



DAVID FIGUEIRA JOSÉ

Licenciado em Engenharia Eletrotécnica

INTELIGÊNCIA ARTIFICIAL NA DETEÇÃO DE DEFEITOS EM PAINÉIS FOTOVOLTAICOS

MESTRADO EM ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES

Universidade NOVA de Lisboa

Draft: 21 de dezembro de 2024

INTELIGÊNCIA ARTIFICIAL NA DETEÇÃO DE DEFEITOS EM PAINÉIS FOTOVOLTAICOS

DAVID FIGUEIRA JOSÉ

Licenciado em Engenharia Eletrotécnica

Orientador: João Francisco Alves Martins
Full Professor, Universidade NOVA de Lisboa

Coorientador: Fernando Manuel Tim Tim Janeiro
Associate Professor, Universidade de Évora

RESUMO

Nos últimos anos tem-se verificado um crescimento exponencial de produção energética através de sistemas fotovoltaicos. Porém, a eficiência destes é bastante vulnerável às condições a que são sujeitos (em especial, às condições climáticas), podendo, por isso, ocorrer diversos defeitos, colocando assim a produção real inferior à esperada. Considerando uma manutenção dos sistemas fotovoltaicos (seja pela análise elétrica do sistema ou presencialmente) feita por um ou vários operadores, a identificação do defeito (tendo em conta a grande variedade possível) poderá demonstrar-se complexa e morosa. Por isso, seria bastante útil uma ferramenta automática, rápida e eficaz que detetasse e classificasse os defeitos para que o utilizador pudesse agir corretamente.

Nesse sentido, o presente trabalho pretende desenvolver uma ferramenta que incorpore vários métodos de deteção e classificação de defeitos. O utilizador, introduzindo os dados do sistema, poderá obter a resposta de diversos métodos sobre o estado do sistema, podendo assim compará-los. Os métodos desenvolvidos pretendem seguir as tecnologias com melhores resultados nesta área (que vão ser explorados neste estudo), associando as análises visuais e elétricas do sistema ao *Deep Learning*. Foram construídos dois métodos elétricos distintos, que, recebendo dados como corrente, tensão, irradiância e temperatura das células, detetam e classificam os defeitos do sistema e, com o mesmo fim, três métodos visuais/térmicos que têm como entrada imagens fotográficas, infravermelhas e eletroluminescentes. A ferramenta que incorpora estes métodos consiste numa *Web App* desenvolvida em *Python* que receberá os dados introduzidos pelo utilizador, fará um processamento e através dos modelos de redes neuronais desenvolvidos (correspondentes aos cinco métodos já indicados) irá classificar o estado de funcionamento do sistema fotovoltaico. Esta ferramenta foi concebida com o objetivo de ser intuitiva, prática ao utilizador e abrangente na identificação dos defeitos.

Com a longa pesquisa para a recolha dos melhores estudos nesta área, e posteriormente na criação da aplicação *Web*, foi constatado que este trabalho apresenta algumas contribuições originais: com conceção de uma ferramenta virtual que incorpore vários métodos de deteção e classificação (elétricos e visuais), o que não foi encontrado paralelo; e com a construção de modelos de redes neuronais com precisões superiores aos melhores

trabalhos estudados no Estado da Arte.

Palavras-chave: detecção de defeitos, inteligência artificial, *deep learning*, sistemas fotovoltaicos, curvas I-V

ABSTRACT

In recent years, there has been an exponential increase in energy production through photovoltaic systems. However, their efficiency is quite vulnerable to the conditions to which they are subjected (especially weather conditions), which can lead to various faults, resulting in actual production being lower than expected. Considering maintenance of photovoltaic systems (whether through electrical analysis of the system or in person) carried out by one or more operators, identifying the defect (taking into account the wide variety possible) may prove to be complex and time-consuming. Therefore, an automatic, fast, and effective tool that detects and classifies faults would be extremely useful, allowing the user to take appropriate action.

In this context, the present work aims to develop a tool that incorporates various methods for detecting and classifying faults. By entering system data, the user can obtain the response of different methods regarding the system's status, allowing for comparison. The methods developed intend to follow the technologies with the best results in this area (which will be explored in this study), associating visual and electrical analyses of the system with Deep Learning. Two distinct electrical methods have been built, which, by receiving data such as current, voltage, irradiance, and cell temperature, detect and classify system faults. Additionally, three visual/thermal methods have been developed that use photographic, infrared, and electroluminescent images as input for the same purpose. The tool that incorporates these methods is a Web App developed in Python, which will receive the data entered by the user, process it, and, through the developed neural network models (corresponding to the five aforementioned methods), classify the operational status of the photovoltaic system. This tool has been designed to be intuitive, user-friendly, and comprehensive in fault identification.

With the long research to collect the best studies in this area, and later in the creation of the Web application, it was found that this work presents some original contributions: with the design of a virtual tool that incorporates several detection and classification methods (electrical and visual), which was not found in the bibliography; and with the construction of neural network models with higher precision than the best works studied in the State of the Art.

Keywords: fault detection, artificial intelligence, deep learning, photovoltaic systems, I-V curves

ÍNDICE

Índice de Figuras	vii
Siglas	ix
1 Introdução	1
2 Estado da Arte	5
2.1 Métodos Visuais e Térmicos	5
2.1.1 Termografia de Infravermelhos	5
2.1.2 Imagens por Eletroluminescência	6
2.2 Métodos Elétricos	6
2.2.1 Medição das Diferenças Diretas	7
2.2.2 Análise das Curvas I-V	7
2.2.3 Análise de Perdas	9
2.2.4 Métodos de Injeção Externa	10
2.2.5 Métodos de detecção de defeitos de terra	10
2.3 Métodos de Inteligência de Artificial	11
2.3.1 Métodos Elétricos	11
2.3.2 Métodos Visuais e Térmicos	16
3 Metodologia Proposta	21
3.1 Métodos Elétricos	21
3.1.1 Análise das Curvas I-V	21
3.1.2 Medição das Diferenças Diretas	24
3.2 Métodos Visuais	25
4 Web App	29
4.1 Utilização da Aplicação	29
4.1.1 Inspeção Visual	30
4.1.2 Análise das Curvas I-V	31

4.1.3	Medição das Diferenças Diretas	33
5	Resultados	35
5.1	Métodos Elétricos	35
5.2	Métodos Visuais	37
5.2.1	Termografia de Infravermelhos	37
5.2.2	Inspeção Visual	40
5.2.3	Imagens por Eletroluminescência	42
6	Conclusão	47
	Bibliografia	49

ÍNDICE DE FIGURAS

2.1	Defeitos refletidos na Curva I-V [3]	8
2.2	Defeitos refletidos na Curva P-V [3]	8
2.3	Parâmetros extraíveis das Curvas I-V [14]	9
2.4	Parâmetros extraíveis das Curvas P-V [14]	9
2.5	Trabalhos que utilizaram <i>Machine Learning</i> (entre 2006 e 2021) [6]	12
3.1	Sistema Fotovoltaico simulado	22
3.2	Evolução I-V do defeitos simulados	23
3.3	Camadas das Redes dos Métodos Elétricos	25
3.4	Camadas das Redes com bases pré-treinadas	26
3.5	Camadas da CNN sequencial simples	27
4.1	Página principal da aplicação	30
4.2	Página principal do método VI	30
4.3	Imagem ambígua de painéis solares com poeira	31
4.4	Resultados do método VI (<i>Web App</i>)	31
4.5	<i>Dataframe</i> e Curva IV (método IVCA, <i>Web App</i>)	32
4.6	Introdução da irradiância e temperatura (<i>Web App</i>)	32
4.7	Resultados do método IVCA (<i>Web App</i>)	32
4.8	Introdução de dados do método DDM	33
4.9	Resultados do método DDM (<i>Web App</i>)	33
5.1	Progressão do Treino do Método IVCA	36
5.2	Matriz de Confusão do Método IVCA (99% de precisão)	36
5.3	Matriz de Confusão do Método DDM (99% de precisão)	36
5.4	Tipos de Defeitos (Método IRTG)	37
5.5	IRTG. Matriz de Confusão (InceptionV3), 83% de precisão	38
5.6	IRTG. Matriz de Confusão (ResNet50), 83% de precisão	38
5.7	IRTG. Matriz de Confusão (VGG16), 81% de precisão	39
5.8	IRTG. Matriz de Confusão (CNN), 79% de precisão	39

5.9	Tipos de Defeitos (Método VI)	40
5.10	VI. Matriz de Confusão (InceptionV3), 89% de precisão	40
5.11	VI. Matriz de Confusão (ResNet50), 86% de precisão	41
5.12	VI. Matriz de Confusão (VGG16), 84% de precisão	41
5.13	VI. Matriz de Confusão (CNN), 72% de precisão	42
5.14	Tipos de Defeitos (Método EI)	43
5.15	EI. Matriz de Confusão (InceptionV3), 94% de precisão	43
5.16	EI. Matriz de Confusão (ResNet50), 92% de precisão	44
5.17	EI. Matriz de Confusão (VGG16), 91% de precisão	44
5.18	EI. Matriz de Confusão (CNN), 86% de precisão	45

SIGLAS

AGAST	<i>Adaptive and Generic Accelerated Segment Test (p. 18)</i>
CNN	<i>Convolutional Neural Network (pp. 14, 16, 18, 26, 37, 40, 42, 45)</i>
CWT	<i>Continuous Wavelet Transform (p. 14)</i>
DDM	<i>Direct Difference Measurement (pp. 7, 21, 24, 25, 33, 35, 37, 47)</i>
EBM	<i>Electrical-based Methods (pp. 5, 6, 11, 47)</i>
ECM	<i>Earth Capacitance Measurement (p. 10)</i>
EI	<i>Electroluminescence Imaging (pp. 5, 6, 11, 17, 18, 21, 28, 42, 45, 47)</i>
EIM	<i>External Injection Method (pp. 7, 10)</i>
GADF	<i>Gramian Angular Difference Field (p. 14)</i>
GBSSL	<i>Graph-based Semi-supervised Learning (p. 11)</i>
GFDI	<i>Ground Fault Detection and Interruption (pp. 10, 11)</i>
GPU	<i>Graphics Processing Unit (pp. 18, 25)</i>
HOG	<i>Histogram of Oriented Gradients (p. 18)</i>
IA	<i>Inteligência Artificial (pp. 11, 16–18, 24)</i>
ICA	<i>Independent Component Analysis (p. 17)</i>
IMD	<i>Insulation Monitoring Device (pp. 10, 11)</i>
IRTG	<i>Infrared Thermography (pp. 5, 6, 11, 16, 17, 21, 25, 27, 37, 45, 47)</i>
IVCA	<i>I-V characteristics analysis (pp. 7, 11, 21, 24, 29, 35, 47)</i>
kNN	<i>K-nearest neighbors (pp. 17, 24)</i>
LSTM	<i>Long Short-term Memory (pp. 24, 35, 47)</i>

MLT	<i>Machine Learning Techniques (p. 11)</i>
NMF	<i>Nonnegative Matrix Factorization (p. 17)</i>
NN	<i>Neural Networks (pp. 11–14, 17, 24)</i>
PCA	<i>Principal Component Analysis (pp. 13, 14, 17)</i>
PHOW	<i>Pyramid Histogram of Visual Words (p. 18)</i>
PLA	<i>Power Loss Analysis (pp. 7, 9)</i>
RBFN	<i>Radial Basis Function Network (p. 15)</i>
RCD	<i>Residual Current Monitoring Devices (pp. 10, 11)</i>
RF	<i>Random Forest (pp. 14, 18)</i>
RP	<i>Recurrence Plot (p. 14)</i>
SIFT	<i>Scale-Invariant Feature Transform (p. 18)</i>
SURF	<i>Speeded-Up Robust Features (p. 18)</i>
SVM	<i>Support Vector Machine (pp. 11, 13–15, 17, 18, 24)</i>
TDR	<i>Time Domain Reflectometry (p. 10)</i>
TSK-FRBS	<i>Takagi-Sugeno-Kahn Fuzzy Rule-based System (p. 16)</i>
VGG	<i>Visual Geometry Group (p. 18)</i>
VI	<i>Visual Inspection (pp. 5, 21, 25, 27, 30, 40, 47)</i>
VTM	<i>Visual and Thermal Methods (pp. 5, 6, 11, 47)</i>

INTRODUÇÃO

Com o mote de combater as alterações climáticas, tem-se, cada vez mais, procurado substitutos eficazes, práticos e económicos para a produção de energia a partir de energias fósseis. Esta mudança na produção enquadra-se na transição energética e pode ser feita a partir de diversas fontes de energia renovável, sendo uma das mais comuns, e promissoras, a energia fotovoltaica. A implementação desta tem sido exponencial ao longo dos anos tendo quase atingido no ano 2022 1200 GW de capacidade de produção instalada globalmente [1]. Também se prevê que em 2050, com este crescimento da produção a partir de energias renováveis, na Europa será possível implementar-se um sistema de energia 100% renovável em que a energia solar fotovoltaica corresponderá a 62% do total [2].

Por isso, será importante que os sistemas fotovoltaicos se apresentem eficientes e a sua manutenção seja rápida e eficaz. Atualmente, estes, devido à sua vulnerabilidade e dependência das condições exteriores, enfrentam vários desafios na sua proteção e manutenção. É uma tecnologia que fica exposta às condições climáticas, à natureza, à intervenção humana (construções que poderão fazer sombra, por exemplo), à sujidade, ao desgaste do tempo, etc. Estas condições afetarão indubitavelmente a produção de energia.

Ora, se se considerar um parque fotovoltaico de dimensões consideráveis que em dado momento tem uma produção inferior à esperada, o primeiro passo a tomar para colmatar este prejuízo seria identificar a causa (ou as várias causas). Mas, previsivelmente, esta tarefa será complexa tendo em conta as dimensões do parque e a vasta quantidade de possíveis causas. Com vista a combater este problema foram, ao longo dos anos, desenvolvidos diversos métodos para a identificação do tipos de defeitos.

Os defeitos do lado DC podem-se dividir em físicos, ambientais e elétricos (Tabela 1.1). Os físicos correspondem aos danos internos das células fotovoltaicas (painel degradado ou defeitos nos conectores). Já a acumulação de pó, a sujidade, o sombreamento e a existência de *hotspots* nos painéis são considerados defeitos ambientais. E, por fim, todos os defeitos relacionados com curtos-circuitos, falhas nos díodos de *bypass*, circuitos abertos e defeitos linha-linha, em arco ou com a terra, são identificados como elétricos [3].

Existem vários métodos e tecnologias para a deteção destes (e outros) defeitos, sendo que nenhum método (individualmente) tem a capacidade de detetar todos os tipos de

Tipos de Defeitos		
Físicos	Internos	defeito no encapsulamento
	Externos	conectores
		vidros quebrados
		célula rachada
		degradação
Ambientais	Temporárias	acumulação de poeira
	Permanentes	sujidade
		pontos quentes
Elétricos	Internos	díodos de <i>bypass</i>
	Externos	circuito aberto
		curto circuito
		defeito de arco
		defeito de terra

Tabela 1.1: Defeitos dos Módulos Fotovoltaicos [3]

defeitos. Neste trabalho serão apresentados e analisados diversos métodos de detecção, localização e classificação destes defeitos, contrapondo os seus prós e contras (a abrangência na detecção dos defeitos, a precisão, a facilidade de implementação, etc.) e tendo em conta a progressão nesta área nos últimos anos.

Atualmente já existe uma bibliografia vasta acerca deste tema e nos últimos anos tem sido notável a introdução de técnicas de *Machine Learning* para a detecção e identificação de defeitos. Estas técnicas permitem, recebendo grandes quantidades de dados, a detecção dos padrões dos defeitos em tempo real, aumentando a precisão e eficácia quando conjugadas com os tradicionais métodos de detecção. Ao analisar continuamente os dados provenientes de sensores, câmaras térmicas e drones, os modelos de *Machine Learning* conseguem identificar anomalias e padrões subtis que podem passar despercebidos por métodos convencionais. Isso resulta em diagnósticos mais rápidos e precisos, permitindo a intervenção imediata e a manutenção preditiva. A inclusão destas técnicas tem sido bastante profícua, como é demonstrado em [4] [5] [6], e não só será estudada no próximo

capítulo como terá um papel essencial na Metodologia Proposta deste trabalho. Também é relevante referir que estas técnicas permitem uma redução dos custos de manutenção, sendo que proporcionam a deteção dos defeitos em tempo real (o que permite uma reação imediata) e de uma forma preventiva (precavendo danos agravados no futuro), para além de que a implementação destas tecnologias tem-se demonstrado prática e de baixo custo.

No próximo capítulo (Estado da Arte) serão apresentados vários trabalhos que representam a diversidade de métodos utilizados para a deteção e identificação dos defeitos dos sistemas fotovoltaicos, sendo divididos em dois grandes grupos, consoante o tipo de dados que utilizam: métodos elétricos e métodos visuais/térmicos. Os primeiros detetam os defeitos com base em dados elétricos (tais como tensão e corrente) podendo também incluir dados como irradiância e temperatura das células, todos estes colhidos a partir de sensores. Os segundos (métodos visuais/térmicos) utilizam imagens fotográficas, infravermelhas ou até eletroluminescentes, captadas por câmaras, drones ou gruas, para detetar os defeitos. Também nesse capítulo serão analisados trabalhos que conjugam estes métodos com técnicas de *Machine Learning*, o que terá um foco especial, sendo que são trabalhos que poderão servir de modelos para a Metodologia Proposta.

No terceiro capítulo deste estudo (Metodologia Proposta), será apresentada a solução ponderada, a partir dos exemplos estudados, para uma ferramenta de deteção e identificação de defeitos de sistemas fotovoltaicos. Esta incorporará diversos métodos, todos baseados em *Deep Learning*, será concebida como *Web App*, permitirá ao utilizador detetar uma vasta gama de defeitos através da introdução de vários tipos de dados e comparar os resultados de cada, sendo que três dos métodos que terá ao seu dispor serão visuais/térmicos e dois elétricos. Por fim, serão apresentados os resultados dos métodos desenvolvidos, o funcionamento da *Web App* e as conclusões do trabalho.

Este trabalho teve como contribuições originais: a construção de vários métodos de deteção e identificação de defeitos em sistemas fotovoltaicos e a incorporação destes numa aplicação virtual de acesso prático e rápido ao operador. Os métodos implementados procuraram superar, ou, pelo menos, igualar os resultados dos melhores trabalhos estudados, o que foi francamente atingido, e a incorporação destes numa única ferramenta virtual é uma contribuição original (visto não se ter encontrado nenhum trabalho que o realizasse).

ESTADO DA ARTE

Os métodos para detecção e identificação dos defeitos podem dividir-se em duas categorias: métodos elétricos, *Electrical-based Methods* (EBM), e métodos visuais e térmicos, *Visual and Thermal Methods* (VTM). Estes últimos normalmente são utilizados para identificar e localizar defeitos físicos (danos nos painéis, degradação, células queimadas, ...) e até sujidade na superfície. Podem até detetar defeitos nos cabos, díodos, junções e conectores. Já os EBM utilizam uma análise das características elétricas para identificar defeitos nos módulos, *strings*, *arrays* (no lado DC) e até após o inversor (lado AC). Estes últimos são considerados, pelos estudos com maior relevância, os métodos que fornecem maior abrangência na detecção e classificação de defeitos [4].

