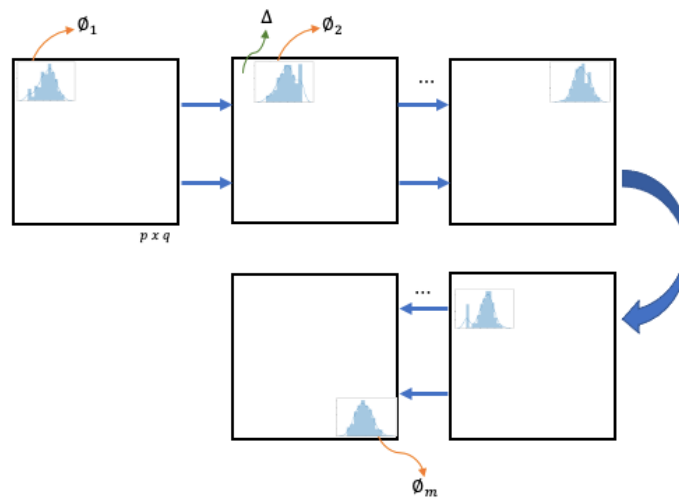
Table 4 - The two different configurations to extract the image features used in this work

	Description Configurations	Output
Configuration 1	$p = q = 64$ $\Delta = 1$ $n = 4$	Feature vector of size 3721 values
Configuration 2	$p = q = 256$ $\Delta = 4$ $n = 4$	Feature vector of size 4096 values

Source: The author (2022).

Once the feature extraction is completed (Figure 11), the next step is to train the ML classification methods. The feature vectors will be used as input to the classification models, each of these vectors represents an image. Before the training process, the SMOTE technique (Figure 12 - see subsection 2.10 for more details) is used to handle the imbalanced dataset problem.

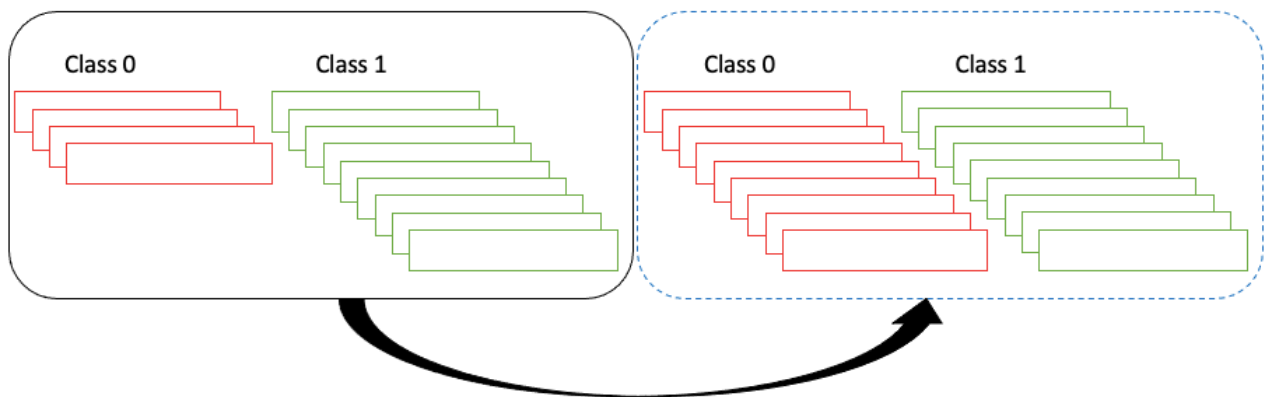
Figure 11 - Feature extraction step using the q-Exponential model



Source: The author (2022).

However, we also perform the classification without the SMOTE, the best result (greater balanced accuracy (BA)) is chosen for each case. Also, before the classification step, we apply a method to reduce the dimensionality, the PCA (see subsection 2.9 for more details). We also performed the classification process without the PCA and compared the results, choosing the option that resulted in the best BA. The conceptual idea is that the images with oil spills will produce feature vectors different from the feature vectors related to the images without oil spills. We expect that this difference will be big enough that the ML techniques can detect the oil spill.. Thus, we believe that the classification models will be able to classify these images with a high balanced accuracy level.

Figure 12 - SMOTE effect



Source: The author (2022).

4 DATASET

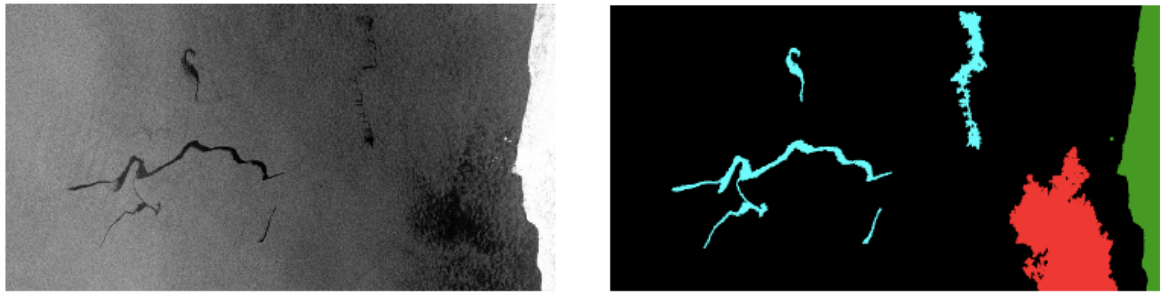
The dataset used in this work were made available by Krestenits et al. [7] and Krestenits et al. [11]. It also includes semantically annotated masks for the researchers to evaluate their experimental results [7]. In general, masks are generated automatically from a segmentation approach, this process could generate some mistakes at the input mask [216]. However, in this work, we are not using the masks to train the models.

The referred dataset containing oil polluted sea areas was collected via the European Space Agency (ESA) database, the Copernicus Open Access Hub, and it has the geographic coordinates and timestamps information about the pollution event provided by the European Maritime Safety Agency (EMSA) through the Clean Sea Net service. Besides, the EMSA confirmed the presence of dark spots depicted in the SAR images as oil spills. The oil pollution records cover a period from 28 September 2015 up to 31 October 2017 while the SAR images were acquired from the Sentinel-1 European Satellite missions [7].

According to Krestenits et al. [7], the ground range coverage of SAR sensor is about 250 km with pixel spacing equal to 10×10 m. Satellites employed for the Sentinel-1 mission are equipped with a SAR system operating at C-band. The polarization of the radar image is dual, i.e., vertical polarization transmitted - horizontal polarization received (VH) and vertical polarization transmitted - vertical polarization received (VV). To make the SAR image dataset only the collected raw data from the VV band were processed, following a series of pre-processing steps in order to extract common visualizations. The image pre-processing scheme includes the following phases:

- (i) Every confirmed oil spill was located according to EMSA records.
- (ii) A region containing oil spills and possibly other contextual information of interest was cropped from the raw SAR image. The cropped image was re-scaled to meet the resolution of 1250×650 pixels.
- (iii) Radiometric calibration was applied in order to project every 1250×650 image into the same plane,
- (iv) A speckle filter was used to suppress the sensor noise scattered in the entire image.
- (v) A linear transformation was applied for dB to real luminosity values conversion.

Figure 13 - The image of the left side is SAR image, and the image of the right side is the corresponding ground truth mask

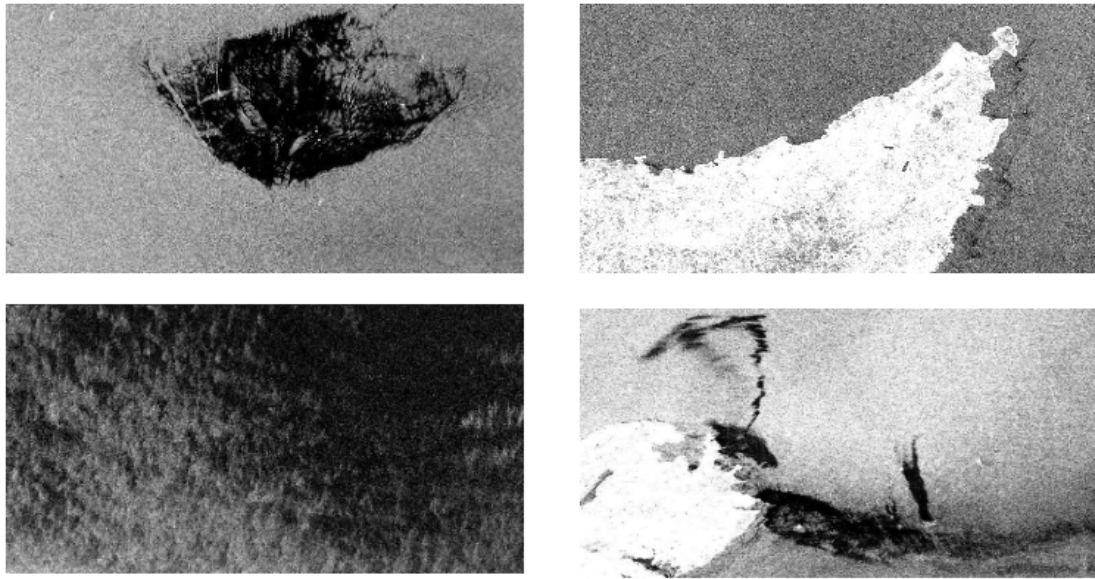


Source: Adapted from Krestenits et al. [7].

The dataset used in this study is composed of 1112 images, where 873 images have oil spills and 239 do not present this phenomenon. The oil spill is identified in the masks by cyan. Besides, the image masks identify other parts that can be of interest, such as: look-alikes (in red); ship (in brown); land (green); and sea surface (black), as can be visualized in Figure 13. However, once this work deals with detection of images that contain oil spills, only two classes were considered: images with oil spills and images without oil spills. In future we intend developing a study multiclass able to identify all the phenomena/objects in the images. Figure 13 presents an example of an extracted SAR image accompanied with its respective ground truth mask; these images are present in the referred dataset used in this paper. In this figure, we can see that the left image has oil spill, look-alikes, ad sea surface.

Besides, another important characteristic of this dataset is that it is quite an imbalanced dataset. An imbalanced dataset is verified when one of the target classes has a small number of instances compared to other classes [200]. In this case, the images with oil spill comprise about 78% of the entire dataset, and the remaining are the images without oil spills. Figure 14 presents a sample of four images without oil spills extracted from the used dataset.

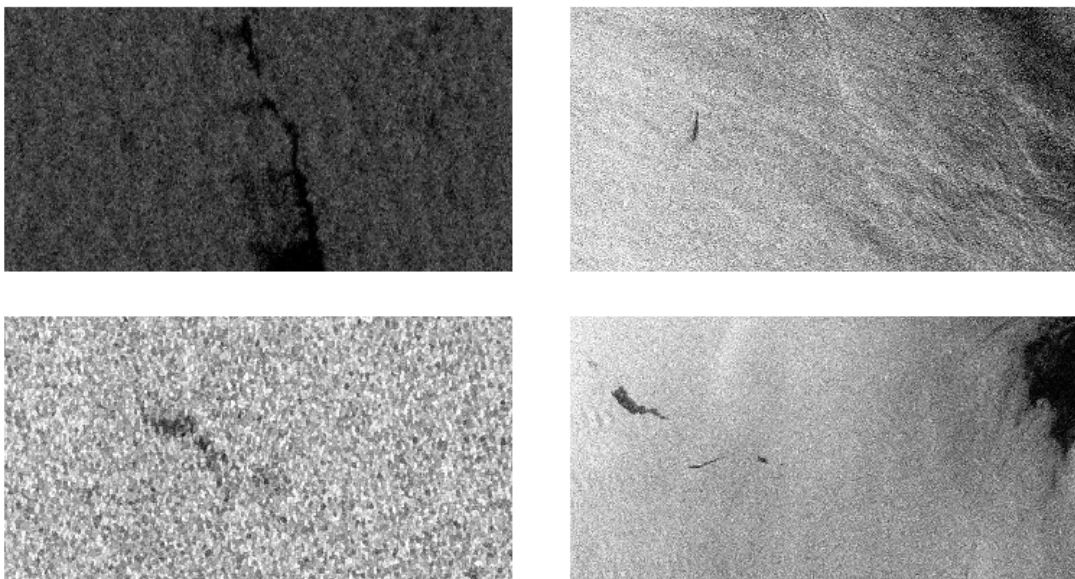
Figure 14 - Sample of images without oil spills



Source: Adapted from Krestenits et al. [7].

On the other hand, Figure 15 presents a sample of four images with oil spills. By visual analysis, we can conclude that the task of identifying the images with and without oil spills is hard, once the oil spill's pattern can be easily confounded by the named look-alikes. Indeed, some researchers of the oil spill detection field confirm that the most challenge verified in the oil spill image identification is the fact that the look-alikes be very similar phenomena compared to the oil spills.

Figure 15 - Sample of images with oil spills

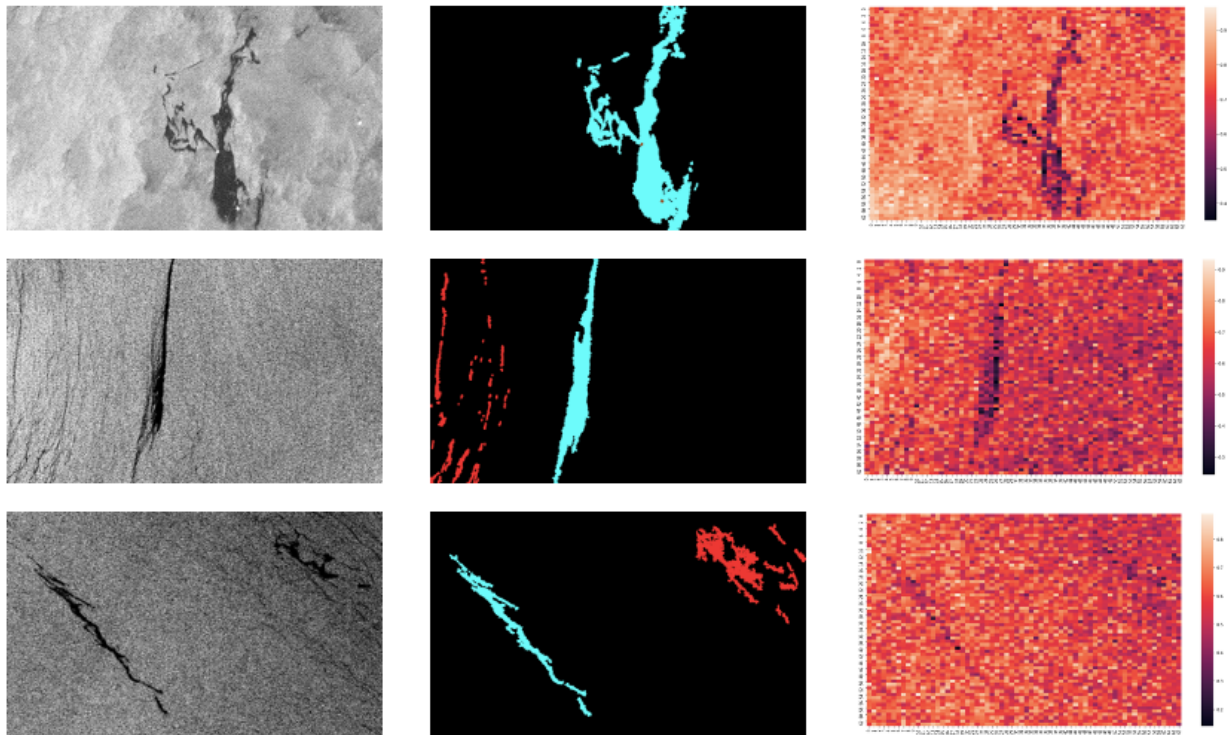


Source: Adapted from Krestenits et al. [7].

4.1 VISUAL ANALYSIS OF THE OIL SPILL DATASET

For visual analysis purposes, we show Figures 16 and 17. We present images with and without oil spills in these figures and the respective masks and heatmaps. The real images and masks are available in the dataset used in this work; the heatmaps were constructed using the proposed q-EFE approach. Such heatmaps were provided by transforming the feature vectors (output of the feature extraction process using the q-EFE (Figure 11)) in matrices preserving the spatial information. We generated the specific heatmaps in Figures 16 and 17 using configuration 2 (Erro! Fonte de referência não encontrada.).

Figure 16 - Images with oil spills and their respective masks and heatmaps

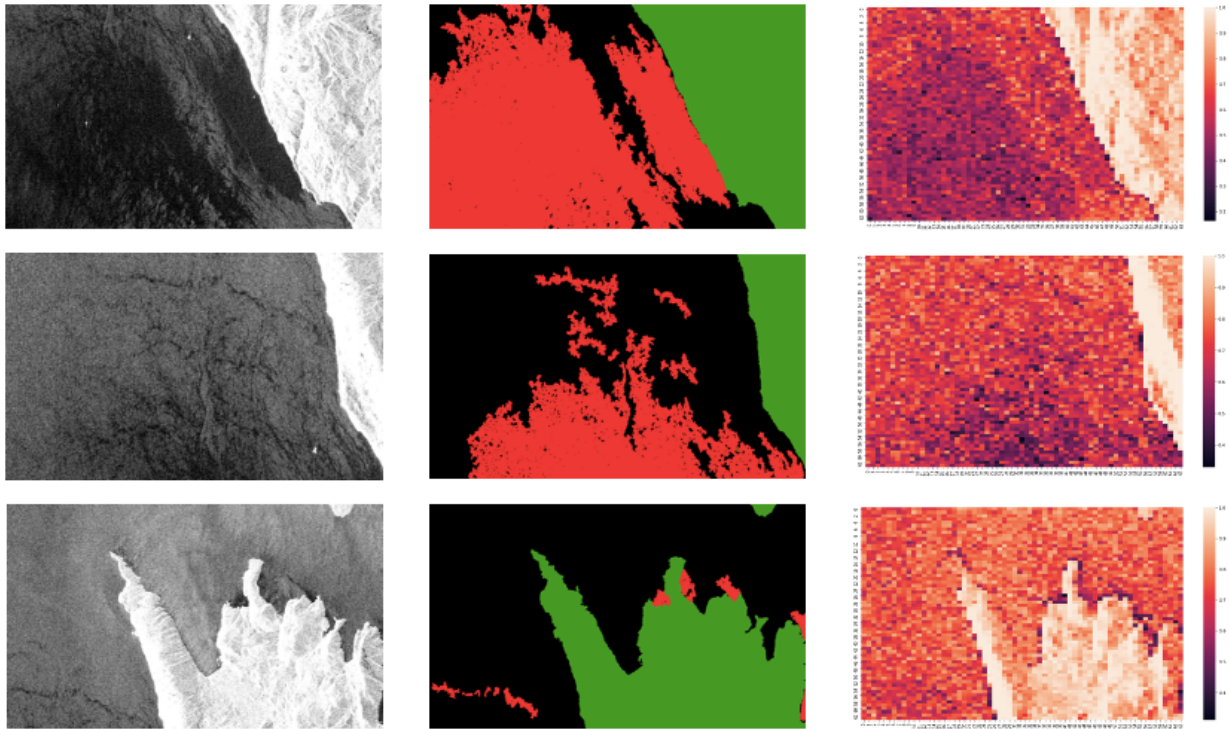


Source: Adapted from Krestenits et al. [7].

In Figure 16, we observe that the oil spill in the original images (left side) are dark spots, considering that such dark spots are represented by the cyan color in the mask (middle). In the heatmaps (right side), the oil spills appear, in general, as darker pixels; this pattern repeats in the three heatmaps displayed. Figure 17 shows the images, masks, and heatmaps without oil spills. In this figure, we see that, for example, land (represented by green in the masks) appears in the heatmaps as lighter pixels. On the other hand, the look-alikes in both classes (with and

without oil spills) appear, in general, also as darker pixels, ratifying the challenge of discriminating between oil spills and look-alikes.

Figure 17 - Images without oil spills and their respective masks and heatmaps



Source: Adapted from Krestenits et al. [7].

5 EXPERIMENTS' RESULTS

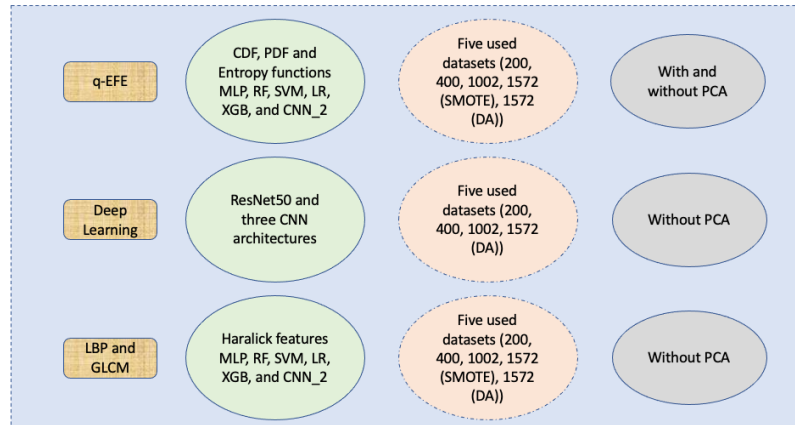
This section will show the obtained results for the several techniques applied in this research. For each specific configuration test, we display the best result; the results of all experiments are in the appendices. Then, the results are analyzed and compared. Considering the ML methods used, **Erro! Fonte de referência não encontrada.** presents the values used in the grid search for all the methods to get the best hyperparameters. Hyperparameters are parameters of the ML and DL models whose values characterize the models actually used in the learning process (e.g., neural network architecture, activation functions, kernel type). Figure 18 presents an outline that synthesizes the ran experiments.

Table 5 - Parameters values for the grid search

Method	Parameters
MLP	Hidden_layer_sizes: [50, 100, 500, 1000] Activation: [identity, logistic, tanh, relu] Solver: [lbgs, SGD, adam] Learning_rate: [(constant), (invscaling), (adaptive)]
RF	N_estimators: [10, 50, 100, 200, 500] Max_features: [auto, sqrt, log2, 5, 10, 30] Max_depth: [2, 8, 16, 32, 64, 128] Min_samples_split: [1, 2, 4, 8, 16, 24] Min_samples_leaf: [1, 2, 5, 10, 15, 30]
SVM	C: [0.1, 1, 100, 1000] Gamma: [auto, 1, 0.01, 0.0001] Kernel: [linear, rbf, sigmoid]
LR	Penalty: [l1, l2] C: [100, 10, 1, 0.1, 0.01, 0.001]
XGB	N_estimators: [100, 500, 1000] Learning_rate: [0.1, 0.05, 0.01] Max_depth: [2, 8, 16, 64, 128] Colsample_bytree: [0.3, 0.8, 1] Gamma: [0, 1, 5]

Source: The author (2022).

Figure 18 - Outline of the experiments

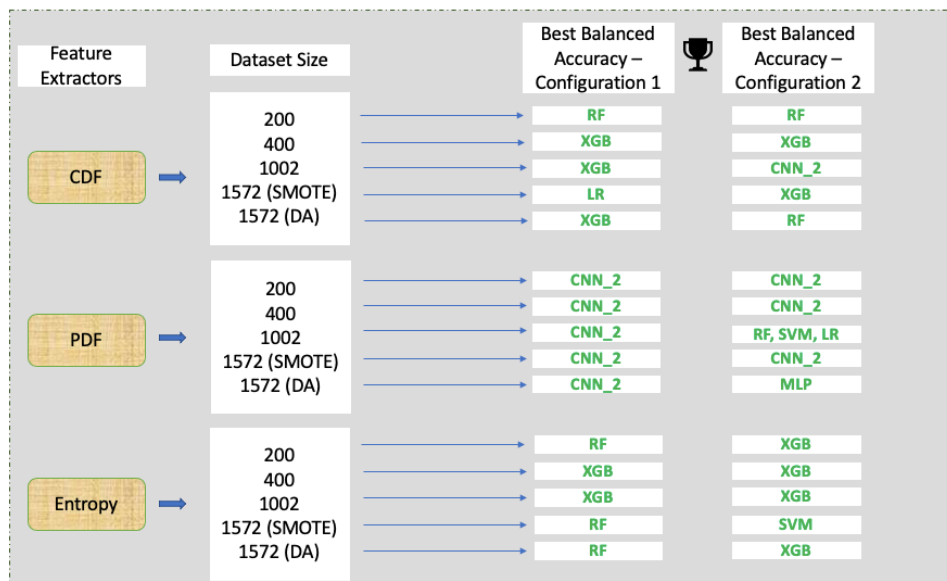


Source: The author (2022).

5.1 q-EFE + ML RESULTS WITH CONFIGURATIONS 1 AND 2

In this subsection, we present the obtained results for the proposed approach, q-Exponential-CV-B, considering configurations 1 and 2, six classification machine learning methods, five dataset sizes to train the models, and three related q-Exponential functions (CDF, PDF, and entropy). In this case, we also tested the three CNN architectures used in this work (Figure 8). However, again, the architecture of the CNN_2 resulted in the best performances from the three CNNs tested. Thus, we only display the CNN_2 results. Figure 19 presents a results summarization of the experiments using configurations 1 and 2. However, the next subsections presents more detailed results.

Figure 19 - Results overview considering configurations 1 and 2



Source: The author (2022).

5.1.1 CDF

Erro! Fonte de referência não encontrada. present the best and worst results using four dataset sizes (200, 400, 1002, 1572 using SMOTE, and 1572 using DA) to generate the respective feature vectors derived from the proposed q-EFE approach using the CDF, considering configurations 1 and 2. This quantity and specific images are the same used to train the CNNs models and to generate the feature vectors of the classical CV techniques. Indeed, the datasets used in this work are the same for all the techniques and classification models tested in this study. For instance, **Erro! Fonte de referência não encontrada.** shows that, considering 200 images and configuration 1, the RF achieved the best BAC (76.11%), and the CNN_2 obtained the worst result, a BAC of 52.47%. Also, using 400 images to generate the respective 400 feature vectors the XGB achieved the best BAC (77.84%), and the LR obtained the worst result, a BAC of only 47.85%. and for 1002 images, the XGB was the best methods. For the experiments using the q-EFE approach, we also applied the PCA to dimensionality reduction before the classification steps. The models that were better without PCA are indicated in the table with “*”.

Table 6 - Metric results achieved with q-EFE approach using four dataset sizes to train the classification models, CDF, and configuration 1 and 2. The best value considering each dataset size and each column is highlighted in bold; worst is underlined. The * means that the results were obtained without PCA.

Dataset size	Configuration	Method	Ac	Pr	Re	Sp	BAC
200	1	RF	0.6727	0.9636	0.6091	0.9130	0.7611
		CNN_2*	0.4000	<u>0.8181</u>	<u>0.3103</u>	<u>0.7391</u>	<u>0.5247</u>
	2	RF	0.5636	0.9148	0.4942	0.8260	0.6601
		CNN_2	<u>0.3545</u>	<u>0.7857</u>	<u>0.2528</u>	<u>0.7391</u>	<u>0.4960</u>
400	1	LR	<u>0.5545</u>	<u>0.7794</u>	0.6091	<u>0.3478</u>	<u>0.4785</u>
		XGB	0.6727	0.9814	<u>0.6022</u>	0.9545	0.7784
	2	LR	0.6181	<u>0.8461</u>	0.6321	<u>0.5652</u>	<u>0.5987</u>
		XGB	0.5636	0.9534	0.4712	0.9130	0.6921
1002	1	RF	<u>0.7454</u>	<u>0.7920</u>	0.9195	0.0869	<u>0.5032</u>
		XGB	0.7818	0.9090	<u>0.8045</u>	0.6956	0.7501
	2	RF	0.7636	<u>0.7904</u>	0.9540	<u>0.0434</u>	<u>0.4987</u>
		XGB	<u>0.7636</u>	0.8144	<u>0.9080</u>	0.2134	0.5627
1572 (SMOTE)	1	RF	0.7272	0.9130	0.7241	0.7391	0.7316
		LR	<u>0.6000</u>	<u>0.7866</u>	<u>0.6781</u>	<u>0.3043</u>	<u>0.4912</u>
	2	SVM	0.7545	<u>0.7830</u>	0.9540	<u>0.0000</u>	<u>0.4770</u>
		XGB	<u>0.6636</u>	0.9032	<u>0.6436</u>	0.7391	0.6914

1572 (DA)	1	XGB	0.7000	0.9090	0.6896	<u>0.7391</u>	0.7143
		CNN_2	<u>0.3363</u>	<u>0.8181</u>	<u>0.2068</u>	0.8260	<u>0.5164</u>
	2	RF	0.7363	<u>0.9027</u>	0.7471	<u>0.6956</u>	0.7213
		CNN_2	<u>0.2727</u>	1.0000	<u>0.0804</u>	1.0000	<u>0.5402</u>

Source: The author (2022).