Neste capítulo serão primeiramente apresentados os métodos visuais/térmicos, de seguida os elétricos e, por fim, métodos de *Machine Learning* baseados em dados visuais, térmicos e elétricos, ou seja, a conjugação entre os métodos anteriores e a Inteligência Artificial. Estes últimos têm recebido bastante destaque nos últimos anos, dado a sua eficácia, e merecerão maior atenção neste capítulo em relação aos restantes métodos.

2.1 Métodos Visuais e Térmicos

Os VTM incluem métodos que utilizam termografia de infravermelhos, *Infrared Thermography* (IRTG), imagens por eletroluminescência, *Electroluminescence Imaging* (EI), inspeção visual, *Visual Inspection* (VI), fluorescência UV (*FL Imaging*), entre outros [4] [7]. Serão somente os dois primeiros métodos analisados neste trabalho, visto que são os mais comumente referidos e os com melhores resultados. Este trabalho, posteriormente, integrou-os na construção de redes neuronais.

2.1.1 Termografia de Infravermelhos

O IRTG é uma tecnologia que utiliza as ondas eletromagnéticas produzidas pelo defeito presente no sistema fotovoltaico para o identificar e localizar. Se houver um curto-circuito, danos na cablagem, nos díodos, nas junções, nos conectores, etc. será produzido, devido ao efeito de Joule, no local do defeito uma zona de maior dissipação de calor. Com isto

será possível detetar e localizar o defeito numa imagem termográfica como é demonstrado em [8]. Em [9] é explorada esta tecnologia, analisando as formas de utilização e identificando os defeitos localizados por esta tecnologia. São apresentados três tipos de utilização: com o próprio utilizador a capturar as imagens; com uma grua; e com um drone. De seguida, são descritos os defeitos passíveis de serem detetados: através das imagens é possível verificar *hotspots* e a ativação dos díodos de *bypass*, que poderão ter origem em sombreamentos parciais; degradação; sujidade; células danificadas; e *strings* e/ou painéis inteiros desconectados, o que poderá indiciar falhas nos conectores, erros na instalação ou a ativação dos fusíveis. Relativamente a estes últimos defeitos, a captura de imagens também pode ser feita aos próprios cabos, conectores, junções e fusíveis de modo a identificar o defeito e quais o terão.

Este último método, sendo passivo, também tem uma versão ativa. É possível através de fontes de calor ou até de vibrações mecânicas induzir a revelação dos defeitos no sistema. São descritos exemplos em [10], em que através do aquecimento são produzidas ondas térmicas no sistema fotovoltaico, que tornarão evidentes os defeitos dos painéis. Contudo, avaliando a exequibilidade (e complexidade) e os benefícios da sua implementação torna-se preferível a utilização da versão passiva.

2.1.2 Imagens por Eletroluminescência

Com a emissão de luz, consequência da reação à passagem de corrente injetada (eletroluminescência), é possível aplicar o EI [11]. Em [12] foi aplicado este método, sendo utilizado um drone para captar as imagens e um pós-processamento através de algoritmos implementados em *Matlab*. Este estudo teve como principal foco a captura e o tratamento das imagens conseguindo imagens de boa qualidade em várias condições de irradiância e temperatura. Esta tecnologia tem sido amplamente conjugada com inteligência artificial, obtendo resultados bastante satisfatórios, como mais à frente será descrito (no capítulo sobre a integração de inteligência artificial).

Por fim, comparando EI com o anterior (IRTG), em [13] são abordados os prós e contras de ambas as tecnologias. Ambas têm em comum as vantagens de não serem intrusivas (funcionam como tecnologia externa ao sistema fotovoltaico) e de serem abrangentes na deteção de defeitos, sendo que a IRTG é apontada como ideal na deteção de *hotspots* e a EI na de rachadura à escala micro. Contudo, o método IRTG demonstrou-se limitado em altas temperaturas e o EI com alguma dependência das condições externas na obtenção de imagens de qualidade.

2.2 Métodos Elétricos

Os EBM são os mais tipicamente usados, mais abrangentes (detetam e classificam um maior número de defeitos comparativamente com os VTM) e os mais estudados, inclusive merecerão muito mais atenção neste estudo. Podem ser resumidos por diferentes

métodos [14] [5]: medição das diferenças diretas, *Direct Difference Measurement (DDM)*; análise das curvas I-V, *I-V characteristics analysis (IVCA)*; análise de perdas, *Power Loss Analysis (PLA)*; métodos de injeção externa, *External Injection Method (EIM)*; e métodos de detecção de defeitos de terra.

2.2.1 Medição das Diferenças Diretas

DDM é o método que utiliza a comparação em tempo real das correntes e tensões em vários pontos do sistemas fotovoltaicos para detetar os defeitos. Sendo um sistema fotovoltaico constituído por vários ramos (*strings*), este método permite a identificação de defeitos através da comparação das medidas elétricas destes ramos (comparando as correntes dos ramos, por exemplo). Um exemplo deste método é o estudo [15], em que são medidas as correntes e as tensões dos ramos do sistema fotovoltaico (sendo que a tensão medida corresponde a uma porção do ramo) e assim, comparando estas medições, é possível detetar e localizar o defeito. É explicado que através da análise das diversas correntes será possível identificar o ramo (ou os ramos) que contem (ou contêm) o defeito e, de seguida, com a comparação das tensões localizar-se-á mais especificamente. Este método permite a classificação de circuitos abertos, curto-circuitos e *hotspots*, mesmo em condições de sombreamento parcial. Métodos semelhantes a este podem ser encontrados em vários outros estudos, como por exemplo em [16]. Neste foram simuladas 3 *strings*, cada uma com inversor, de 8 módulos em diversas situações de sombreamento, curto-circuito e circuito aberto e comparadas as medições de tensão e corrente. Demonstrou-se a possibilidade de detetar, localizar e classificar defeitos como: curtos-circuitos e circuitos abertos em várias condições de irradiância, seja em irradiância uniforme ou não-uniforme (em diversas situações de sombreamento parcial).

Por fim, em [17] foram analisadas e interpretadas as curvas I-V e P-V, permitindo a identificação de vários tipos de defeitos. Na conclusão deste estudo é afirmado que o próximo passo será a implementação de um algoritmo inteligente que, através das curvas I-V e P-V, detete e localize os defeitos. Esta previsão revelou-se acertada, visto que, para além de atualmente já existir ampla bibliografia nesse sentido, este próprio trabalho pretende explorar uma solução nessa área. Precisamente, neste estudo, serão analisados vários trabalhos que incorporam diversos métodos baseados em inteligência artificial.

2.2.2 Análise das Curvas I-V

A deteção de defeitos através da análise das curvas I-V e P-V, por sua vez, pode-se considerar um método à parte (**IVCA**), dada a larga extensão de bibliografia relacionada. Neste estudo convirá distinguir este método dos demais dado que os seus fundamentos serão utilizados na solução escolhida. Através deste método tem sido possível a deteção e classificação da maior parte dos defeitos, e a sua potencialidade é desde logo notada pelo reflexo que os defeitos têm nas curvas I-V e P-V (Figuras 2.1 e 2.2). A maior parte dos defeitos são facilmente distinguíveis através da análise das curvas, o que é demonstrado, por exemplo,

em [18]. Neste trabalho são simulados em *Matlab* vários tipos de curto-circuito (no díodo de bypass, na célula ou no módulo), diversos graus de sombreamento (conjugado com outros defeitos) e vários tipos de degradação. Estas curvas poderão ser utilizadas na sua totalidade (todos os dados) ou poderão ser extraídos certos valores representativos das suas evoluções (presentes nas Figuras 2.3 e 2.4), que geralmente são: a corrente e a tensão no ponto de potência máxima (I_{mp} e V_{mp} , respetivamente); corrente curto-circuito (I_{sh}); e tensão em circuito aberto (V_{oc}). Com estes também será possível estabelecer-se um padrão e identificar o estado do sistema fotovoltaico (se se encontra em funcionamento normal, ou com qualquer tipo de defeito).

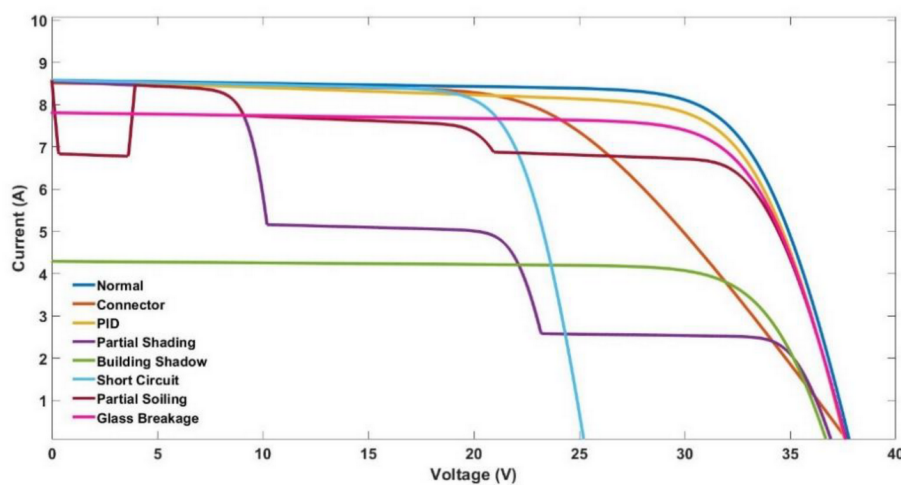


Figura 2.1: Defeitos refletidos na Curva I-V [3]

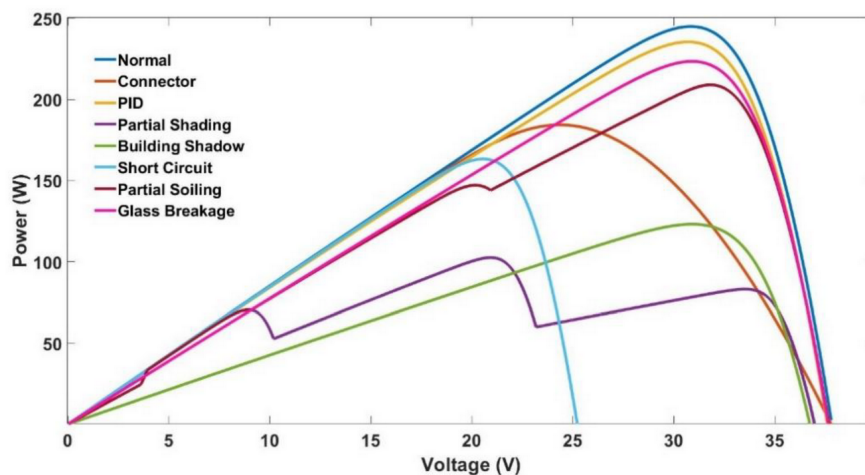


Figura 2.2: Defeitos refletidos na Curva P-V [3]

Contudo, criar um sistema automático que prontamente identifique esta extensa diversidade de defeitos (representados por tantos tipos de curvas I-V e P-V) é uma tarefa desafiante e árdua se se não tiver como auxílio algum tipo de inteligência artificial. E por isso mesmo, neste e noutros métodos, esta tecnologia tem sido abundantemente utilizada,

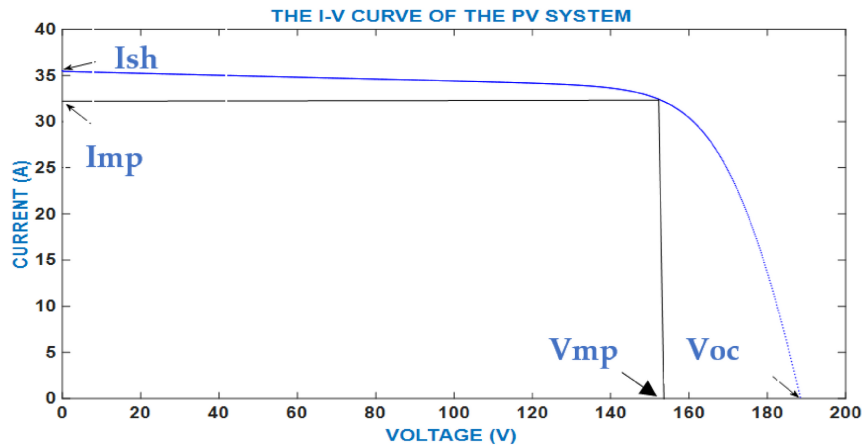


Figura 2.3: Parâmetros extraíveis das Curvas I-V [14]

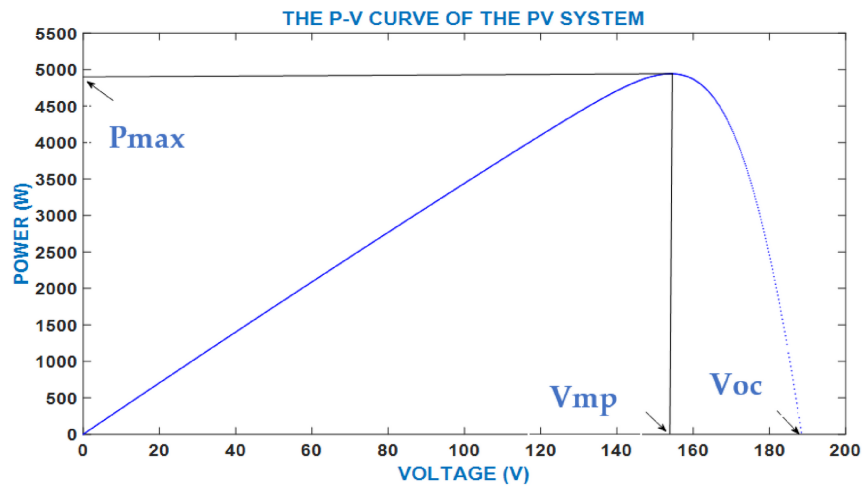


Figura 2.4: Parâmetros extraíveis das Curvas P-V [14]

originando vasta bibliografia. A junção entre análise das curvas I-V e P-V e a inteligência artificial, na detecção e classificação dos defeitos fotovoltaicos, é um dos temas centrais deste trabalho e será afluído mais à frente.

2.2.3 Análise de Perdas

Outro método usado para combater esta problemática é o [PLA](#). Este, como o nome indica, utiliza a análise das perdas, causadas pelos defeitos, para os detetar. Um exemplo claro deste método é dado por [19], que calculando a potência máxima que o sistema fotovoltaico pode dar e comparando-a com o que é medido identifica o defeito. Neste trabalho é usado o ambiente *Matlab* na simulação do sistema com e sem defeitos. Para calcular a evolução (curva) da potência esperada são consideradas a irradiância e a temperatura ambiente em tempo real de 3 dias e, aplicando falhas noutros sistemas paralelos, são simuladas as potências reais. Pela análise entre as potências esperada e medida serão detetados os defeitos.

Um exemplo mais elaborado deste método é descrito em [20]. Neste caso além de

detetados os defeitos através da comparação entre potência esperada e obtida, também são identificados os tipos de defeitos. É utilizado *Matlab* como ferramenta de simulação e diagnóstico e são testados os vários tipos de falhas experimentalmente (com um sistema fotovoltaico). Os algoritmos desenvolvidos são de três tipos: um que detetará o defeito e o classificará como contínuo ou temporário e outros dois que especificarão o tipo de defeito (se for contínuo será possível sujidade generalizada ou localizada, *hotspot* ou degradação, em caso de defeito temporário será classificado como um determinado tipo de sombreamento).

2.2.4 Métodos de Injeção Externa

Em [21] é utilizado um EIM. Neste é utilizado a tecnologia *Time Domain Reflectometry* (TDR) para detetar, localizar e classificar o defeito. Esta consiste na injeção de um sinal na linha de transmissão (neste caso será uma linha que atravessa os módulos fotovoltaicos) e a comparação com o sinal refletido (resultante da passagem pelo circuito). A análise da forma de onda do sinal de resposta detetará o defeito e denunciá-lo o seu tipo e a sua localização. Porém, neste estudo, só são identificados curtos-circuitos e circuitos abertos e é limitado a ambientes não iluminados (*indoor*). Estas limitações foram identificadas e, passados dois anos após a publicação deste estudo, foi, pelos mesmos autores, melhorado o projeto. Desta forma foi tornado possível a identificação de defeitos de desconexão e degradação nos módulos e foi testado o método num ambiente *outdoor* [22], testando o método em condições reais. Contudo, esta tecnologia é bastante influenciável pelas condições da instalação fotovoltaica (a cablagem, os componentes e a montagem dos módulos), o que requer uma aferição das condições iniciais, através de TDR, logo após a implementação do sistema fotovoltaico.

Outro exemplo de EIM é o *Earth Capacitance Measurement* (ECM). Este consiste na medição da capacidade entre o sistema e a terra. É aplicada uma tensão alternada no sistema e com a medição da corrente que flui para a terra é calculada a capacidade. Esta tecnologia foi testada e comparada com a anterior (TDR) em [23]. Segundo este estudo, a utilização de ECM é mais adequada para a deteção de desconexões entre módulos, enquanto que TDR para a deteção e localização de degradação. Ambas têm a vantagem de não serem influenciadas pela variação da irradiância.

2.2.5 Métodos de deteção de defeitos de terra

Existem também métodos desenvolvidos para detetar os defeitos de terra (entre os condutores e a terra). Estes são especializados na deteção de falhas específicas e por isso conseguem atingir um bom nível de eficácia. Estes são: *Ground Fault Detection and Interruption* (GFDI); *Residual Current Monitoring Devices* (RCD); e *Insulation Monitoring Device* (IMD) [24]. Tendo o sistema um circuito por onde circule a corrente de terra é necessário que, se esta exceder certos limites, o sistema reaja identificando o defeito e isolando o *string* correspondente. Este corte é feito por um fusível dimensionado especificamente

para estes contextos. Este método é o [GFDI](#). Contudo, o dimensionamento deste fusível pode tornar-se complexo devido à humidade do local, resistência do condutor de terra, possíveis interferências magnéticas e temperatura ambiente. Todos estes fatores poderão influenciar a corrente e, por isso, queimar o fusível sem a existência de um defeito. Outro método de deteção de defeitos de terra é o [RCD](#). Este tem um funcionamento semelhante a um diferencial, ou seja, identificará o defeito se num dado *array* a corrente de entrada for diferente da de saída (o que significa que também poderá detetar defeitos linha-linha, entre condutores). Por fim, se não existir, ou não estiver em funcionamento, um sistema de terra, existe um método para identificar o defeito: [IMD](#). Este mede a resistência entre os condutores e a terra e alerta o sistema se esta diminuir até certos valores. Contudo, à semelhança do [GFDI](#), este também é suscetível às condições externas. Para combater esta influência foi considerada uma variável que, variando em função das condições externas, ajusta o valor da resistência máxima admissível em [24].

2.3 Métodos de Inteligência de Artificial

Por fim, é necessário analisar o conjunto de métodos mais importante para este trabalho, o mais promissor e prolífero dos últimos anos: *Machine Learning Techniques* (MLT) e em especial uma subcategoria deste, *Deep Learning*. Estes baseiam-se, muito resumidamente, no desenvolvimento de algoritmos treinados para identificar padrões entre as entradas do sistema, *inputs*, (por exemplo, correntes e tensões dos módulos) e saídas (tipos de defeitos). Também é possível utilizar esta tecnologia para pré-processamento dos conjuntos de dados, extraindo destes os seus parâmetros característicos, de modo a melhorar a eficácia e velocidade no treino dos algoritmos. A esta função dá-se o nome de extração de características (*features*).