5.1.2 PDF

Erro! Fonte de referência não encontrada. present the results using four dataset sizes (200, 400, 1002, 1572 using SMOTE, and 1572 using DA) to generate the respective feature vectors derived from the proposed q-EFE approach using the PDF, considering configurations 1 and 2. In this situation, for instance, using 1002 images and configuration 1, the CNN_2 without PCA reached the best result and the MLP presented the worst performance.

Table 7 - Metric results achieved with q-EFE approach using four dataset sizes to train the classification models, PDF, and configurations 1 and 2. The best value considering each dataset size and each column is highlighted in bold; worst is underlined. The * means that the results were obtained without PCA

Dataset size	Configuration	Method	<i>Ac</i>	<i>Pr</i>	<i>Re</i>	<i>Sp</i>	<i>BAC</i>
200	1	RF	<u>0.2090</u>	<u>0.0000</u>	<u>0.0000</u>	1.0000	<u>0.5000</u>
		SVM	<u>0.2090</u>	<u>0.0000</u>	<u>0.0000</u>	1.0000	<u>0.5000</u>
		LR	<u>0.2090</u>	<u>0.0000</u>	<u>0.0000</u>	1.0000	<u>0.5000</u>
		XGB	<u>0.2090</u>	<u>0.0000</u>	<u>0.0000</u>	1.0000	<u>0.5000</u>
		CNN_2*	0.7000	0.8750	0.7241	<u>0.6086</u>	0.6664
	2	MLP	<u>0.2454</u>	<u>0.6428</u>	<u>0.1034</u>	0.7826	<u>0.4430</u>
		CNN_2	0.7090	0.8571	0.7586	<u>0.5217</u>	0.6401
400	1	MLP	<u>0.2454</u>	<u>0.6428</u>	<u>0.1034</u>	0.7826	<u>0.4430</u>
		CNN_2*	0.6909	0.8840	0.7011	<u>0.6521</u>	0.6766
	2	MLP	<u>0.2363</u>	<u>0.6153</u>	<u>0.0919</u>	0.7826	<u>0.4372</u>
		CNN_2	0.6545	0.9016	0.6321	<u>0.7391</u>	0.6856
1002	1	MLP	<u>0.6727</u>	<u>0.8311</u>	0.7356	<u>0.4347</u>	<u>0.5852</u>
		CNN_2*	0.7000	0.8857	<u>0.7126</u>	0.6521	0.6824
	2	RF	<u>0.2090</u>	<u>0.0000</u>	<u>0.0000</u>	1.0000	0.5000
		SVM	<u>0.2090</u>	<u>0.0000</u>	<u>0.0000</u>	1.0000	0.5000
		LR	<u>0.2090</u>	<u>0.0000</u>	<u>0.0000</u>	1.0000	0.5000
		XGB	<u>0.2090</u>	<u>0.0000</u>	<u>0.0000</u>	1.0000	0.5000
		CNN_2	<u>0.2272</u>	<u>0.6000</u>	<u>0.0689</u>	<u>0.8260</u>	<u>0.4475</u>
1572 (SMOTE)	1	MLP	<u>0.2363</u>	<u>0.6153</u>	<u>0.0919</u>	0.7826	<u>0.4372</u>
		CNN_2	0.7000	0.8648	0.7356	<u>0.5652</u>	0.6504
	2	MLP	<u>0.2363</u>	<u>0.6363</u>	<u>0.0804</u>	<u>0.8260</u>	<u>0.4532</u>
		CNN_2	0.6818	0.8939	0.6781	<u>0.6956</u>	0.6869
1572 (DA)	1	MLP	0.6909	<u>0.8533</u>	0.7356	<u>0.5217</u>	<u>0.6286</u>
		CNN_2*	<u>0.5181</u>	0.9047	<u>0.4367</u>	<u>0.8260</u>	0.6314
	2	MLP	0.6454	<u>0.8636</u>	0.6551	<u>0.6086</u>	0.6319
		CNN_2	<u>0.4454</u>	0.9062	<u>0.3333</u>	0.8695	<u>0.6014</u>

Source: The author (2022).

5.1.3 Entropy

Erro! Fonte de referência não encontrada. present the results using four dataset sizes (200, 400, 1002, 1572 using SMOTE, and 1572 using DA) to generate the respective feature vectors derived from the proposed q-EFE approach using the Entropy, considering configurations 1 and 2.

Table 8 - Metric results achieved with q-EFE approach using four dataset sizes to train the classification models, Entropy, and configurations 1 and 2. The best value considering each dataset size and each column is highlighted in bold; worst is underlined. The * means that the results were obtained without PCA

Dataset size	Configuration	Method	<i>Ac</i>	<i>Pr</i>	<i>Re</i>	<i>Sp</i>	<i>BAC</i>
200	1	RF	<u>0.6454</u>	0.8428	<u>0.6781</u>	0.5217	0.5999
		CNN_2	0.7909	<u>0.7909</u>	1.0000	<u>0.0000</u>	<u>0.5000</u>
	2	XGB	0.6272	0.8484	0.6436	<u>0.5652</u>	0.6044
		CNN_2	<u>0.2090</u>	<u>0.0000</u>	<u>0.0000</u>	1.0000	<u>0.5000</u>
400	1	SVM	0.7818	<u>0.7889</u>	0.9885	<u>0.0000</u>	<u>0.4942</u>
		XGB	<u>0.6545</u>	0.8450	0.6896	0.5217	0.6056
	2	XGB	<u>0.6272</u>	0.8382	<u>0.6551</u>	0.5217	0.5884
		CNN_2	0.7909	<u>0.7909</u>	1.0000	<u>0.0000</u>	<u>0.5000</u>
1002	1	RF	<u>0.7636</u>	<u>0.7904</u>	0.9540	<u>0.0434</u>	<u>0.4987</u>
		XGB	<u>0.7636</u>	0.8144	<u>0.9080</u>	0.2134	0.5627
	2	MLP	<u>0.7545</u>	<u>0.7830</u>	0.9540	<u>0.0000</u>	<u>0.4770</u>
		XGB	<u>0.7545</u>	0.8260	<u>0.8735</u>	0.3043	0.5889
1572 (SMOTE)	1	MLP	<u>0.2545</u>	0.7272	<u>0.0919</u>	0.8695	<u>0.4807</u>
		RF	0.6727	0.8311	0.7356	<u>0.4347</u>	0.5852
	2	MLP	0.2818	<u>0.7857</u>	0.1264	0.8695	<u>0.4980</u>
		SVM	0.7181	0.8255	0.8160	0.3474	0.5819
1572 (DA)	1	RF	0.4454	0.8611	0.3563	0.7826	0.5694
		LR	<u>0.3181</u>	<u>0.7307</u>	<u>0.2183</u>	<u>0.6956</u>	<u>0.4570</u>
	2	MLP	0.2272	0.7500	0.0344	0.9565	<u>0.4955</u>
		XGB	0.5545	0.8958	0.4942	0.7826	0.6384

Source: The author (2022).

5.3 LBP + ML AND GLCM + ML RESULTS

In this subsection, we present the experiment results for the oil spill detection using two methods that involves a classical computer vision technique, LBP and GLCM, described in

subsection 2.6. We used the traditional CV techniques to extract features from the SAR oil spill images, and, then, used six machine learning methods, including one deep learning model (CNN), to classify the images. These methods are also used to classify the feature vectors provided by the q-EFE approach proposed in this study. Also, we tested the three CNN architectures used in this work (Figure 8), however, as the architecture of the CNN_2 resulted in the best performances between the CNNs we only display the results of this model. Tables below only shows the best and worst results for each experiment. Table 9 present the results using four dataset sizes (200, 400, 1002, 1572 using SMOTE, and 1572 using DA) to generate the respective feature vectors derived from the LBP CV technique.

Table 9 - Metric results achieved with q-EFE approach using four dataset sizes to train the classification models and LBP. The best value considering each dataset size and each column is highlighted in bold; worst is underlined. The * means that the results were obtained without PCA

Dataset size	Method	<i>Ac</i>	<i>Pr</i>	<i>Re</i>	<i>Sp</i>	<i>BAC</i>
200	LR_LBP	0.8000	0.8155	<u>0.9655</u>	0.1739	0.5697
	CNN_2_LBP	<u>0.7727</u>	<u>0.7870</u>	0.9770	<u>0.0000</u>	<u>0.4885</u>
400	MLP_LBP	<u>0.7909</u>	0.8404	<u>0.9080</u>	0.3478	0.6279
	SVM_LBP	<u>0.7909</u>	<u>0.7909</u>	1.0000	<u>0.0000</u>	<u>0.5000</u>
1002	MLP_LBP	0.7909	<u>0.7909</u>	1.0000	<u>0.0000</u>	<u>0.5000</u>
	RF_LBP	0.7909	<u>0.7909</u>	1.0000	<u>0.0000</u>	<u>0.5000</u>
	SVM_LBP	0.7909	<u>0.7909</u>	1.0000	<u>0.0000</u>	<u>0.5000</u>
	XGB_LBP	<u>0.7727</u>	0.8163	<u>0.9195</u>	0.2173	0.5684
	CNN_2_LBP	0.7909	<u>0.7909</u>	1.0000	<u>0.0000</u>	<u>0.5000</u>
1572 (SMOTE)	MLP_LBP	<u>0.7727</u>	0.8780	<u>0.8275</u>	0.5652	0.6964
	CNN_2_LBP	<u>0.5818</u>	<u>0.7594</u>	<u>0.6896</u>	0.1739	<u>0.4317</u>
1572 (DA)	RF_LBP	0.7909	<u>0.8076</u>	0.9655	<u>0.1304</u>	<u>0.5479</u>
	LR_LBP	0.8272	0.8469	0.9540	0.3478	0.6509

Source: The author (2022).

In Table 10, we present the experiment results for the oil spill detection using GLCM and Haralick features. Table 10 present the results using four dataset sizes (200, 400, 1002, 1572 using SMOTE, and 1572 using DA) to generate the respective feature vectors derived from the GLCM CV technique.

Table 10 - Metric results achieved with q-EFE approach using four dataset sizes to train the classification models and GLCM. The best value considering each dataset size and each column is highlighted in bold; worst is underlined. The * means that the results were obtained without PCA

Dataset size	Method	<i>Ac</i>	<i>Pr</i>	<i>Re</i>	<i>Sp</i>	<i>BAC</i>
200	MLP_GLCM	<u>0.2363</u>	<u>0.6666</u>	<u>0.0689</u>	0.8695	<u>0.4692</u>

	XGB_GLCM	0.6909	0.9344	0.6551	0.8260	0.7406
400	MLP_GLCM	<u>0.2372</u>	<u>0.6700</u>	<u>0.0700</u>	0.8700	<u>0.4700</u>
	XGB_GLCM	0.6953	0.9362	0.6598	<u>0.8297</u>	0.7447
1002	MLP_GLCM	<u>0.7909</u>	<u>0.7909</u>	1.0000	<u>0.0000</u>	<u>0.5000</u>
	RF_GLCM	0.8181	0.8453	<u>0.9425</u>	0.3478	0.6451
	SVM_GLCM	<u>0.7909</u>	<u>0.7909</u>	1.0000	<u>0.0000</u>	<u>0.5000</u>
1572 (SMOTE)	MLP_GLCM	0.7909	<u>0.7909</u>	1.0000	<u>0.0000</u>	<u>0.5000</u>
	RF_GLCM	0.7909	0.8902	<u>0.8390</u>	0.6086	0.7238
1572 (DA)	SVM_GLCM	<u>0.7909</u>	<u>0.7909</u>	1.0000	<u>0.0000</u>	<u>0.5000</u>
	LR_GLCM	0.8545	0.8817	<u>0.9425</u>	0.5217	0.7321

Source: The author (2022).

5.4 CNN AND RESNET50 RESULTS WITH ORIGINAL IMAGES

In this subsection, the results reached using the CNNs and ResNet50 models are shown. Figure 8 shows the applied CNN architectures for comparison purposes. Table 11 show the metrics results achieved by CNN_1, CNN_2, CNN_3 and ResNet50. In this table, we present the best and worst performances for the three CNNs and the Resnet, considering four dataset sizes to train the models (200, 400, 1002, and 1572 images). The set with 1002 images is the original train dataset, the others have the same number of images in each class. For the cases with 200 and 400 images, the examples were randomly chosen. The set with 1572 images was augmented by Data Augmentation (DA). The test set is the same for all the experiments of this work, which includes 110 images, being 87 with oil spills and 23 without oil spills.

Table 11 - Metric results achieved by the three CNNs using four dataset sizes for training, including using Data Augmentation (DA). The best value of each column is highlighted in bold and the worst is underlined

CNN_1	<i>Ac</i>	<i>Pr</i>	<i>Re</i>	<i>Sp</i>	<i>BAC</i>
200	<u>0.5909</u>	0.9375	<u>0.5172</u>	0.8695	0.6934
1002	0.8090	<u>0.8113</u>	0.9885	<u>0.1304</u>	<u>0.5594</u>
CNN_2	<i>Ac</i>	<i>Pr</i>	<i>Re</i>	<i>Sp</i>	<i>BAC</i>
400	<u>0.5636</u>	0.9333	0.4827	0.8695	0.6761
1002	0.7909	<u>0.7909</u>	1.0000	<u>0.0000</u>	<u>0.5000</u>
CNN_3	<i>Ac</i>	<i>Pr</i>	<i>Re</i>	<i>Sp</i>	<i>BAC</i>
1002	0.8272	0.8469	0.9540	0.3478	0.6509
1572 (DA)	<u>0.7454</u>	<u>0.7809</u>	<u>0.9425</u>	<u>0.0000</u>	<u>0.4712</u>
RenNet50	<i>Ac</i>	<i>Pr</i>	<i>Re</i>	<i>Sp</i>	<i>BAC</i>
200	<u>0.7636</u>	0.7961	<u>0.9425</u>	0.0869	0.5147
1572	0.7818	<u>0.7889</u>	0.9885	<u>0.0000</u>	<u>0.4942</u>

Source: The author (2022).

5.5 COMPARISON OF RESULTS

Tables 12 and 13 presents *BAC* mean and standard deviation provided by the q-EFE approach, using configurations 1 and 2, and four data set sizes to train the models (200, 400, 1002, 1572), the sizes 200 and 400 images were chosen to test small balanced datasets, 1002 images comprise the original and imbalanced dataset, and 1572 images is a training dataset composed by the original data plus DA. The mentioned training datasets do not include the test set, which consist of 110 images (88 labeled as ‘oil spills’ and 22 labeled as ‘no oil spills’). All the models analyzed here considered the same test set. Also, all the training processes took 10% of the training data to validation. Ten rounds provided these measures. For the training sets of sizes 200 and 400, we randomly chose an equal number of images from both classes from the original training data set of size 1002, with 786 images “with oil spill” and 216 “without oil spill”. We used DA to complete the training dataset with 1572 images, where 786 images have oil spills and 786 have not, thus only the minority class (“without oil spill”) were augmented.

Table 12 - Descriptive statistics for the balanced accuracy *BAC* obtained in 10 rounds, considering the four training set sizes, the ML models, and configuration 1. The best value of each column and different training sizes are highlighted in bold and the worst is underlined for mean, median and, standard deviation

Dataset size	Model	Mean	Median	Standard Deviation	Maximum	Minimum
200	MLP	0.6333	<u>0.6312</u>	<u>0.0755</u>	0.7496	0.5000
	RF	0.6979	0.7025	0.0173	0.7208	0.6684
	SVM	0.6934	0.6926	0.0037	0.6996	0.6869
	LR	<u>0.6240</u>	0.6380	0.0256	0.6524	0.5814
	XGB	0.6800	0.6921	0.0340	0.7278	0.6224
400	MLP	0.5657	0.5628	<u>0.0355</u>	0.6221	0.4897
	RF	0.6834	0.6965	0.0273	0.7188	0.6364
	SVM	0.7022	0.6984	0.0094	0.7258	0.6926
	LR	<u>0.4804</u>	<u>0.4836</u>	0.0260	0.5277	0.4395
	XGB	0.6724	0.6735	0.0204	0.7073	0.6409
1002	MLP	<u>0.5000</u>	<u>0.5000</u>	0.0000	0.5000	0.50000
	RF	0.5197	0.5278	0.0237	0.5422	0.5000
	SVM	0.5043	0.5000	0.0137	0.5434	0.5000
	LR	0.5278	0.5319	0.0149	0.5377	0.5000
	XGB	0.7348	0.7341	<u>0.0292</u>	0.7833	0.6894
1572	MLP	0.6679	0.6631	0.0284	0.7328	0.6344
	RF	0.6544	0.6420	<u>0.0379</u>	0.7488	0.6151
	SVM	0.7468	0.7488	0.0065	0.7501	0.7283
	LR	<u>0.6263</u>	<u>0.6266</u>	0.0125	0.6484	0.5992
	XGB	0.7399	0.7344	0.0123	0.7648	0.7258

Source: The author (2022).

For the results presented in Tables 12 and 13, configurations 1 and 2, it was used PCA to dimensionality reduction, the features vectors were reduced to 20 components, other component sizes were tested (10, 40, 50, 100) with inferior performances. In Table 12, the results concern the proposed CV methodology based on q -EFE using configuration 1, for 200 training images, RF presented the higher mean for the BAc (69.79%), closely followed by SVM (mean BAc of 69.34%). Also, SVM presented the best performances for 400 and 1572 training images (mean BAc of 74.68% for 1572 images). XGB achieved the best result for the imbalanced dataset (1002 images) and it was the second-best for 1572 images training. Besides, LR achieved the worst BAc for 1572 images (62.63% of BAc. Also, for 1002 images, the MLP had a very poor BAc (50.00%), which means that the model was incapable of distinguishing the images and, consequently, considering all the test examples as the same class (i.e., “with oil spill”). This behavior may lead to conclude that the MLP has a limited ability to deal with the imbalanced data sets. SVM had the smallest standard deviations for most situations, which means that this model is not much sensitive to small input variations. In Table 13, we present the results of configuration 2 used with the q -EFE. Again, SVM was the method that presented the best performances, except for the imbalanced dataset (1002 training images). In this case, the XGB was again the best method. Besides, the results of configuration 1 were superior compared to the results reached by configuration 2, once the best model considering configuration 2 achieved a mean BAc only close to 70%, against a mean BAc close of 75% of the best model considering configuration 1.

Table 13 - Descriptive statistics for the balanced accuracy BAC obtained in 10 rounds, considering the four training set sizes, the ML models, and configuration 2. The best value of each column and different training sizes are highlighted in bold and the worst is underlined for mean, median and, standard deviation

Dataset size	Model	Mean	Median	Standard Deviation	Maximum	Minimum
200	MLP	0.6146	0.6261	<u>0.0601</u>	0.6889	0.5179
	RF	0.6635	0.6636	0.0260	0.6934	0.6051
	SVM	0.6652	0.6496	0.0506	0.7488	0.6119
	LR	<u>0.6010</u>	<u>0.6019</u>	0.0265	0.6409	0.6921
	XGB	0.6457	0.6445	0.0239	0.6921	0.6186
400	MLP	0.5980	0.6027	<u>0.0583</u>	0.6606	0.4500
	RF	0.6893	0.6792	0.0265	0.7291	0.6499
	SVM	0.7018	0.7057	0.0101	0.7086	0.6799
	LR	<u>0.5506</u>	<u>0.5664</u>	0.0358	0.5802	0.4657
	XGB	0.6684	0.6761	0.0423	0.7266	0.5879
1002	MLP	0.5000	<u>0.5000</u>	0.0000	0.5000	0.5000
	RF	<u>0.4984</u>	0.5009	0.0208	0.5249	0.4642
	SVM	0.5178	0.5022	<u>0.0566</u>	0.6779	0.4827
	LR	0.5319	0.5319	0.0000	0.5319	0.5319

	XGB	0.6418	0.6561	0.0505	0.6996	0.4827
1572	MLP	0.6298	0.6254	0.0230	0.6689	0.5934
	RF	<u>0.5649</u>	<u>0.5185</u>	<u>0.0986</u>	0.7156	0.4482
	SVM	0.7069	0.7066	0.0150	0.7328	0.6709
	LR	0.6360	0.6324	0.0122	0.6541	0.6202
	XGB	0.6821	0.6788	0.0205	0.7098	0.6421

Source: The author (2022).

On the other hand, Table 14 presents the performances obtained by the ResNet50, CNNs, LBP + ML, and the GLCM + ML approaches. All models were trained and tested with the same datasets used with the proposed *q-EFE* approach. The performances show that LBP provided the worst results in two situations (200 and 1002 images), and ResNet50 reached the worst results in the other two cases (400 and 1572 images). On the other hand, the methods with GLCM provided better results compared with LBP and DL (ResNet50 and CNNs) for all dataset sizes. XGB using 200 images provided the best GLCM result (74.06%), approaching SVM with *q-EFE*, configuration 1, and 1572 images (74.68%). Also, in Table 14, we do not report standard deviation related to SVM, LR, and XGB because, as we use the same input (feature vectors) and the same seed (random state) in the grid search step, the results for these methods do not change. MLP and RF are initialized randomly, and, even for the same input, the results can vary. For the results corresponding to the *q-EFE* approach, there is a variation for SVM, LR, and XGB because we perform a PCA before each round, and the input data may change.

Table 14 - Mean and standard deviation for BAC obtained in 10 rounds MLP, RF, ResNet50 and CNNs, and BAC reached by SVM, LR, and XGB considering the four training set sizes and the LBP and GLCM CV techniques. The best value of each column and different training sizes are highlighted in bold, and the worst is underlined for mean and standard deviation.

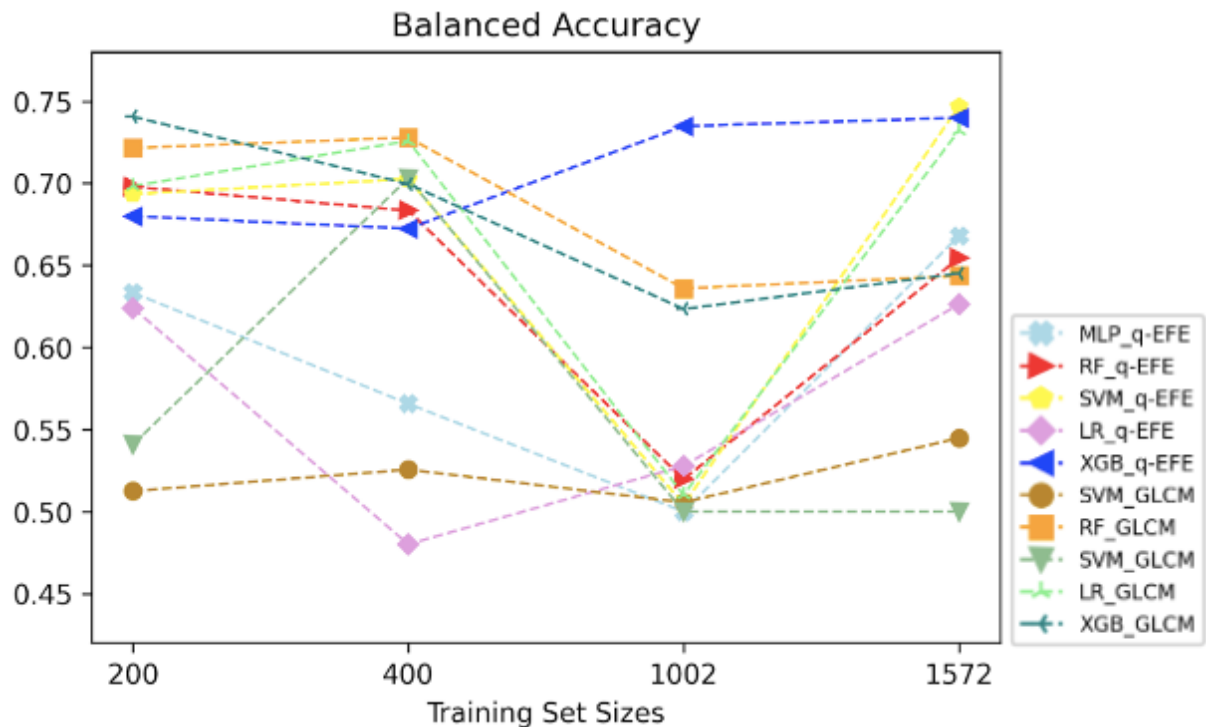
Dataset size	Model	Mean \pm standard deviation	Dataset size	Mean \pm standard deviation
200	MLP_LBP	0.5104 \pm 0.0331	400	0.5015 \pm 0.0050
	RF_LBP	<u>0.5000</u> \pm 0.0000		0.5013 \pm 0.0121
	SVM_LBP	<u>0.5000</u>		<u>0.5000</u>
	LR_LBP	0.5697		0.5422
	XGB_LBP	0.5147		0.5697
	MLP_GLCM	0.5125 \pm 0.0397		0.5256 \pm 0.0539
	RF_GLCM	0.7215 \pm 0.0136		0.7279 \pm 0.0219
	SVM_GLCM	0.5409		0.7031
	LR_GLCM	0.6984		0.7258
	XGB_GLCM	0.7406		0.6996
	CNN_1	0.6873 \pm 0.0219		0.6359 \pm 0.0253
	CNN_2	0.6725 \pm <u>0.0621</u>		0.6859 \pm 0.0380
	CNN_3	0.6559 \pm 0.0573		0.5927 \pm <u>0.0647</u>
	ResNet50	0.5147 \pm 0.0266		<u>0.4953</u> \pm 0.0343
1002	MLP_LBP	0.5000 \pm 0.0000	1572	0.5747 \pm <u>0.0673</u>
	RF_LBP	<u>0.4994</u> \pm 0.0018		0.5727 \pm 0.0513
	SVM_LBP	0.5000		0.6349

LR_LBP	0.5582	0.6509
XGB_LBP	0.5684	0.6496
MLP_GLCM	0.5056 ± 0.0165	0.5448 ± 0.0545
RF_GLCM	0.6360 ± 0.0185	0.6436 ± 0.0036
SVM_GLCM	0.5000	0.5000
LR_GLCM	0.5102	0.7321
XGB_GLCM	0.6234	0.6451
CNN_1	0.6325 ± 0.0300	0.6299 ± 0.0254
CNN_2	0.6557 ± 0.0483	0.5715 ± 0.0472
CNN_3	0.5473 ± 0.0679	0.5169 ± 0.0257
ResNet50	0.5110 ± 0.0186	<u>0.4942</u> ± 0.0224

Source: The author (2022).

Figure 20 presents the BAc means for q -EFE using configuration 1 and the results with GLCM. We plotted such results because they were the best ones. In general, the GLCM presented the higher BAc means for 200 and 400 images, and the q -EFE presented the higher BAc means for 1002 and 1572 images. However, the SVM using the q -EFE outperformed the other methods with mean BAc of 74.68% for 1572 images.

Figure 20 - Average balanced accuracy of 10 rounds for the test set, considering the four dataset sizes, the q -EFE and the GLCM



Source: The author (2022).

The results indicate that the SVM had the best performance for detecting oil spills in SAR imagery, followed by XGB, using the proposed q -EFE approach, configuration 1 and DA. The

comparisons indicate that this new methodology for identifying oil spills in images is promising and represents an advance in this field.

Besides, we made a test to evaluate the generalization power of the best models of configuration 1 (SVM and XGB). In the grayscale conversion step (Figure 10), we used a continuous grayscale instead of a discrete one to convert the images from the test set. We obtained a BAc of 78.97% using SVM and a BAc of 73.86% using XGB. Taking the regular accuracy into consideration, we obtained 82.72% using SVM and 66.36% with XGB. Thus, the SVM was more powerful in the generalization task than the XGB. The generalization ability assessment is essential, as, in practice, new images may have unseen patterns in the training stage.