Neste capítulo serão apresentados vários trabalhos que combinam estas técnicas com os métodos apresentados anteriormente (visuais, térmicos e elétricos). Primeiro serão descritos vários [EBM](#) (em especial a análise das curvas I-V, [IVCA](#)) e, de seguida, quatro trabalhos que utilizam [VTM](#) (2 de [IRTG](#) e 2 de [EI](#)), sendo que todos estes conjugam os seus métodos com técnicas de [Inteligência Artificial](#) (IA).

2.3.1 Métodos Elétricos

Nesta secção é tido em conta o estudo disponível em [6], onde é apresentada a implementação de estudos que conjugam *Machine Learning* e os métodos elétricos de deteção e classificação de defeitos, entre 2006 a 2021. Neste é declarado que os métodos de *Machine Learning* utilizados para deteção e diagnóstico podem ser divididos em: Redes Neurais; *Wavelet-based*; *Fuzzy-based*; *Decision Tree*; *Support Vector Machine* (SVM); *Graph-based Semi-supervised Learning* (GBSSL); *Regression-based*; e outros (Figura 2.5). Contudo serão somente analisadas as tecnologias com maior predominância (*Neural Networks* (NN), *Wavelet-based*, [SVM](#) e *Fuzzy-based*), com especial foco nos trabalhos de redes neurais, visto que não

só são os mais utilizados como também servem como exemplo na Metodologia Proposta deste presente trabalho.

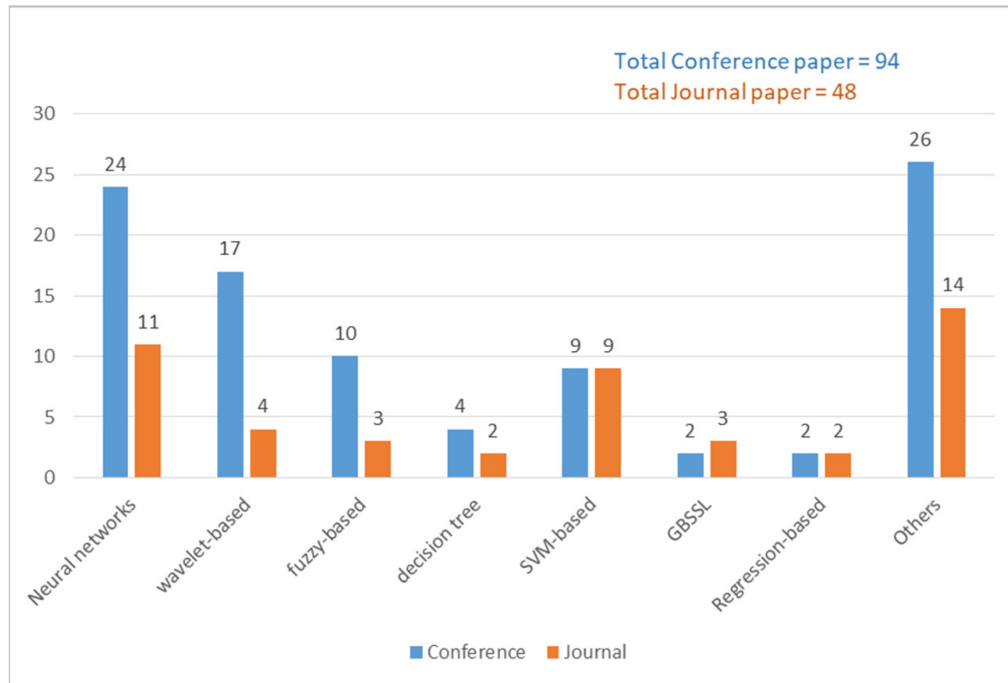


Figura 2.5: Trabalhos que utilizaram *Machine Learning* (entre 2006 e 2021) [6]

2.3.1.1 Redes Neurais

Os métodos baseados em **NN**, também referidos como Redes Neurais Artificiais, são os mais encontrados na literatura e demonstram também grande eficácia, o que também motivou a sua utilização na Metodologia Proposta. Estes, como o nome indica, foram inspirados na estrutura do sistema nervoso animal, e permitem, com bastante profundidade, a captação e memorização de padrões dos mais diversos conjuntos de dados. Estas redes neurais são constituídas por unidades de processamento interconectadas (chamadas de neurónios) organizadas por 3 tipos de camadas: camada de entrada (*Input Layer*); camadas ocultas (*Hidden Layers*); e camada de saída (*Output Layer*). A primeira tem como função receber os dados de entrada e a última apresentar o resultado de todo o processo que, no caso da deteção de defeitos, seria a classificação do estado do sistema fotovoltaico. Já as camadas ocultas são responsáveis pela maior parte do processamento e extração de informação, permitindo à rede a captação de padrões. Para formar uma rede neuronal é necessário, antes de mais, dividir a base de dados em dois conjuntos: de treino e teste/validação (para que o modelo seja testado com dados que ainda não conhece). O treino da rede neuronal é, muito sucintamente, constituída por duas fases principais: propagação direta (*Forward Propagation*) e propagação inversa (*Back Propagation*). Na primeira os dados são transmitidos desde a camada de entrada até à de saída, os neurónios recebem a informação dos anteriores, processam-na e transmitem aos seguintes. No final

desta sequência, na camada de saída, a rede gera a classificação (do estado do sistema fotovoltaico, neste caso) e a diferença entre o estado previsto e o estado real é usada para reajustar as conexões entre neurónios retrospectivamente, ou seja, há uma propagação inversa (da camada de saída para a camada de entrada). Desta forma, a rede vai reajustando os seus parâmetros ao longo das iterações, durante o treino.

Em [25] é utilizado uma abordagem em **NN** para identificação de 8 tipos de defeitos, sendo os dados obtidos experimentalmente. Também foram simuladas várias situações (de defeito e em operação normal) em *Matlab* para validação dos resultados práticos. As entradas utilizadas foram as curvas I-V, a irradiância e a temperatura do módulo, ou seja, tendo esta rede treinada será possível, com base nestas entradas, detetar e identificar os tipos de defeitos.

Também em [26] tem-se um exemplo semelhante, mas com resultados práticos de um ano de implementação: foram detetados sombreamentos; degradações; problemas no inversor; fusíveis queimados; entre outras, com uma precisão bastante alta (a precisão mais baixa é a do sombreamento e é de 87,9%). Este estudo mostrou que, com a implementação desta tecnologia, a operação e manutenção do sistema fotovoltaico foi otimizada, aumentando em 4,7% o seu rendimento energético diário (após um ano de implementação).

Outro exemplo com resultados bastante positivos (precisões acima de 90%) é [27], sendo que o conjunto de defeitos detetados abrange diversas formas de sombreamento, curto-circuitos nas células e falhas nos díodos, perfazendo no total 9 tipos de defeitos. Neste trabalho também é comparado o desempenho das **NN** com o de outro algoritmo (mais rudimentar), confirmando a superioridade do primeiro. O conjunto de dados é extenso e provém das simulações em *Matlab*.

Por fim, os estudos com os melhores resultados são [14] e [28]. O primeiro [14] demonstra uma performance de 99,9% na deteção de 8 estados (sem defeito, sombreamento parcial, circuito aberto, degradação, defeito linha-linha, de ponte, no díodo de bypass e a combinação entre linha-linha e degradação). Neste trabalho são simulados 4560 amostras em *Matlab Simulink* utilizando irradiâncias e temperaturas reais, são desenhadas as curvas I-V e P-V e, a partir destas, extraídas as entradas para treinar a **NN**. Constituem-se 3 cenários: no primeiro as entradas são a potência máxima, a tensão em aberto e a corrente de curto-circuito; no segundo acrescenta-se a tensão e corrente no ponto ótimo; e no terceiro cenário soma-se, relativamente ao segundo, a temperatura e irradiância. Estes obtiveram 82,2%, 94,7% e 99,9% de precisão, respetivamente, o que permite provar a importância da inclusão das entradas dos segundo e terceiro cenários.

Já o segundo [28] consegue atingir uma precisão de 100% utilizando **NN**. Neste trabalho foram usados **NN** e **SVM** como algoritmos de classificação de estados (operação normal, 2 tipos de sombreamento parcial, de degradação e de curto-circuito e 1 tipo de circuito aberto) com várias técnicas de extração de características e utilizando *Principal Component Analysis (PCA)*, em simulação e experimentalmente. Foram consideradas 4 combinações de classificador e tipo de extração de características: **SVM** com 8 parâmetros (irradiância,

temperatura ambiente, corrente e tensão em potência máxima, corrente de curto-circuito, tensão em aberto e resistências shunt e série); **NN** com curvas I-V (diretamente); **NN** com *Recurrence Plot* (RP); e **NN** com *Gramian Angular Difference Field* (GADF). Em simulação, os resultados foram, respetivamente, 94,8%, 99,92%, 99,96% e 100% de precisão. De seguida, foi comprovada a importância da utilização do **PCA**, comparando os resultados com e sem este e verificando um aumento na precisão e uma melhoria substancial no tempo de treino e pré-processamento. Por fim, estes métodos foram aplicados experimentalmente comprovando, e até superando, os resultados simulados.

Outro tipo de **NN** utilizado nestes trabalhos é *Convolutional Neural Network* (**CNN**) (*Convolutional Neural Network*). Este é especialmente dedicado à visão computacional sendo que permite a extração de características espaciais e processamento de imagens e, por isso, a sua utilização é muito mais frequente nos métodos visuais/térmicos, visto que os dados destes são imagens. No entanto, no contexto da deteção de defeitos a partir de dados elétricos pode revelar-se bastante útil, principalmente, na extração de características. Se estes dados puderem ser traduzidos em escalogramas 2D, como em [29], ou graficamente, como em [30], as **CNN** poderão obter resultados positivos.

Em [29] são recolhidos os dados elétricos (corrente, tensão e potência no ponto máximo, corrente de curto-circuito, tensão em aberto, temperatura e irradiância) simulados em *Matlab Simulink* (3456 amostras), representando 6 estados (sem defeito, sombreamento parcial e defeito durante sombreamento, defeito linha-linha, circuito aberto e defeito de arco). De seguida, com este conjunto de dados 1D, e através de *Continuous Wavelet Transform* (**CWT**), são produzidos escalogramas 2D. Desta forma é possível utilizar uma **CNN** de duas dimensões, neste caso a pré-treinada *AlexNet*, para extrair as características (determinar as entradas mais influentes). A este conjunto de entradas foram aplicadas várias técnicas de classificação em *Deep Learning* que vão desde **SVM** ou *Random Forest* (RF) até à própria pré-treinada *AlexNet*, atingindo precisões de 73%. Para provar a melhoria de eficácia trazida pela extração de características (através da **CNN** 2D) foram também testados os métodos mais comuns (usando o conjunto de dados 1D) atingindo uma precisão máxima de cerca de 70%. Esta melhoria deve-se ao processamento executado pela **CNN** 2D que atribui mais peso aos dados relevantes e menos aos irrelevantes (extraíndo as características convenientes). Este método poderá demonstrar-se bastante útil em cenários de elevado ruído.

Em [30] também é utilizada uma **CNN**, mas, neste caso, para detetar e identificar distúrbios energéticos como cavas, transitórios e interrupções. Neste estudo é, primeiramente, desenvolvida esta tecnologia para classificar 4 classes (funcionamento normal, aparecimento de uma cava, transitório ou interrupção), obtendo uma precisão de 99,83%, e de seguida incluindo a distinção entre início e fim de cava ou interrupção (perfazendo assim 8 classes). Apesar desta inclusão a precisão não sofreu grandes perturbações, descendo para os 94,54%. Também é relevante notar que estes resultados foram obtidos a partir de conjuntos de dados (reais) com muito mais amostras representativas do funcionamento

normal do que do funcionamento com defeitos (representando assim melhor as condições reais, visto que o sistema normalmente se encontra sem defeitos). Esta influência é estudada neste trabalho comparando resultados com conjuntos de dados balanceados (ou seja, com um número de amostras igual para cada classe).

2.3.1.2 *Wavelet-based*

Assim como o último exemplo, este método baseado em *wavelet* é utilizado na extração de características com uma elevada taxa de sucesso. Em [31] é simulado um sistema fotovoltaico em *Matlab Simulink*, produzindo cerca de 7200 amostras representativos de 14 estados (13 tipos de defeitos mais o estado de normal operação). Após a recolha deste conjunto de dados, é utilizada uma abordagem *wavelet-based* para extrair as características e de seguida é feita a classificação usando uma *Radial Basis Function Network (RBFN)*. Este trabalho obteve uma precisão de 97% na identificação dos defeitos, o que é um resultado excelente tendo em conta a extensa lista de tipos de defeitos.

Outro exemplo da utilização do método *wavelet-based* é [32], em que são comparados três métodos na deteção e classificação: dois de *wavelet-based* e um de NN. Foram recolhidos dados experimentais e, a partir destes, extraídos 500 amostras, sendo representados 5 estados (normal, curto-circuito, circuito aberto, degradação e sombreamento parcial) cada um com 80 amostras. De cada amostra foram extraídas as características consideradas relevantes (corrente e tensão em ponto de máxima potência, corrente de curto-circuito e tensão em aberto). Por fim, foram aplicados os métodos, obtendo os seguintes resultados de precisão: 96% e 83% para os métodos *wavelet-based* e 79% para o NN.

2.3.1.3 *Support Vector Machine*

Também é possível utilizar algoritmos baseados em *SVM*, como é exemplo [33]. Neste são recolhidas 400 amostras simuladas, extraindo as características das curvas I-V (corrente de curto-circuito, tensão em aberto e corrente e tensão no ponto de máxima potência), de seguida, utilizaram-se estes dados como entradas da *SVM* que detetará e classificará os defeitos (sendo, neste caso, curto-circuito, circuito aberto e sombreamento parcial). Para otimizar os parâmetros deste algoritmo de diagnóstico foi implementado o *k-fold cross-validation*. Este trabalho atingiu uma precisão na fase de testes de 97%.

Outro exemplo da utilização desta tecnologia é [34], em que são medidas a irradiância, temperatura ambiente e corrente e tensão do *array*, e pretende-se detetar e classificar 6 estados (operação normal, defeitos linha-linha um e dois módulos, sombreamento parcial, circuito aberto numa *string* e defeito nos sensores de corrente e tensão). As entradas utilizadas são: a temperatura das células; a corrente; a tensão; e o *Gamma*, que é divisão entre a potência do *array* (calculada a partir da corrente e tensão) e a irradiância. Este estudo obteve uma precisão de deteção de 99,4% e de diagnóstico de 98%.

SVM também é amplamente usada na deteção de defeitos especificamente, por exemplo em [35], [36] e [37] para o defeito linha-linha e em [38] e [39] para sombreamento parcial.

2.3.1.4 Fuzzy-based

Por fim, também é possível aplicar algoritmos *fuzzy* para detetar e classificar defeitos nos sistemas fotovoltaicos. Assim como nos métodos anteriores, também neste caso é frequente encontrar estudos que usam algoritmos *fuzzy* para identificar defeitos específicos, por exemplo: em [40] e [41] são utilizados na deteção de defeitos de arco elétrico e em [42] e [43] na de sombreamento parcial.

Contudo, existem também trabalhos mais abrangentes na classificação de defeitos, exemplo disto é o [44]. Neste é utilizado o método *fuzzy* conhecido como *Takagi-Sugeno-Kahn Fuzzy Rule-based System (TSK-FRBS)*, em que a partir da medição da corrente e tensão à saída do inversor, da irradiância e da temperatura ambiente é calculada a potência esperada em tempo real e comparada com a potência produzida. Com a diferença entre estes dois valores será possível detetar e identificar os defeitos, ou a ausência destes, que poderão ser: células danificadas numa *string* em uma ou em zonas distintas; infiltrações de água; e sombreamento parcial. Os dados foram obtidos de fevereiro a julho obtendo precisões médias de 91,1% e 88,7% (*True Positive Rates* e *True Negative Rates*, respetivamente).

2.3.2 Métodos Visuais e Térmicos

Na deteção de defeitos de sistemas fotovoltaicos através de métodos visuais/térmicos, a sua conjugação com métodos de [IA](#) tem tendido, em larga medida, para a utilização de [CNN](#). Estas redes neuronais, adaptadas a dados 2D, conseguem atingir desempenhos elevados na deteção de padrões visuais e no processamento de dados espaciais, e, por isso, em conjugação com os métodos visuais, conseguem obter resultados bastante positivos. Devido à sua extensa utilização e aos seus resultados, não só todos os trabalhos apresentados neste capítulo utilizam [CNN](#), como também serão implementadas na Metodologia Proposta.

2.3.2.1 Termografia de Infravermelhos

Em [45] é descrito todo o processo de construção de uma rede neuronal com base em [IRTG](#), desde a recolha e tratamento da amostra até ao desenvolvimento da rede, que deteta e classifica 6 defeitos: sujidade, rachaduras, *hotpots*, riscos, danos generalizados e bordas danificadas. Primeiramente é apresentado o processo de captação das imagens infravermelhas, que é baseado em [IRTG](#) ativo (que utiliza fontes de calor para produzir radiação infravermelha), de seguida são testadas e comparadas várias redes neuronais e, por fim, após a seleção da rede com melhores resultados, é analisado o impacto da utilização de algoritmos de extração de características no processamento das imagens (antes do treino da rede). Neste estudo também são sustentadas as vantagens do método de captação de imagens escolhido (*Eddy Current Thermography*) face aos métodos tradicionais.

A comparação de resultados das várias redes neuronais teve como principal objetivo pôr em contraste as redes pré-treinadas, neste caso, GoogLeNet, VGG-16 e LeNet-5, e duas

redes tradicionais, uma [NN](#) e uma de [SVM](#) adaptadas. O resultado desta comparação foi largamente favorável às redes pré-treinadas, todas obtendo resultados acima de 90% de *True Positive Rate* na classificação, em especial a GoogLeNet, que obteve praticamente 98%. Sendo esta última a rede com melhores resultados, foi utilizada para a demonstração do impacto da introdução de algoritmos de extração de características. Estes algoritmos têm como função extrair as componentes principais das amostras (neste caso, imagens) simplificando assim os dados utilizados no treino das redes, o que eventualmente permitirá uma redução do tempo de treino e uma melhoria na precisão das redes na classificação das imagens. Neste caso, foram utilizados: [PCA](#), [Independent Component Analysis \(ICA\)](#) e [Nonnegative Matrix Factorization \(NMF\)](#). E, segundo os resultados do estudo, estes algoritmos justificam a sua utilização, sendo que tanto o tempo de treino diminuiu e a precisão aumentou.

Outro exemplo da conjunção entre [IRTG](#) e [IA](#) é [46]. Neste trabalho é, ao contrário do anterior, utilizado [IRTG](#) passivo em que as imagens são captadas por um drone em pleno parque fotovoltaico. Após a recolha das imagens foram constituídos duas bases de dados, com o objetivo de criar duas redes neuronais: uma que deteta a existência de defeito (identifica dois estados, operação normal e com defeito) e outra que classifica o defeito (módulo curto-circuitado, acumulação de poeira, sombreamento e díodo de bypass danificado). De seguida, o tamanho destas bases de dados foi aumentado 15 vezes através da criação de novas imagens a partir das já existentes. Este método de expansão da base de dados, que tem como objetivo diversificar e aumentar os dados, denomina-se por *augmentation* e, neste caso, consiste em rodar, intensificar ou diminuir o brilho, aproximar ou afastar, espelhar e deslocar as imagens recolhidas de modo a criar mais imagens para treino das redes neuronais. Por fim, foram concebidas 3 técnicas de *Machine Learning* ([K-nearest neighbors \(kNN\)](#), [SVM](#) e *Catboost*) e 2 de *Deep Learning* (uma rede simples e uma pré-treinada, neste caso, a VGG-16) para a classificação de ambas as situações (averiguação da existência de defeito e classificação do defeito) e postas em contraste para comparar os seus desempenhos. Com este estudo foi, portanto, demonstrado a relevância da introdução de *Deep Learning* visto que, em comparação com as técnicas de *Machine Learning*, esta apresentou resultados de precisão bastante satisfatórios (98,7% em ambas as situações de classificação), sendo que a rede pré-treinada VGG-16 obteve praticamente 100% nas duas situações. Esta última, posteriormente, foi considerada e utilizada no presente estudo, como é descrito na Metodologia Proposta.

2.3.2.2 Imagens por Eletroluminescência

A conjugação dos métodos [EI](#) e as técnicas de *Machine Learning*, executada com bastante êxito, é verificável em [47]. Este trabalho estabelece 4 fases para a deteção e classificação de defeitos: a recolha de imagens eletroluminescentes; o pré-processamento das imagens; a extração de características; e classificação do estado do sistema fotovoltaico. Foram capturadas 982 imagens de módulos e, após uma filtragem (para remover imagens de

baixa qualidade), foi constituída uma base de dados de 753 imagens. De seguida, foram identificados 4 defeitos: 3 tipos de danos físicos (A, B e C, consoante o grau de danificação) e falhas nos dedos da célula (estes são os contactos metálicos que conduzem a energia gerada aos barramentos da célula). Com a base de dados formada, inicia-se o pré-processamento dos dados, que, neste caso, consiste em: normalizar as imagens, de modo que a todas sejam excluídas características irrelevantes para classificação (como o brilho e o que aparece atrás dos módulos); de seguida, é realizada a extração de células, que permite a individualização de cada célula numa só imagem (multiplicando assim a base de dados para 47244); e, por fim, a extração da região relevante. Este último passo permite uma simplificação significativa das imagens das células, realçando os defeitos presentes, com a transformação destas em imagens binárias. Com esta base de dados, já filtrada e transformada, é então executada a extração de características, à semelhança de outros trabalhos aqui já apresentados. Neste caso, as características foram calculadas e estabelecidas sem o auxílio de nenhum algoritmo, tendo assim sido formadas 25 características. Por fim, foram utilizadas 2 técnicas de *Machine Learning* (RF e SVM), obtendo resultados bastante positivos: RF com 96,7% de precisão e SVM com praticamente 100%.

Outro exemplo desta combinação (EI e IA) é encontrado em [48]. Neste trabalho, à semelhança do anterior, também é relatada a captura de imagens de módulos e a posterior extração de células para formar a base de dados, contudo o resto do processo (até à classificação de defeitos) é distinta. Após a criação da base de dados, é iniciado um pré-processamento muito mais apurado do que o do estudo anterior, embora o primeiro passo seja semelhante (a normalização das imagens): neste pré-processamento é dado bastante peso à extração de características sendo, por isso, apresentados vários algoritmos que executem essa função: KAZE; *Adaptive and Generic Accelerated Segment Test* (AGAST); *Histogram of Oriented Gradients* (HOG); *Pyramid Histogram of Visual Words* (PHOW); *Scale-Invariant Feature Transform* (SIFT); *Speeded-Up Robust Features* (SURF); e *Visual Geometry Group* (VGG). Todos estes são utilizados na visão computacional para extração de características e reconhecimento de objetos, sendo este conjunto bastante representativo da diversidade de técnicas utilizadas nesta área. Estes foram atentamente analisados e classificados de acordo com a sua capacidade de detetar características e pontos chave nas células. Por fim, foram concebidos vários classificadores de dois tipos: de *Machine Learning* (neste caso, SVM) com dados pós-extração de características e de *Deep Learning* (redes neuronais convolucionais). Do primeiro tipo, o que demonstrou maior performance (precisão) foi SVM com extração de características de KAZE e VGG (82,44%), contudo, ainda assim inferior à resultante de CNN (88,42%), demonstrando assim que uma CNN, mesmo sem extração de características prévia, consegue ser superior. No entanto, esta apesar de conseguir apresentar um melhor desempenho, também tem um custo superior: obriga a utilização de unidades de processamento gráfico, *Graphics Processing Unit* (GPU). Por isso, o método SVM com KAZE/VGG apresenta-se, em comparação, uma opção com elevada eficácia e ainda com custos computacionais reduzidos. Também é necessário realçar que neste trabalho não são classificados os tipos de defeitos, em vez

disto as células são classificadas por probabilidade/gravidade de defeito (de 0% a 100%).

METODOLOGIA PROPOSTA

Neste capítulo será apresentada a metodologia adotada para desenvolver uma ferramenta de detecção e diagnóstico de defeitos, no lado DC, dos sistemas fotovoltaicos. Esta ferramenta incorporará vários métodos e o utilizador poderá assim obter e comparar os vários resultados. Os métodos escolhidos foram dois elétricos (*IVCA* e *DDM*) e três visuais (*IRTG*, *EI* e *VI*), todos com base em redes neuronais. Neste capítulo serão então descritos os processos de construção destas redes neuronais, sendo que no capítulo 5 (Resultados) serão apresentados os resultados destas (a precisões na detecção dos defeitos). O processo de construção e a utilização da *Web App* são descritos no capítulo 4 (*Web App*).

3.1 Métodos Elétricos

3.1.1 Análise das Curvas I-V

O método com base na análise das curvas I-V foi integralmente desenvolvido em *Matlab*, desde a simulação dos dados até à construção da rede neuronal. A simulação dos dados (visto que não foi possível a produção de dados reais e não foram encontradas bases de dados online) foi feita em *Simulink*, assim como vários trabalhos aqui apresentados. Sendo três deles tomados como referências para a simulação dos defeitos [3], [14] e [18].

Em *Matlab Simulink* foram construídos os sistemas fotovoltaicos (com e sem defeitos) através de bibliotecas que permitem uma modelação simples destes sistemas. O exemplo principal destas é o bloco *PV-Array (Simscape Electrical)*, que com a configuração dos parâmetros (*strings* em paralelo e em série, células por módulo, corrente de curto-circuito, tensão em aberto, intervalo de temperatura das células, etc.), com a introdução de valores de temperatura e irradiância (entradas) é possível a medição da corrente e tensão aos seus terminais (saída) e, assim, a simulação da sua característica elétrica (curvas I-V e P-V). Mais especificamente, é possível visualizar na Figura 3.1 a configuração do sistema fotovoltaico. Este é constituído por 5 *strings* cada uma com 5 módulos (bloco *PV Array*) e recebe como *inputs* a temperatura e a irradiância. Por fim, são medidas a corrente, tensão e calculada a potência, que servirão como *outputs* para o *workspace* do *Matlab*.

Nesse sentido, foram criados os ficheiros *Simulink*, cada um correspondente a um

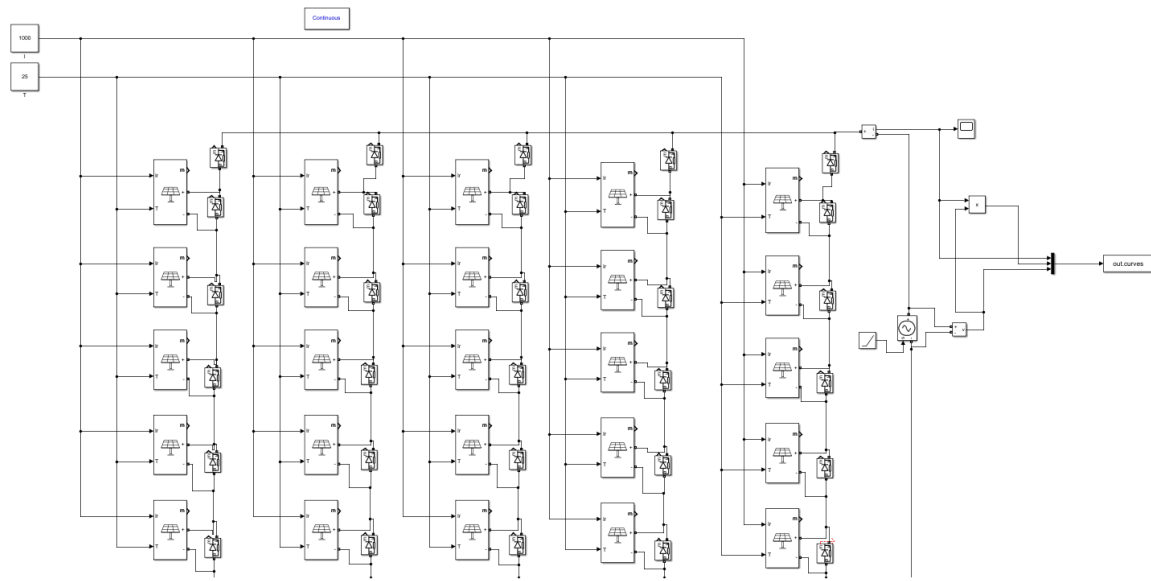


Figura 3.1: Sistema Fotovoltaico simulado

estado do sistema (em funcionamento normal, com sombreamento parcial, com sujidade, etc.). Para a simulação dos defeitos podem ser consultados inúmeros trabalhos (visto que, como foi descrito no estado da arte, a utilização de *Matlab* é frequente), sendo para o atual estudo considerados os seguintes: [3], [14] e [18]. Estes demonstram a simulação de vários defeitos e por vezes formas diferentes de simular os mesmos defeitos. Avaliando todos os defeitos possíveis de reproduzir foram selecionados os seguintes:

- Sombreamento parcial: o sistema fotovoltaico é dividido em 3 regiões, sem sombreamento, com 30% sombreamento e com 60%. Todas estas regiões representam um terço do total e o seu efeito de sombreamento é conseguido através de filtros sobre a irradiância.
- Sujidade: da mesma forma que no sombreamento parcial, o sistema fotovoltaico é dividido em 3 regiões, porém com filtros de 90% e 80% (em vez de 60% e 30%).
- Sombreamento de edifício: é aplicado um filtro de 50% na irradiância em um ou dois módulos.
- Vidro danificado: é filtrada a irradiância em 91% em todo o sistema fotovoltaico.
- Conectores danificados: é adicionada uma resistência de 5 Ohm em série com o sistema.
- Degradação: é colocada uma resistência de 1000 Ohm em paralelo com o sistema.
- Circuito aberto: uma *string* é desconectada.
- Curto-circuito: são curto-circuitados 1 a 4 módulos da mesma *string*.

Para a simulação e recolha dos dados foi desenvolvido um *script* em *Matlab* (.m) para cada estado a simular. Este *script* atribuirá em cada iteração um par de valores de temperatura e irradiância e guardará os resultados das simulações.

Todos os *scripts* foram escritos de forma idêntica, exceto os relativos aos defeitos por sombreamento parcial e sujidade. Estes não devem conter somente um valor de irradiância,

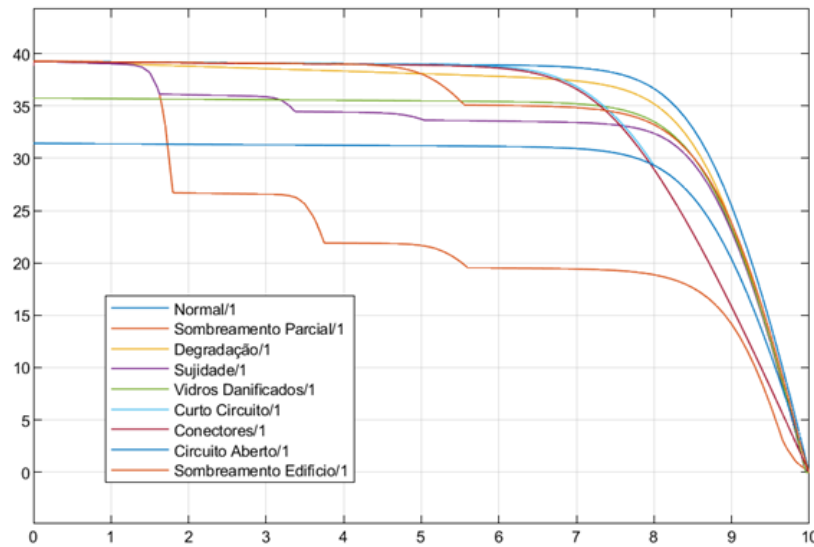


Figura 3.2: Evolução I-V do defeitos simulados

mas sim três valores, ou seja, em vez de se atribuir uma só irradiância, deve-se repartir três irradiâncias (100%, 90% e 80% da irradiância, no caso de sombreamento parcial) por todo o sistema fotovoltaico. Para isso, o *script* nestes dois casos distribui equitativa e aleatoriamente estes três valores para cada um dos 25 *PV Arrays* em cada iteração. Assim, em cada iteração é possível formar uma nova configuração de sombreamento parcial e de sujidade, o que permite diversificar o mais possível a amostra (neste caso, as evoluções das curvas I-V). Também há que realçar que neste aspeto (diversificação da amostra) também foram evoluídas outras simulações: na simulação do sistema em curto-circuito foram realizados 4 tipos de curto-circuito (em 1, 2, 3 e 4 módulos, numa *string*), o que representa 4 curvas I-V distintas, e no caso de sombreamento por edifício foi considerado o sombreamento tanto de dois módulos da mesma *string* como entre um de uma e outro de outra (e até o sombreamento somente de um módulo).

De modo a criar uma base de dados que represente o melhor possível os dados reais, foram produzidos muito mais dados relativos à situação de normal funcionamento (18000 casos) do que dos restantes estados (1000 casos cada). Desta forma considera-se que o estado largamente predominante durante operação é o estado de normal funcionamento.

Os dados extraídos são correspondentes às curvas na sua totalidade, ou seja, não são extraídos os seus parâmetros característicos, mas sim a curva na sua inteira evolução. Esta opção revelou-se adequada em [28], sendo que houve mais precisão no segundo caso (entrada: curvas I-V) do que no primeiro (entradas: 8 parâmetros característicos). Para acelerar este processo de recolha, dada a necessidade de produzir uma extensa base de dados, foi utilizado o *Parallel Computing Toolbox* (do *Matlab*) para que as várias simulações possam correr em paralelo. Por fim, foi verificado que na simulação de curvas mais complexas o número de pontos era largamente superior do que as restantes, sendo um exemplo disto a comparação entre as curvas de sombreamento parcial (podendo

chegar aos 600 pontos) e as de funcionamento normal (muitas vezes não chegando aos 100 pontos). Prevendo um custo no tempo de treino da rede neuronal e na precisão resultante, esta disparidade foi resolvida através da interpolação de todas as curvas da base de dados para serem representadas por apenas 50 pontos.

Por fim, com o auxílio da *Deep Learning Toolbox*, foi configurada, desenvolvida e testada uma rede neuronal *Long Short-term Memory (LSTM)*. Esta rede recebe como entrada 5 sequências de dados: irradiância; temperatura das células; e as evoluções da corrente, tensão e potência. A partir desta base de dados foram formados o conjunto de dados para treino da rede (90% do total), feito em 70 épocas, e o conjunto para teste/validação (10%). Foi, portanto, criada uma rede neuronal para a detecção e classificação de 8 defeitos (9 estados, contando com a operação sem defeitos) composta por 5 camadas (Figura 3.3):

- Camada de Entrada: aqui são recebidos os dados, neste caso, 5 sequências de dados.
- Camada Bidirecional *LSTM* (BiLSTM): esta é a camada principal da rede. Nesta são processadas as sequências e criadas as memórias (curta e longa), que controlarão o fluxo de informações de forma a manter a relevância das informações passadas. Por ser bidirecional, esta camada processa as sequências nos dois sentidos, do início ao fim (*forward*) e do fim ao início (*backward*). Esta camada conta com 150 unidades ocultas.
- Camada totalmente conectada: nesta é transformada a saída da anterior (BiLSTM) em função das 9 classes (9 possíveis estados do sistema fotovoltaico). Estas 9 classes são representadas por 9 neurónios.
- Camada Softmax: aqui são convertidos os resultados da camada anterior em probabilidades para cada classe.
- Camada de Classificação: por fim, nesta camada é classificado o estado do sistema.

3.1.2 Medição das Diferenças Diretas

A implementação deste método foi baseada em [49]. Neste trabalho são recolhidas as tensões e correntes de duas *strings* e, juntamente com a temperatura e irradiância, é determinado o estado do sistema, ou seja, pela diferença direta das medições das *strings* (DDM) foi possível classificar o defeito. Para este efeito foram recolhidas estas medições durante 16 dias, de seguida foram catalogadas em função do estado do sistema (funcionamento normal, curto-circuito, circuito aberto, degradação e sombreado) e, por fim, foram desenvolvidos 4 métodos de IA para a detecção e classificação do estado do sistema fotovoltaico (kNN, *Decision Trees*, SVM e NN). Neste estudo, o método com maior precisão (na fase de validação) é NN com 95,44%. Por isso, o presente estudo, não só seguiu este método como procurou melhorá-lo (aumentando a precisão).

Sendo que os dados usados em [49] foram disponibilizados publicamente, já em formato *Matlab* (.mat), foi adaptada a rede neuronal desenvolvida para o método IVCA, alterando o número de unidades ocultas na Camada BiLSTM para 100 e o número de épocas para 40.

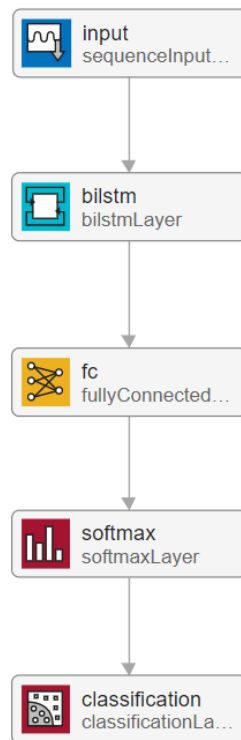


Figura 3.3: Camadas das Redes dos Métodos Elétricos

3.2 Métodos Visuais

Assim como no método apresentado anteriormente ([DDM](#)), os três métodos visuais implementados neste trabalho necessitaram de base de dados disponíveis online, sendo uma fonte bastante útil o site [50] (donde foram obtidos os dados para os métodos [IRTG](#) e [VI](#)). Todos estes métodos utilizam redes neurais convolucionais, e para a seleção da melhor rede para cada caso foram testadas e comparadas várias redes convencionais (sequenciais, com as camadas típicas) e pré-treinadas. Toda a conceção das redes foi efetuada em Python, no Google Colab (permitindo assim a utilização de [GPU](#) externos, o que acelera consideravelmente o treino das redes), essencialmente utilizando as bibliotecas Tensorflow (para a construção de redes neurais), NumPy e Pandas (que permitem a manipulação e organização dos dados).

Para cada método visual foram concebidas várias redes neurais, desde redes pré-treinadas a redes sequenciais, e, comparando os resultados de cada, selecionada a rede ideal para cada caso. As redes neurais pré-treinadas são disponibilizadas pela Keras e, avaliando o peso e a precisão de cada, foram selecionadas 3: ResNet50, VGG16 e InceptionV3. Redes como EfficientNet e DenseNet, por exemplo, após pequenos testes demonstraram-se não só morosas como relativamente ineficientes. Para o treino e comparação de resultados junta-se também uma rede sequencial composta por 6 blocos convolucionais.