Also, we performed an analysis of the scalability of the q-EFE. Table 15 shows the required time to extract features and train the best-tested models presented in Figure 20, and the DL models applied for comparison purposes, according to the software configurations previously presented. Thus, analyzing Table 15, we can see that the q-EFE is the method requiring the greatest computational effort in terms of time. However, the time issue is justified by the performance reached with q-EFE. And this disadvantage is only verified for the feature extraction and to train the model. As with all ML-based models, once the model is already trained, the test step is quick.

Table 15 - Required time to extract features and train the best models presented in this work

Models (1572 images)	SVM q-EFE	LR_GLCM	CNN_1	CNN_2	CNN_3	ResNet50
Time in seconds	81744	110	600	1300	1800	2800

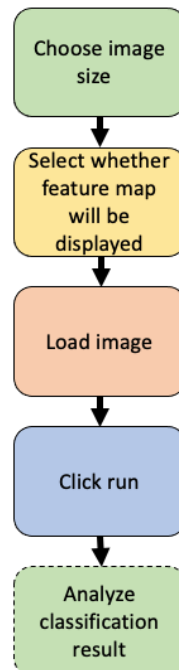
Source: The author (2022).

6 WEB APP FOR OIL SPILL DETECTION

We also designed a simple web app for oil spill detection in this work. This web app relies on a novel feature extraction technique based on the q-Exponential combined with SVM or LR. The following steps are necessary to use the web app (Figure 21):

- First, the image is loaded in a chosen size by the user. This resize process is performed by the “Pillow” library using the “resize” function. Then, the image is converted to a grayscale representation utilizing a function (mean) of the “numpy” library.
- Then, the user must select whether the feature map related to the inputted image should be displayed after the q-Exponential-based feature extraction.
- To display this feature map and to perform the classification using the SVM or LR model, we apply the referred technique to extract features based on the q-Exponential CDF. Then, the chosen ML model is used to perform the classification and predict the class.
- Finally, the user will see the classification results displayed.

Figure 21 - Steps of use of the Oil Spill Detection Web App



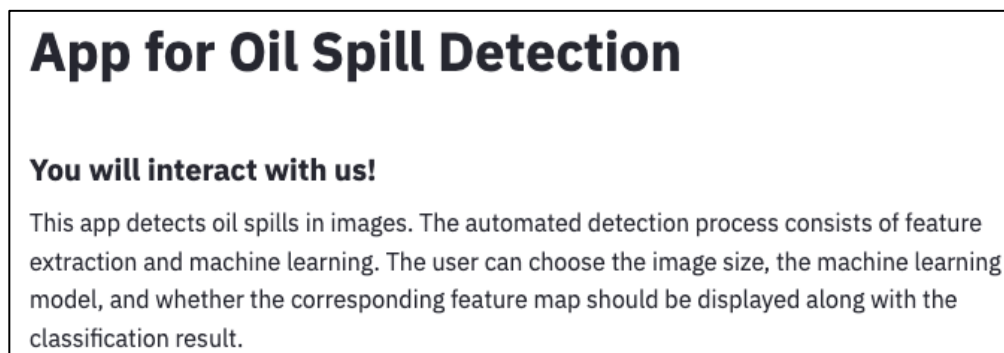
Source: The author (2022).

In this context, the feature extraction performed in this proposed system is the very same presented in Section 3, which is based on the q-Exponential distribution computes the CDF

using patches of the image. We maximize the q-Exponential log-likelihood function for each patch to estimate the corresponding parameters. We then take the mean pixel of the patch and calculate the corresponding CDF. If the original q-Exponential log-likelihood presents estimation problems, the corrected function is used, as detailed in [203]. After this process, we have the feature vectors used to train and test an SVM or an LR model.

In Figure 22, we show the web app initial view, with a summary description for the user.

Figure 22 - This is the first view encountered by the user of the web app for oil spill detection



App for Oil Spill Detection

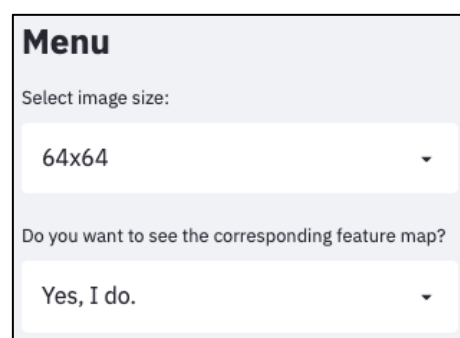
You will interact with us!

This app detects oil spills in images. The automated detection process consists of feature extraction and machine learning. The user can choose the image size, the machine learning model, and whether the corresponding feature map should be displayed along with the classification result.

Source: The author (2022).

In Figure 23, we present how the user must start to interact with the system. He must choose the image size (“64x64” or “256x256”). If the user does not choose any of these two options, the system will set the default value of the image size, which is “64x64”. Then, the user must choose whether they want to see the corresponding feature map. Again, if the user does not choose any of the two options, the system will set the default choice for this question, which is “Yes, I do.”

Figure 23 - View of the left side of the oil spill web app, where the selects the image size and chooses whether the feature map should or not be displayed



Menu

Select image size:

64x64

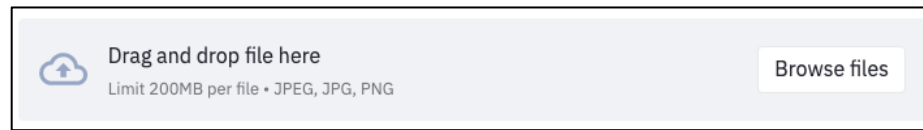
Do you want to see the corresponding feature map?

Yes, I do.

Source: The author (2022).

Then, the user must upload an image from their computer. Figure 24 presents the local to upload the image.

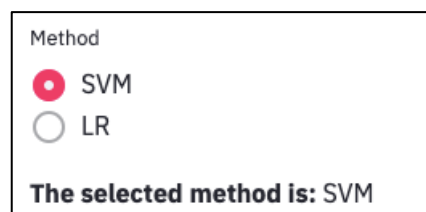
Figure 24 - Local where the user must upload an image



Source: The author (2022).

After the upload step, the user chooses which machine learning method will perform the classification task (Figure 25).

Figure 25 - Selection of the machine learning method



Source: The author (2022).

Next, to start the feature extraction and classification processes, the user must click in the “Run” button (Figure 26).

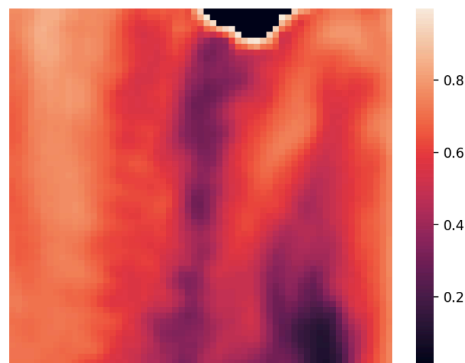
Figure 26 - Button Run to start the process



Source: The author (2022).

When the button “Run ” is clicked, the uploaded image is shown on the screen, as the image size. The uploaded image will be resized according to the image size chosen by the user in the initial process. After that, the process of feature extraction begins and the user must wait for the process to be completed to see the result. While the feature extraction is processed, appears on the screen a message for the user to wait for the end of the process. If the user did choose to see the correspondent feature map, it would be presented on the screen before the classification result is shown. Then, the classification result appears for the user plus the predicted class and probability of such result using an SVM or LR pre-trained model. Figure 27 presents a feature map generated by our proposed interface.

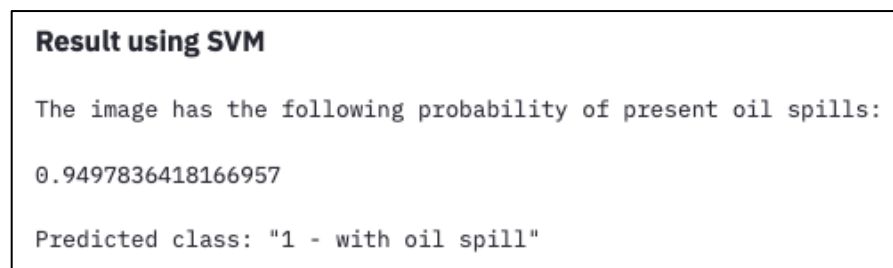
Figure 27 - Feature map generated by the oil spill detection web app



Source: The author (2022).

Besides, Figure 28 shows how the classification result, the chosen machine model, the probability of such outcome, and the predicted class are displayed for the user.

Figure 28 - Classification result, probability of the result according to pre-trained ML model, and predicted class



Source: The author (2022).

7 CONCLUSION

This work proposed a new images feature extractor based on q-Exponential distribution that aimed to detect oil spills in SAR images. Indeed, the results obtained by our proposed methodology outperformed other well-known and succeed computer vision and deep learning models.

Oil spills in seas and oceans are a specific environmental disaster caused by humans with a high potential for damage. Such events can cause impacts that can be seen in the marine environment, human health, and the economy. In this context, the whole environment can be negatively and extensively affected. It may cause the death of thousands of plants and animals, and contaminate huge areas, including the atmospheric air. Besides, oil spills can directly affect the health of the oil clean-up workers. They can also cause significant economic exes with clean-up activities, government tax, and losses of fishermen and tourism businesses. Because of this, it is essential for rapid oil spill identification. In this context, this research proposed a novel feature extractor based on the q-Exponential probabilistic distribution to comprise, along with machine learning models, a computer vision methodology to detect oil spills automatically and, hence, avoid more significant catastrophic impacts.

The oil spill detection is generally made using SAR images. This specific radar is very useful for the oil spill detection task because it is a kind of radar capable of differentiating the oil and seawater characteristics. However, even using SAR imagery, one big challenge in this task involves the common look-alike occurrences. Such phenomenon is denominated as similar to oil spill events, but it is natural phenomena such as wind shadow and wave movements.

In this context, our proposed feature extractor based on the q-Exponential distribution (q-EFE) was developed to handle the look-alike problems and detect the oil spills accurately. The q-Exponential model is used in the reliability field also because of its capability to fit rare events well (power-law behavior). In reliability, rare events are events with a low occurrence probability, e.g., systems that rarely fail. In the oil spill context, the oil spills generally appear in the images as small dark spots; such image pieces can be considered rare events because they are small parts compared to the whole image. This is the motivation to use the q-Exponential distribution in this work. The quantitative measuring of the oil spill average size and its relation with the whole image is a topic of ongoing research.

Thus, we computed the CDF, PDF, and entropy functions according to the methodology proposed in this work, detailed the feature extraction from SAR images, and performed the classification. We executed several experiments using such q-Exponential related functions, and the CDF achieved the best results compared to PDF and entropy functions. These best results can be explained by the fact that CDF values are naturally within the interval $[0, 1]$, which could be important to the tested classifiers.

Besides, we also performed several experiments using ResNet50, other deep learning models (three architectures of CNNs) and two classical CV features extractors (LBP and GLCM). Also, we applied five ML models (MLP, RF, SVM, LR, and XGB) and one DL model (CNN) combined with the proposed q-EFE and combined with LBP and GLCM. The CDF outperformed the applied methods considering the entire dataset without data augmentation techniques (1002 images) and considering the dataset plus data augmentation (1572 images). The SVM and XGB were the best classifiers tested; SVM was slightly superior. We also performed a generalization test considering these two best ML models, SVM and XGB, and the SVM was superior.

The results indicate that this new methodology for identifying oil spills in images is promising and represents an advance in this field. It can be used online to monitor oil spills in seas, making possible damage reduction considering a risk framework and the safety barriers to prevent and/or mitigate oil spill consequences. The results showed that there is space for improvements in the *BAC* levels. We believe that, with more data, we can improve the achieved performances.

7.1 WORK LIMITATIONS

Limitations encountered during the development process of this work are summarized below. We considered:

- The grayscale range (0 - 255);
- One color channel;
- Patches of size 4 x 4;
- A single dataset. Therefore, despite being promising, the obtained results are specifically related to it and are not readily extendible to other datasets.

7.2 FUTURE WORKS

In developing this work, we found some new perspectives to improve the results achieved in this research. In the following, we list topics of ongoing and/or future research:

- To test other configurations beyond the tested in this research (different patch sizes, different strides, image sizes, color ranges, color channels, etc.), aiming for better results;
- To apply the proposed approach to other oil spill datasets, including datasets constructed by images generated with another type of radars and possibly datasets of images generated by drones;
- To test other deep learning models such as Generative Adversarial Networks (GANs), Long Short-Term Memory (LSTM), and others.
- To research works that propose semiautomatic methodologies to identify oil spills;
- To test other probabilistic distributions as replacement of the q-Exponential distribution to try to find unknown possibilities of feature extraction power and to perform comparisons;
- To develop an approach to perform semantic segmentation using the proposed feature extractor and others for binary and multiclass problems;
- To perform tests using images generated by other types of radar, such as Side Looking Airborne Radar (SLAR) and compare with the SAR imagery results;
- To test unsupervised methods (e.g., Variational Autoencoder) to handle the imbalanced dataset problem.
- To perform a study to measure the oil spill mean size and compare it with the whole image.
- To evaluate the use of Unmanned Aerial Vehicle (UAV) to generate oil spills images;
- To apply the proposed approach to address images from other risk and reliability contexts, such as concrete images for crack detection.
- To improve the proposed web app for oil spill detection to make possible the oil spill source identification.

REFERENCES

- [1] I. L. Briggs and B. C. Briggs, 'Chapter 10 - Petroleum Industry Activities and Human Health', in *The Political Ecology of Oil and Gas Activities in the Nigerian Aquatic Ecosystem*, P. E. Ndimele, Ed. Academic Press, 2018, pp. 143–147. doi: 10.1016/B978-0-12-809399-3.00010-0.
- [2] R. Huz, M. Lastra, and J. López, 'Other Environmental Health Issues: Oil Spill', 2018. doi: 10.1016/B978-0-12-409548-9.11156-X.
- [3] C. Brekke and A. H. S. Solberg, 'Oil spill detection by satellite remote sensing', *Remote Sens. Environ.*, vol. 95, no. 1, pp. 1–13, Mar. 2005, doi: 10.1016/j.rse.2004.11.015.
- [4] D. Ansell, B. Dicks, C. Guenette, T. Moller, R. Santner, and I. White, 'A Review of the Problems Posed By Spills of Heavy Fuel Oils', *Int. Oil Spill Conf. Proc.*, vol. 2001, Mar. 2001, doi: 10.7901/2169-3358-2001-1-591.
- [5] H. Singh, N. Bhardwaj, S. K. Arya, and M. Khatri, 'Environmental impacts of oil spills and their remediation by magnetic nanomaterials', *Environ. Nanotechnol. Monit. Manag.*, vol. 14, p. 100305, Dec. 2020, doi: 10.1016/j.enmm.2020.100305.
- [6] P. Krata and J. Jachowski, 'Towards a modification of a regulatory framework aiming at bunker oil spill prevention from ships – A design aspect of bunker tanks vents location guided by CFD simulations', *Reliab. Eng. Syst. Saf.*, vol. 208, p. 107370, Dec. 2020, doi: 10.1016/j.ress.2020.107370.
- [7] M. Krestenitis, G. Orfanidis, K. Ioannidis, K. Avgerinakis, S. Vrochidis, and I. Kompatsiaris, 'Oil Spill Identification from Satellite Images Using Deep Neural Networks', *Remote Sens.*, vol. 11, p. 1762, Jul. 2019, doi: 10.3390/rs11151762.
- [8] J. Beyer, H. C. Trannum, T. Bakke, P. V. Hodson, and T. K. Collier, 'Environmental effects of the Deepwater Horizon oil spill: A review', *Mar. Pollut. Bull.*, vol. 110, no. 1, pp. 28–51, Sep. 2016, doi: 10.1016/j.marpolbul.2016.06.027.
- [9] X. Shi, Y. Wang, M. Luo, and C. Zhang, 'Assessing the feasibility of marine oil spill contingency plans from an information perspective', *Saf. Sci.*, vol. 112, pp. 38–47, Feb. 2019, doi: 10.1016/j.ssci.2018.09.014.
- [10] J. Yang *et al.*, 'Decision Fusion of Deep Learning and Shallow Learning for Marine Oil Spill Detection', *Remote Sens.*, vol. 14, no. 3, Art. no. 3, Jan. 2022, doi: 10.3390/rs14030666.
- [11] M. Krestenitis, G. Orfanidis, K. Ioannidis, K. Avgerinakis, S. Vrochidis, and I. Kompatsiaris, *Early Identification of Oil Spills in Satellite Images Using Deep CNNs*. 2019, p.

435. doi: 10.1007/978-3-030-05710-7_35.

[12] T. Webler and F. Lord, 'Planning for the Human Dimensions of Oil Spills and Spill Response', *Environ. Manage.*, vol. 45, pp. 723–38, Feb. 2010, doi: 10.1007/s00267-010-9447-9.

[13] M. A. D'Andrea and G. K. Reddy, 'The Development of Long-Term Adverse Health Effects in Oil Spill Cleanup Workers of the Deepwater Horizon Offshore Drilling Rig Disaster', *Front. Public Health*, vol. 6, 2018, doi: 10.3389/fpubh.2018.00117.

[14] L. C. Ribeiro, K. de Souza, E. Domingues, and A. Magalhaes, 'Blue water turns black: economic impact of oil spill on tourism and fishing in Brazilian Northeast', *Curr. Issues Tour.*, vol. 24, May 2020, doi: 10.1080/13683500.2020.1760222.

[15] K. Topouzelis, 'Oil Spill Detection by SAR Images: Dark Formation Detection, Feature Extraction and Classification Algorithms', *Sensors*, vol. 8, Oct. 2008, doi: 10.3390/s8106642.

[16] M. E. de Araújo, C. W. N. Ramalho, and P. W. de Melo, 'Artisanal fishers, consumers and the environment: immediate consequences of the oil spill in Pernambuco, Northeast Brazil', *Cad. Saude Publica*, vol. 36, no. 1, p. e00230319, 2020, doi: 10.1590/0102-311X00230319.

[17] D. C. Zacharias, C. M. Gama, and A. Fornaro, 'Mysterious oil spill on Brazilian coast: Analysis and estimates', *Mar. Pollut. Bull.*, vol. 165, p. 112125, Apr. 2021, doi: 10.1016/j.marpolbul.2021.112125.

[18] D. C. Zacharias, C. M. Gama, J. Harari, R. P. da Rocha, and A. Fornaro, 'Mysterious oil spill on the Brazilian coast – Part 2: A probabilistic approach to fill gaps of uncertainties', *Mar. Pollut. Bull.*, vol. 173, p. 113085, Dec. 2021, doi: 10.1016/j.marpolbul.2021.113085.

[19] W. Røed and T. Bjerga, *Holistic understanding and clarification of environmental safety barriers in the oil and gas industry*. 2017, p. 191. doi: 10.1201/9781315210469-164.

[20] S. Sklet, 'Safety barriers: Definition, classification, and performance', *J. Loss Prev. Process Ind.*, vol. 19, no. 5, pp. 494–506, Sep. 2006, doi: 10.1016/j.jlp.2005.12.004.

[21] R. Bubbico, S. Lee, D. Moscati, and N. Paltrinieri, 'Dynamic assessment of safety barriers preventing escalation in offshore Oil&Gas', *Saf. Sci.*, vol. 121, pp. 319–330, Jan. 2020, doi: 10.1016/j.ssci.2019.09.011.

[22] H. Meng, G. Chen, X. Liu, and Y. Zhu, 'An oil recovery technique for emergency response in deepwater blowout accidents', *Saf. Sci.*, vol. 113, pp. 134–143, Mar. 2019, doi: 10.1016/j.ssci.2018.11.011.

[23] G. de A. Carvalho, P. J. Minnett, N. F. F. Ebecken, and L. Landau, 'Classification of Oil Slicks and Look-Alike Slicks: A Linear Discriminant Analysis of Microwave, Infrared, and Optical Satellite Measurements', *Remote Sens.*, vol. 12, no. 13, Art. no. 13, Jan. 2020, doi:

10.3390/rs12132078.

- [24] W. Tian, Y. Shao, J. Yuan, S. Wang, and Y. Liu, ‘An experiment for oil spill recognition using RADARSAT-2 image’, in *2010 IEEE International Geoscience and Remote Sensing Symposium*, Jul. 2010, pp. 2761–2764. doi: 10.1109/IGARSS.2010.5652898.
- [25] A. Akkartal and F. Sunar, ‘THE USAGE OF RADAR IMAGES IN OIL SPILL DETECTION’, 2008. [/paper/THE-USAGE-OF-RADAR-IMAGES-IN-OIL-SPILL-DETECTION-Akkartal-Sunar/393716a468006991b6cd8caa492421b5e28799ca](#) (accessed Apr. 29, 2021).
- [26] Y. Shu, J. Li, H. Yousif, and G. Gomes, ‘Dark-spot detection from SAR intensity imagery with spatial density thresholding for oil-spill monitoring’, *Remote Sens. Environ.*, vol. 114, no. 9, pp. 2026–2035, Sep. 2010, doi: 10.1016/j.rse.2010.04.009.
- [27] Z. Jiao, C. Jia, and Y. Cai, ‘A new approach to oil spill detection that combines deep learning with unmanned aerial vehicles’, *Comput. Ind. Eng.*, vol. 135, Nov. 2018, doi: 10.1016/j.cie.2018.11.008.
- [28] S. Singha, T. Bellerby, and O. Trieschmann, ‘Satellite Oil Spill Detection Using Artificial Neural Networks’, *Sel. Top. Appl. Earth Obs. Remote Sens. IEEE J. Of*, vol. 6, pp. 2355–2363, Dec. 2013, doi: 10.1109/JSTARS.2013.2251864.
- [29] S. Tong, X. Liu, Q. Chen, Z. Zhang, and G. Xie, ‘Multi-feature based ocean oil spill detection for polarimetric SAR data using random forest and the self-similarity parameter’, *Remote Sens.*, vol. 11, no. 4, Art. no. 4, 2019, doi: 10.3390/rs11040451.
- [30] T. De Kerf, J. Gladines, S. Sels, and S. Vanlanduit, ‘Oil Spill Detection Using Machine Learning and Infrared Images’, *Remote Sens.*, vol. 2020, p. 4090, Dec. 2020, doi: 10.3390/rs12244090.
- [31] T. Ojala, M. Pietikainen, and T. Maenpaa, ‘Multiresolution gray-scale and rotation invariant texture classification with local binary patterns’, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002, doi: 10.1109/TPAMI.2002.1017623.
- [32] R. Haralick, K. Shanmugam, and I. Dinstein, ‘Textural Features for Image Classification’, *IEEE Trans Syst Man Cybern*, vol. SMC-3, pp. 610–621, Jan. 1973.
- [33] S. Murala, R. P. Maheshwari, and B. Raman, ‘Local Tetra Patterns: A New Feature Descriptor for Content-Based Image Retrieval’, *IEEE Trans. Image Process. Publ. IEEE Signal Process. Soc.*, vol. 21, pp. 2874–86, May 2012, doi: 10.1109/TIP.2012.2188809.
- [34] M. Q. Huang, J. Ninić, and Q. B. Zhang, ‘BIM, machine learning and computer vision techniques in underground construction: Current status and future perspectives’, *Tunn. Undergr. Space Technol.*, vol. 108, p. 103677, Feb. 2021, doi: 10.1016/j.tust.2020.103677.

- [35] A. Darko, A. P. C. Chan, M. A. Adabre, D. J. Edwards, M. R. Hosseini, and E. E. Ameyaw, 'Artificial intelligence in the AEC industry: Scientometric analysis and visualization of research activities', *Autom. Constr.*, vol. 112, p. 103081, Apr. 2020, doi: 10.1016/j.autcon.2020.103081.
- [36] M. R. Ibrahim, J. Haworth, and T. Cheng, 'Understanding cities with machine eyes: A review of deep computer vision in urban analytics', *Cities*, vol. 96, p. 102481, Jan. 2020, doi: 10.1016/j.cities.2019.102481.
- [37] D. Mery, E. Svec, M. Arias, V. Rizzo, J. Saavedra, and S. Banerjee, 'Modern Computer Vision Techniques for X-Ray Testing in Baggage Inspection', *IEEE Trans. Syst. Man Cybern. Syst.*, vol. PP, pp. 1–11, Dec. 2016, doi: 10.1109/TSMC.2016.2628381.
- [38] D. Liu, D. Zhang, Y. Song, H. Huang, and W. Cai, 'Panoptic Feature Fusion Net: A Novel Instance Segmentation Paradigm for Biomedical and Biological Images', *IEEE Trans. Image Process.*, vol. 30, pp. 2045–2059, 2021, doi: 10.1109/TIP.2021.3050668.
- [39] M. Ghahremani, Y. Liu, and B. Tiddeman, 'FFD: Fast Feature Detector', *IEEE Trans. Image Process.*, vol. 30, pp. 1153–1168, 2021, doi: 10.1109/TIP.2020.3042057.
- [40] X. Xiao, Y. Chen, Y.-J. Gong, and Y. Zhou, 'Low-Rank Preserving t-Linear Projection for Robust Image Feature Extraction', *IEEE Trans. Image Process.*, vol. 30, pp. 108–120, 2021, doi: 10.1109/TIP.2020.3031813.
- [41] K. Murphy, A. Torralba, D. Eaton, and W. Freeman, *Object Detection and Localization Using Local and Global Features*, vol. 4170. 2006, p. 400. doi: 10.1007/11957959_20.
- [42] N. O. Mahony *et al.*, 'Deep Learning vs. Traditional Computer Vision', *ArXiv191013796 Cs*, vol. 943, 2020, doi: 10.1007/978-3-030-17795-9.
- [43] C. Zhang *et al.*, 'Interpreting and Improving Adversarial Robustness of Deep Neural Networks With Neuron Sensitivity', *IEEE Trans. Image Process.*, vol. 30, pp. 1291–1304, 2021, doi: 10.1109/TIP.2020.3042083.
- [44] S. Nikan *et al.*, 'PWD-3DNet: A Deep Learning-Based Fully-Automated Segmentation of Multiple Structures on Temporal Bone CT Scans', *IEEE Trans. Image Process.*, vol. 30, pp. 739–753, 2021, doi: 10.1109/TIP.2020.3038363.
- [45] X. Huang, B. Zhang, W. Perrie, Y. Lu, and C. Wang, 'A novel deep learning method for marine oil spill detection from satellite synthetic aperture radar imagery', *Mar. Pollut. Bull.*, vol. 179, p. 113666, Jun. 2022, doi: 10.1016/j.marpolbul.2022.113666.
- [46] G. Marcus, 'Deep Learning: A Critical Appraisal', Jan. 2018.
- [47] J. Gros, C. Reddy, R. Nelson, S. Socolofsky, and J. Arey, 'Simulating gas-liquid-water partitioning and fluid properties of petroleum mixtures under pressure: Implications for deep-

sea blowouts’, *Environ. Sci. Technol.*, vol. 50, Apr. 2016, doi: 10.1021/acs.est.5b04617.

[48] J. F. Karinen, ‘Petroleum in the deep sea environment: Potential for damage to biota’, *Environ. Int.*, vol. 3, no. 2, pp. 135–144, Jan. 1980, doi: 10.1016/0160-4120(80)90047-1.

[49] F. Aguilera, J. Mendez, E. Pásaro, and B. Laffon, ‘Review on the effects of exposure to spilled oils on human health’, *J. Appl. Toxicol. JAT*, vol. 30, pp. 291–301, May 2010, doi: 10.1002/jat.1521.

[50] C. Clayton, ‘PETROLEUM GEOLOGY | Chemical and Physical Properties’, in *Encyclopedia of Geology*, 2005, pp. 248–260. doi: 10.1016/B0-12-369396-9/00246-X.

[51] L. L. Barbosa, F. V. C. Kock, R. C. Silva, J. C. C. Freitas, V. Lacerda, and E. V. R. Castro, ‘Application of Low-Field NMR for the Determination of Physical Properties of Petroleum Fractions’, *Energy Fuels*, vol. 27, no. 2, pp. 673–679, Feb. 2013, doi: 10.1021/ef301588r.

[52] C. Russo *et al.*, ‘Investigation on chemical and structural properties of coal- and petroleum-derived pitches and implications on physico-chemical properties (solubility, softening and coking)’, *Fuel*, vol. 245, pp. 478–487, Jun. 2019, doi: 10.1016/j.fuel.2019.02.040.

[53] National Research Council (US) Committee on Oil in the Sea: Inputs, Fates, and Effects, *Oil in the Sea III: Inputs, Fates, and Effects*. Washington (DC): National Academies Press (US), 2003. Accessed: Apr. 27, 2021. [Online]. Available: <http://www.ncbi.nlm.nih.gov/books/NBK220703/>

[54] U. H. Yim *et al.*, ‘Fingerprint and weathering characteristics of stranded oils after the Hebei Spirit oil spill’, *J. Hazard. Mater.*, vol. 197, pp. 60–9, Dec. 2011, doi: 10.1016/j.jhazmat.2011.09.055.

[55] A. Hoang, N. Duong, Nguyen, P. Viet, and D. Nguyen, ‘A Report of Oil Spill Recovery Technologies’, *Int. J. Appl. Eng. Res.*, vol. 13, No. 7, Jan. 2018.

[56] Doerffer, J. W., ‘Oil Spill Response in the Marine Environment - 1st Edition’. <https://www.elsevier.com/books/oil-spill-response-in-the-marine-environment/doerffer/978-0-08-041000-5> (accessed Apr. 27, 2021).

[57] H. Santos Silva *et al.*, ‘The role of metalloporphyrins on the physical-chemical properties of petroleum fluids’, *Fuel*, vol. 188, pp. 374–381, Jan. 2017, doi: 10.1016/j.fuel.2016.10.065.

[58] P. Eckle, P. Burgherr, and E. Michaux, ‘Risk of large oil spills: a statistical analysis in the aftermath of Deepwater Horizon’, *Environ. Sci. Technol.*, vol. 46, no. 23, pp. 13002–13008, Dec. 2012, doi: 10.1021/es3029523.

[59] S. Chang, J. Stone, K. Demes, and M. Piscitelli, ‘Consequences of oil spills: a review

and framework for informing planning’, *Ecol. Soc.*, vol. 19, no. 2, May 2014, doi: 10.5751/ES-06406-190226.

[60] M. R. Garcia, A. P. Cattani, P. da Cunha Lana, R. C. L. Figueira, and C. C. Martins, ‘Petroleum biomarkers as tracers of low-level chronic oil contamination of coastal environments: A systematic approach in a subtropical mangrove’, *Environ. Pollut.*, vol. 249, pp. 1060–1070, Jun. 2019, doi: 10.1016/j.envpol.2019.03.006.

[61] A. Bayat, S. F. Aghamiri, A. Moheb, and G. R. Vakili-Nezhaad, ‘Oil Spill Cleanup from Sea Water by Sorbent Materials’, *Chem. Eng. Technol.*, vol. 28, no. 12, pp. 1525–1528, 2005, doi: <https://doi.org/10.1002/ceat.200407083>.

[62] N. Brakorenko and T. Korotchenko, ‘Impact of petroleum products on soil composition and physical-chemical properties’, *IOP Conf. Ser. Earth Environ. Sci.*, vol. 33, p. 012028, Mar. 2016, doi: 10.1088/1755-1315/33/1/012028.

[63] O. G. Brakstad, A. Lewis, and C. J. Beegle-Krause, ‘A critical review of marine snow in the context of oil spills and oil spill dispersant treatment with focus on the Deepwater Horizon oil spill’, *Mar. Pollut. Bull.*, vol. 135, pp. 346–356, Oct. 2018, doi: 10.1016/j.marpolbul.2018.07.028.

[64] F. Nirchio, S. Tomaso, W. Biamino, E. Parisato, P. Trivero, and A. Giancaspro, ‘Oil spills automatic detection from SAR images’, *Int. J. Remote Sens. - INT J REMOTE SENS*, vol. 26, Mar. 2005, doi: 10.1080/01431160512331326558.

[65] ‘The Global Challenges Foundation’, *The Global Challenges Foundation*. <https://globalchallenges.org/about/the-global-challenges-foundation/> (accessed Jan. 16, 2022).

[66] ‘Sabesp » Meio Ambiente » Reciclagem de óleo’. <https://site.sabesp.com.br/site/interna/Default.aspx?secaoId=82> (accessed Aug. 07, 2022).

[67] R. K. Gupta, G. J. Dunderdale, M. W. England, and A. Hozumi, ‘Oil/water separation techniques: a review of recent progresses and future directions’, *J. Mater. Chem. A*, vol. 5, no. 31, pp. 16025–16058, Aug. 2017, doi: 10.1039/C7TA02070H.

[68] T. Dalton and D. Jin, ‘Extent and frequency of vessel oil spills in US marine protected areas’, *Mar. Pollut. Bull.*, vol. 60, pp. 1939–45, Nov. 2010, doi: 10.1016/j.marpolbul.2010.07.036.

[69] J. M. Capuzzo, ‘Biological effects of petroleum hydrocarbons on marine organisms: Integration of experimental results and predictions of impacts’, *Mar. Environ. Res.*, vol. 17, no. 2, pp. 272–276, Jan. 1985, doi: 10.1016/0141-1136(85)90104-7.

[70] M. Islam and M. Tanaka, ‘Impacts of pollution on coastal and marine ecosystems including coastal and marine fisheries and approach for management: A review and synthesis’,

Mar. Pollut. Bull., vol. 48, pp. 624–49, May 2004, doi: 10.1016/j.marpolbul.2003.12.004.

[71] R. S. Judson *et al.*, ‘Analysis of eight oil spill dispersants using rapid, in vitro tests for endocrine and other biological activity’, *Environ. Sci. Technol.*, vol. 44, no. 15, pp. 5979–5985, Aug. 2010, doi: 10.1021/es102150z.

[72] D. N. Major and H. Wang, ‘How public health impact is addressed: a retrospective view on three different oil spills’, *Toxicol. Environ. Chem.*, vol. 94, no. 3, pp. 442–467, Mar. 2012, doi: 10.1080/02772248.2012.654633.

[73] L. Sandrini-Neto, C. C. Martins, and P. C. Lana, ‘Are intertidal soft sediment assemblages affected by repeated oil spill events? A field-based experimental approach’, *Environ. Pollut.*, vol. 213, pp. 151–159, Jun. 2016, doi: 10.1016/j.envpol.2016.02.014.

[74] G. Miller, D. Hackett, and C. Wolfe, *Living in the Environment*, 4th edition. Canada: Nelson Cengage Adapted, 2016.

[75] Cohen Maurie, ‘(1) (PDF) Technological Disasters and Natural Resource Damage Assessment: An Evaluation of the Exxon Valdez Oil Spill’. https://www.researchgate.net/publication/227638672_Technological_Disasters_and_Natural_Resource_Damage_Assessment_An_Evaluation_of_the_Exxon_Valdez_Oil_Spill (accessed Apr. 27, 2021).

[76] S.-M. Cheong, ‘Fishing and Tourism Impacts in the Aftermath of the Hebei-Spirit Oil Spill’, *J. Coast. Res.*, vol. 28, pp. 1648–1653, Nov. 2012, doi: 10.2307/23353616.

[77] B. W. Ritchie, J. C. Crotts, A. Zehrer, and G. T. Volsky, ‘Understanding the Effects of a Tourism Crisis: The Impact of the BP Oil Spill on Regional Lodging Demand’, *J. Travel Res.*, vol. 53, no. 1, pp. 12–25, Jan. 2014, doi: 10.1177/0047287513482775.

[78] M. de O. Soares *et al.*, ‘Oil spill in South Atlantic (Brazil): Environmental and governmental disaster’, *Mar. Policy*, vol. 115, p. 103879, May 2020, doi: 10.1016/j.marpol.2020.103879.

[79] H. Escobar, ‘Mystery oil spill threatens marine sanctuary in Brazil’, *Science*, vol. 366, no. 6466, pp. 672–672, Nov. 2019, doi: 10.1126/science.366.6466.672.

[80] IBAMA, ‘IBAMA (Instituto Brasileiro do Meio Ambiente e dos Recursos Naturais e Renováveis). Áreas com localidades oleadas no Nordeste brasileiro.’, *IBDMAR*, Oct. 05, 2019. <http://www.ibdmar.org/2019/10/poluicao-por-oleo-em-praias-do-nordeste-brasileiro-podem-ter-origem-no-alto-mar/> (accessed Apr. 27, 2021).

[81] E. Haddad, A. Porsse, and W. Rabahy, ‘Domestic tourism and regional inequality in Brazil’, *Tour. Econ.*, vol. 19, pp. 173–186, Feb. 2013, doi: 10.5367/te.2013.0185.

[82] T. C. Hansel, H. J. Osofsky, J. D. Osofsky, and A. Speier, ‘Longer-Term Mental and

- Behavioral Health Effects of the Deepwater Horizon Gulf Oil Spill’, *J. Mar. Sci. Eng.*, vol. 3, no. 4, Art. no. 4, Dec. 2015, doi: 10.3390/jmse3041260.
- [83] H. J. Osofsky, T. C. Hansel, J. D. Osofsky, and A. Speier, ‘Factors Contributing to Mental and Physical Health Care in a Disaster-Prone Environment’, *Behav. Med. Wash. DC*, vol. 41, no. 3, pp. 131–137, 2015, doi: 10.1080/08964289.2015.1032201.
- [84] H. Osofsky, J. Osofsky, and T. Hansel, ‘Deepwater Horizon Oil Spill: Mental Health Effects on Residents in Heavily Affected Areas’, *Disaster Med. Public Health Prep.*, vol. 5, pp. 280–6, Dec. 2011, doi: 10.1001/dmp.2011.85.
- [85] D. Dave and A. E. Ghaly, *Remediation Technologies for Marine Oil Spills: A Critical Review and Comparative Analysis*.
- [86] A. Szymanek, “‘Defence-in-Depth’ Strategy in Transport Risk Management”, in *Transport Systems Telematics*, Berlin, Heidelberg, 2010, pp. 51–58. doi: 10.1007/978-3-642-16472-9_5.
- [87] Y. Liu, ‘Safety barriers: Research advances and new thoughts on theory, engineering and management’, *J. Loss Prev. Process Ind.*, vol. 67, p. 104260, Sep. 2020, doi: 10.1016/j.jlp.2020.104260.
- [88] N. Bhardwaj and A. N. Bhaskarwar, ‘A review on sorbent devices for oil-spill control’, *Environ. Pollut.*, vol. 243, pp. 1758–1771, Dec. 2018, doi: 10.1016/j.envpol.2018.09.141.
- [89] K. Øien, S. Hauge, F. Størseth, and R. Tinmannsvik, ‘Towards a holistic approach for barrier management in the petroleum industry’, *undefined*, 2015, Accessed: Apr. 30, 2021. [Online]. Available: [/paper/Towards-a-holistic-approach-for-barrier-management-%C3%98ien-Hauge/d072458717869d450f073f0dee871dd99391bba4](#)
- [90] S. Jacobsen, K. Haver, O. Gudmestad, and Ø. Tuntland, *Overview of Measures Specifically Designed to Prevent Oil Pollution in the Arctic Marine Environment from Offshore Petroleum Activities*. 2016. doi: 10.4043/27432-MS.
- [91] Z.-C. Zhu, C.-W. Chu, H.-T. Bian, and J.-C. Jiang, ‘An integration method using distributed optical fiber sensor and Auto-Encoder based deep learning for detecting sulfurized rust self-heating of crude oil tanks’, *J. Loss Prev. Process Ind.*, vol. 74, p. 104623, Jan. 2022, doi: 10.1016/j.jlp.2021.104623.
- [92] Fingas, M., *Oil Spill Science and Technology*. Elsevier, 2011. doi: 10.1016/C2009-0-19703-9.
- [93] P. E. Ndimele, *The Political Ecology of Oil and Gas Activities in the Nigerian Aquatic Ecosystem*. 2017, p. 467.
- [94] M. Fingas and C. Brown, ‘A Review of Oil Spill Remote Sensing’, *Sensors*, vol. 18, p.

91, Dec. 2017, doi: 10.3390/s18010091.

[95] D. P. French-McCay, 'Oil spill impact modeling: Development and validation', *Environ. Toxicol. Chem.*, vol. 23, no. 10, pp. 2441–2456, 2004, doi: <https://doi.org/10.1897/03-382>.

[96] J. Fritt-Rasmussen and P. J. Brandvik, 'Measuring ignitability for in situ burning of oil spills weathered under Arctic conditions: From laboratory studies to large-scale field experiments', *Mar. Pollut. Bull.*, vol. 62, no. 8, pp. 1780–1785, Aug. 2011, doi: 10.1016/j.marpolbul.2011.05.020.

[97] ITOPF, 'TIP 05: Use of skimmers in oil pollution response - ITOPF'. <https://www.itopf.org/knowledge-resources/documents-guides/document/tip-05-use-of-skimmers-in-oil-pollution-response/> (accessed May 07, 2021).

[98] R. M. Atlas and T. C. Hazen, 'Oil Biodegradation and Bioremediation: A Tale of the Two Worst Spills in U.S. History', *Environ. Sci. Technol.*, vol. 45, no. 16, pp. 6709–6715, Aug. 2011, doi: 10.1021/es2013227.

[99] S. Tewari and A. Sirvaiya, 'OIL SPILL REMEDIATION AND ITS REGULATION', *Int. J. Res. Sci. Eng.*, vol. 1, pp. 2394–8299, Oct. 2015.

[100] H. Sun, H. Liu, S. Wang, and Y. Liu, 'Remediation of oil spill-contaminated sands by chemical-free microbubbles generated in tap and saline water', *J. Hazard. Mater.*, vol. 366, pp. 124–129, Mar. 2019, doi: 10.1016/j.jhazmat.2018.11.102.

[101] M. Fingas and C. Brown, 'Review of oil spill remote sensing', *Mar. Pollut. Bull.*, vol. 83, no. 1, pp. 9–23, Jun. 2014, doi: 10.1016/j.marpolbul.2014.03.059.

[102] D. Sundaravadivelu, M. T. Suidan, A. D. Venosa, and P. I. Rosales, 'Characterization of solidifiers used for oil spill remediation', *Chemosphere*, vol. 144, pp. 1490–1497, Feb. 2016, doi: 10.1016/j.chemosphere.2015.10.030.

[103] Nomack, Read 'An Ecosystem Services Approach to Assessing the Impacts of the Deepwater Horizon Oil Spill in the Gulf of Mexico' at *NAP.edu*. doi: 10.17226/18387.

[104] G. Olah and A. Molnar, 'Hydrocarbon Chemistry, 2nd Edition | Wiley', *Wiley.com*. <https://www.wiley.com/en-sg/Hydrocarbon+Chemistry%2C+2nd+Edition-p-9780471433484> (accessed May 07, 2021).

[105] M. Leung, 'Bioremediation: Techniques for Cleaning up a mess', vol. 2, p. 5, 2004.

[106] M. M. Amro, *Treatment Techniques of Oil-Contaminated Soil and Water Aquifers*.

[107] J. Fussell and D. Rundquist, 'On Defining Remote Sensing', *Photogramm. Eng.*, p. 5, 1986.

[108] R. Al-Ruzouq *et al.*, 'Sensors, Features, and Machine Learning for Oil Spill Detection

- and Monitoring: A Review', *Remote Sens.*, vol. 12, no. 20, Art. no. 20, Jan. 2020, doi: 10.3390/rs12203338.
- [109] Marzioletti P; Laneve G, '(1) Oil spill monitoring on water surfaces by radar L, C and X band SAR imagery: A comparison of relevant characteristics | Request PDF'. https://www.researchgate.net/publication/309776900_Oil_spill_monitoring_on_water_surfaces_by_radar_L_C_and_X_band_SAR_imagery_A_comparison_of_relevant_characteristics (accessed Apr. 30, 2021).
- [110] S. A. Khoshnevis and S. Ghorshi, 'A tutorial on tomographic synthetic aperture radar methods', *SN Appl. Sci.*, vol. 2, no. 9, p. 1504, Aug. 2020, doi: 10.1007/s42452-020-03298-6.
- [111] O. Ponce, P. Prats-Iraola, R. Scheiber, A. Reigber, and A. Moreira, 'First Airborne Demonstration of Holographic SAR Tomography With Fully Polarimetric Multicircular Acquisitions at L-Band', *IEEE Trans. Geosci. Remote Sens.*, vol. 54, Jul. 2016, doi: 10.1109/TGRS.2016.2582959.
- [112] S. K. Chaturvedi, S. Banerjee, and S. Lele, 'An assessment of oil spill detection using Sentinel 1 SAR-C images', *J. Ocean Eng. Sci.*, vol. 5, no. 2, pp. 116–135, Jun. 2020, doi: 10.1016/j.joes.2019.09.004.
- [113] A. Solberg, 'Remote Sensing of Ocean Oil-Spill Pollution', *Proc. IEEE*, vol. 100, pp. 2931–2945, Oct. 2012, doi: 10.1109/JPROC.2012.2196250.
- [114] P. Liu, cf Zhao, X. Li, M. He, and W. Pichel, 'Identification of ocean oil spills in SAR imagery based on fuzzy logic algorithm', *Int. J. Remote Sens. - INT J REMOTE SENS*, vol. 31, pp. 4819–4833, Sep. 2010, doi: 10.1080/01431161.2010.485147.
- [115] M. Ma, J. Liang, M. Guo, Y. Fan, and Y. Yin, 'SAR image segmentation based on Artificial Bee Colony algorithm', *Appl Soft Comput*, 2011, doi: 10.1016/j.asoc.2011.05.039.
- [116] Norouzi, Akbarizadeh, and Eftekhari, '(2) (PDF) A hybrid feature extraction method for SAR image registration'. https://www.researchgate.net/publication/325632306_A_hybrid_feature_extraction_method_for_SAR_image_registration (accessed May 18, 2021).
- [117] G. Chierchia, D. Cozzolino, G. Poggi, and L. Verdoliva, 'SAR image despeckling through convolutional neural networks', *ArXiv170400275 Cs*, May 2017, Accessed: May 18, 2021. [Online]. Available: <http://arxiv.org/abs/1704.00275>
- [118] F. Lattari, B. Gonzalez Leon, F. Asaro, A. Rucci, C. Prati, and M. Matteucci, 'Deep Learning for SAR Image Despeckling', *Remote Sens.*, vol. 11, no. 13, Art. no. 13, Jan. 2019, doi: 10.3390/rs11131532.
- [119] K. Topouzelis, V. Karathanassi, P. Pavlakis, and D. Rokos, 'Detection and

- discrimination between oil spills and look-alike phenomena through neural networks’, *ISPRS J. Photogramm. Remote Sens.*, vol. 62, no. 4, pp. 264–270, Sep. 2007, doi: 10.1016/j.isprsjprs.2007.05.003.
- [120] Mejail, Jacobo-Berlles, and Frery, ‘(2) (PDF) Parametric roughness estimation in amplitude SAR images under the multiplicative model’. https://www.researchgate.net/publication/39195356_Parametric_roughness_estimation_in_amplitude_SAR_images_under_the_multiplicative_model (accessed May 18, 2021).
- [121] Y. Fan, X. Rui, G. Zhang, T. Yu, X. Xu, and S. Poslad, ‘Feature Merged Network for Oil Spill Detection Using SAR Images’, *Remote Sens.*, vol. 13, no. 16, Art. no. 16, Jan. 2021, doi: 10.3390/rs13163174.
- [122] F. Nunziata and M. Migliaccio, ‘Oil Spill Monitoring and Damage Assessment via PolSAR Measurements’, *Aquat. Procedia*, vol. 3, pp. 95–102, Mar. 2015, doi: 10.1016/j.aqpro.2015.02.232.
- [123] M. J. Collins, M. Denbina, B. Minchew, C. E. Jones, and B. Holt, ‘On the Use of Simulated Airborne Compact Polarimetric SAR for Characterizing Oil–Water Mixing of the Deepwater Horizon Oil Spill’, *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 8, no. 3, pp. 1062–1077, Mar. 2015, doi: 10.1109/JSTARS.2015.2401041.
- [124] A. Rogalski, ‘Infrared detectors: an overview’, *Infrared Phys. Technol.*, vol. 43, no. 3, pp. 187–210, Jun. 2002, doi: 10.1016/S1350-4495(02)00140-8.
- [125] A. Pisano, F. Bignami, and R. Santoleri, ‘Oil Spill Detection in Glint-Contaminated Near-Infrared MODIS Imagery’, *Remote Sens.*, vol. 7, no. 1, Art. no. 1, Jan. 2015, doi: 10.3390/rs70101112.
- [126] J. W. Salisbury, D. M. D’Aria, and F. F. Sabins, ‘Thermal infrared remote sensing of crude oil slicks’, *Remote Sens. Environ.*, vol. 45, no. 2, pp. 225–231, Aug. 1993, doi: 10.1016/0034-4257(93)90044-X.
- [127] Y. Chen, L. Zhang, J. Hu, Z. Liu, and K. Xu, ‘Emergency response recommendation for long-distance oil and gas pipeline based on an accident case representation model’, *J. Loss Prev. Process Ind.*, vol. 77, p. 104779, Jul. 2022, doi: 10.1016/j.jlp.2022.104779.
- [128] R. N. Vasconcelos *et al.*, ‘Oil Spill Detection and Mapping: A 50-Year Bibliometric Analysis’, *Remote Sens.*, vol. 12, no. 21, Art. no. 21, Jan. 2020, doi: 10.3390/rs12213647.
- [129] V. Karathanassi, K. Topouzelis, P. Pavlakis, and D. Rokos, ‘An object-oriented methodology to detect oil spills’, *Int. J. Remote Sens. - INT J REMOTE SENS*, vol. 27, pp. 5235–5251, Dec. 2006, doi: 10.1080/01431160600693575.
- [130] F. Del Frate, A. Petrocchi, J. Lichtenegger, and G. Calabresi, ‘Neural networks for oil

- spill detection using ERS-SAR data', *Geosci. Remote Sens. IEEE Trans. On*, vol. 38, pp. 2282–2287, Oct. 2000, doi: 10.1109/36.868885.
- [131] B. Fiscella, A. Giancaspro, F. Nirchio, P. Pavese, and P. Trivero, 'Oil spill detection using marine SAR images', *Int. J. Remote Sens. - INT J REMOTE SENS*, vol. 21, pp. 3561–3566, Dec. 2000, doi: 10.1080/014311600750037589.
- [132] M. Kubat, R. C. Holte, and S. Matwin, 'Machine Learning for the Detection of Oil Spills in Satellite Radar Images', *Mach. Learn.*, vol. 30, no. 2, pp. 195–215, Feb. 1998, doi: 10.1023/A:1007452223027.
- [133] M. Marghany, 'RADARSAT automatic algorithms for detecting coastal oil spill pollution', *Int. J. Appl. Earth Obs. Geoinformation*, vol. 3, pp. 191–196, Dec. 2001, doi: 10.1016/S0303-2434(01)85011-X.
- [134] A. Almulihi, F. Alharithi, S. Bourouis, R. Alroobaea, Y. Pawar, and N. Bouguila, 'Oil Spill Detection in SAR Images Using Online Extended Variational Learning of Dirichlet Process Mixtures of Gamma Distributions', *Remote Sens.*, vol. 13, no. 15, Art. no. 15, Jan. 2021, doi: 10.3390/rs13152991.
- [135] W. Wang *et al.*, 'A fast, edge-preserving, distance-regularized model with bilateral filtering for oil spill segmentation of SAR images', *J. Oceanol. Limnol.*, vol. 39, Jan. 2021, doi: 10.1007/s00343-020-0105-7.
- [136] Y. Li, X. Lyu, A. C. Frery, and P. Ren, 'Oil Spill Detection with Multiscale Conditional Adversarial Networks with Small-Data Training', *Remote Sens.*, vol. 13, no. 12, Art. no. 12, Jan. 2021, doi: 10.3390/rs13122378.
- [137] M. R. A. Conceição *et al.*, 'SAR Oil Spill Detection System through Random Forest Classifiers', *Remote Sens.*, vol. 13, no. 11, Art. no. 11, Jan. 2021, doi: 10.3390/rs13112044.
- [138] G. Li, Y. Li, Y. Hou, X. Wang, and L. Wang, 'Marine Oil Slick Detection Using Improved Polarimetric Feature Parameters Based on Polarimetric Synthetic Aperture Radar Data', *Remote Sens.*, vol. 13, no. 9, Art. no. 9, Jan. 2021, doi: 10.3390/rs13091607.
- [139] B. Wang *et al.*, 'A Spectral-Spatial Features Integrated Network for Hyperspectral Detection of Marine Oil Spill', *Remote Sens.*, vol. 13, no. 8, Art. no. 8, Jan. 2021, doi: 10.3390/rs13081568.
- [140] M. Mohammadi, A. Sharifi, M. Hosseingholizadeh, and A. Tariq, 'Detection of Oil Pollution Using SAR and Optical Remote Sensing Imagery: A Case Study of the Persian Gulf', *J. Indian Soc. Remote Sens.*, Jun. 2021, doi: 10.1007/s12524-021-01399-2.
- [141] S. Temitope Yekeen, A. Balogun, and K. B. Wan Yusof, 'A novel deep learning instance segmentation model for automated marine oil spill detection', *ISPRS J. Photogramm.*

- Remote Sens.*, vol. 167, pp. 190–200, Sep. 2020, doi: 10.1016/j.isprsjprs.2020.07.011.
- [142] M. S. Ozigis, J. D. Kaduk, C. H. Jarvis, P. da Conceição Bispo, and H. Balzter, ‘Detection of oil pollution impacts on vegetation using multifrequency SAR, multispectral images with fuzzy forest and random forest methods’, *Environ. Pollut.*, vol. 256, p. 113360, Jan. 2020, doi: 10.1016/j.envpol.2019.113360.
- [143] K. Manivannan, P. Aggarwal, V. Devabhaktuni, A. Kumar, D. Nims, and P. Bhattacharya, ‘Particulate matter characterization by gray level co-occurrence matrix based support vector machines’, *J. Hazard. Mater.*, vol. 223–224, pp. 94–103, Jul. 2012, doi: 10.1016/j.jhazmat.2012.04.056.
- [144] C. G. Eichkitz, J. Amtmann, and M. G. Schreilechner, ‘Calculation of grey level co-occurrence matrix-based seismic attributes in three dimensions’, *Comput. Geosci.*, vol. 60, pp. 176–183, Oct. 2013, doi: 10.1016/j.cageo.2013.07.006.
- [145] M. Nixon, *Feature Extraction and Image Processing*, 2^o edição. New York; Boulder: Academic Press, 2008.
- [146] T. Ojala, M. Pietikäinen, and D. Harwood, ‘A comparative study of texture measures with classification based on featured distributions’, *Pattern Recognit.*, vol. 29, no. 1, pp. 51–59, Jan. 1996, doi: 10.1016/0031-3203(95)00067-4.
- [147] L. Liu, L. Zhao, Y. Long, G. Kuang, and P. Fieguth, ‘Extended local binary patterns for texture classification’, *Image Vis. Comput.*, vol. 30, no. 2, pp. 86–99, Feb. 2012, doi: 10.1016/j.imavis.2012.01.001.
- [148] F. Liu, Z. Tang, and J. Tang, ‘WLBP: Weber local binary pattern for local image description’, *Neurocomputing*, vol. 120, pp. 325–335, Nov. 2013, doi: 10.1016/j.neucom.2012.06.061.
- [149] A. Suruliandi, K. Meena, and R. Rose, ‘Local binary pattern and its derivatives for face recognition’, *Comput. Vis. IET*, vol. 6, pp. 480–488, Sep. 2012, doi: 10.1049/iet-cvi.2011.0228.
- [150] L. Houam, A. Hafiane, A. Boukrouche, E. Lespessailles, and R. Jennane, ‘One dimensional local binary pattern for bone texture characterization’, *Pattern Anal. Appl.*, vol. 17, no. 1, pp. 179–193, Feb. 2014, doi: 10.1007/s10044-012-0288-4.
- [151] N. Chatlani and J. J. Soraghan, ‘Local binary patterns for 1-D signal processing’, in *2010 18th European Signal Processing Conference*, Aug. 2010, pp. 95–99.
- [152] Z. Guo, L. Zhang, and D. Zhang, ‘A Completed Modeling of Local Binary Pattern Operator for Texture Classification’, *IEEE Trans. Image Process. Publ. IEEE Signal Process. Soc.*, vol. 19, pp. 1657–63, Mar. 2010, doi: 10.1109/TIP.2010.2044957.
- [153] WANG Guo-de, ZHANG Pei-lin 1, REN Guo-quan, KOU Xi, ‘Texture Feature