As redes neurais concebidas com base em redes pré-treinadas têm a seguinte

estrutura (Figura 3.4):

- Camada de entrada (*Input Layer*): aqui as imagens são recebidas e processadas para as dimensões de entrada da rede, que neste caso são de (244, 244, 3), ou seja, 244 pixels de altura e largura e com 3 canais de cor de imagem (RGB).
- Camadas de extração de características (*Functional*, na Figura 3.4): nestas camadas está o modelo base da rede, que neste caso será uma das redes pré-treinadas (ResNet50, VGG16 ou InceptionV3). Aqui são extraídas as características das imagens.
- Camada de *Pooling Global*: recebendo os mapas de características produzidos na camada anterior, esta reduzirá a dimensão destes mapas e formará um valor médio para cada, produzindo assim uma representação simplificada e compacta dos resultados da camada anterior.
- Camada de *Dropout*: nesta camada é definida a percentagem de neurónios que serão desativados a cada iteração, aleatoriamente. Esta prática permite evitar o *overfitting*, que ocorre quando o modelo, por excesso de complexidade, passa a memorizar para além dos padrões necessários (ruído ou dados peculiares).
- Camada de saída (*Dense*): por fim, nesta última camada é feita a classificação e é formada por neurónios totalmente conectados com a camada anterior. Estes neurónios terão o número correspondente ao número de classes possíveis de classificação, ou seja, cada neurónio determinará a probabilidade relativa a cada classe. Assim, o estado identificado em cada imagem será o que obtiver maior probabilidade.

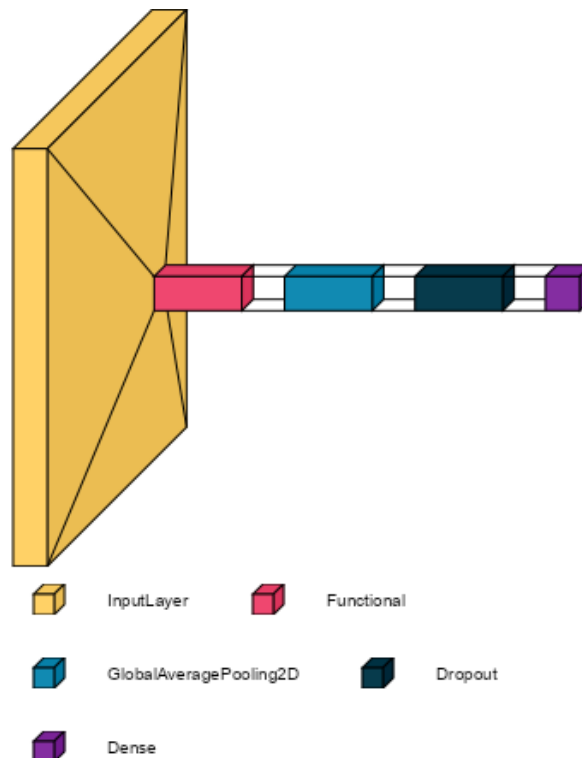


Figura 3.4: Camadas das Redes com bases pré-treinadas

Já a [CNN](#) sequencial (Figura 3.5), tendo em comum as camadas de entrada e saída, é

muito mais simples e leve (o que confere peso e complexidade nas anteriores é o modelo base constituído pelas redes pré-treinadas). Esta rede, após a camada de entrada, tem 6 blocos de 3 camadas (convolução, normalização e *pooling*), em que, sequencialmente, vão aumentando os filtros da camada de convolução (para aumentar a capacidade de captura de detalhes). Nestes blocos as camadas de normalização têm como função estabilizar as saídas das camadas de convolução, acelerando o treino e melhorando a eficácia, já as camadas de *pooling* reduzem as dimensões das características extraídas pelas camadas de convolução. Após todos estes blocos, foi introduzida uma camada *Flatten* que converterá as saídas 2D das camadas anteriores para 1D e permitirá uma melhor transição para as seguintes camadas: camadas densas (ou totalmente conectadas, *Fully Conected*). Estas, em conjunto com as de *Dropout*, configuram o final da rede (imediatamente antes da camada de saída), sendo que as densas vão progressivamente reduzindo o número de neurónios, preparando os dados para classificação.

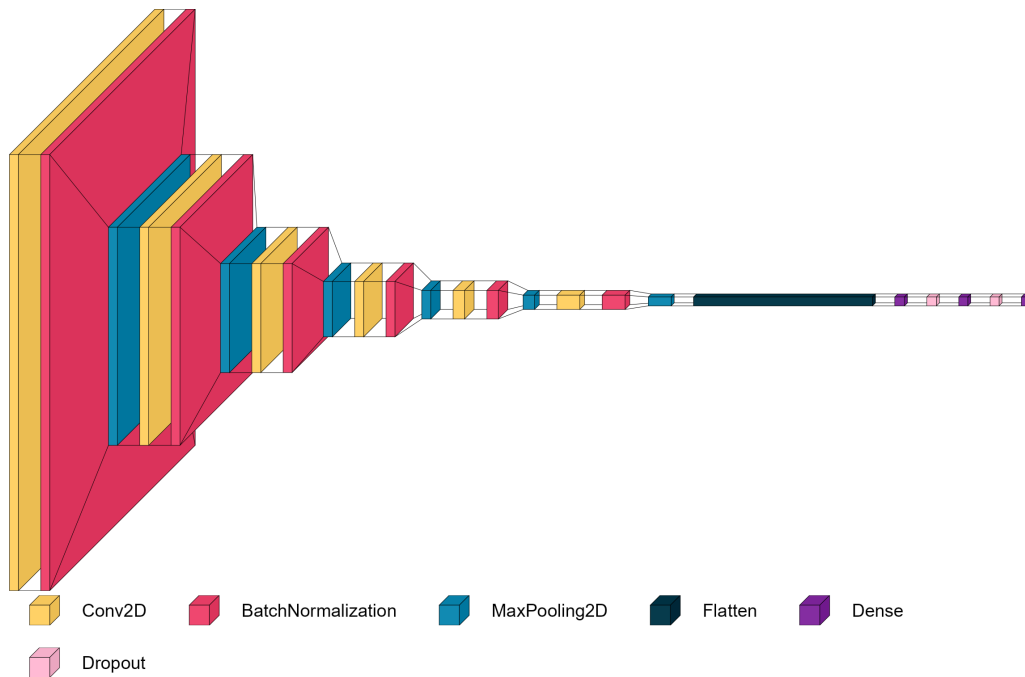


Figura 3.5: Camadas da CNN sequencial simples

Para a implementação do métodos **IRTG** foi utilizada a base de dados descrita em [51], que consiste em imagens infravermelhas de módulos divididas em 10 estados: *Hotspot* numa célula; *Hotspots* em várias células; rachadura; sombreamento; díodo de *bypass* ativado; vários díodos de *bypass* ativados; vegetação a cobrir o módulo; sujidade; módulo queimado; e sem defeito. Esta base de dados encontra-se disponível online [52] e pretende representar os estados dos módulos ao longo do tempo, o que implica que seja desequilibrada, ou seja, que apresente mais imagens dos estados predominantes (o estado de funcionamento normal tem 10000 imagens enquanto, por exemplo, o sombreamento tem 1056).

O método VI, assim como o anterior, socorreu-se de uma base de dados online [53].

Esta contém 891 imagens visuais (fotográficas) de painéis solares divididas em 6 classes: 3 tipos de sujidade, poeira, neve e dejetos de aves (*Bird Drop*); 2 tipos de danos (elétricos e físicos); e sem qualquer defeito. Diferentemente do caso anterior, esta base de dados é pouco desequilibrada, contudo, visto que mesmo a classe que deveria ser largamente predominante (imagens de operação normal dos painéis) tem poucas imagens (194), optou-se por se manter este leve desequilíbrio.

Ao contrário dos restantes métodos visuais, a base de dados usada para o método [EI](#) foi obtida diretamente dos autores. Estes disponibilizaram a base de dados que criaram e descreveram em [54]. Esta é a base de dados mais extensa dos três métodos sendo constituída por 15853 imagens para treino e validação do modelo divididas em 13 classes. Contudo, visto que 5 (das 13) classes tinham uma baixíssima representatividade (cada uma menos de 80 imagens), optou-se pela utilização de apenas 8 classes (Figura 5.14): curto-circuito; rachadura; dedos; deslocação vertical e horizontal; núcleo negro; mancha espessa; e sem defeito. Antes do treino dos modelos é desde logo notável a semelhança entre células em boas condições e com manchas espessas, o que, como se verificará mais adiante, dificulta a identificação deste defeito por parte do modelo. Esta base de dados é bastante desequilibrada, sendo que as imagens de células sem defeito são 72% (11353 imagens) do total.

WEB APP

Por fim, desenvolvidos os 5 modelos de redes neurais convolucionais, foi então construída a *Web App* que os integra e que os disponibiliza ao utilizador. Esta foi feita em Streamlit, uma biblioteca de código aberto de Python, o que facilitou a integração dos modelos visuais (visto que foram concebidos em Python). Porém os métodos elétricos, desenvolvidos em Matlab, necessitaram de uma adaptação: foram transformados num formato ONNX. Este é um formato aberto para representação de modelos de *Machine Learning* que permite a sua utilização em diversas plataformas. De seguida, também em Python, foi necessário desenvolver um pré-processamento de imagens (no caso dos métodos visuais) e de dados (métodos elétricos) para que a aplicação, independentemente do tamanho das imagens ou do conjunto de dados que o utilizador introduzir, consiga utilizar sempre os modelos. Sabe-se que os modelos visuais têm de receber imagens de 244 pixéis de altura e largura e que o modelo elétrico *IVCA* recebe 5 *arrays* de 50 pontos. No caso dos métodos visuais, a aplicação, aceitando qualquer imagem, transforma-a em (244, 244, 3). No caso do método *IVCA*, são pedidas as evoluções da corrente em função da tensão num formato CSV e os valores da irradiância e temperatura, e a aplicação, calculando a evolução da potência e interpolando o número de pontos recebido para o número pretendido, configura um novo conjunto de dados que poderá ser usado no modelo.

4.1 Utilização da Aplicação

A aplicação é muito simples e direta. Na sua página inicial apresenta os métodos de deteção disponíveis, em que permite ao utilizador obter uma descrição geral de cada método (assim como o tipo de dados que terá de introduzir para usar cada método) e, à esquerda desta página, o utilizador poderá acessar a página correspondente a cada método (Figura 4.1). De seguida, selecionado o método, é pedida a introdução dos dados de entrada (sejam dados visuais ou elétricos, dependendo do método) e, após a introdução, em poucos segundos, a aplicação apresentará os resultados graficamente, indicando a probabilidade de cada defeito. Para exemplificar será aqui apresentado uma classificação

de um método visual (neste caso, VI) e dos dois elétricos.

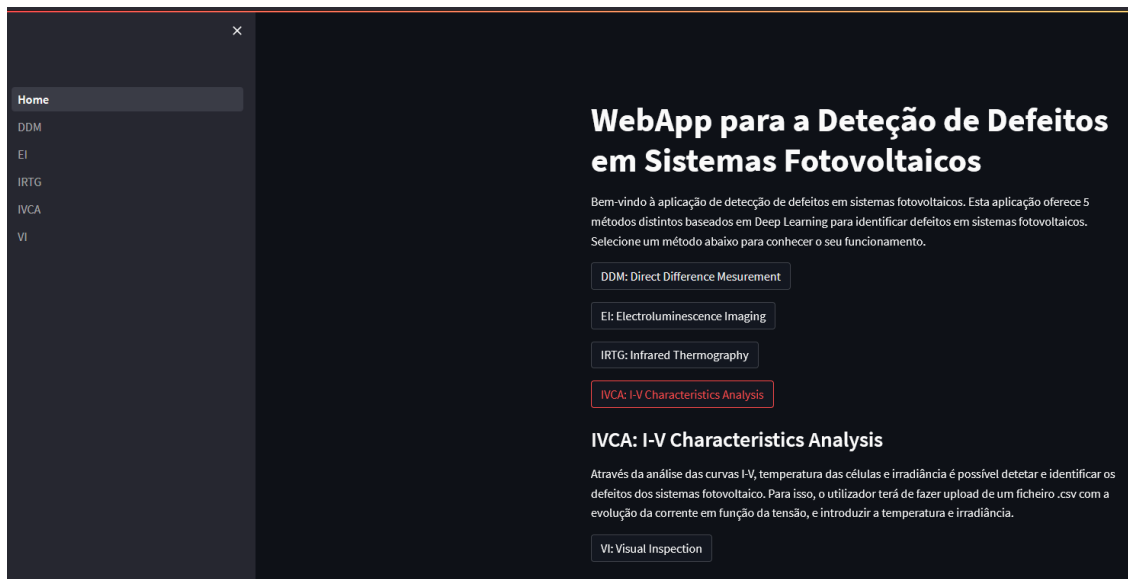


Figura 4.1: Página principal da aplicação

4.1.1 Inspeção Visual

Em todos os métodos visuais, o utilizador, através de *drag and drop* ou procurando nos documentos, terá de fazer *upload* da imagem a classificar (Figura 4.2). Para exemplificar foi escolhida uma imagem ambígua (Figura 4.3), em que os painéis se encontram levemente com poeira e com uma pequena marca que poderá ser identificada como dejetos de ave. Porém, avaliando previamente a imagem (antes do treino dos modelos), foi considerada como sendo uma imagem de painéis com poeira, tendo em conta que é o defeito que mais sobressai.

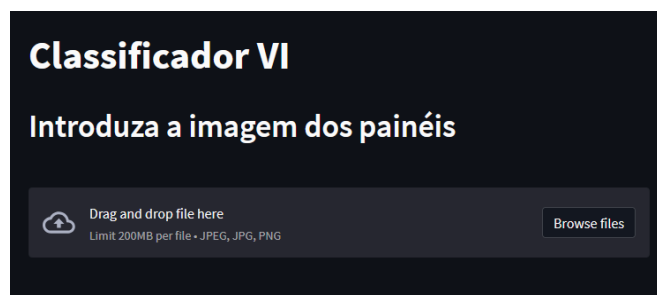


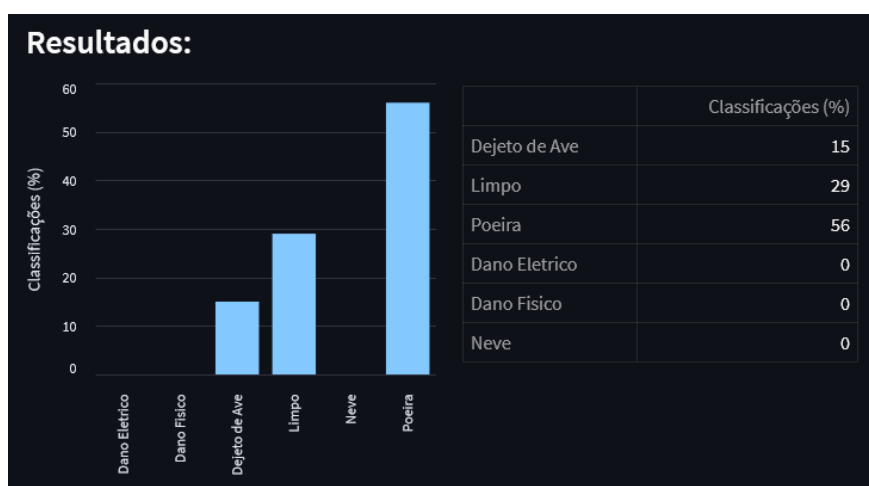
Figura 4.2: Página principal do método VI

Após a introdução da imagem por parte do utilizador, a aplicação calcula (em cerca de 5 segundos), através do modelo implementado, a probabilidade de cada estado dos painéis (Figura 4.4).

Observando os resultados é possível constatar a eficácia na classificação. Esta imagem, sendo bastante ambígua, permitiu verificar que o modelo reconheceu o padrão de três



Figura 4.3: Imagem ambígua de painéis solares com poeira

Figura 4.4: Resultados do método VI (*Web App*)

estados numa só imagem e ainda conseguiu arbitrar qual é o estado predominante.

4.1.2 Análise das Curvas I-V

Na utilização deste método, o utilizador necessitará da evolução da corrente em função da tensão em formato CSV e os valores da irradiância e temperatura. A aplicação apenas permite o *upload* de um ficheiro CSV para ler os dados elétricos (curva I-V) e, após a introdução destes, serão lidos e apresentados num *dataframe* e formada uma representação gráfica destes (curva I-V), como apresentado na Figura 4.5.

De seguida, será indicado ao utilizador a introdução dos valores da temperatura e irradiância (Figura 4.6) e, por fim, são apresentados os resultados de classificação (Figura 4.7).

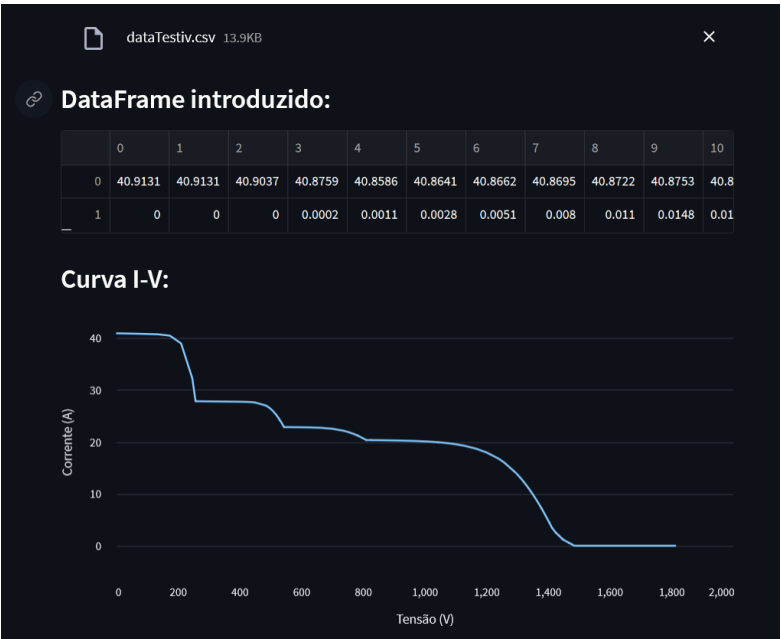


Figura 4.5: Dataframe e Curva IV (método IVCA, Web App)

The screenshot shows a web application form with two input fields. The first field is labeled 'Introduza o valor da irradiância:' and contains the value '932.00'. The second field is labeled 'Introduza o valor da temperatura:' and contains the value '75.00'. Below the fields is a 'Submeter' button. At the bottom, the input values are displayed: 'Irradiância 932.0' and 'Temperatura 75.0'.

Figura 4.6: Introdução da irradiância e temperatura (Web App)



Figura 4.7: Resultados do método IVCA (Web App)

4.1.3 Medição das Diferenças Diretas

O método **DDM** é o mais simples e manual dos 5 métodos. Sendo que as entradas do modelo são somente 6 números (correntes e tensões das duas *strings*, temperatura e irradiância). Desta forma o utilizador somente necessita de introduzir estes 6 valores (Figura 4.8) e terá imediatamente o resultado da classificação do modelo (Figura 4.9). Neste teste na aplicação os dados introduzidos (representativos da classe “sombreamento”) foram recolhidos aleatoriamente do conjunto de dados para teste/validação da modelo.