- Extraction Method Fused with LBP and GLCM'. <http://www.ecice06.com/EN/10.3969/j.issn.1000-3428.2012.11.061> (accessed May 01, 2021).
- [154] Davoudi Rouzbeh; Gregory R. MillerJ. Nathan Kutz, '(1) (PDF) Structural Load Estimation Using Machine Vision and Surface Crack Patterns for Shear-Critical RC Beams and Slabs'. https://www.researchgate.net/publication/324981051_Structural_Load_Estimation_Using_Machine_Vision_and_Surface_Crack_Patterns_for_Shear-Critical_RC_Beams_and_Slabs (accessed May 01, 2021).
- [155] Hassan Ramchoun; Mohammed Amine; Mohammed Amine Janati Idriss; Mohamed Ettaouil, '(2) (PDF) Multilayer Perceptron: Architecture Optimization and Training'. https://www.researchgate.net/publication/292996667_Multilayer_Perceptron_Architecture_Optimization_and_Training (accessed May 04, 2021).
- [156] S. Haykin, *Neural Networks and Learning Machines*, 3rd edition. New York: Pearson, 2008.
- [157] M. W. Gardner and S. R. Dorling, 'Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences', *Atmos. Environ.*, vol. 32, no. 14, pp. 2627–2636, Aug. 1998, doi: 10.1016/S1352-2310(97)00447-0.
- [158] L. Breiman, 'Random Forests | SpringerLink'. <https://link.springer.com/article/10.1023/A:1010933404324> (accessed May 08, 2021).
- [159] S. Athey, J. Tibshirani, and S. Wager, 'Generalized Random Forests', *ArXiv161001271 Econ Stat*, Apr. 2018, Accessed: May 08, 2021. [Online]. Available: <http://arxiv.org/abs/1610.01271>
- [160] C. Tang, D. Garreau, and U. von Luxburg, 'When do random forests fail?', *Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, Accessed: May 08, 2021. [Online]. Available: <https://papers.nips.cc/paper/2018/hash/204da255aea2cd4a75ace6018fad6b4d-Abstract.html>
- [161] S. Wager and S. Athey, 'Estimation and Inference of Heterogeneous Treatment Effects using Random Forests: Journal of the American Statistical Association: Vol 113, No 523'. <https://www.tandfonline.com/doi/abs/10.1080/01621459.2017.1319839?journalCode=uasa20> (accessed May 08, 2021).
- [162] C. Campbell and Y. Ying, 'Learning with Support Vector Machines', *Synth. Lect. Artif. Intell. Mach. Learn.*, vol. 5, no. 1, pp. 1–95, Feb. 2011, doi: 10.2200/S00324ED1V01Y201102AIM010.
- [163] G. Mountrakis, J. Im, and C. Ogole, 'Support vector machines in remote sensing: A review', *ISPRS J. Photogramm. Remote Sens.*, vol. 66, no. 3, pp. 247–259, May 2011, doi:

- 10.1016/j.isprsjprs.2010.11.001.
- [164] J. Suykens and J. Vandewalle, ‘Least Squares Support Vector Machine Classifiers’, *Neural Process. Lett.*, vol. 9, pp. 293–300, Jun. 1999, doi: 10.1023/A:1018628609742.
- [165] J. A. K. Suykens, ‘Support Vector Machines: A Nonlinear Modelling and Control Perspective’, *Eur. J. Control*, vol. 7, no. 2, pp. 311–327, Jan. 2001, doi: 10.3166/ejc.7.311-327.
- [166] D. Pregibon, ‘Logistic Regression Diagnostics’, *Ann. Stat.*, vol. 9, no. 4, pp. 705–724, Jul. 1981, doi: 10.1214/aos/1176345513.
- [167] F. Y. Hsieh, D. Bloch, and M. Larsen, ‘A Simple Method of Sample Size Calculation for Linear and Logistic Regression’, *Stat. Med.*, vol. 17, pp. 1623–34, Aug. 1998, doi: 10.1002/(SICI)1097-0258(19980730)17:143.0.CO;2-S.
- [168] Armitage, Berry, and Geoffrey, ‘Statistical Methods in Medical Research, 4th Edition | Wiley’, *Wiley.com*. <https://www.wiley.com/en-ae/Statistical+Methods+in+Medical+Research%2C+4th+Edition-p-9780632052578> (accessed May 08, 2021).
- [169] T. Chen and C. Guestrin, *XGBoost: A Scalable Tree Boosting System*. 2016, p. 794. doi: 10.1145/2939672.2939785.
- [170] L. T. Le, H. Nguyen, J. Zhou, J. Dou, and H. Moayedi, ‘Estimating the Heating Load of Buildings for Smart City Planning Using a Novel Artificial Intelligence Technique PSO-XGBoost’, *Appl. Sci.*, vol. 9, no. 13, Art. no. 13, Jan. 2019, doi: 10.3390/app9132714.
- [171] S. Dhaliwal, A. Nahid, and R. Abbas, ‘Effective Intrusion Detection System Using XGBoost’, Jul. 2018.
- [172] J. F. Barraza, E. L. Droguett, V. M. Naranjo, and M. R. Martins, ‘Capsule Neural Networks for structural damage localization and quantification using transmissibility data’, *Appl. Soft Comput.*, vol. 97, p. 106732, Dec. 2020, doi: 10.1016/j.asoc.2020.106732.
- [173] M. Långkvist, L. Karlsson, and A. Loutfi, ‘A review of unsupervised feature learning and deep learning for time-series modeling’, *Pattern Recognit. Lett.*, vol. 42, pp. 11–24, Jun. 2014, doi: 10.1016/j.patrec.2014.01.008.
- [174] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, ‘Gradient-based learning applied to document recognition’, *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998, doi: 10.1109/5.726791.
- [175] M. Petrochuk, ‘hparams’.
- [176] V. Christlein, L. Spranger, M. Seuret, A. Nicolaou, P. Král, and A. Maier, ‘Deep Generalized Max Pooling’, in *2019 International Conference on Document Analysis and Recognition (ICDAR)*, Sep. 2019, pp. 1090–1096. doi: 10.1109/ICDAR.2019.00177.

- [177] M. Abadi *et al.*, *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*. 2015.
- [178] A. Gulli and S. Pal, *Deep Learning with Keras: Implementing deep learning models and neural networks with the power of Python*. Birmingham Mumbai: Packt Publishing, 2017.
- [179] K. He, X. Zhang, S. Ren, and J. Sun, ‘Deep Residual Learning for Image Recognition’, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.
- [180] N. Sharma, V. Jain, and A. Mishra, ‘An Analysis Of Convolutional Neural Networks For Image Classification’, *Procedia Comput. Sci.*, vol. 132, pp. 377–384, Jan. 2018, doi: 10.1016/j.procs.2018.05.198.
- [181] K. Simonyan and A. Zisserman, ‘Very Deep Convolutional Networks for Large-Scale Image Recognition’, *ArXiv 14091556*, Sep. 2014.
- [182] A. K. Sharma, A. Nandal, A. Dhaka, D. Koundal, D. C. Bogatinoska, and H. Alyami, ‘Enhanced Watershed Segmentation Algorithm-Based Modified ResNet50 Model for Brain Tumor Detection’, *BioMed Res. Int.*, vol. 2022, p. e7348344, Feb. 2022, doi: 10.1155/2022/7348344.
- [183] M. Elpeltagy and H. Sallam, ‘Automatic prediction of COVID– 19 from chest images using modified ResNet50’, *Multimed. Tools Appl.*, vol. 80, no. 17, pp. 26451–26463, Jul. 2021, doi: 10.1007/s11042-021-10783-6.
- [184] ‘EBSCOhost | 149566568 | Classification of Pneumonia Cell Images Using Improved ResNet50 Model.’
<https://web.p.ebscohost.com/abstract?direct=true&profile=ehost&scope=site&authtype=crawler&jrnl=07650019&AN=149566568&h=XjEqQfP1kR%2fgEgRRfewe7QKMmunXoUT0%2bwpBv0JMwj9BSTW%2bZbSGmwgdbU10olx2l5r32bTiKD5VUe0Z1JqT6Q%3d%3d&crl=c&resultNs=AdminWebAuth&resultLocal=ErrCrlNotAuth&crlhashurl=login.aspx%3fdirect%3dtrue%26profile%3dehost%26scope%3dsite%26authtype%3dcrawler%26jrnl%3d07650019%26AN%3d149566568> (accessed May 10, 2022).
- [185] I. Z. Mukti and D. Biswas, ‘Transfer Learning Based Plant Diseases Detection Using ResNet50’, in *2019 4th International Conference on Electrical Information and Communication Technology (EICT)*, Dec. 2019, pp. 1–6. doi: 10.1109/EICT48899.2019.9068805.
- [186] Y. Chu, X. Yue, L. Yu, M. Sergei, and Z. Wang, ‘Automatic Image Captioning Based on ResNet50 and LSTM with Soft Attention’, *Wirel. Commun. Mob. Comput.*, vol. 2020, p. e8909458, Oct. 2020, doi: 10.1155/2020/8909458.
- [187] A. Shabbir *et al.*, ‘Satellite and Scene Image Classification Based on Transfer Learning

- and Fine Tuning of ResNet50', *Math. Probl. Eng.*, vol. 2021, p. e5843816, Jul. 2021, doi: 10.1155/2021/5843816.
- [188] P. Yang, C. Dong, X. Zhao, and X. Chen, 'The Surface Damage Identifications of Wind Turbine Blades Based on ResNet50 Algorithm', in *2020 39th Chinese Control Conference (CCC)*, Jul. 2020, pp. 6340–6344. doi: 10.23919/CCC50068.2020.9189408.
- [189] S. Karamizadeh, S. M. Abdullah, A. A. Manaf, M. Zamani, and A. Hooman, 'An Overview of Principal Component Analysis', *J. Signal Inf. Process.*, vol. 04, no. 03, pp. 173–175, 2013, doi: 10.4236/jsip.2013.43B031.
- [190] H. Abdi and L. J. Williams, 'Principal component analysis', *WIREs Comput. Stat.*, vol. 2, no. 4, pp. 433–459, 2010, doi: <https://doi.org/10.1002/wics.101>.
- [191] R. Bro and A. K. Smilde, 'Principal component analysis', *Anal. Methods*, vol. 6, no. 9, pp. 2812–2831, Apr. 2014, doi: 10.1039/C3AY41907J.
- [192] C. Park and H. Park, 'Fingerprint classification using fast Fourier transform and nonlinear discriminant analysis', *Pattern Recognit.*, vol. 38, pp. 495–503, Apr. 2005, doi: 10.1016/j.patcog.2004.08.013.
- [193] Y. Wang, H. Yao, and S. Zhao, 'Auto-encoder based dimensionality reduction', *Neurocomputing*, vol. 184, pp. 232–242, Apr. 2016, doi: 10.1016/j.neucom.2015.08.104.
- [194] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, 'AutoAugment: Learning Augmentation Policies from Data', *ArXiv180509501 Cs Stat*, Apr. 2019, Accessed: May 18, 2021. [Online]. Available: <http://arxiv.org/abs/1805.09501>
- [195] D. S. Park *et al.*, 'SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition', *Interspeech 2019*, pp. 2613–2617, Sep. 2019, doi: 10.21437/Interspeech.2019-2680.
- [196] L. Yaeger, R. Lyon, and B. Webb, 'Effective Training of a Neural Network Character Classifier for Word Recognition', *Adv. Neural Inf. Process. Syst.*, vol. 9, 1996, Accessed: May 18, 2021. [Online]. Available: <https://proceedings.neurips.cc/paper/1996/hash/81e5f81db77c596492e6f1a5a792ed53-Abstract.html>
- [197] A. Krizhevsky, I. Sutskever, and G. Hinton, 'ImageNet Classification with Deep Convolutional Neural Networks', *Neural Inf. Process. Syst.*, vol. 25, Jan. 2012, doi: 10.1145/3065386.
- [198] L. Taylor and G. Nitschke, 'Improving Deep Learning using Generic Data Augmentation', *ArXiv170806020 Cs Stat*, Aug. 2017, Accessed: May 18, 2021. [Online]. Available: <http://arxiv.org/abs/1708.06020>

- [199] N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer, ‘SMOTE: Synthetic Minority Over-sampling Technique’, *J Artif Intell Res JAIR*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.
- [200] C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, *Safe-Level-SMOTE: Safe-Level-Synthetic Minority Over-Sampling TEchnique for Handling the Class Imbalanced Problem*, vol. 5476. 2009, p. 482. doi: 10.1007/978-3-642-01307-2_43.
- [201] M. Kubat, ‘Addressing the Curse of Imbalanced Training Sets: One-Sided Selection’, *Fourteenth Int. Conf. Mach. Learn.*, Jun. 2000.
- [202] Tetko, Livingstone, and Luik, ‘Neural network studies. 1. Comparison of overfitting and overtraining | Journal of Chemical Information and Modeling’. <https://pubs.acs.org/doi/10.1021/ci00027a006> (accessed May 17, 2021).
- [203] A. C. S. V. de Negreiros, I. D. Lins, M. J. das C. Moura, and E. L. Droguett, ‘Reliability data analysis of systems in the wear-out phase using a (corrected) q-Exponential likelihood’, *Reliab. Eng. Syst. Saf.*, vol. 197, p. 106787, May 2020, doi: 10.1016/j.ress.2019.106787.
- [204] D. Firth, ‘Bias Reduction of Maximum Likelihood Estimates’, *Biometrika*, vol. 80, no. 1, pp. 27–38, 1993, doi: 10.2307/2336755.
- [205] Jeffreys, H., ‘An invariant form for the prior probability in estimation problems | Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences’. <https://royalsocietypublishing.org/doi/10.1098/rspa.1946.0056> (accessed May 01, 2021).
- [206] J. M. Amigó, S. G. Balogh, and S. Hernández, ‘A Brief Review of Generalized Entropies’, *Entropy*, vol. 20, no. 11, Art. no. 11, Nov. 2018, doi: 10.3390/e20110813.
- [207] J. A. Nelder and R. Mead, ‘A Simplex Method for Function Minimization’, *Comput. J.*, vol. 7, no. 4, pp. 308–313, Jan. 1965, doi: 10.1093/comjnl/7.4.308.
- [208] R. C. P. Marques, F. N. s Medeiros, and J. Nobre, ‘SAR Image Segmentation Based on Level Set Approach and $\{ \text{cal G} \}_A^0$ Model’, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, pp. 046–2057, Oct. 2012, doi: 10.1109/TPAMI.2011.274.
- [209] D. M. Pianto and F. Cribari-Neto, ‘Dealing with monotone likelihood in a model for speckled data’, *Comput. Stat. Data Anal.*, vol. 55, no. 3, pp. 1394–1409, Mar. 2011, doi: 10.1016/j.csda.2010.09.029.
- [210] P. Sergio Rodrigues, G. Wachs-Lopes, R. Morello Santos, E. Coltri, and G. Antonio Giraldi, ‘A q-Extension of Sigmoid Functions and the Application for Enhancement of Ultrasound Images’, *Entropy*, vol. 21, no. 4, Art. no. 4, Apr. 2019, doi: 10.3390/e21040430.
- [211] S. N. Chaudhri, N. S. Rajput, S. H. Alsamhi, A. V. Shvetsov, and F. A. Almalki, ‘Zero-Padding and Spatial Augmentation-Based Gas Sensor Node Optimization Approach in

- Resource-Constrained 6G-IoT Paradigm', *Sensors*, vol. 22, no. 8, Art. no. 8, Jan. 2022, doi: 10.3390/s22083039.
- [212] R. Sales Filho, E. Droguett, I. Lins, M. Moura, M. Amiri, and R. Azevedo, 'Stress-Strength Reliability Analysis with Extreme Values based on q -Exponential Distribution: Stress-Strength Reliability and q -Exponential Distribution', *Qual. Reliab. Eng. Int.*, vol. 33, Jan. 2016, doi: 10.1002/qre.2020.
- [213] J. Ludescher and A. Bunde, 'Universal behavior of the interoccurrence times between losses in financial markets: Independence of the time resolution', *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.*, vol. 90, p. 062809, Dec. 2014, doi: 10.1103/PhysRevE.90.062809.
- [214] L. Malacarne, R. Mendes, and E. Lenzi, 'q-Exponential Distribution in Urban Agglomeration', *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.*, vol. 65, p. 017106, Feb. 2002, doi: 10.1103/PhysRevE.65.017106.
- [215] D. C. Haworth, 'Progress in probability density function methods for turbulent reacting flows', *Prog. Energy Combust. Sci.*, vol. 36, no. 2, pp. 168–259, Apr. 2010, doi: 10.1016/j.peccs.2009.09.003.
- [216] Y. Li, J. Zhang, W. Zhao, W. Jiang, and H. Lu, 'Inductive Guided Filter: Real-Time Deep Matting with Weakly Annotated Masks on Mobile Devices', in *2020 IEEE International Conference on Multimedia and Expo (ICME)*, Jul. 2020, pp. 1–6. doi: 10.1109/ICME46284.2020.9102873.

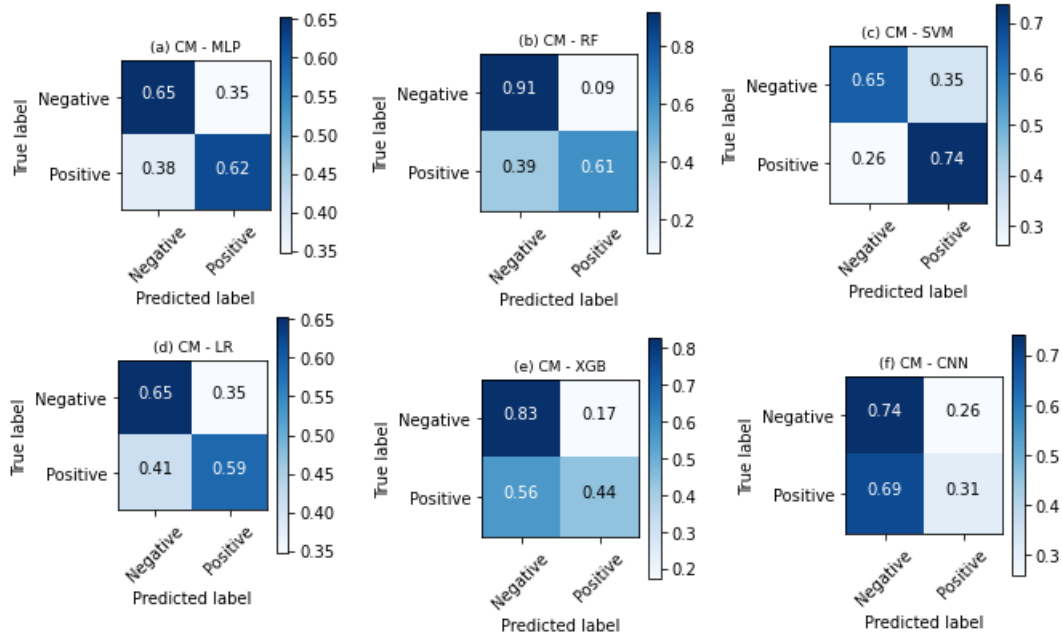
APPENDICE A – TEST RESULTS

In this appendix Section we present the confusion matrices for all the experiments and the tables that bring the parameters used in the ML methods according to the grid search performances for all the ran experiments of this research.

Part 1 - Proposed Approach

In this part 1 we present the confusion matrices and the parameters chosen by the grid search for the proposed q-EFE approach, considering configurations 1 and 2.

Figure A 1: Confusion matrices of the results obtained by the q-EFE approach using 200 images to train the classification models, the CDF, and configuration 1.



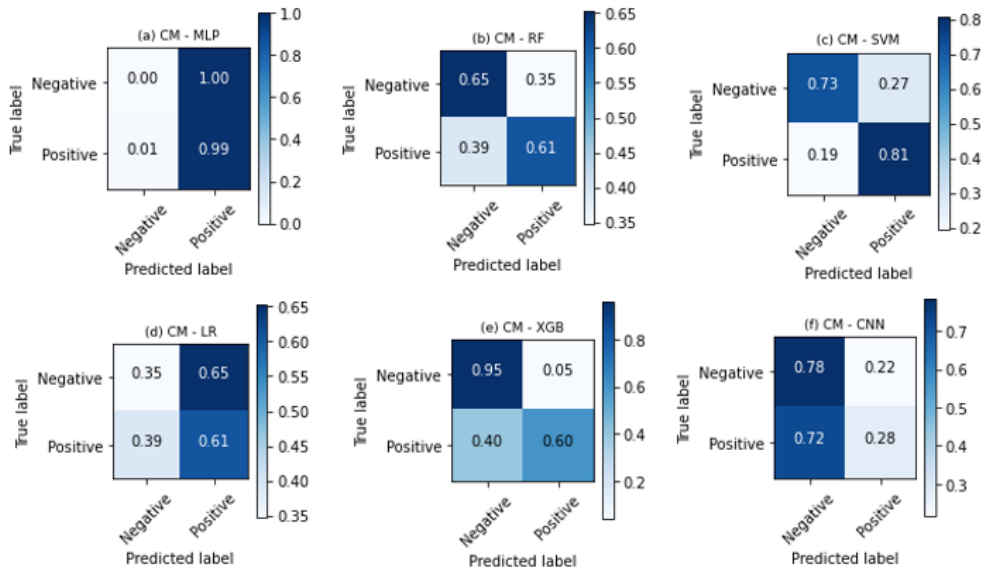
Source: The author (2022).

Table A - 1: Parameters that were chosen in the grid search for q-EFE, CDF, configuration 1, and 200 images.

MLP	RF	SVM	LR	XGB
'solver': 'adam', 'learning_rate': 'invscaling', 'hidden_layer_sizes': (200), 'activation': 'tanh'.	'n_estimators': 200, 'min_samples_split': 8, 'min_samples_leaf': 1, 'max_features': 'log2', 'max_depth': 2.	'kernel': 'rbf', 'gamma': 0.01, 'C': 0.1.	'penalty': 'l2', 'C': 100.	'n_estimators': 100, 'max_depth': 8, 'learning_rate': 0.01, 'gamma': 1, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 2: Confusion matrices of the results obtained by the q-EFE approach using 400 images to train the classification models, the CDF, and configuration 1



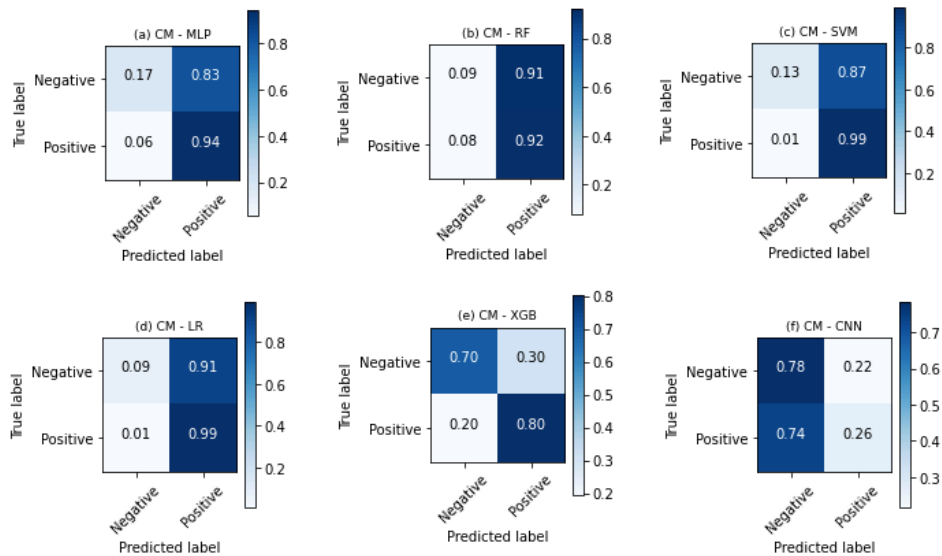
Source: The author (2022).

Table A - 2: Parameters that were chosen in the grid search for q-EFE, CDF, configuration 1, and 400 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'invscaling' 'hidden_layer_sizes': (100), 'activation': 'logistic'.	'n_estimators': 50, 'min_samples_split': 16, 'min_samples_leaf': 1, 'max_features': 'log2', 'max_depth': 2.	'kernel': 'rbf', 'gamma': 0.01, 'C': 1.	'penalty': 'l2', 'C': 0.1.	'n_estimators': 100, 'max_depth': 2, 'learning_rate': 0.01, 'gamma': 5, 'colsample_bytree': 0.8.

Source: The author (2022).

Figure A 3: Confusion matrices of the results obtained by the q-EFE approach using 1002 images to train the classification models, the CDF, and the configuration 1



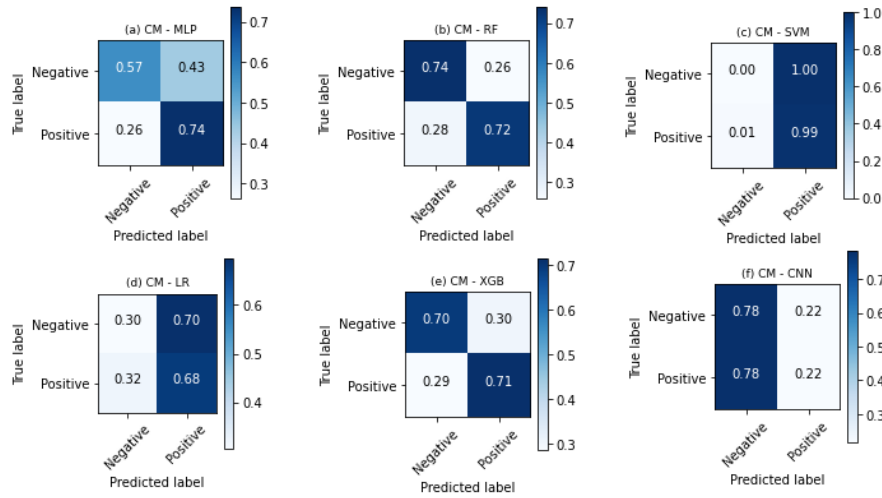
Source: The author (2022).

Table A - 3: Parameters that were chosen in the grid search for q-EFE, CDF, configuration 1, and 1002 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'adaptive', 'hidden_layer_sizes': (1000), 'activation': 'identity'.	'n_estimators': 100, 'min_samples_split': 2, 'min_samples_leaf': 2, 'max_features': 'sqrt', 'max_depth': 2.	kernel': 'linear', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 0.1.	'n_estimators': 100, 'max_depth': 2, 'learning_rate': 0.01, 'gamma': 5, 'colsample_bytree': 1.

Source: The author (2022).