Figura 4.8: Introdução de dados do método DDM

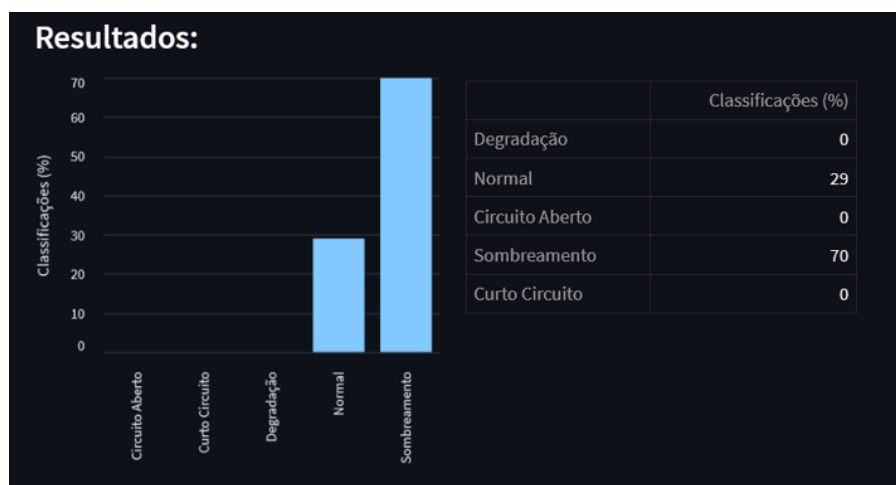


Figura 4.9: Resultados do método DDM (*Web App*)

Toda a *Web App* pode ser largamente melhorada esteticamente, e no seu design, porém, neste teste pretende-se, principalmente, formar uma aplicação simples, prática e testar as suas funcionalidades de deteção e identificação de defeitos nos sistemas fotovoltaicos. Nesse sentido, a *Web App* cumpre o seu propósito.

RESULTADOS

Neste capítulo serão apresentados e analisados os resultados, em termos de precisão, das redes neurais de cada método. O cálculo da precisão de cada rede é feito pela divisão entre o número de amostras bem classificadas e o número de todas as amostras (do conjunto de teste/validação). Este cálculo tem a desvantagem de apresentar valores demasiado gerais, sendo que a rede, tendo uma precisão bastante elevada numa classe de defeitos com uma larga representatividade na amostra total (por exemplo, a classe de funcionamento normal tem um número de amostras bastante superior às restantes classes), pode apresentar uma precisão geral alta embora tendo classes de defeitos, com menor representatividade, com precisões baixas. Por isso, para uma análise mais individualizada a cada defeito, também serão apresentadas matrizes de confusão para cada rede.

5.1 Métodos Elétricos

Como foi descrito no capítulo anterior (Metodologia Proposta) a rede usada para os métodos elétricos é baseada em [LSTM](#) bidirecional. Esta, sendo relativamente simples, não necessitou de mais camadas porque desde logo apresentou resultados de excelente qualidade.

Relativamente ao método [IVCA](#), o treino da rede foi executado em 115 épocas (Figura 5.1) e, de seguida, foi construída uma *Confusion Matrix* com o conjunto de validação (Figura 5.2). Os resultados ficaram próximos da perfeição, obtendo 98,73% de precisão geral, o que tendo em conta a quantidade e diversidade da base de dados enaltece ainda mais este resultado.

O modelo do método [DDM](#) foi treinado em somente 40 épocas e o conjunto de teste/precisão obteve 98,94% de precisão (na validação), atingindo também precisões individuais (de cada classe de defeitos) bastante altas (Figura 5.3). Este resultado foi excelente e conseguiu superar os resultados atingidos em [49].

Em conclusão, os modelos dos métodos elétricos apresentaram resultados excelentes e ambos conseguem competir com os melhores modelos apresentados no Estado da Arte (capítulo 3). O modelo [IVCA](#) tem uma base de dados extensa e diversa (9 classes), tendo

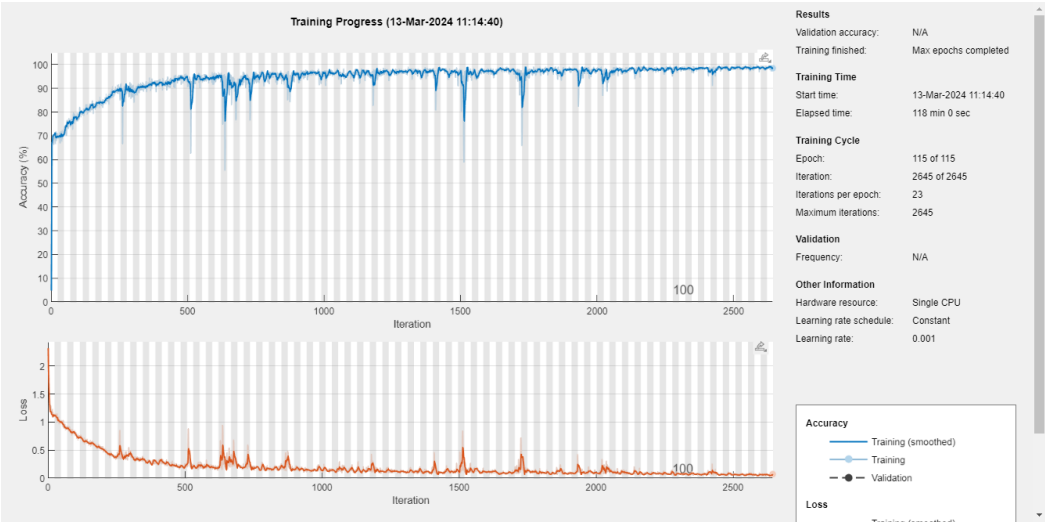


Figura 5.1: Progressão do Treino do Método IVCA

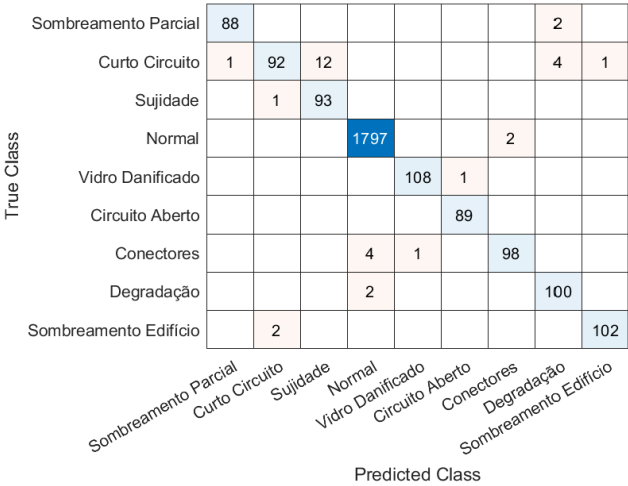


Figura 5.2: Matriz de Confusão do Método IVCA (99% de precisão)

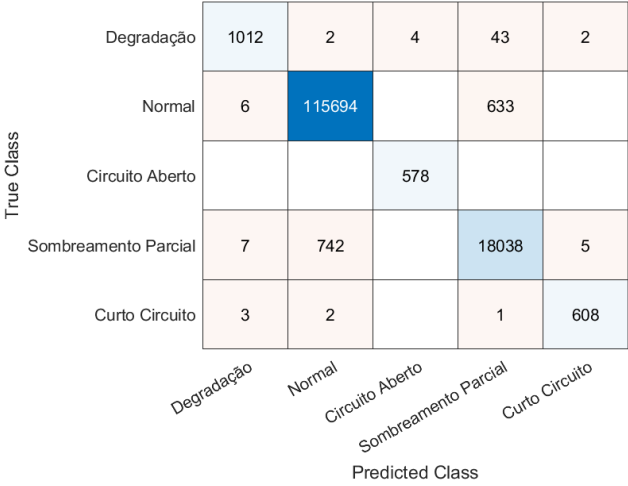


Figura 5.3: Matriz de Confusão do Método DDM (99% de precisão)

defeitos como o de sombreamento parcial ou de curto-circuito várias configurações, ou seja, estes defeitos também conseguem ser bastante diversificados, e, no entanto, este obteve precisões próximas de 100%. Também o modelo de [DDM](#) atingiu este nível de resultados superando assim, como era seu objetivo, os resultados obtidos em [49], sendo que este trabalho teve a mesma base de dados.

5.2 Métodos Visuais

Como foi descrito no capítulo anterior, para a construção dos modelos dos métodos visuais foram definidas 4 Redes Neurais Convolucionais ([CNN](#)), onde também foram descritas as suas camadas, 3 baseadas em redes pré-treinadas (InceptionV3, ResNet50 e VGG16) e uma CNN sequencial simples. Estas foram testadas em todos os métodos.

5.2.1 Termografia de Infravermelhos

Relativamente ao método [IRTG](#), este conta com 10 classes, presentes na Figura 5.4, e para o treino e teste/validação das 4 [CNN](#), foram numeradas de 0 a 9, em que, seguindo a ordem presente na Figura 5.4, da esquerda para a direita, *Hotspot* numa célula é o 0 e *Vegetação* é 9. Após o treino das [CNN](#), foram constituídas as seguintes matrizes de confusão a partir do conjunto de teste/validação.

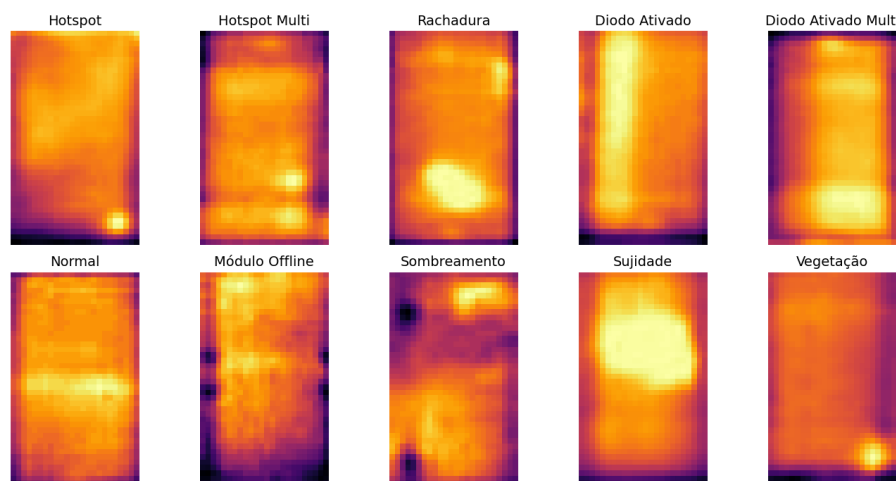


Figura 5.4: Tipos de Defeitos (Método IRTG)

Como é verificável nos resultados, a escassez de dados em algumas classes pode ter prejudicado o treino e, por isso, as precisões de algumas classes. O exemplo claro disto é a precisão na classificação das imagens de módulos com sujidade: todas as redes tiveram resultados péssimos nesta classificação. Ainda assim a deteção da ativação de vários díodos de *bypass* (que também é representada por poucas imagens) obteve resultados bastante positivos. O que também permite concluir que a qualidade das imagens da classe “sujidade” poderá ter influenciado o treino dos modelos.

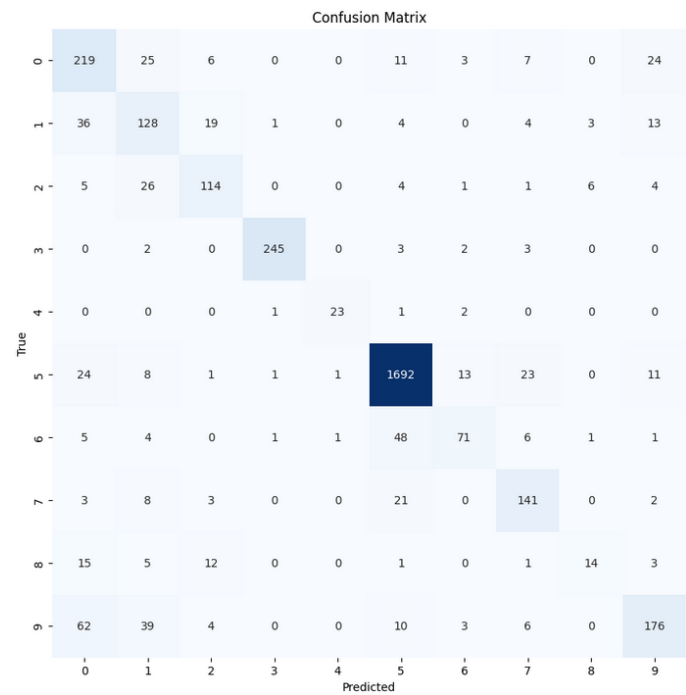


Figura 5.5: IRTG. Matriz de Confusão (InceptionV3), 83% de precisão

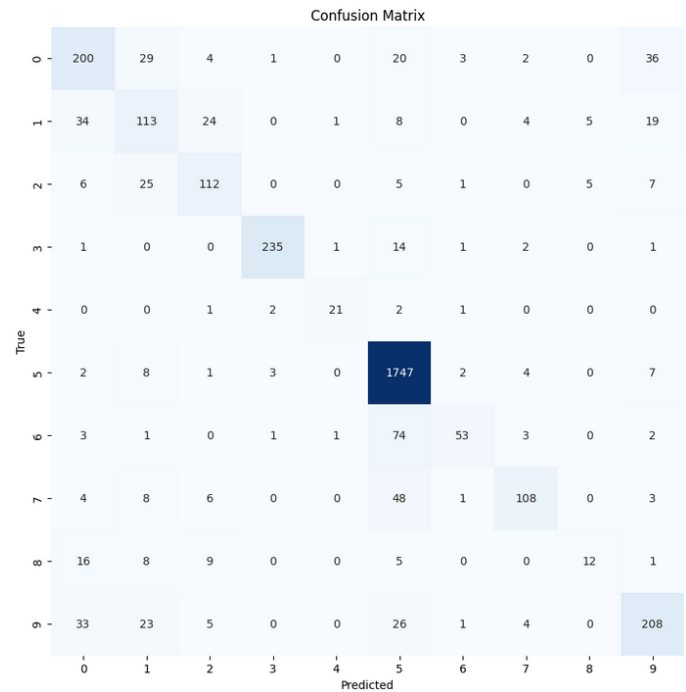


Figura 5.6: IRTG. Matriz de Confusão (ResNet50), 83% de precisão

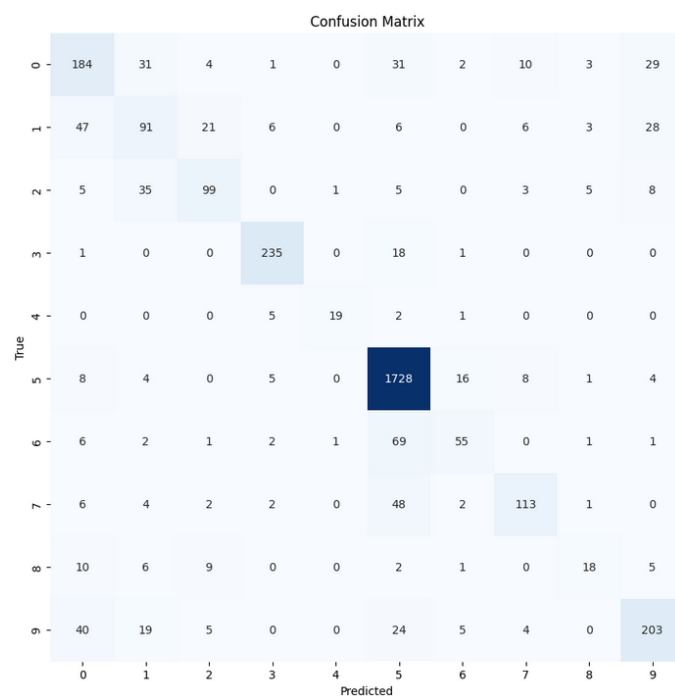


Figura 5.7: IRTG. Matriz de Confusão (VGG16), 81% de precisão

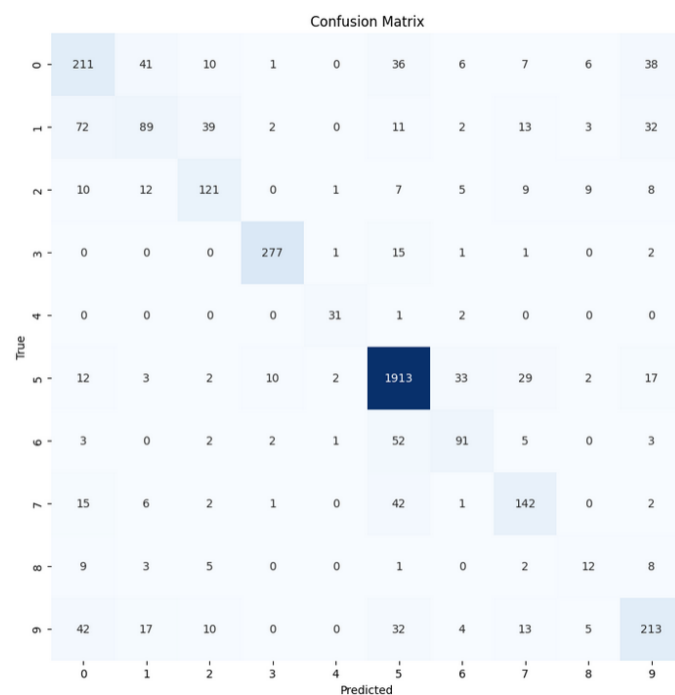


Figura 5.8: IRTG. Matriz de Confusão (CNN), 79% de precisão

Este método foi o que obteve menor precisão dos 3 métodos visuais em todas as redes exceto no caso da **CNN** sequencial (Figura 5.8). Esta, apesar de ser consideravelmente mais simples (e por isso bastante mais leve) que as restantes, demonstrou uma precisão próxima das restantes (79%). Ainda assim, foi a pior das 4, sendo que a InceptionV3 (Figura 5.5) e a ResNet50 (Figura 5.6) obtiveram 83% e a VGG16 (Figura 5.7) 81% de precisão.

5.2.2 Inspeção Visual

Para a construção do método **VI** foram constituídas 6 classes, presentes na Figura 5.9, as 4 **CNN** foram treinadas e, de seguida, testadas com conjunto de validação. Para a numeração das classes foi seguida a ordem presente na Figura 5.9, da esquerda para a direita, sendo a classe “Dejeto de Ave” o número 0 e a “Neve” o 5. Os resultados do teste com o conjunto de validação foram estruturados nas seguintes matrizes de confusão.