Figure A 4: Confusion matrices of the results obtained by the q-EFE approach using 1572 feature vectors (using SMOTE) to train the classification models, the CDF, and the configuration 1



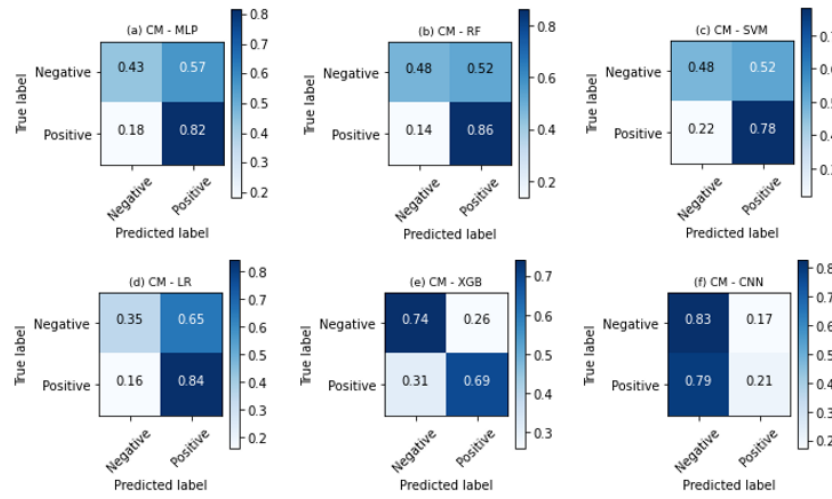
Source: The author (2022).

Table A - 4: Parameters that were chosen in the grid search for q-EFE, CDF, configuration 1, and 1002 images plus SMOTE (1572 images in total)

MLP	RF	SVM	LR	XGB
'solver': 'adam', 'learning_rate': 'invscaling', 'hidden_layer_sizes': (1000) 'activation': 'relu'.	'n_estimators': 200, 'min_samples_split': 2, 'min_samples_leaf': 1, 'max_features': 'sqrt', 'max_depth': 8.	kernel': 'rbf', 'gamma': 1, 'C': 1.	'penalty': 'l2', 'C': 100.	n_estimators': 500, 'max_depth': 64, 'learning_rate': 0.05, 'gamma': 0, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 5: Confusion matrices of the results obtained by the q-EFE approach using 1572 images (with DA) to train the classification models, the CDF, and the configuration 1



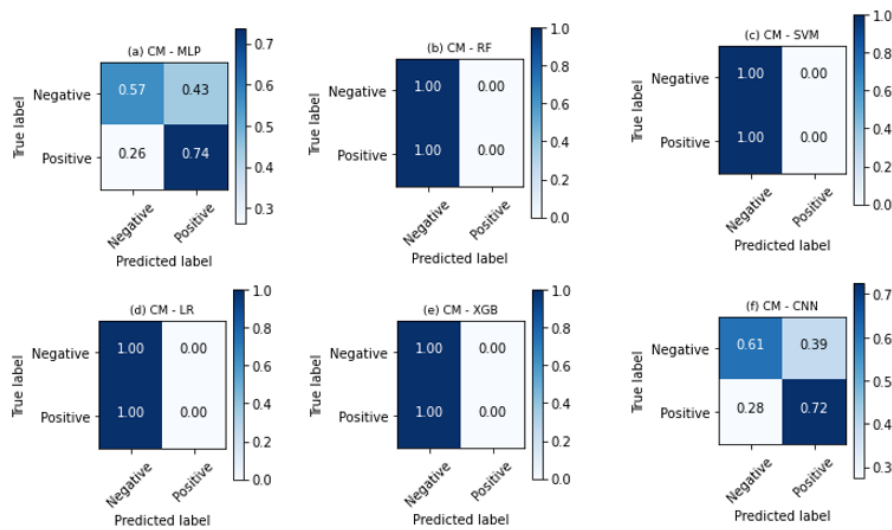
Source: The author (2022).

Table A - 5: Parameters that were chosen in the grid search for q-EFE, CDF, configuration 1, and 1002 images plus DA (1572 images in total).

MLP	RF	SVM	LR	XGB
'solver': 'adam', 'learning_rate': 'adaptive', 'hidden_layer_sizes': (200), 'activation': 'tanh'	'n_estimators': 50, 'min_samples_split': 8, 'min_samples_leaf': 15, 'max_features': 'log2', 'max_depth': 2.	'kernel': 'rbf', 'gamma': 0.0001, 'C': 1000.	'penalty': 'l2', 'C': 100.	'n_estimators': 500, 'max_depth': 16, 'learning_rate': 0.1, 'gamma': 0, 'colsample_bytree': 1.

Source: The author (2022).

Figure A 6: Confusion matrices of the results obtained by the q-EFE approach using 200 images to train the classification models, the PDF, and the configuration 1



Source: The author (2022).

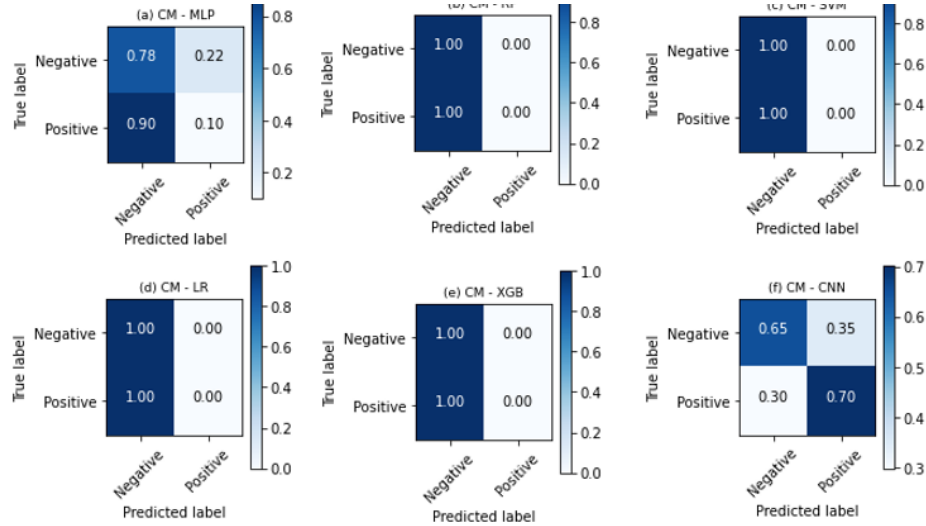
Table A - 6: Parameters that were chosen in the grid search for q-EFE, PDF, configuration 1, and 200 images.

MLP	RF	SVM	LR	XGB
-----	----	-----	----	-----

'solver': 'sgd', 'learning_rate': 'constant', 'hidden_layer_sizes': (50), 'activation': 'identity'.	'n_estimators': 200, 'min_samples_split': 8, 'min_samples_leaf': 15, 'max_features': 'sqrt', 'max_depth': 32.	kernel': 'linear', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 100.	n_estimators': 500, 'max_depth': 64, 'learning_rate': 0.05, 'gamma': 1, 'colsample_bytree': 0.3.
--	---	---	-------------------------------	--

Source: The author (2022).

Figure A 7: Confusion matrices of the results obtained by the q-EFE approach using 400 images to train the classification models, the PDF, and the configuration 1.



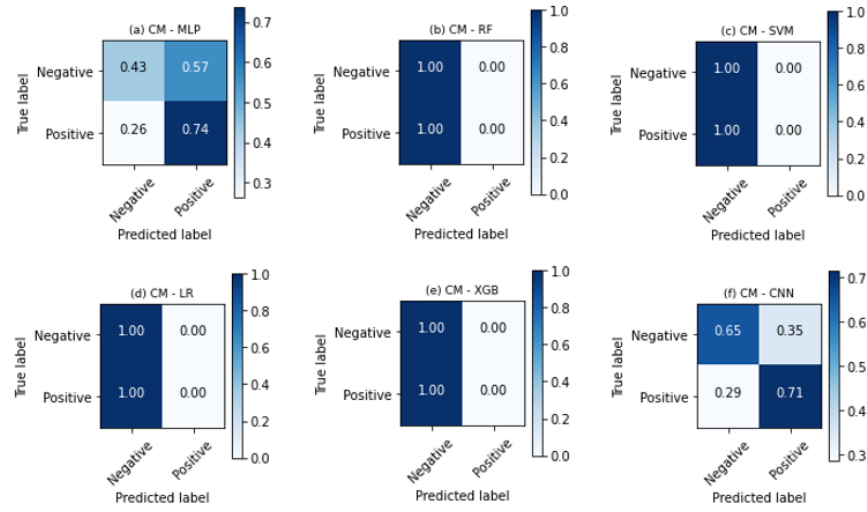
Source: The author (2022).

Table A - 7: Parameters that were chosen in the grid search for q-EFE, PDF, configuration 1, and 400 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'constant', 'hidden_layer_sizes': (50), 'activation': 'identity'	'n_estimators': 200, 'min_samples_split': 8, 'min_samples_leaf': 15, 'max_features': 'sqrt', 'max_depth': 32.	'kernel': 'linear', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 100.	n_estimators': 500, 'max_depth': 64, 'learning_rate': 0.05, 'gamma': 1, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 8: Confusion matrices of the results obtained by the q-EFE approach using 1002 images to train the classification models, the PDF, and the configuration 1.



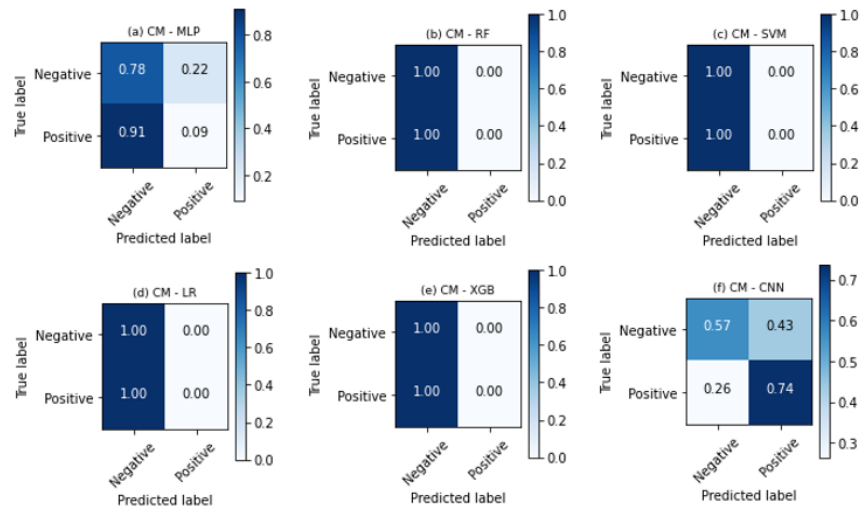
Source: The author (2022).

Table A - 8: Parameters that were chosen in the grid search for q-EFE, PDF, configuration 1, and 1002 images.

MLP	RF	SVM	LR	XGB
'solver': 'lbfgs', 'learning_rate': 'constant', 'hidden_layer_sizes': (50), 'activation': 'identity'.	'n_estimators': 200, 'min_samples_split': 8, 'min_samples_leaf': 15, 'max_features': 'sqrt', 'max_depth': 8.	'kernel': 'rbf', 'gamma': 0.01, 'C': 1.	'penalty': 'l2', 'C': 100.	'n_estimators': 500, 'max_depth': 64, 'learning_rate': 0.05, 'gamma': 1, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 9: Confusion matrices of the results obtained by the q-EFE approach using 1572 feature vectors (using SMOTE) to train the classification models, the PDF, and the configuration 1.



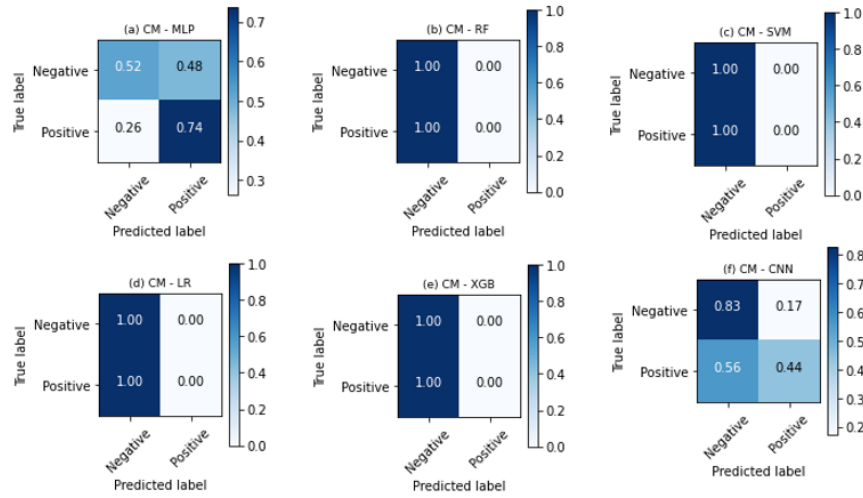
Source : The author (2022).

Table A - 9: Parameters that were chosen in the grid search for q-EFE, PDF, configuration 1, and 1002 images plus SMOTE (1572 images in total).

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'constant', 'hidden_layer_sizes': (50), 'activation': 'identity'	'n_estimators': 200, 'min_samples_split': 8, 'min_samples_leaf': 15, 'max_features': 'sqrt', 'max_depth': 32.	kernel': 'linear', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 100.	n_estimators': 500, 'max_depth': 64, 'learning_rate': 0.05, 'gamma': 1, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 10: Confusion matrices of the results obtained by the q-EFE approach using 1572 images (with DA) to train the classification models, the PDF, and the configuration 1.



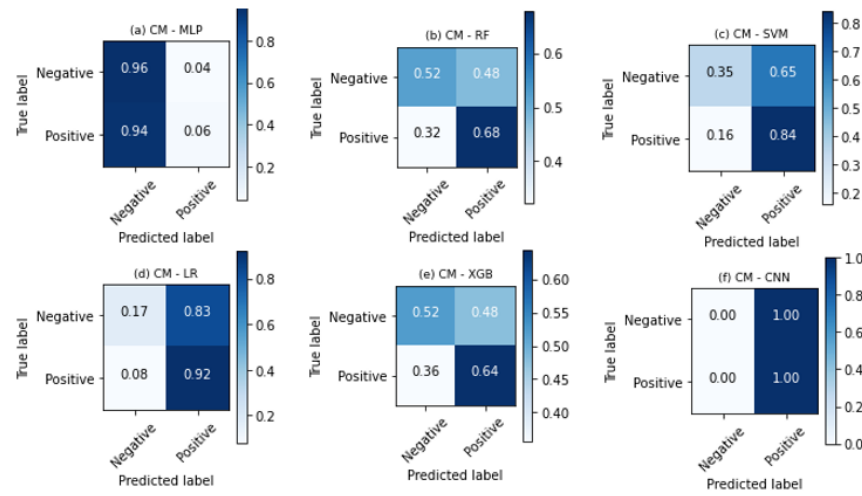
Source: The author (2022).

Table A - 10: Parameters that were chosen in the grid search for q-EFE, PDF, configuration 1, and 1002 images plus DA (1572 images in total).

MLP	RF	SVM	LR	XGB
'solver': 'lbfgs', 'learning_rate': 'adaptive', 'hidden_layer_sizes': (1000), 'activation': 'relu'	'n_estimators': 50, 'min_samples_split': 8, 'min_samples_leaf': 15 'max_features': 'sqrt' 'max_depth': 8.	kernel': 'linear', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2' 'C': 0.1.	n_estimators': 500, 'max_depth': 16, 'learning_rate': 0.1, 'gamma': 1, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 11: Confusion matrices of the results obtained by the q-EFE approach using 200 images to train the classification models, the entropy, and the configuration 1.



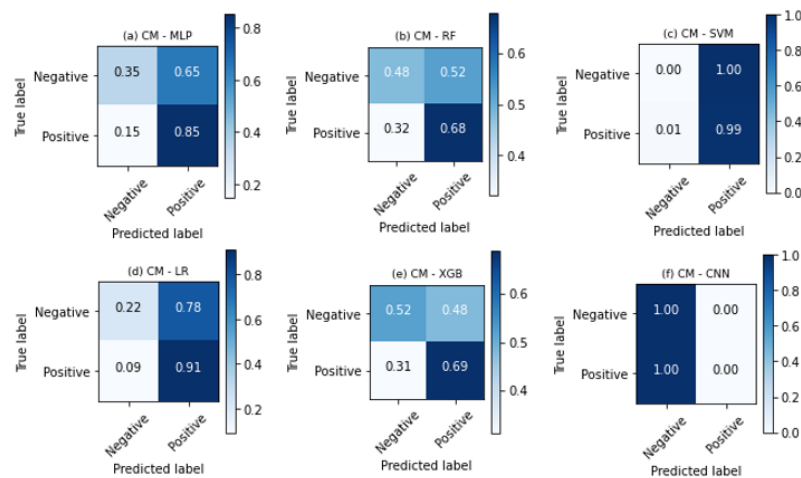
Source: The author (2022).

Table A - 11: Parameters that were chosen in the grid search for q-EFE, entropy function, configuration 1, and 200 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': adaptive, 'hidden_layer_sizes': (50), 'activation': 'relu'.	'n_estimators': 200, 'min_samples_split': 8, 'min_samples_leaf': 15, 'max_features': 'sqrt', 'max_depth': 8.	kernel': 'rbf', 'gamma': 0.01, 'C': 0.1.	'penalty': 'l2', 'C':100.	n_estimators': 500, 'max_depth': 16, 'learning_rate': 0.05, 'gamma': 1, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 12: Confusion matrices of the results obtained by the q-EFE approach using 400 images to train the classification models, the PDF, and the configuration 1.



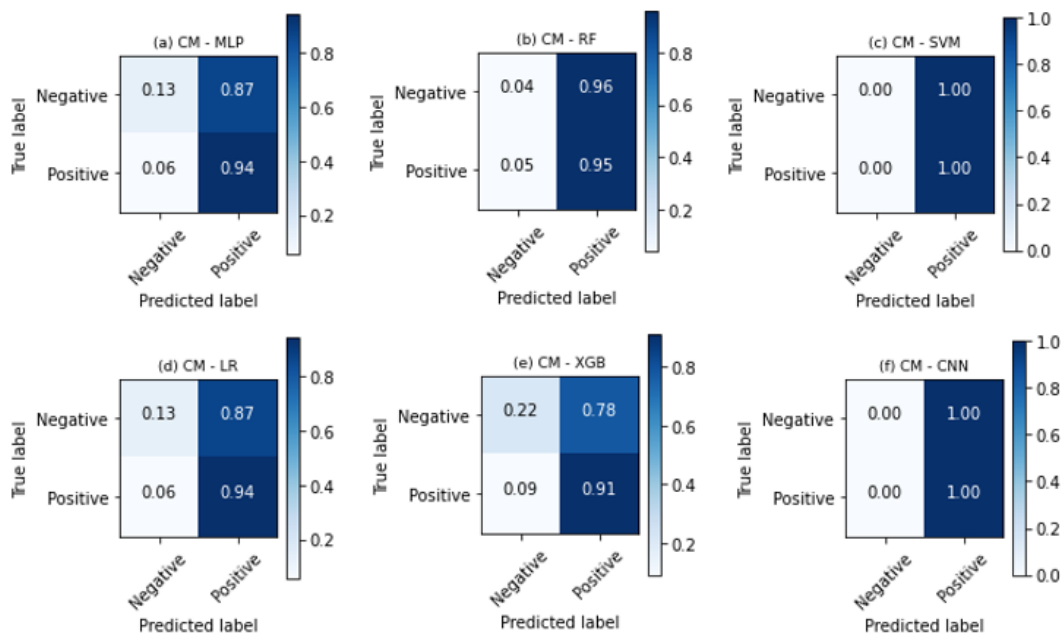
Source: The author (2022).

Table A - 12: Parameters that were chosen in the grid search for q-EFE, entropy function, configuration 1, and 200 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'adaptive', 'hidden_layer_sizes': (50), 'activation': 'tanh'.	'n_estimators': 50, 'min_samples_split': 24, 'min_samples_leaf': 10, 'max_features': 'log2', 'max_depth': 2.	kernel': 'rbf', 'gamma': 0.01, 'C': 1.	'penalty': 'l2', 'C': 1.	n_estimators': 500, 'max_depth': 2, 'learning_rate': 0.01, 'gamma': 1, 'colsample_bytree': 0.8.

Source: The author (2022).

Figure A 13: Confusion matrices of the results obtained by the q-EFE approach using 1002 images to train the classification models, the entropy, and the configuration 1.



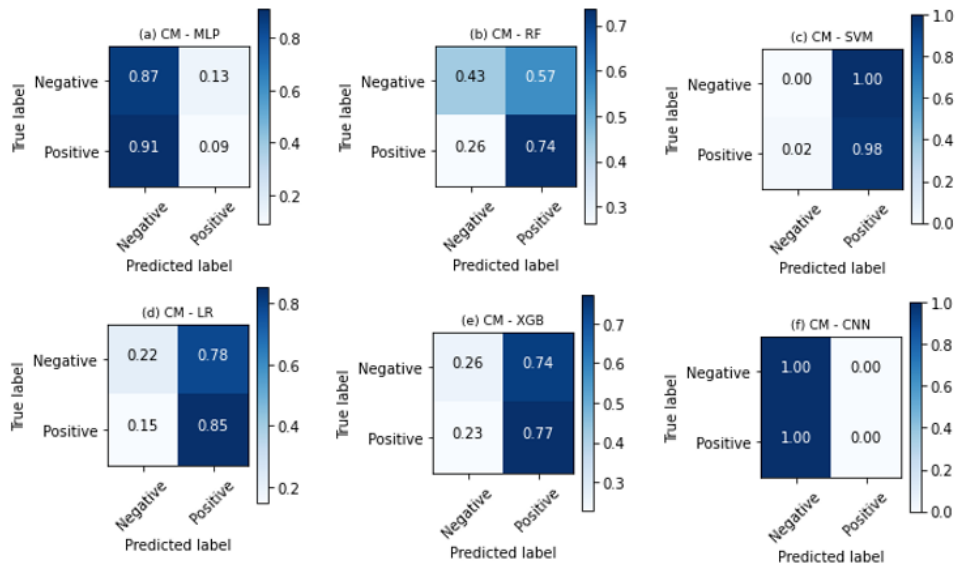
Source: The author (2022).

Table A - 13: Parameters that were chosen in the grid search for q-EFE, entropy function, configuration 1, and 1002 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'constant', 'hidden_layer_sizes': (50), 'activation': 'logistic'.	'n_estimators': 200, 'min_samples_split': 8, 'min_samples_leaf': 1, 'max_features': 'log2', 'max_depth': 2.	kernel': 'rbf', 'gamma': 'autp', 'C': 0.1.	'penalty': 'l2', 'C': 0.001.	n_estimators': 100, 'max_depth': 128, 'learning_rate': 0.01, 'gamma': 0, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 14: Confusion matrices of the results obtained by the q-EFEapproach using 1572 feature vectors (using SMOTE) to train the classification models, the entropy, and the configuration 1.



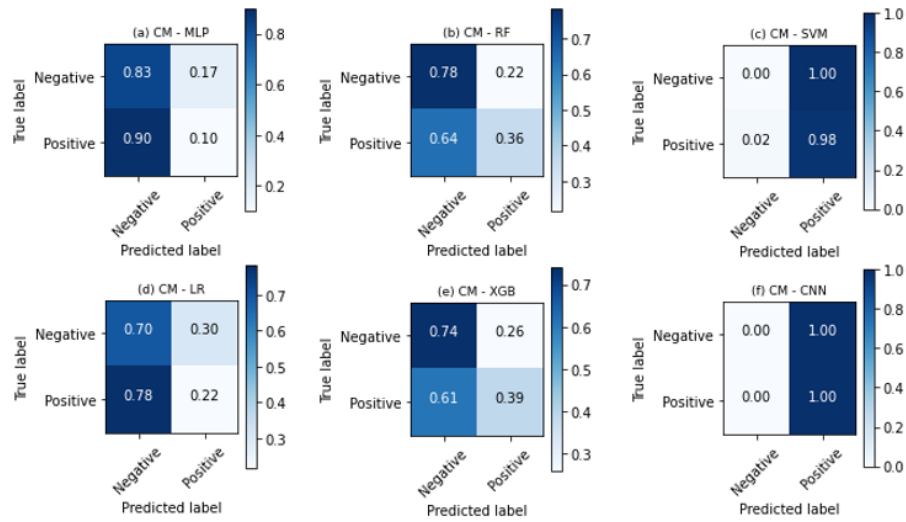
Source: The author (2022).

Table A - 14: Parameters that were chosen in the grid search for q-EFE, entropy function, configuration 1, and 1002 images plus SMOTE (1572 images in total).

MLP	RF	SVM	LR	XGB
'solver': 'adam', 'learning_rate': 'constant', 'hidden_layer_sizes': (1000), 'activation': 'identity'.	'n_estimators': 500, 'min_samples_split': 4, 'min_samples_leaf': 1, 'max_features': 'sqrt', 'max_depth': 32.	kernel: 'rbf', 'gamma': 0.01, 'C': 1.	'penalty': 'l2', 'C': 0.1.	n_estimators': 1000, 'max_depth': 16, 'learning_rate': 0.05, 'gamma': 0, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 15: Confusion matrices of the results obtained by the q-EFEapproach using 1572 images (with DA) to train the classification models, the entropy, and the configuration 1.



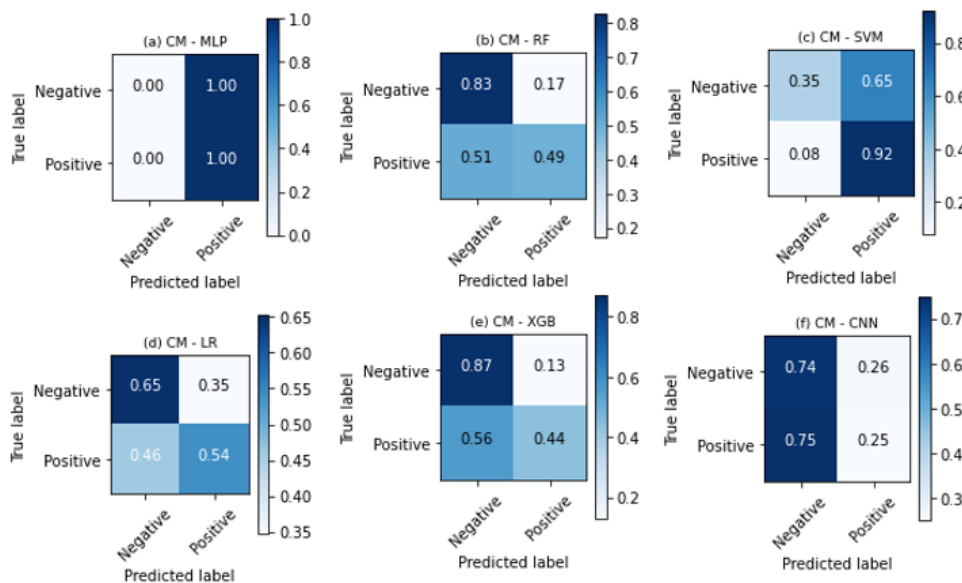
Source: The author (2022).

Table A - 15: Parameters that were chosen in the grid search for q-EFE, entropy function, configuration 1, and 1002 images plus DA (1572 images in total).

MLP	RF	SVM	LR	XGB
'solver': 'adam', 'learning_rate': 'constant', 'hidden_layer_sizes': (1000), 'activation': 'relu'.	'n_estimators': 100, 'min_samples_split': 8, 'min_samples_leaf': 2, 'max_features': 'sqrt', 'max_depth': 16.	kernel': 'rbf', 'gamma': 0.01, 'C': 1.	'penalty': 'l2', 'C': 0.001.	'n_estimators': 100, 'max_depth': 16, 'learning_rate': 0.01, 'gamma': 0, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 16: Confusion matrices of the results obtained by the q-EFE approach using 200 images to train the classification models, the CDF, and the configuration 2.



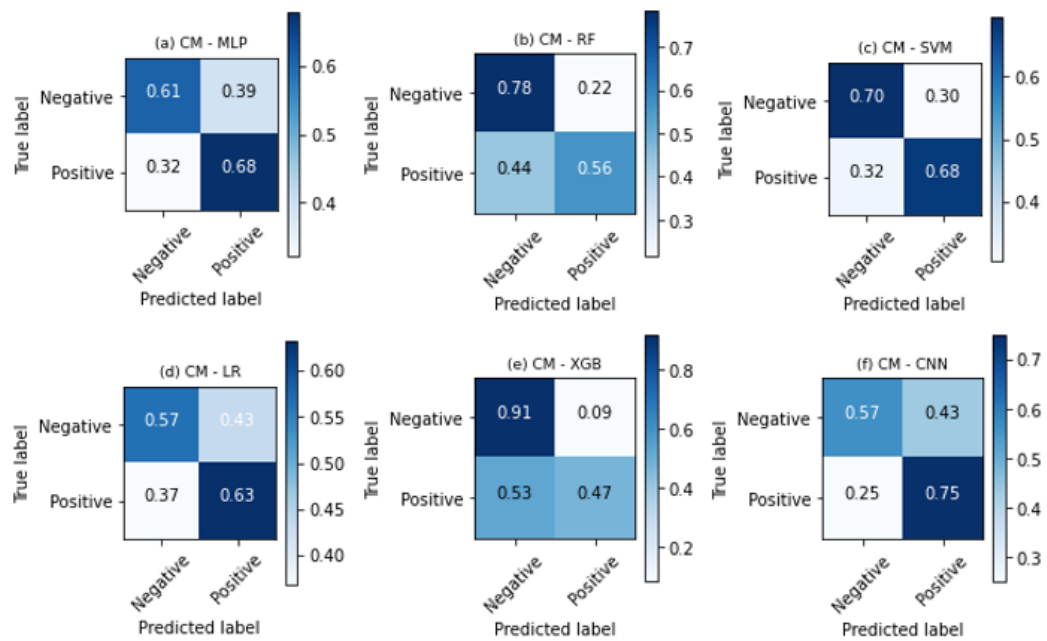
Source: The author (2022).

Table A - 16: Parameters that were chosen in the grid search for q-EFE, CDF, configuration 2, and 200 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'invscaling', 'hidden_layer_sizes': (200), 'activation': 'logistic'.	'n_estimators': 50, 'min_samples_split': 2, 'min_samples_leaf': 2, 'max_features': 10, 'max_depth': 2.	kernel': 'rbf', 'gamma': 0.0001, 'C': 0.1.	'penalty': 'l2', 'C': 0.1.	n_estimators': 100, 'max_depth': 128, 'learning_rate': 0.01, 'gamma': 5, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 17: Confusion matrices of the results obtained by the q-EFE approach using 400 images to train the classification models, the CDF, and the configuration 2.



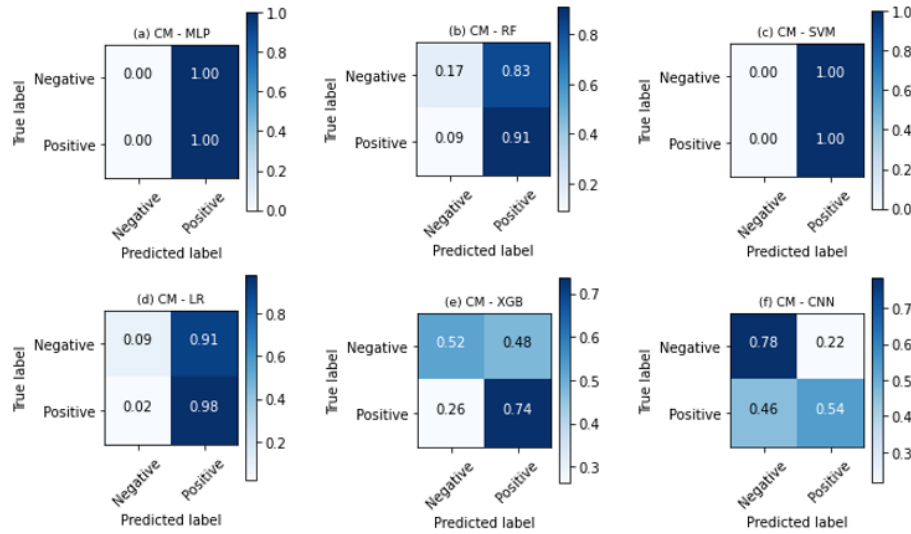
Source: The author (2022).

Table A - 17: Parameters that were chosen in the grid search for q-EFE, CDF, configuration 2, and 400 images.

MLP	RF	SVM	LR	XGB
'solver': 'adam', 'learning_rate': 'invscaling', 'hidden_layer_sizes': (50), 'activation': 'tanh'.	'n_estimators': 200, 'min_samples_split': 16, 'min_samples_leaf': 1, 'max_features': 10, 'max_depth': 2.	kernel': 'rbf', 'gamma': 0.01, 'C': 1.	'penalty': 'l2', 'C': 100.	n_estimators': 1000, 'max_depth': 8, 'learning_rate': 0.05, 'gamma': 5, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 18: Confusion matrices of the results obtained by the q-EFE approach using 1002 images to train the classification models, the CDF, and the configuration 2.



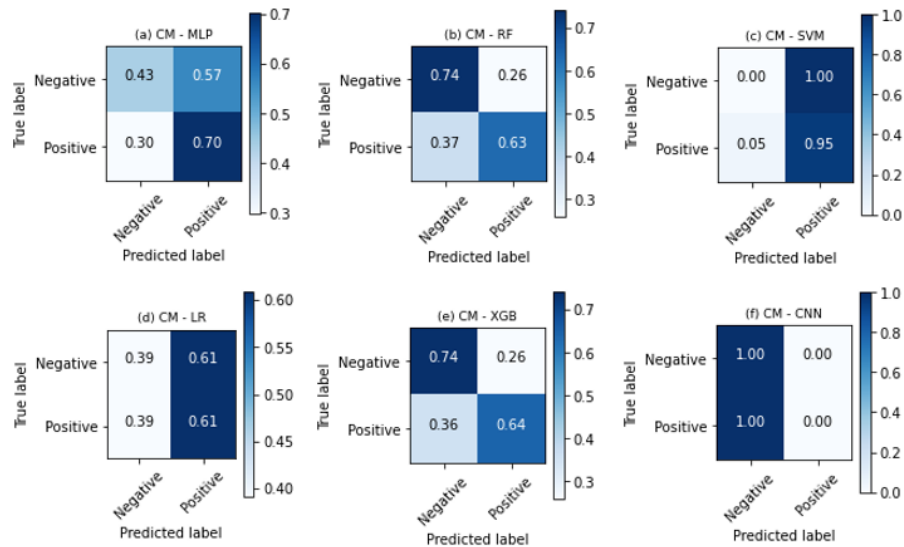
Source: The author (2022).

Table A - 18: Parameters that were chosen in the grid search for q-EFE, CDF, configuration 2, and 1002 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'constant', 'hidden_layer_sizes': (50), 'activation': 'logistic'.	'n_estimators': 100, 'min_samples_split': 16, 'min_samples_leaf': 10, 'max_features': 'log2', 'max_depth': 2.	'kernel': 'rbf', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 0.01.	'n_estimators': 100, 'max_depth': 2, 'learning_rate': 0.01, 'gamma': 5, 'colsample_bytree': 1.

Source: The author (2022).

Figure A 19: Confusion matrices of the results obtained by the q-EFE approach using 1572 feature vectors (using SMOTE) to train the classification models, the CDF, and the configuration 2.



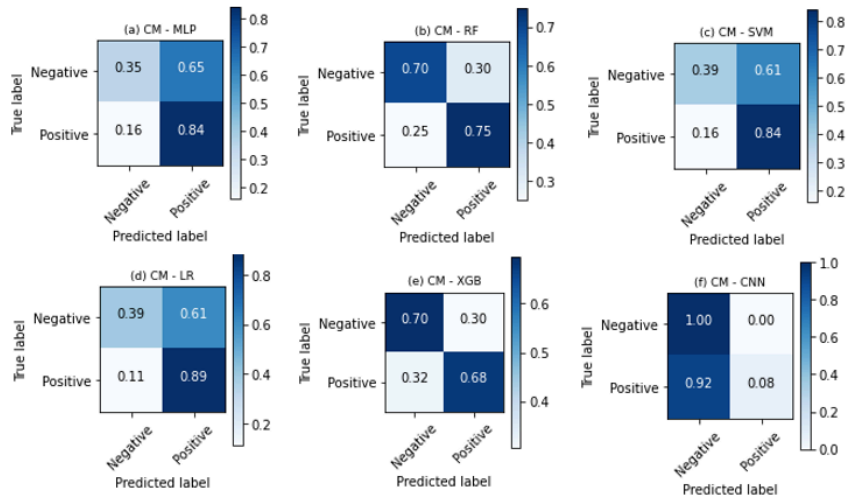
Source: The author (2022).

Table A - 19: Parameters that were chosen in the grid search for q-EFE, CDF, configuration 2, and 1002 images plus SMOTE (1572 images in total).

MLP	RF	SVM	LR	XGB
'solver': 'adam', 'learning_rate': 'adaptive', 'hidden_layer_sizes': (200), 'activation': 'tanh'.	'n_estimators': 500, 'min_samples_split': 4, 'min_samples_leaf': 1, 'max_features': 'sqrt', 'max_depth': 32.	'kernel': 'rbf', 'gamma': 1, 'C': 1.	'penalty': 'l2', 'C': 0.1.	'n_estimators': 500, 'max_depth': 16, 'learning_rate': 0.05, 'gamma': 0, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 20: Confusion matrices of the results obtained by the q-EFE approach using 1572 images (with DA) to train the classification models, the CDF, and the configuration 2.



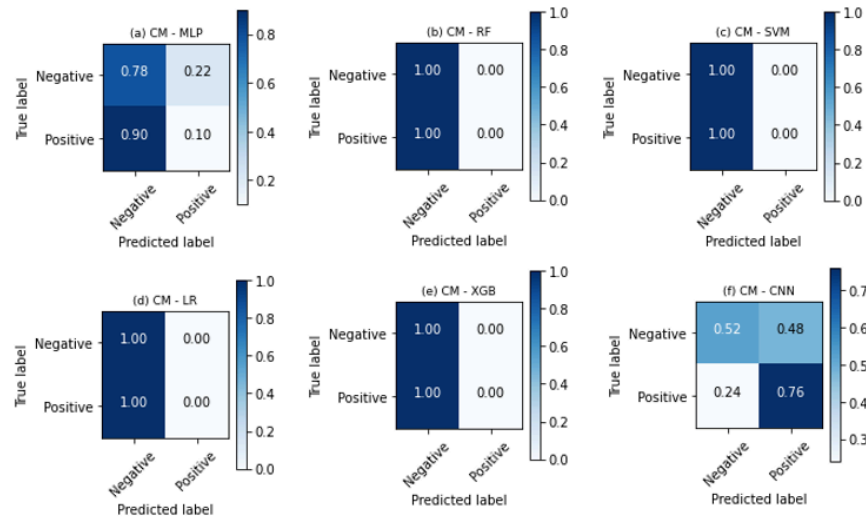
Source: The author (2022).

Table A - 20: Parameters that were chosen in the grid search for q-EFE, CDF, configuration 1, and 1002 images plus DA (1572 images in total).

MLP	RF	SVM	LR	XGB
'solver': 'adam', 'learning_rate': 'constant', 'hidden_layer_sizes': (100), 'activation': 'tanh'.	'n_estimators': 200, 'min_samples_split': 2, 'min_samples_leaf': 1, 'max_features': 'sqrt', 'max_depth': 8.	'kernel': 'rbf', 'gamma': 0.0001, 'C': 1000.	'penalty': 'l2', 'C': 10.	'n_estimators': 100, 'max_depth': 8, 'learning_rate': 0.01, 'gamma': 0, 'colsample_bytree': 0.8.

Source: The author (2022).

Figure A 21; Confusion matrices of the results obtained by the q-EFEapproach using 200 images to train the classification models, the PDF, and the configuration 2.



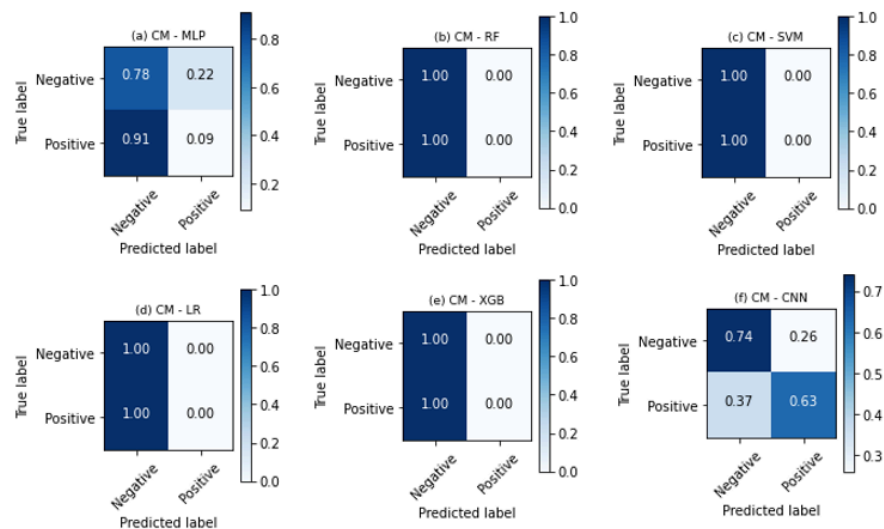
Source: The author (2022).

Table A - 21: Parameters that were chosen in the grid search for q-EFE, PDF, configuration 2, and 200 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'constant', 'hidden_layer_sizes': (50), 'activation': 'identity'.	'n_estimators': 200, 'min_samples_split': 8, 'min_samples_leaf': 15, 'max_features': 'sqrt', 'max_depth': 32.	'kernel': 'linear', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 100.	'n_estimators': 500, 'max_depth': 64, 'learning_rate': 0.05, 'gamma': 1, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 22: Confusion matrices of the results obtained by the q-EFE approach using 400 images to train the classification models, the PDF, and the configuration 2.



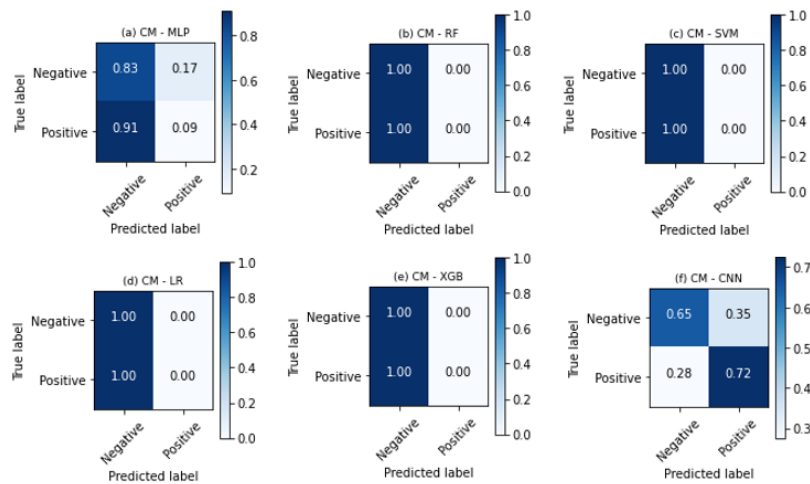
Source: The author (2022).

Table A - 22: Parameters that were chosen in the grid search for q-EFE, PDF, configuration 2, and 400 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'constant', 'hidden_layer_sizes': (50), 'activation': 'identity'.	'n_estimators': 200, 'min_samples_split': 8, 'min_samples_leaf': 15, 'max_features': 'sqrt', 'max_depth': 32.	kernel': 'linear', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 100.	n_estimators': 500, 'max_depth': 64, 'learning_rate': 0.05, 'gamma': 1, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 23: Confusion matrices of the results obtained by the q-EFE approach using 1002 images to train the classification models, the PDF, and the configuration 2



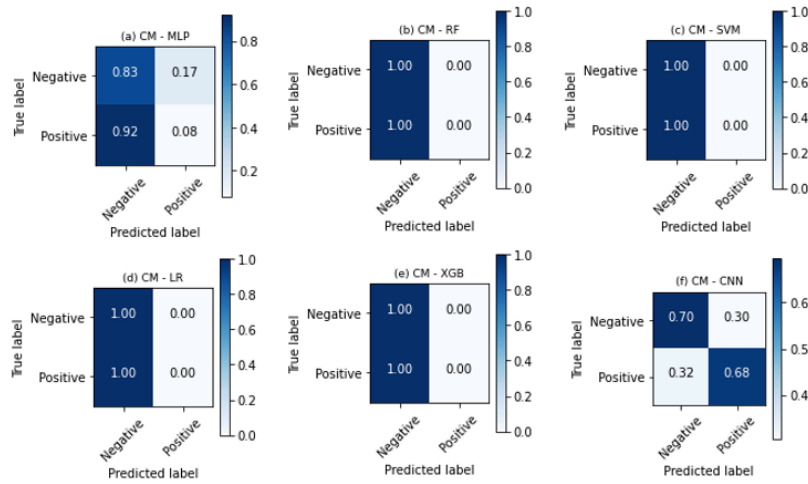
Source: The author (2022).

Table A - 23: Parameters that were chosen in the grid search for q-EFE, PDF, configuration 2, and 1002 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'constant', 'hidden_layer_sizes': (50), 'activation': 'identity'.	'n_estimators': 200, 'min_samples_split': 8, 'min_samples_leaf': 15, 'max_features': 'sqrt', 'max_depth': 32.	kernel': 'linear', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 100.	n_estimators': 500, 'max_depth': 64, 'learning_rate': 0.05, 'gamma': 1, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 24: Confusion matrices of the results obtained by the q-EFE approach using 1572 feature vectors (using SMOTE) to train the classification models, the PDF, and the configuration 2.



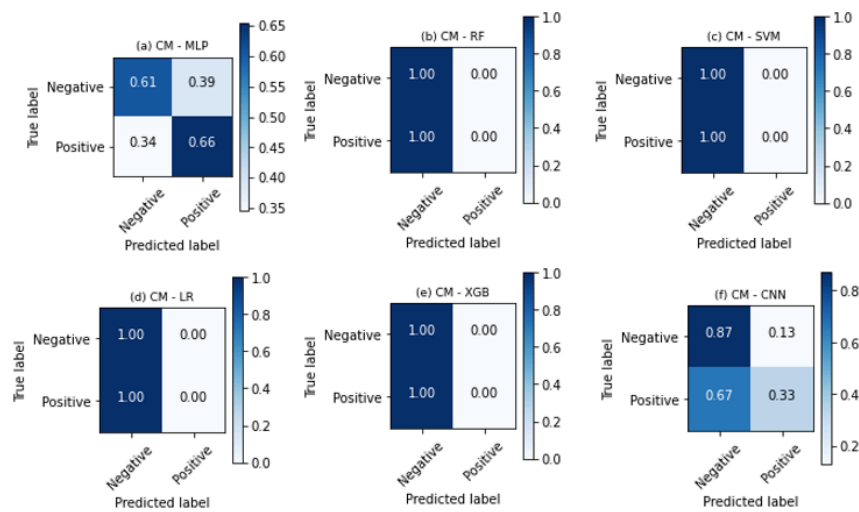
Source: The author (2022).

Table A - 24: Parameters that were chosen in the grid search for q-EFE, PDF, configuration 2, and 1002 images plus SMOTE (1572 images in total).

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'constant', 'hidden_layer_sizes': (50), 'activation': 'identity'.	'n_estimators': 200, 'min_samples_split': 8, 'min_samples_leaf': 15, 'max_features': 'sqrt', 'max_depth': 32.	'kernel': 'linear', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 100.	'n_estimators': 500, 'max_depth': 64, 'learning_rate': 0.05, 'gamma': 1, 'colsample_bytree': 0.3

Source: The author (2022).

Figure A 25: Confusion matrices of the results obtained by the q-EFE approach using 1572 images (with DA) to train the classification models, the PDF, and the configuration 2.



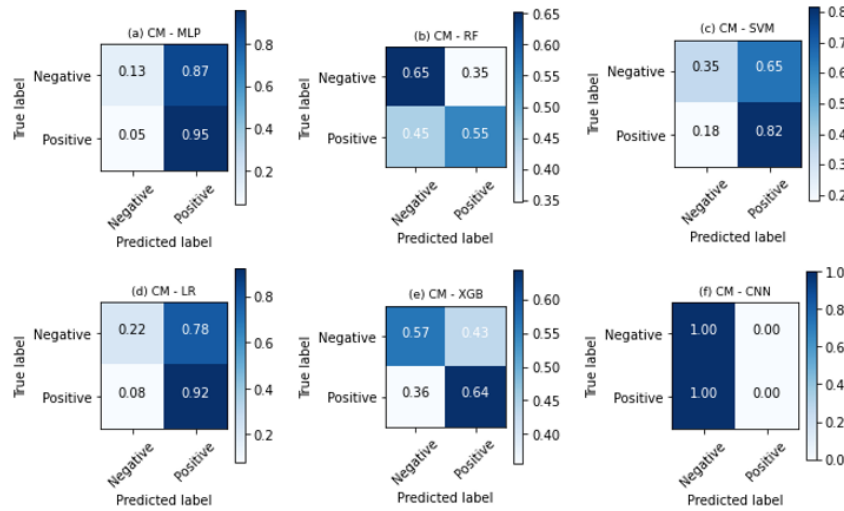
Source: The author (2022).

Table A - 25: Parameters that were chosen in the grid search for q-EFE, PDF, configuration 1, and 1002 images plus DA (1572 images in total).

MLP	RF	SVM	LR	XGB
'solver': 'lbfgs', 'learning_rate': 'constant', 'hidden_layer_sizes': (50), 'activation': 'identity'.	'n_estimators': 50, 'min_samples_split': 16, 'min_samples_leaf': 1, 'max_features': 'log2', 'max_depth': 2.	kernel': 'linear', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 100.	n_estimators': 500, 'max_depth': 64, 'learning_rate': 0.05, 'gamma': 1, 'colsample_bytree': 0.3

Source: The author (2022).

Figure A 26: Confusion matrices of the results obtained by the q-EFE approach using 200 images to train the classification models, the entropy, and the configuration 2.



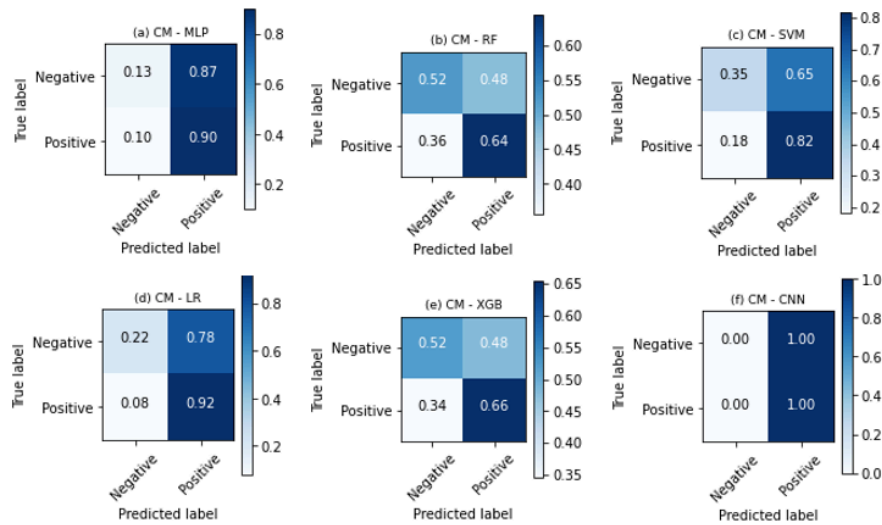
Source: The author (2022).

Table A - 26: Parameters that were chosen in the grid search for q-EFE, entropy function, configuration 2, and 200 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'constant', 'hidden_layer_sizes': (1000) 'activation': 'identity'.	'n_estimators': 10, 'min_samples_split': 4, 'min_samples_leaf': 10, 'max_features': 'sqrt', 'max_depth': 8.	kernel': 'rbf', 'gamma': 0.01, 'C': 0.1.	'penalty': 'l2', 'C': 0.001.	n_estimators': 100, 'max_depth': 2, 'learning_rate': 0.1, 'gamma': 5, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 27: Confusion matrices of the results obtained by the q-EFE approach using 400 images to train the classification models, the entropy, and the configuration 2.



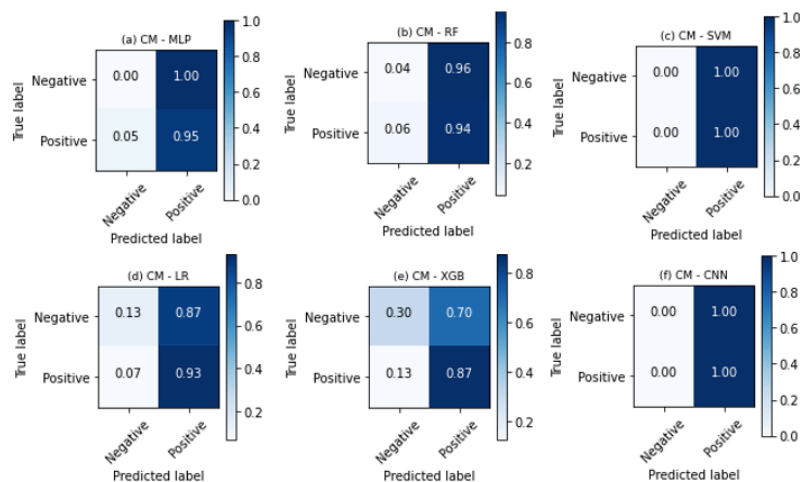
Source: The author (2022).

Table A - 27: Parameters that were chosen in the grid search for q-EFE, entropy function, configuration 2, and 400 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'invscaling', 'hidden_layer_sizes': (500), 'activation': 'tanh'.	'n_estimators': 10, 'min_samples_split': 16, 'min_samples_leaf': 5, 'max_features': 'auto', 'max_depth': 2.	kernel': 'rbf', 'gamma': 0.01, 'C': 0.1.	'penalty': 'l2', 'C': 0.001.	n_estimators': 1000, 'max_depth': 2, 'learning_rate': 0.1, 'gamma': 5, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 28: Confusion matrices of the results obtained by the q-EFE approach using 1002 images to train the classification models, the entropy, and the configuration 2.



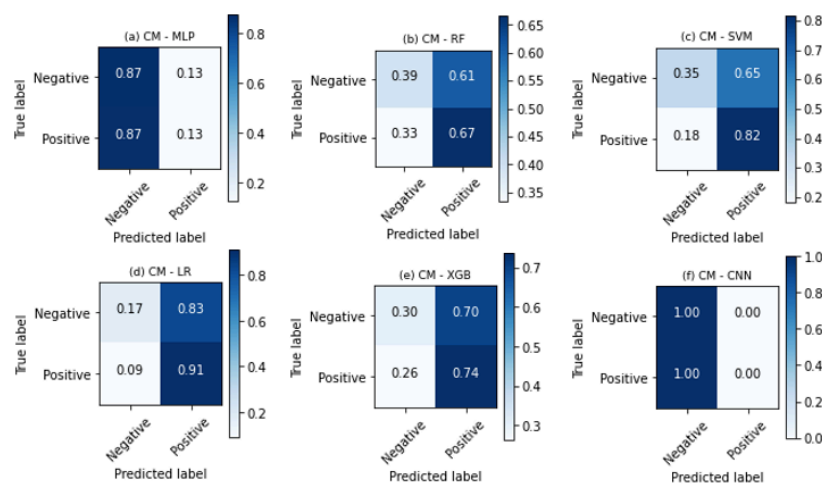
Source: The author (2022).

Table A - 28: Parameters that were chosen in the grid search for q-EFE, entropy function, configuration 2, and 1002 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'adaptive', 'hidden_layer_sizes': (50), 'activation': 'logistic'.	'n_estimators': 50, 'min_samples_split': 2, 'min_samples_leaf': 30, 'max_features': 'auto', 'max_depth': 2.	kernel': 'rbf', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 0.001.	n_estimators': 100, 'max_depth': 2, 'learning_rate': 0.01, 'gamma': 5, 'colsample_bytree': 1.

Source: The author (2022).

Figure A 29: Confusion matrices of the results obtained by the q-EFE approach using 1572 feature vectors (using SMOTE) to train the classification models, the entropy, and the configuration 2.