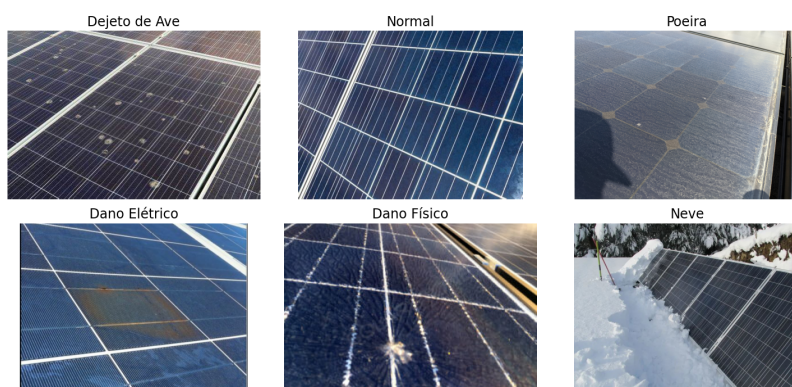


Figura 5.9: Tipos de Defeitos (Método VI)

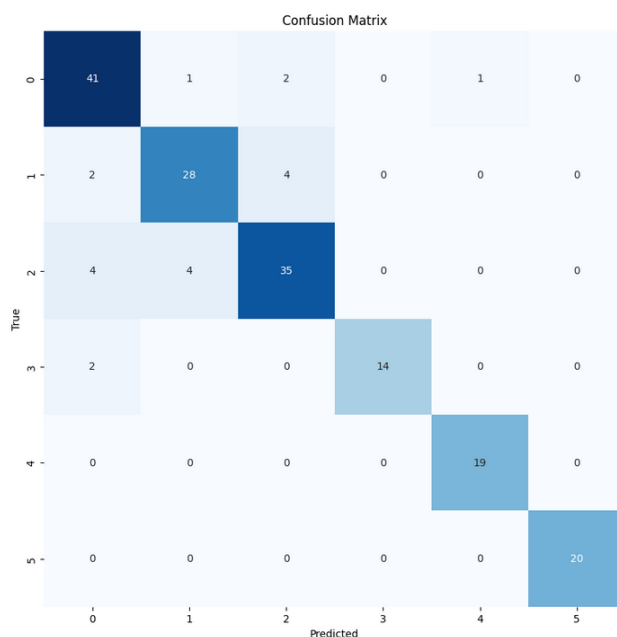


Figura 5.10: VI. Matriz de Confusão (InceptionV3), 89% de precisão

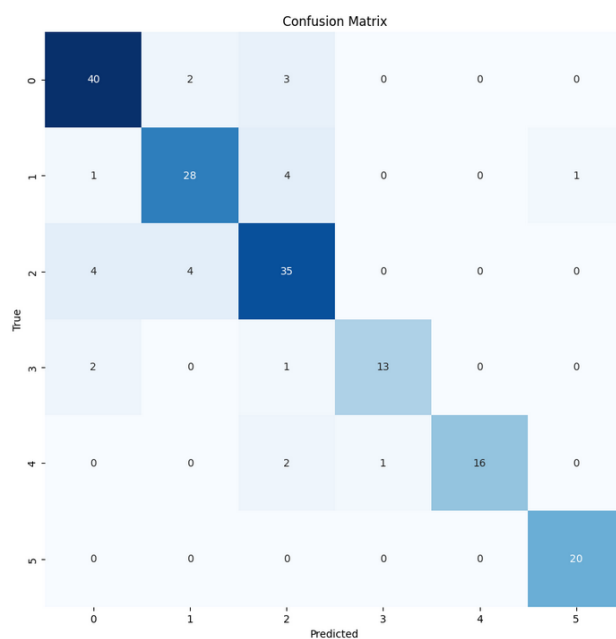


Figura 5.11: VI. Matriz de Confusão (ResNet50), 86% de precisão

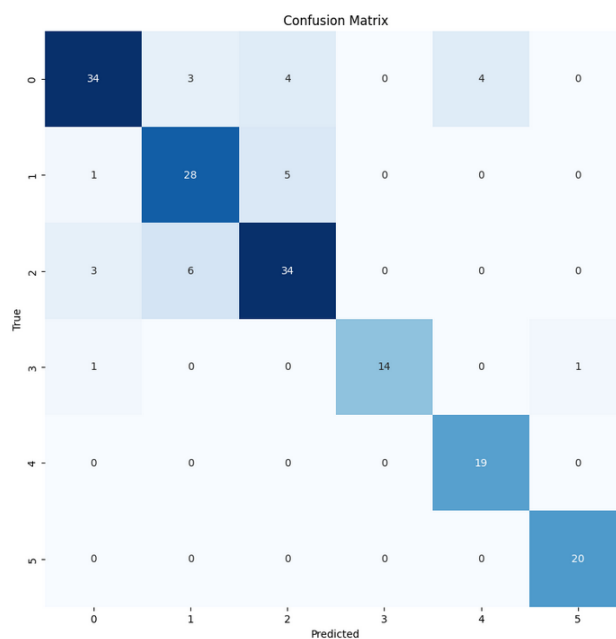


Figura 5.12: VI. Matriz de Confusão (VGG16), 84% de precisão

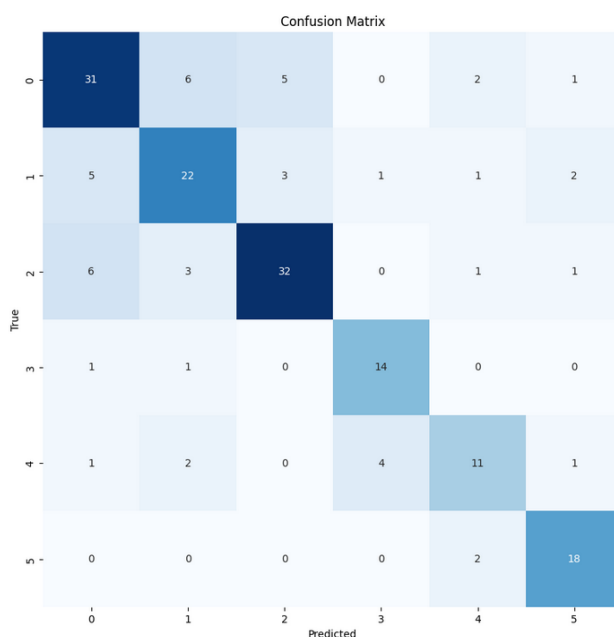


Figura 5.13: VI. Matriz de Confusão (CNN), 72% de precisão

Em contraste com os conjuntos de dados dos outros métodos, o deste era bastante reduzido, o que não proporcionou péssimos resultados, antes pelo contrário. Neste caso os modelos com base em redes pré-treinadas obtiveram precisões bastante elevadas, destacando-se a de InceptionV3 (Figura 5.10), com 89%, das restantes (ResNet50, Figura 5.11, com 86% e VGG16, Figura 5.12, com 84%). Já a CNN sequencial não foi além dos 72% (Figura 5.13).

5.2.3 Imagens por Eletroluminescência

Relativamente ao método EI, foram constituídas 8 classes, presentes na Figura 5.14, e a partir desta foram numeradas, à semelhança dos restantes métodos, da esquerda para a direita (sendo a classe “Núcleo Negro” o 0 e a “Deslocação Vertical” o 7). Após o treino das 4 CNN, com o conjunto de teste/validação, foram formadas as seguintes matrizes de confusão.

Como anteriormente foi indicado (na Metodologia Proposta), a identificação do defeito “mancha espessa” não obteve um resultado positivo na validação de todas as redes, com exceção da InceptionV3 (Figura 5.15), que, não sendo muito bom, cumpriu um mínimo aceitável. Esta, mais uma vez, demonstrou ser a melhor opção, destacando-se das demais com 94% de precisão (no total), com elevadas precisões na maior parte dos defeitos e, juntamente com a ResNet50, com 100% na deteção de curtos-circuitos e deslocações horizontais. A ResNet50 (Figura 5.16) e a VGG16 (Figura 5.17) obtiveram também resultados bastante positivos (no total), tendo 92% e 91% respetivamente. Por fim, os piores resultados foram produzidos pela CNN sequencial (Figura 5.18), com péssimos resultados na deteção de dedos e manchas espessas (e não satisfatórios na de rachaduras), ainda assim obtendo uma precisão geral de 86% (o que só foi possível com a elevada

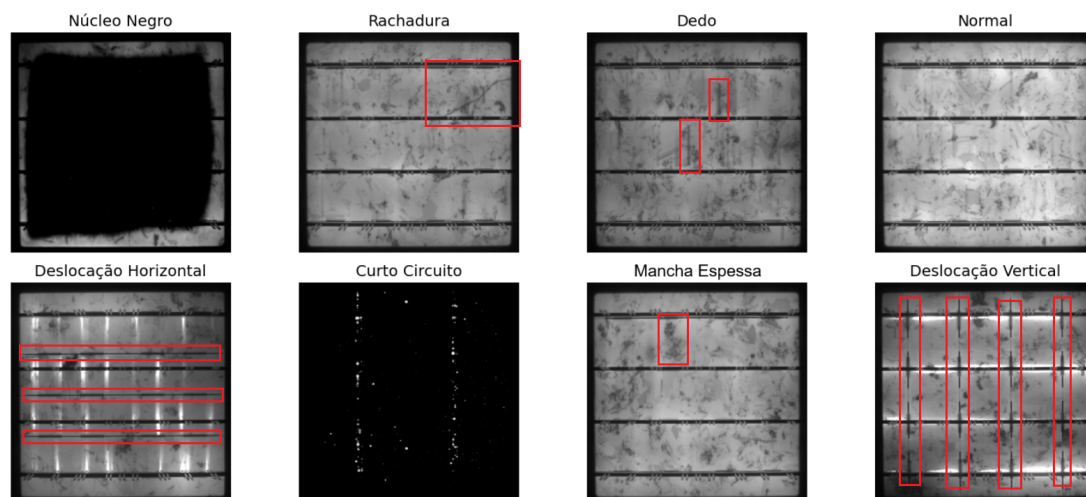


Figura 5.14: Tipos de Defeitos (Método EI)

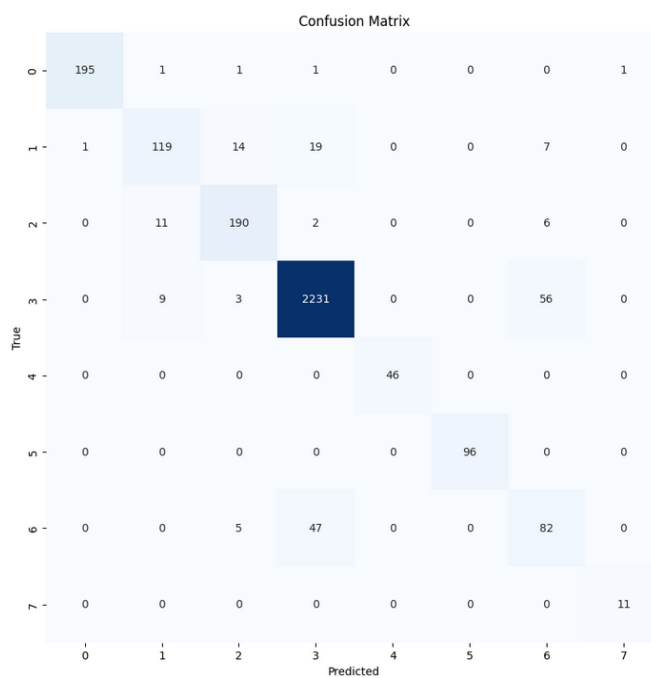


Figura 5.15: EI. Matriz de Confusão (InceptionV3), 94% de precisão

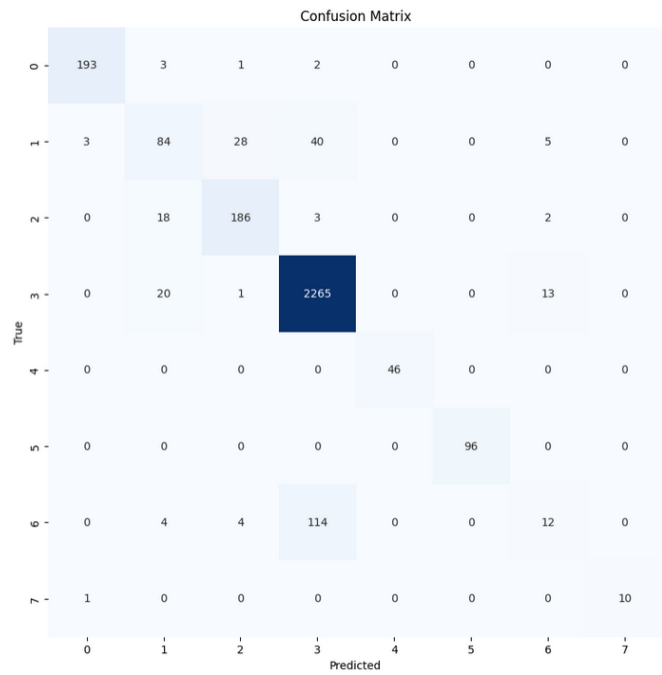


Figura 5.16: EI. Matriz de Confusão (ResNet50), 92% de precisão

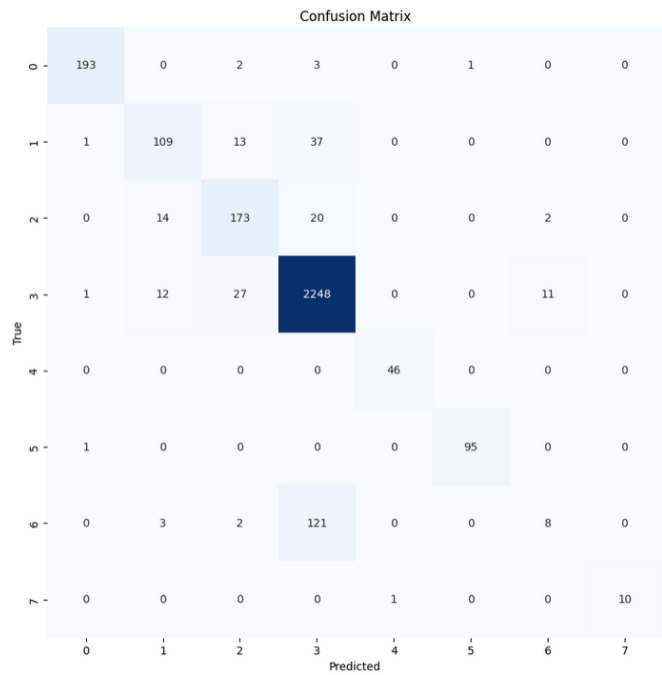


Figura 5.17: EI. Matriz de Confusão (VGG16), 91% de precisão

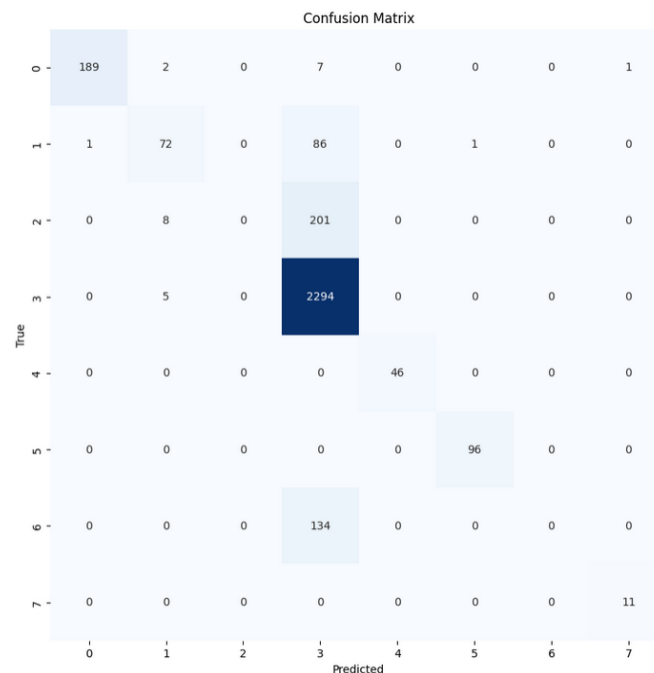


Figura 5.18: EI. Matriz de Confusão (CNN), 86% de precisão

eficácia na deteção dos restantes defeitos).

Assim, de acordo com os resultados, para todos os métodos os modelos com melhor performance foram os de rede pré-treinada InceptionV3. Porém, no método [IRTG](#) se se optar pela [CNN](#) sequencial não se verificará uma perda de desempenho substancial (em comparação com o modelo InceptionV3) e tem-se a vantagem de utilizar um modelo mais leve. Também se pode concluir que os modelos conseguiram detetar a esmagadora maior parte dos defeitos porém, se se descartar certos defeitos (se se formar novos modelos sem os defeitos com pouca precisão) poderão, ainda assim, ser formados modelos bastante abrangentes na deteção e identificação de defeitos em sistemas fotovoltaicos, sendo que nos métodos visuais com menor precisão, [EI](#) e [IRTG](#), são usadas 10 e 8 classes, respetivamente.

CONCLUSÃO

Como foi descrito na Introdução deste documento, urge a conceção e implementação de métodos eficazes, práticos e económicos para a deteção e classificação dos sistemas fotovoltaicos, melhorando assim a sua manutenção e, conseqüentemente, corrigindo o seu rendimento ao longo dos anos de utilização. Nos últimos anos têm-se verificado diversas opções neste sentido (abrangendo várias áreas) e no segundo capítulo (Estado da Arte) do presente trabalho foram analisadas. Foram recolhidos estudos de referência das principais áreas: [VTM](#) e [EBM](#), atribuindo uma maior atenção a estes métodos combinados com inteligência artificial (em especial, redes neuronais).

Foi então proposta, no terceiro capítulo, a conceção de uma ferramenta que possibilitará ao utilizador usar diversos métodos para a deteção e classificação dos estados do sistema, de modo a que o utilizador possa introduzir vários tipos de dados. Foram identificados e estudados os trabalhos de [VTM](#) e [EBM](#) com os melhores resultados e foram elaboradas 5 redes neuronais, 3 com base em métodos visuais ([EI](#), [VI](#) e [IRTG](#)) e 2 em métodos elétricos ([IVCA](#) e [DDM](#)). Para cada [VTM](#) foram construídas 4 redes neuronais (1 rede neuronal convolucional simples e 3 pré-treinadas), sendo posteriormente selecionada a rede com maior precisão, que foi, em todos os métodos, a rede pré-treinada InceptionV3 com 94% em [EI](#), 89% em [VI](#) e 83% em [IRTG](#). Contudo, apesar de precisões gerais elevadas, as redes dos métodos de [EI](#) e [IRTG](#) obtiveram baixas precisões em defeitos específicos, na deteção de manchas espessas e sujidade nos módulos, respetivamente, o que se pode dever à escassez de imagens destes defeitos no treino das redes e/ou à fraca qualidade de algumas imagens. Ainda assim, se se descartarem estas classes, estes métodos continuarão com uma abrangência elevada na deteção de defeitos, visto que em [IRTG](#) são considerados 10 estados (9 defeitos mais o estado sem defeito) e em [EI](#) 8. As redes neuronais dos métodos elétricos ([IVCA](#) e [DDM](#)) são baseadas em [LSTM](#) bidirecional e obtiveram resultados bastante próximos da perfeição. O método [IVCA](#), tendo em conta a sua abrangência e precisão, pode ser considerado o método com melhores resultados (dos 5 implementados): permite identificar 9 estados com uma precisão geral de quase 99%. Esta rede rivaliza com os melhores trabalhos analisados no Estado da Arte (que utilizam [IVCA](#)). O método de [DDM](#), detetando 5 estados, atingiu uma precisão geral de, praticamente, 99%. A rede neuronal

deste método foi construída com a base de dados utilizada em [49] e a sua conceção teve como principal objetivo superar os resultados deste, o que foi manifestamente conseguido visto que a precisão geral obtida em [49] é de 93% e no presente trabalho de 99%.