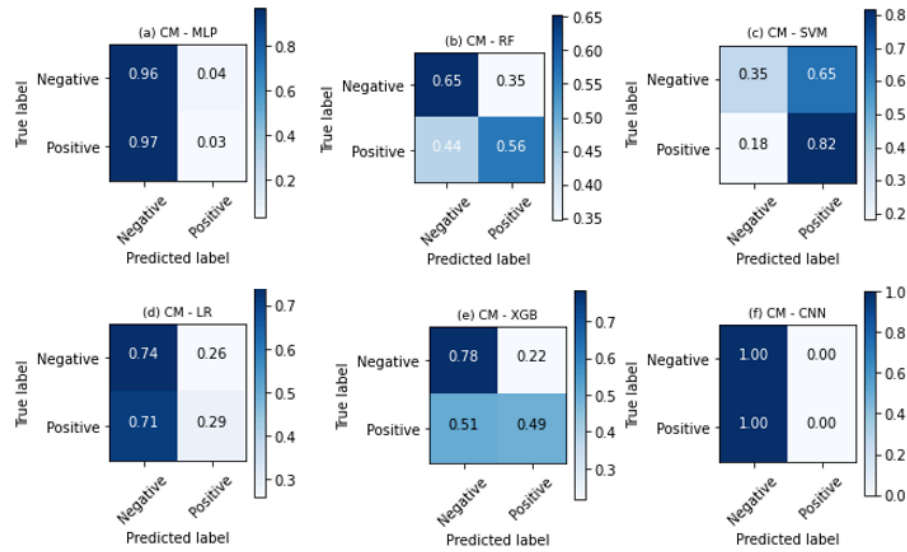
Source: The author (2022).

Table A - 29: Parameters that were chosen in the grid search for q-EFE, entropy function, configuration 2, and 1002 images plus SMOTE (1572 images in total).

MLP	RF	SVM	LR	XGB
'solver': 'adam', 'learning_rate': 'invscaling', 'hidden_layer_sizes': (100), 'activation': 'relu'.	'n_estimators': 100, 'min_samples_split': 2, 'min_samples_leaf': 2, 'max_features': 'auto', 'max_depth': 32.	kernel': 'rbf', 'gamma': 0.01, 'C': 0.1.	'penalty': 'l2', 'C': 0.001.	n_estimators': 100, 'max_depth': 128, 'learning_rate': 0.01, 'gamma': 0, 'colsample_bytree': 0.8.

Source: The author (2022).

Figure A 30: Confusion matrices of the results obtained by the q-EFE approach using 1572 images (with DA) to train the classification models, the entropy, and the configuration 2.



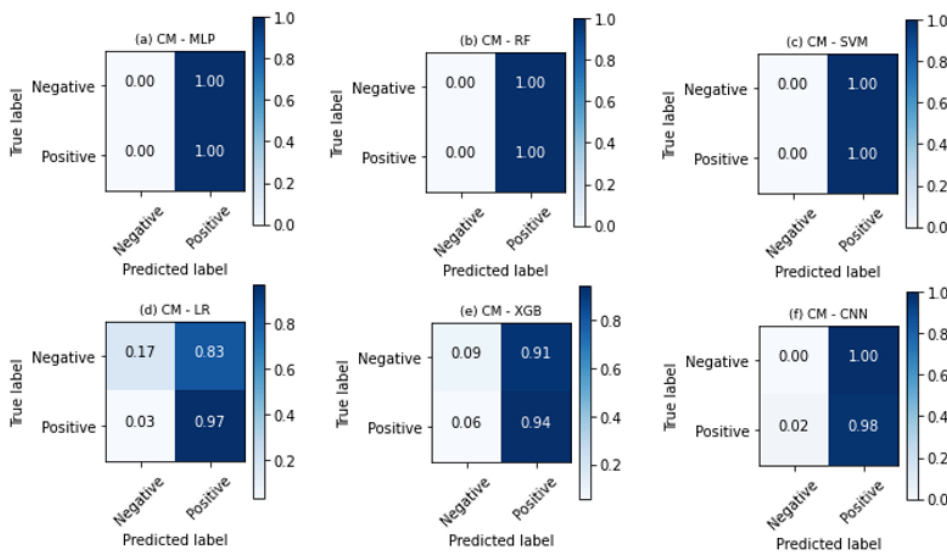
Source: The author (2022).

Table A - 30: Parameters that were chosen in the grid search for q-EFE, entropy function, configuration 2, and 1002 images plus DA (1572 images in total).

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'constant', 'hidden_layer_sizes': (1000), 'activation': 'logistic'.	'n_estimators': 500, 'min_samples_split': 2, 'min_samples_leaf': 5, 'max_features': 'auto', 'max_depth': 8.	kernel: 'rbf', 'gamma': 0.01, 'C': 0.1.	'penalty': 'l2', 'C': 10.	'n_estimators': 500, 'max_depth': 8, 'learning_rate': 0.1, 'gamma': 0, 'colsample_bytree': 0.8.

Source: The author (2022).

Figure A 31: Confusion matrices of the results obtained by the LBP, using the 200 images to train the classification methods



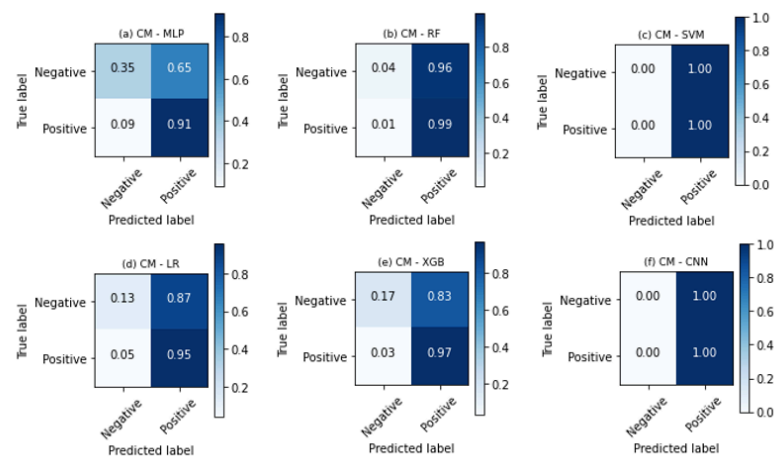
Source: The author (2022).

Table A - 31: Parameters that were chosen in the grid search for LBP and 200 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'constant', 'hidden_layer_sizes': (50), 'activation': 'identity'.	'n_estimators': 50, 'min_samples_split': 16, 'min_samples_leaf': 30, 'max_features': 'sqrt', 'max_depth': 2.	'kernel': 'linear', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 0.1.	'n_estimators': 100, 'max_depth': 128, 'learning_rate': 0.01, 'gamma': 5, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 32: Confusion matrices of the results obtained by the LBP, using the 400 images to train the classification methods.



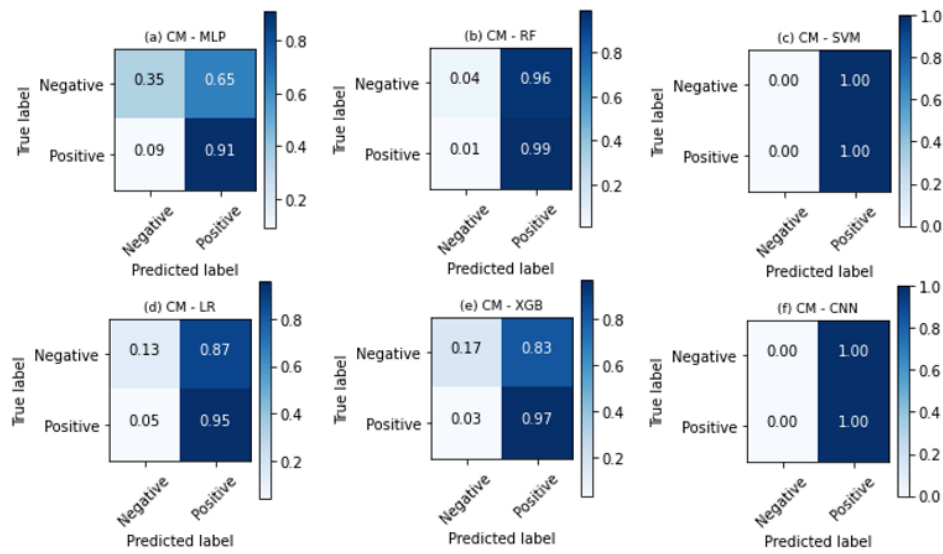
Source: The author (2022).

Table A - 32: Parameters that were chosen in the grid search for LBP and 400 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'adptive', 'hidden_layer_sizes': (50), 'activation': 'identity'.	'n_estimators': 100, 'min_samples_split': 2, 'min_samples_leaf': 2, 'max_features': 'sqrt', 'max_depth': 2.	'kernel': 'linear', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 0.01.	'n_estimators': 100, 'max_depth': 2, 'learning_rate': 0.1, 'gamma': 5, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 33: Confusion matrices of the results obtained by the LBP, using the entire dataset to train the classification methods, without SMOTE and DA.



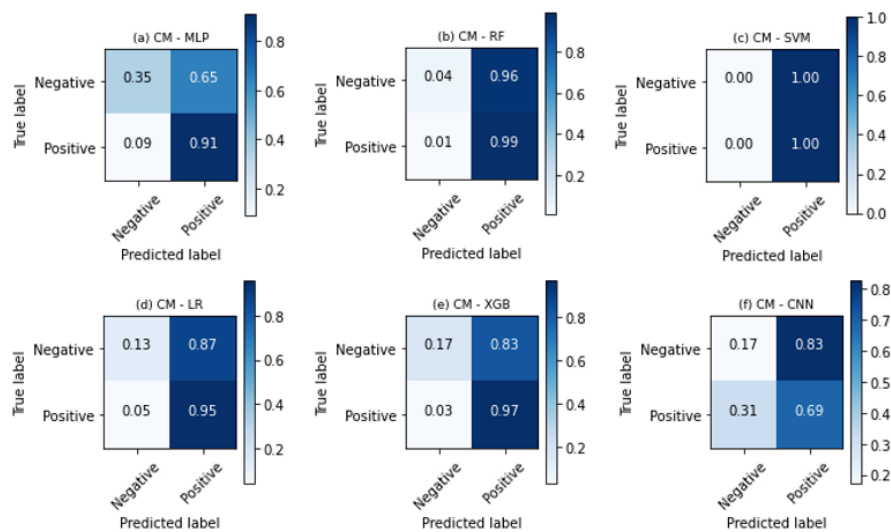
Source: The author (2022).

Table A - 33: Parameters that were chosen in the grid search for LBP and 1002 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'adptive', 'hidden_layer_sizes': (50), 'activation': 'identity'.	'n_estimators': 100, 'min_samples_split': 2, 'min_samples_leaf': 2, 'max_features': 'sqrt', 'max_depth': 2.	'kernel': 'linear', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 0.01.	'n_estimators': 100, 'max_depth': 2, 'learning rate': 0.1, 'gamma': 5, 'colsample_bytree': 0.3.

Source: The author (2022).

Figure A 34: Confusion matrices of the results obtained by the LBP, using the entire dataset plus SMOTE (1572 feature vectors in total to train the classification methods).



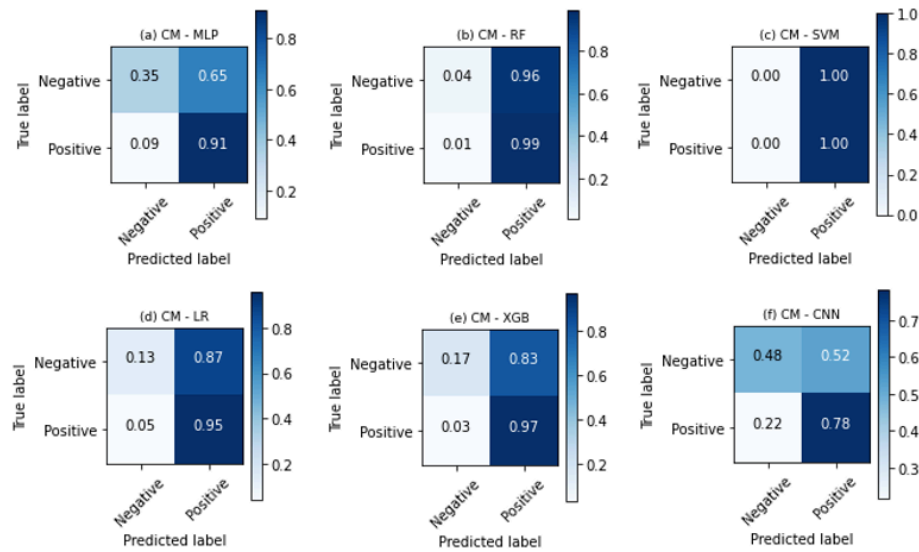
Source: The author (2022).

Table A - 34: Parameters that were chosen in the grid search for LBP and 1002 images plus SMOTE (1572 images in total).

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'invscaling', 'hidden_layer_sizes': (50), 'activation': 'relu'.	'n_estimators': 50, 'min_samples_split': 4, 'min_samples_leaf': 1, 'max_features': 5, 'max_depth': 128,	'kernel': 'rbf', 'gamma': 'auto', 'C': 1.	'penalty': 'l2', 'C': 0.001.	'n_estimators': 1000, 'max_depth': 64, 'learning_rate': 0.01, 'gamma': 0, 'colsample_bytree': 0.8.

Source: The author (2022).

Figure A 35: Confusion matrices of the results obtained by the LBP for texture analysis, using the entire dataset plus DA (1572 images in total to train the classification)



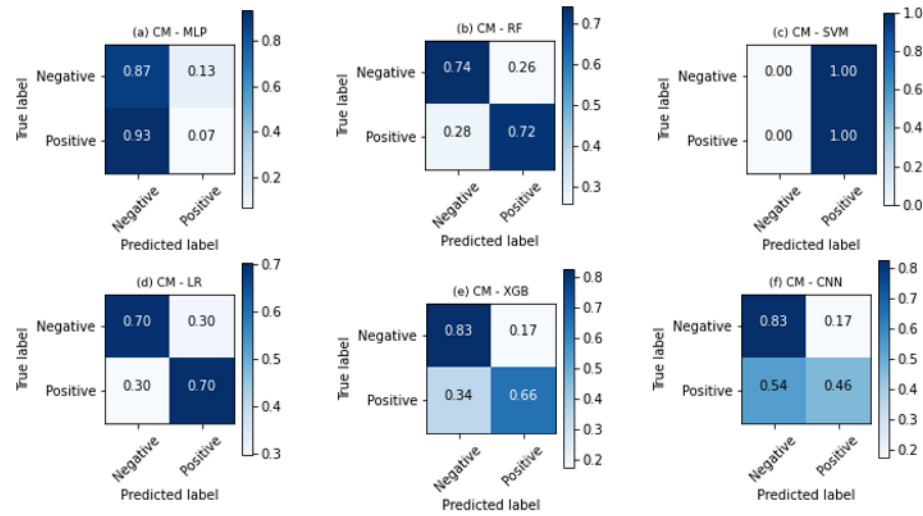
Source: The author (2022).

Table A - 35: Parameters that were chosen in the grid search for LBP and 1002 images plus DA (1572 images in total).

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': constant', 'hidden_layer_sizes': (100), 'activation': 'relu'	'n_estimators': 200, 'min_samples_split': 16, 'min_samples_leaf': 1, 'max_features': 10, 'max_depth': 2.	'kernel': 'sigmoid', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 0.001.	'n_estimators': 100, 'max_depth': 2, 'learning_rate': 0.01, 'gamma': 5, 'colsample_bytree': 1.

Source: The author (2022).

Figure A 36: Confusion matrices of the results obtained by the GLCM, using the 200 images to train the classification methods.



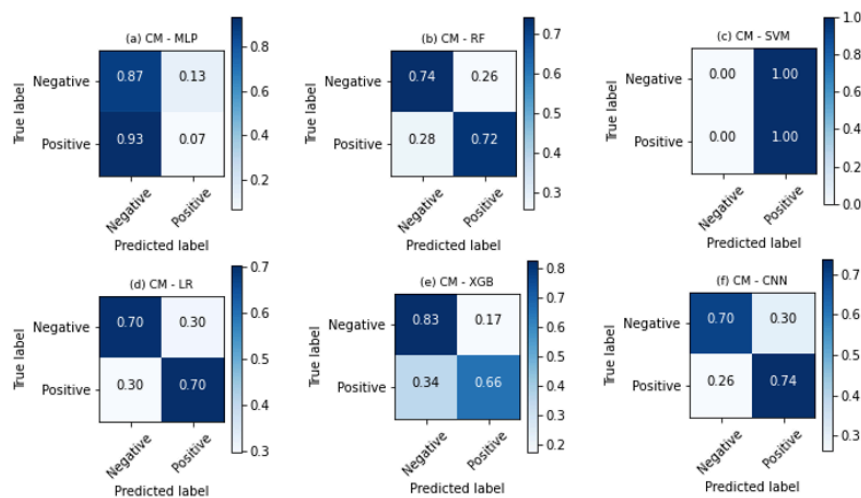
Source: The author (2022).

Table A - 36: Parameters that were chosen in the grid search for GLCM and 200 images.

MLP	RF	SVM	LR	XGB
'solver': 'adam', 'learning_rate': 'invscaling', 'hidden_layer_sizes': (1000), 'activation': 'relu'.	'n_estimators': 10, 'min_samples_split': 4, 'min_samples_leaf': 30, 'max_features': '5', 'max_depth': 16.	'kernel': 'sigmoid', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 1.	'n_estimators': 1000, 'max_depth': 2, 'learning_rate': 0.1, 'gamma': 0, 'colsample_bytree': 0.8.

Source: The author (2022).

Figure A 37: Confusion matrices of the results obtained by the GLCM, using the 400 images to train the classification methods.



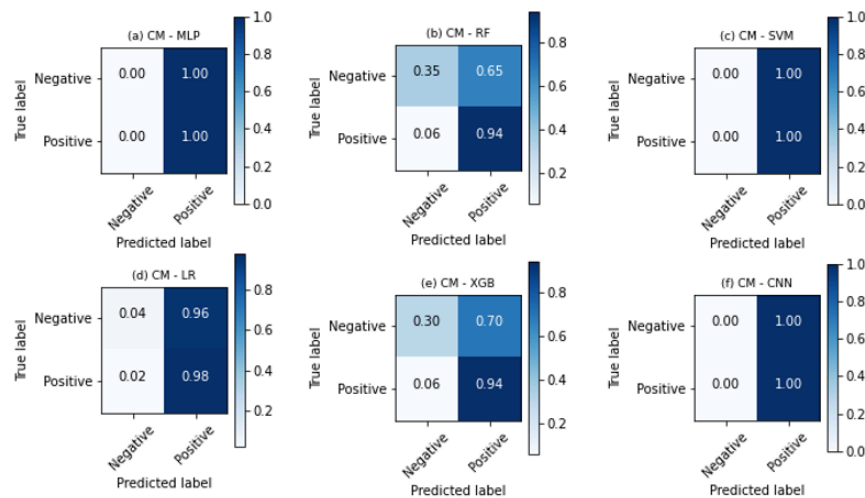
Source: The author (2022).

Table A - 37: Parameters that were chosen in the grid search for GLCM and 400 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'adptive', 'hidden_layer_sizes': (1000), 'activation': 'relu'.	'n_estimators': 10, 'min_samples_split': 2, 'min_samples_leaf': 10, 'max_features': '5', 'max_depth': 16.	'kernel': 'sigmoid', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 0.1.	'n_estimators': 500, 'max_depth': 64, 'learning_rate': 0.05, 'gamma': 5, 'colsample_bytree': 1.

Source: The author (2022).

Figure A 38: Confusion matrices of the results obtained by the GLCM, using the entire dataset to train the classification methods, without SMOTE and DA.



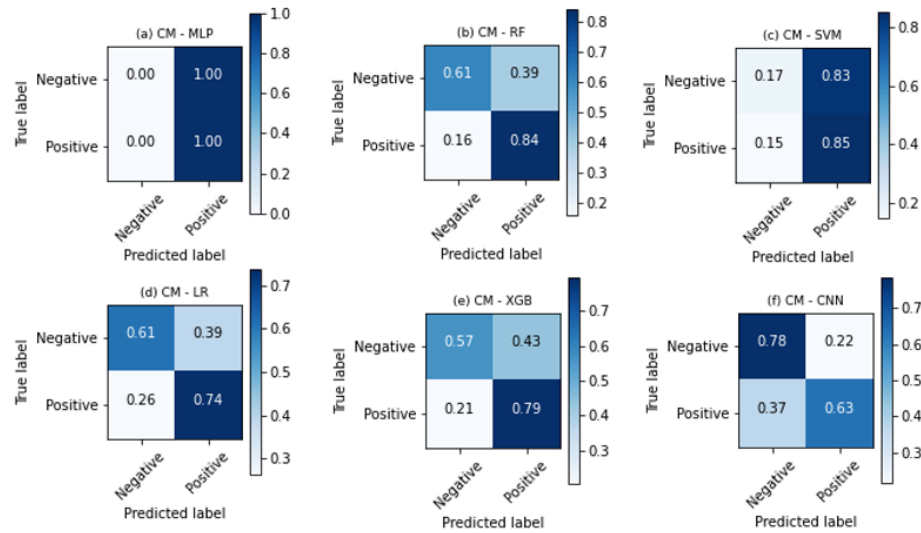
Source: The author (2022).

Table A - 38: Parameters that were chosen in the grid search for GLCM and 1002 images.

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'invscaling', 'hidden_layer_sizes': (500), 'activation': 'identity'.	'n_estimators': 200, 'min_samples_split': 24, 'min_samples_leaf': 30, 'max_features': 'log2', 'max_depth': 2.	'kernel': 'linear', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 0.1.	'n_estimators': 100, 'max_depth': 2, 'learning_rate': 0.01, 'gamma': 5, 'colsample_bytree': 1.

Source: The author (2022).

Figure A 39: Confusion matrices of the results obtained by the GLCM, using the entire dataset plus SMOTE (1572 feature vectors in total to train the classification methods).



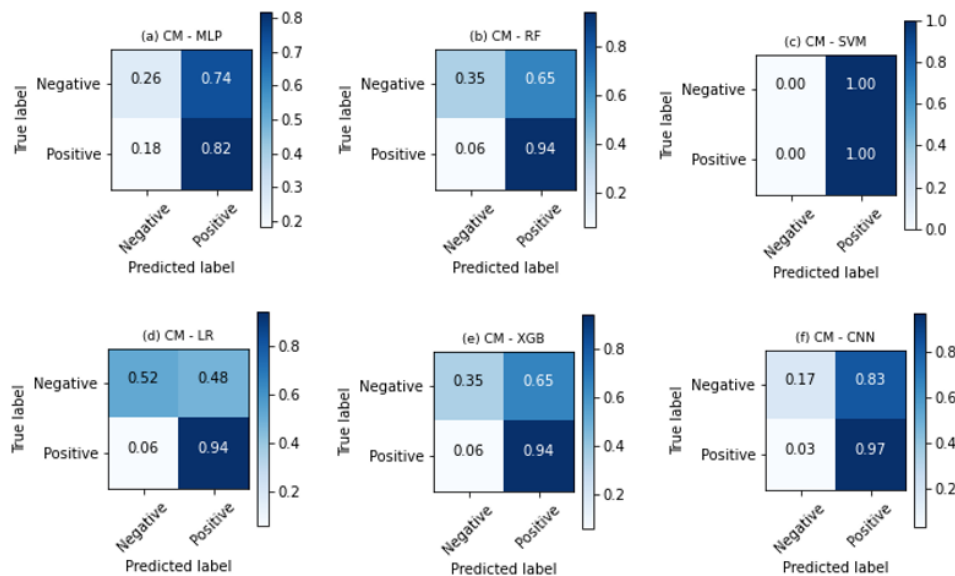
Source: The author (2022).

Table A - 39: Parameters that were chosen in the grid search for GLCM and 1002 images plus SMOTE (1572 images in total).

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': 'invscaling', 'hidden_layer_sizes': (50), 'activation': 'relu'.	'n_estimators': 50, 'min_samples_split': 4, 'min_samples_leaf': 1, 'max_features': 5, 'max_depth': 128,	'kernel': 'rbf', 'gamma': 'auto', 'C': 1.	'penalty': 'l2', 'C': 0.001.	'n_estimators': 1000, 'max_depth': 64, 'learning_rate': 0.01, 'gamma': 0, 'colsample_bytree': 0.8.

Source: The author (2022).

Figure A 40: Confusion matrices of the results obtained by the GLCM for texture analysis, using the entire dataset plus DA (1572 images in total to train the classification).



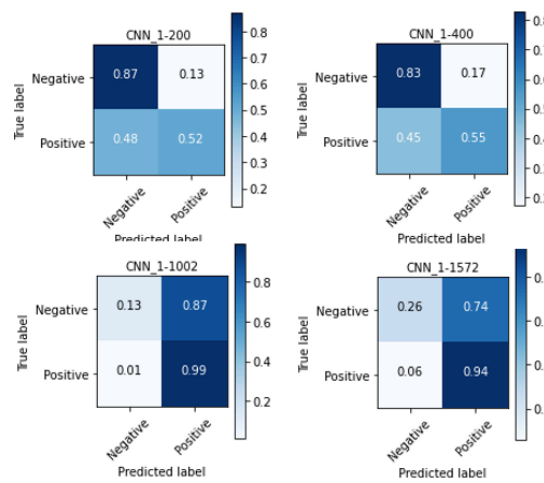
Source: The author (2022).

Table A - 40: Parameters that were chosen in the grid search for GLCM and 1002 images plus DA (1572 images in total).

MLP	RF	SVM	LR	XGB
'solver': 'sgd', 'learning_rate': constant', 'hidden_layer_sizes': (100), 'activation': 'relu'	'n_estimators': 50, 'min_samples_split': 16, 'min_samples_leaf': 1, 'max_features': 5, 'max_depth': 2.	'kernel': 'sigmoid', 'gamma': 'auto', 'C': 0.1.	'penalty': 'l2', 'C': 100.	'n_estimators': 100, 'max_depth': 2, 'learning_rate': 0.10, 'gamma': 5, 'colsample_bytree': 1.

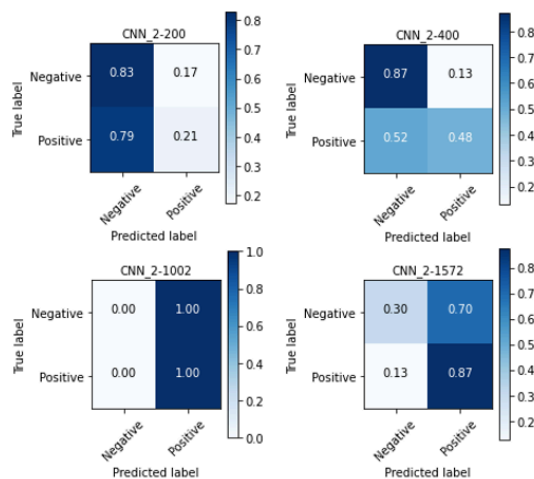
Source: The author (2022).

Figure A 41: Confusion matrices of the results obtained by CNN_1 for the different dataset sizes.



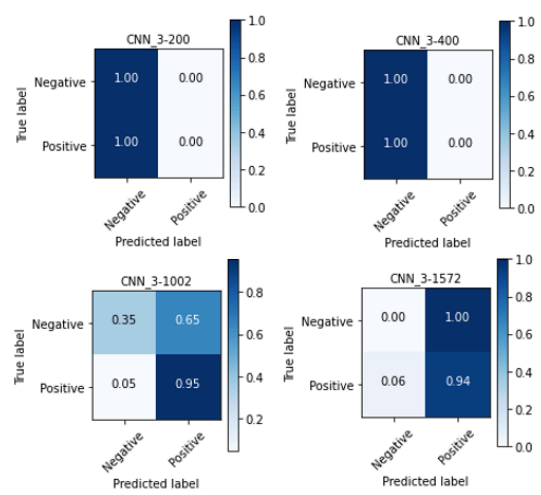
Source: The author (2022).

Figure A 42: Confusion matrices of the results obtained by CNN_2 for the different dataset sizes.



Source: The source (2022).

Figure A 43: Confusion matrices of the results obtained by CNN_3 for the different dataset sizes



Source: The author (2022).