Por fim, foi concebida uma *Web App* que integrou todos os 5 métodos, o que permite uma vasta abrangência na deteção de defeitos e a possibilidade do utilizador comparar vários resultados (provenientes de tecnologias distintas). Nesta é solicitada a introdução dos dados (consoante o método escolhido) para a classificação do estado do sistema fotovoltaico e, quase imediatamente, são apresentados os resultados da classificação. Assim, os operadores/técnicos responsáveis terão uma ferramenta que os auxiliará na manutenção dos sistemas fotovoltaicos. Se os técnicos, depois de verificarem que a energia produzida não corresponde à esperada e, visual ou analiticamente (dados elétricos), não identificarem o defeito (ou quiserem uma confirmação), terão uma *Web App* bastante abrangente e precisa que responderá imediatamente e a partir de vários métodos (o que permitirá a comparação e confirmação de resultados). Esta ferramenta também poderá servir como base de um sistema de monitorização contínua: sendo que somente necessita dos dados de entrada (mesmo não processados) para detetar os defeitos, se se acoplar à *Web App* um sistema de extração contínua de dados (sejam dados visuais ou elétricos), esta poderá monitorizar, em tempo real, o estado do sistema fotovoltaico. Esta aplicação cumpriu os seus objetivos, propostos para este trabalho, possibilitando ao utilizador a deteção de uma vasta gama de defeitos de uma forma prática, simples e intuitiva.

BIBLIOGRAFIA

- [1] *Fast Lexical Analyser*. URL: <https://www.statista.com/statistics/280220/global-cumulative-installed-solar-pv-capacity/> (ver p. 1).
- [2] M. Ram et al. «European Energy System Based on 100% Renewable Energy – Transport Sector». Em: *Mobilität der Zukunft: Intermodale Verkehrskonzepte*. Ed. por W. Siebenpfeiffer. Berlin, Heidelberg: Springer Berlin Heidelberg, 2021, pp. 519–562. ISBN: 978-3-662-61352-8. DOI: 10.1007/978-3-662-61352-8_36. URL: https://doi.org/10.1007/978-3-662-61352-8_36 (ver p. 1).
- [3] M. Hojabri et al. «IoT-Based PV Array Fault Detection and Classification Using Embedded Supervised Learning Methods». Em: *Energies* 15.6 (2022). ISSN: 1996-1073. DOI: 10.3390/en15062097. URL: <https://www.mdpi.com/1996-1073/15/6/2097> (ver pp. 1, 2, 8, 21, 22).
- [4] Y.-Y. Hong e R. A. Pula. «Methods of photovoltaic fault detection and classification: A review». Em: *Energy Reports* 8 (2022), pp. 5898–5929. ISSN: 2352-4847. DOI: <https://doi.org/10.1016/j.egyrs.2022.04.043>. URL: <https://www.sciencedirect.com/science/article/pii/S2352484722008022> (ver pp. 2, 5).
- [5] S. R. Madeti e S. Singh. «A comprehensive study on different types of faults and detection techniques for solar photovoltaic system». Em: *Solar Energy* 158 (2017), pp. 161–185. ISSN: 0038-092X. DOI: <https://doi.org/10.1016/j.solener.2017.08.069>. URL: <https://www.sciencedirect.com/science/article/pii/S0038092X17307508> (ver pp. 2, 7).
- [6] A. Abubakar, C. F. M. Almeida e M. Gemignani. «Review of Artificial Intelligence-Based Failure Detection and Diagnosis Methods for Solar Photovoltaic Systems». Em: *Machines* 9.12 (2021). ISSN: 2075-1702. DOI: 10.3390/machines9120328. URL: <https://www.mdpi.com/2075-1702/9/12/328> (ver pp. 2, 11, 12).
- [7] G. M. El-Banby et al. «Photovoltaic System Fault Detection Techniques: A Review». Em: *Neural Computing and Applications* 35.35 (2023-12), pp. 24829–24842. ISSN: 1433-3058. DOI: 10.1007/s00521-023-09041-7. URL: <https://doi.org/10.1007/s00521-023-09041-7> (ver p. 5).

- [8] J. Haney e A. Burstein. «PV system operations and Maintenance Fundamentals». Em: (2013). URL: <http://www.solarabcs.org/about/publications/reports/operations-maintenance/pdfs/SolarABCs-35-2013.pdf> (ver p. 6).
- [9] M. Cubukcu e A. Akanalci. «Real-time inspection and determination methods of faults on photovoltaic power systems by thermal imaging in Turkey». Em: *Renewable Energy* 147 (2020), pp. 1231–1238. ISSN: 0960-1481. DOI: <https://doi.org/10.1016/j.renene.2019.09.075>. URL: <https://www.sciencedirect.com/science/article/pii/S0960148119314065> (ver p. 6).
- [10] X. Maldague, F. Galmiche e A. Ziadi. «Advances in pulsed phase thermography». Em: *Infrared Physics and Technology* 43.3 (2002), pp. 175–181. ISSN: 1350-4495. DOI: [https://doi.org/10.1016/S1350-4495\(02\)00138-X](https://doi.org/10.1016/S1350-4495(02)00138-X). URL: <https://www.sciencedirect.com/science/article/pii/S135044950200138X> (ver p. 6).
- [11] T. Kirchartz et al. «Reciprocity between electroluminescence and quantum efficiency used for the characterization of silicon solar cells». Em: *Progress in Photovoltaics: Research and Applications* 17.6 (2009), pp. 394–402. DOI: <https://doi.org/10.1002/pip.895>. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/pip.895>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/pip.895> (ver p. 6).
- [12] G. Alves dos Reis Benatto et al. «Drone-Based Daylight Electroluminescence Imaging of PV Modules». Em: *IEEE Journal of Photovoltaics* 10.3 (2020), pp. 872–877. DOI: [10.1109/JPHOTOV.2020.2978068](https://doi.org/10.1109/JPHOTOV.2020.2978068) (ver p. 6).
- [13] T. Hussain et al. «A Review on Defect Detection of Electroluminescence-Based Photovoltaic Cell Surface Images Using Computer Vision». Em: *Energies* 16.10 (2023). ISSN: 1996-1073. DOI: [10.3390/en16104012](https://doi.org/10.3390/en16104012). URL: <https://www.mdpi.com/1996-1073/16/10/4012> (ver p. 6).
- [14] A. A. Al-Katheri et al. «Application of Artificial Intelligence in PV Fault Detection». Em: *Sustainability* 14.21 (2022). ISSN: 2071-1050. DOI: [10.3390/su142113815](https://doi.org/10.3390/su142113815). URL: <https://www.mdpi.com/2071-1050/14/21/13815> (ver pp. 7, 9, 13, 21, 22).
- [15] M. Alajmi e I. Abdel-Qader. «Fault detection and localization in solar photovoltaic arrays using the current-voltage sensing framework». Em: *2016 IEEE International Conference on Electro Information Technology (EIT)*. 2016, pp. 0307–0312. DOI: [10.1109/EIT.2016.7535257](https://doi.org/10.1109/EIT.2016.7535257) (ver p. 7).
- [16] N. Gokmen et al. «An efficient fault diagnosis method for PV systems based on operating voltage-window». Em: *Energy Conversion and Management* 73 (2013), pp. 350–360. ISSN: 0196-8904. DOI: <https://doi.org/10.1016/j.enconman.2013.05.015>. URL: <https://www.sciencedirect.com/science/article/pii/S0196890413002744> (ver p. 7).

- [17] M. Davarifar, A. Rabhi e A. E. Hajjaji. «Comprehensive Modulation and Classification of Faults and Analysis Their Effect in DC Side of Photovoltaic System». Em: (2013). URL: <http://www.scirp.org/journal/paperinformation.aspx?paperid=37970> (ver p. 7).
- [18] A. Fezzani et al. «Modeling and analysis of the photovoltaic array faults». Em: *2015 3rd International Conference on Control, Engineering and Information Technology (CEIT)*. 2015, pp. 1–9. DOI: 10.1109/CEIT.2015.7232983 (ver pp. 8, 21, 22).
- [19] A. Houssein et al. «Monitoring and fault diagnosis of photovoltaic panels». Em: *2010 IEEE International Energy Conference*. 2010, pp. 389–394. DOI: 10.1109/ENERGYCON.2010.5771711 (ver p. 9).
- [20] J. Solórzano e M. Egido. «Automatic fault diagnosis in PV systems with distributed MPPT». Em: *Energy Conversion and Management* 76 (2013), pp. 925–934. ISSN: 0196-8904. DOI: <https://doi.org/10.1016/j.enconman.2013.08.055>. URL: <https://www.sciencedirect.com/science/article/pii/S019689041300530X> (ver p. 9).
- [21] T. Takashima et al. «Experimental Studies of Failure Detection Methods in PV Module Strings». Em: *2006 IEEE 4th World Conference on Photovoltaic Energy Conference*. Vol. 2. 2006, pp. 2227–2230. DOI: 10.1109/WCPEC.2006.279952 (ver p. 10).
- [22] T. Takashima, J. Yamaguchi e M. Ishida. «Fault detection by signal response in PV module strings». Em: *2008 33rd IEEE Photovoltaic Specialists Conference*. 2008, pp. 1–5. DOI: 10.1109/PVSC.2008.4922843 (ver p. 10).
- [23] T. Takashima et al. «Experimental studies of fault location in PV module strings». Em: *Solar Energy Materials and Solar Cells* 93.6 (2009). 17th International Photovoltaic Science and Engineering Conference, pp. 1079–1082. ISSN: 0927-0248. DOI: <https://doi.org/10.1016/j.solmat.2008.11.060>. URL: <https://www.sciencedirect.com/science/article/pii/S0927024808004583> (ver p. 10).
- [24] M. K. Alam et al. «A Comprehensive Review of Catastrophic Faults in PV Arrays: Types, Detection, and Mitigation Techniques». Em: *IEEE Journal of Photovoltaics* 5.3 (2015), pp. 982–997. DOI: 10.1109/JPHOTOV.2015.2397599 (ver pp. 10, 11).
- [25] W. Chine et al. «A novel fault diagnosis technique for photovoltaic systems based on artificial neural networks». Em: *Renewable Energy* 90 (2016), pp. 501–512. ISSN: 0960-1481. DOI: <https://doi.org/10.1016/j.renene.2016.01.036>. URL: <https://www.sciencedirect.com/science/article/pii/S0960148116300362> (ver p. 13).
- [26] M. Chang et al. «PV O&M OPTIMIZATION BY AI PRACTICE». Em: 2019-09. DOI: 10.4229/EUPVSEC20192019-5B0.5.3 (ver p. 13).

- [27] A. A. Djalab et al. «Robust Method for Diagnosis, Detection of Faults in Photovoltaic Systems Using Artificial Neural Networks». Em: *Periodica Polytechnica Electrical Engineering, Computer Science* 64.3 (2020), 291–302. DOI: 10.3311/PPee.14828. URL: <https://pp.bme.hu/eecs/article/view/14828> (ver p. 13).
- [28] B. Li et al. «Fault diagnosis of photovoltaic panels using full I–V characteristics and machine learning techniques». Em: *Energy Conversion and Management* 248 (2021-11), p. 114785. DOI: 10.1016/j.enconman.2021.114785. URL: <https://hal.science/hal-03415367> (ver pp. 13, 23).
- [29] F. Aziz et al. «A Novel Convolutional Neural Network-Based Approach for Fault Classification in Photovoltaic Arrays». Em: *IEEE Access* 8 (2020), pp. 41889–41904. DOI: 10.1109/ACCESS.2020.2977116 (ver p. 14).
- [30] N. M. Rodrigues, F. M. Janeiro e P. M. Ramos. «Deep Learning for Power Quality Event Detection and Classification Based on Measured Grid Data». Em: *IEEE Transactions on Instrumentation and Measurement* 72 (2023), pp. 1–11. DOI: 10.1109/TIM.2023.3293555 (ver p. 14).
- [31] V. S. B. Kurukuru et al. «A Novel Fault Classification Approach for Photovoltaic Systems». Em: *Energies* 13.2 (2020). ISSN: 1996-1073. DOI: 10.3390/en13020308. URL: <https://www.mdpi.com/1996-1073/13/2/308> (ver p. 15).
- [32] X. Li et al. «Fault diagnostic method for PV array based on improved wavelet neural network algorithm». Em: *Proceeding of the 11th World Congress on Intelligent Control and Automation*. 2014, pp. 1171–1175. DOI: 10.1109/WCICA.2014.7052884 (ver p. 15).
- [33] J. Wang et al. «Fault diagnosis method of photovoltaic array based on support vector machine». Em: *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects* 45.2 (2023), pp. 5380–5395. DOI: 10.1080/15567036.2019.1671557. eprint: <https://doi.org/10.1080/15567036.2019.1671557>. URL: <https://doi.org/10.1080/15567036.2019.1671557> (ver p. 15).
- [34] M. M. Badr et al. «Fault Detection and Diagnosis for Photovoltaic Array Under Grid Connected Using Support Vector Machine». Em: *2019 IEEE Conference on Power Electronics and Renewable Energy (CPERE)*. 2019, pp. 546–553. DOI: 10.1109/CPERE45374.2019.8980103 (ver p. 15).
- [35] A. Eskandari et al. «Autonomous Monitoring of Line-to-Line Faults in Photovoltaic Systems by Feature Selection and Parameter Optimization of Support Vector Machine Using Genetic Algorithms». Em: *Applied Sciences* 10.16 (2020). ISSN: 2076-3417. DOI: 10.3390/app10165527. URL: <https://www.mdpi.com/2076-3417/10/16/5527> (ver p. 15).
- [36] Z. Yi e A. H. Etemadi. «A novel detection algorithm for Line-to-Line faults in Photovoltaic (PV) arrays based on support vector machine (SVM)». Em: *2016 IEEE Power and Energy Society General Meeting (PESGM)*. 2016, pp. 1–4. DOI: 10.1109/PESGM.2016.7742026 (ver p. 15).

- [37] A. Eskandari, J. Milimonfared e M. Aghaei. «Line-line fault detection and classification for photovoltaic systems using ensemble learning model based on I-V characteristics». Em: *Solar Energy* 211 (2020), pp. 354–365. ISSN: 0038-092X. DOI: <https://doi.org/10.1016/j.solener.2020.09.071>. URL: <https://www.sciencedirect.com/science/article/pii/S0038092X20310240> (ver p. 15).
- [38] H.-T. Yau et al. «Cloud Monitoring for Solar Plants with Support Vector Machine Based Fault Detection System». Em: *Mathematical Problems in Engineering* 2014 (2014-07), p. 564517. ISSN: 1024-123X. DOI: [10.1155/2014/564517](https://doi.org/10.1155/2014/564517). URL: <https://doi.org/10.1155/2014/564517> (ver p. 15).
- [39] J. A. Hernández-Pérez et al. «Optimal Parameter Selection for Support Vector Machine Based on Artificial Bee Colony Algorithm: A Case Study of Grid-Connected PV System Power Prediction». Em: *Computational Intelligence and Neuroscience* 2017 (2017-08), p. 7273017. ISSN: 1687-5265. DOI: [10.1155/2017/7273017](https://doi.org/10.1155/2017/7273017). URL: <https://doi.org/10.1155/2017/7273017> (ver p. 15).
- [40] B. Grichting, J. Goette e M. Jacomet. «Cascaded fuzzy logic based arc fault detection in photovoltaic applications». Em: *2015 International Conference on Clean Electrical Power (ICCEP)*. 2015, pp. 178–183. DOI: [10.1109/ICCEP.2015.7177620](https://doi.org/10.1109/ICCEP.2015.7177620) (ver p. 16).
- [41] L. Wang et al. «Arc Fault Detection Algorithm Based on Variational Mode Decomposition and Improved Multi-Scale Fuzzy Entropy». Em: *Energies* 14.14 (2021). ISSN: 1996-1073. DOI: [10.3390/en14144137](https://doi.org/10.3390/en14144137). URL: <https://www.mdpi.com/1996-1073/14/14/4137> (ver p. 16).
- [42] D. N. Murugesan, R. Anitha e M. Ganesan. «N-Semi Regular Graph-Based Fuzzy Semi-supervised Learning Approach for Fault Detection and Classification in Photovoltaic Arrays». Em: *International Journal of Advanced Research in Engineering and Technology* (2020-10). URL: <https://ssrn.com/abstract=3711496> (ver p. 16).
- [43] E. Natsheh e S. Samara. «Tree Search Fuzzy NARX Neural Network Fault Detection Technique for PV Systems with IoT Support». Em: *Electronics* 9.7 (2020). ISSN: 2079-9292. DOI: [10.3390/electronics9071087](https://doi.org/10.3390/electronics9071087). URL: <https://www.mdpi.com/2079-9292/9/7/1087> (ver p. 16).
- [44] P. Ducange et al. «An intelligent system for detecting faults in photovoltaic fields». Em: *2011 11th International Conference on Intelligent Systems Design and Applications*. 2011, pp. 1341–1346. DOI: [10.1109/ISDA.2011.6121846](https://doi.org/10.1109/ISDA.2011.6121846) (ver p. 16).
- [45] B. Du et al. «Intelligent Classification of Silicon Photovoltaic Cell Defects Based on Eddy Current Thermography and Convolution Neural Network». Em: *IEEE Transactions on Industrial Informatics* 16.10 (2020), pp. 6242–6251. DOI: [10.1109/TII.2019.2952261](https://doi.org/10.1109/TII.2019.2952261) (ver p. 16).

- [46] S. Boubaker et al. «Assessment of Machine and Deep Learning Approaches for Fault Diagnosis in Photovoltaic Systems Using Infrared Thermography». Em: *Remote Sensing* 15.6 (2023). ISSN: 2072-4292. DOI: 10.3390/rs15061686. URL: <https://www.mdpi.com/2072-4292/15/6/1686> (ver p. 17).
- [47] C. Mantel et al. «Machine learning prediction of defect types for electroluminescence images of photovoltaic panels». Em: *Applications of Machine Learning*. Ed. por M. E. Zelinski et al. Vol. 11139. International Society for Optics e Photonics. SPIE, 2019, p. 1113904. DOI: 10.1117/12.2528440. URL: <https://doi.org/10.1117/12.2528440> (ver p. 17).
- [48] S. Deitsch et al. «Automatic classification of defective photovoltaic module cells in electroluminescence images». Em: *Solar Energy* 185 (2019), pp. 455–468. ISSN: 0038-092X. DOI: <https://doi.org/10.1016/j.solener.2019.02.067>. URL: <https://www.sciencedirect.com/science/article/pii/S0038092X19302014> (ver p. 18).
- [49] A. E. Lazzaretti et al. «A Monitoring System for Online Fault Detection and Classification in Photovoltaic Plants». Em: *Sensors* 20.17 (2020). ISSN: 1424-8220. DOI: 10.3390/s20174688. URL: <https://www.mdpi.com/1424-8220/20/17/4688> (ver pp. 24, 35, 37, 48).
- [50] *Kaggle*. URL: <https://www.kaggle.com> (ver p. 25).
- [51] M. Millendorf, E. Obropta e N. Vadhavkar. «Infrared solar module dataset for anomaly detection». Em: *Proc. Int. Conf. Learn. Represent.* 2020. URL: <https://ai4earthscience.github.io/iclr-2020-workshop/papers/ai4earth22.pdf> (ver p. 27).
- [52] M. Millendorf, E. Obropta e N. Vadhavkar. *Infrared Solar Modules*. 2023. DOI: 10.34740/KAGGLE/DSV/6866566. URL: <https://www.kaggle.com/dsv/6866566> (ver p. 27).
- [53] *Solar Panel Images Clean and Faulty Images*. URL: <https://www.kaggle.com/datasets/pythonafroz/solar-panel-images/data> (ver p. 27).
- [54] B. Su, Z. Zhou e H. Chen. «PVEL-AD: A Large-Scale Open-World Dataset for Photovoltaic Cell Anomaly Detection». Em: *IEEE Transactions on Industrial Informatics* 19.1 (2023), pp. 404–413. DOI: 10.1109/TII.2022.3162846 (ver p. 28).



