



**DANIELLE ABREU FORTUNATO**

**MODELAGEM GRANULAR CONVOLUCIONAL EVOLUTIVA  
PARA CLASSIFICAÇÃO DE FLUXO DE IMAGENS**

**LAVRAS – MG**

**2024**

**DANIELLE ABREU FORTUNATO**

**MODELAGEM GRANULAR CONVOLUCIONAL EVOLUTIVA PARA  
CLASSIFICAÇÃO DE FLUXO DE IMAGENS**

Dissertação apresentada à Universidade Federal de Lavras, como parte das exigências do Programa de Pós-Graduação em Engenharia de Sistemas e Automação, área de concentração em Sistemas Inteligentes, para a obtenção do título de Mestre.

Profa. DSc. Sílvia Costa Ferreira  
Orientadora

Prof. DSc. Daniel Furtado Leite  
Coorientador

Prof. DSc. Patrícia Ferreira Ponciano Ferraz  
Coorientadora

**LAVRAS – MG**

**2024**

**Ficha catalográfica elaborada pelo Sistema de Geração de Ficha Catalográfica da Biblioteca  
Universitária da UFLA, com dados informados pelo(a) próprio(a) autor(a).**

Fortunato, Danielle Abreu

Modelagem Granular Convolutacional Evolutiva para Classificação de Fluxo de Imagens / Danielle Abreu Fortunato. – Lavras : UFLA, 2024.

68 p. : il.

Orientador(a): Profa. DSc. Sílvia Costa Ferreira.

Coorientador(a): Prof. DSc. Daniel Furtado Leite.

Dissertação (mestrado acadêmico) –Universidade Federal de Lavras, 2024.

Bibliografia.

1. Visão Computacional. 2. Reconhecimento de Imagens. 3. Sistemas Inteligentes Evolutivos. I. Ferreira, Silvia Costa. II. Leite, Daniel Furtado. III. Ferraz, Patrícia Ferreira Ponciano. IV. Título.

**DANIELLE ABREU FORTUNATO**

**MODELAGEM GRANULAR CONVOLUCIONAL EVOLUTIVA PARA  
CLASSIFICAÇÃO DE FLUXO DE IMAGENS**

**EVOLVING CONVOLUTIONAL GRANULAR MODELING FOR IMAGE STREAM  
CLASSIFICATION**

Dissertação apresentada à Universidade Federal de Lavras, como parte das exigências do Programa de Pós-Graduação em Engenharia de Sistemas e Automação, área de concentração em Sistemas Inteligentes, para a obtenção do título de Mestre.

APROVADA em 04 de Março de 2024.

Prof. DSc. Daniel Furtado Leite	Paderborn University, Germany
Prof. DSc. Danilo Alves de Lima	UFLA
Profa. DSc. Sílvia Costa Ferreira	UFLA
Profa. DSc. Tatiane Carvalho Alvarenga	UEM

Profa. DSc. Sílvia Costa Ferreira  
Orientadora

Prof. DSc. Daniel Furtado Leite  
Co-Orientador

Prof. DSc. Patrícia Ferreira Ponciano Ferraz  
Co-Orientadora

**LAVRAS – MG  
2024**

*Dedico este trabalho a Deus, fonte de toda sabedoria e inspiração. Agradeço por guiar meus passos ao longo desta jornada acadêmica, concedendo-me força, discernimento e perseverança. Que este trabalho possa refletir não apenas meu esforço, mas também Sua graça e orientação constante. Em cada página escrita, reconheço Sua presença e agradeço por todas as bênçãos recebidas. Que seja feita a Sua vontade em tudo o que faço. Amém.*

## **AGRADECIMENTOS**

Agradeço a Deus e a Nossa Senhora Aparecida por me darem forças e me permitirem chegar até aqui.

Aos professores Dra. Sílvia Costa Ferreira e Dr. Daniel Furtado Leite, pelo apoio e conhecimentos compartilhados.

À toda minha família, meus pais, meu irmão e meu avô pelo suporte e base fundamental na minha vida.

A minha avó Ana da Costa Lima (in memorian), por ter me ensinado o valor de se fazer as coisas da melhor maneira possível e por me olhar de seu lugar.

Aos meus amigos Luciana e Rafael pela amizade, tempo e paciência dedicados a mim.

À Universidade Federal de Lavras (UFLA) e ao Programa de Pós-Graduação em Engenharia de Sistemas e Automação (PPGESISA), pela oportunidade concedida para a realização do mestrado.

Aos colegas do PPGESISA que sempre estiveram ao meu lado nos momentos de estudo e apresentação de trabalhos.

Aos professores do PPGESISA, pelas oportunidades concedidas e ensinamentos compartilhados.

*Saiba que, aos olhos de Deus, nada é pequeno se tudo o que fizermos for feito por amor.  
(Santa Teresinha do Menino Jesus)*

## RESUMO

Os avanços recentes em aprendizado de máquina para visão computacional e classificação de imagens trouxeram, dentre outros desafios, os seguintes: (i) a explicabilidade ou interpretabilidade do modelo neural profundo para classificação; e (ii) a capacidade de aprendizado contínuo *online* do modelo após sua implementação em um ambiente dinâmico, conforme observado em um fluxo de imagens. Neste trabalho, é apresentada uma estrutura de Rede Neural Granular Evolutiva Convolutiva que se destina a avançar a compreensão e aplicação de aprendizado de máquina em visão computacional, especificamente no reconhecimento e classificação de imagens. A rede é equipada com um algoritmo incremental, que aborda ambas as questões (i) e (ii), proporcionando um nível mais elevado de interpretabilidade ao modelo neural e permitindo aprendizado contínuo ao longo da vida. A modelagem proposta, denominada *Convolutional Evolutionary Granular Neural Network* (CEGNN) combina parte de uma Rede Neural Convolutiva (CNN) denominada VGG-16 com uma rede granular evolutiva (EGNN). A estrutura conexionista e os parâmetros de grânulos de informação da EGNN são gradualmente desenvolvidos e atualizados com base na análise dos componentes principais (PCAs) de variáveis latentes que podem representar características que não são diretamente observáveis, como bordas, texturas, formas ou objetos, extraídas do fluxo de imagens. Em particular, a CNN VGG-16 é explorada para gerar um espaço característico compacto, o qual refere-se a uma representação de características de dados em um espaço de menor dimensionalidade que preserva as informações relevantes para uma tarefa específica, como classificação de imagens, enquanto a EGNN, composta por grânulos fuzzy trapezoidais e funções de agregação T-norma e S-norma, é utilizada para capturar padrões e classificar imagens. A técnica de Análise de Componentes Principais (PCA) é implementada no ponto de integração entre a VGG-EGNN, visando representar as características abstratas que influenciam os dados observados, reduzir o tempo de processamento dos dados e do treinamento *online*. Essa abordagem não só permite lidar eficientemente com imagens ou *frames* de vídeo em frequências relativamente mais elevadas, mas também destaca que a precisão e a interpretabilidade do modelo global são aprimoradas pela reconfiguração das conexões resultantes da transformação PCA no espaço latente. Isso é possível, pois ao se reduzir a dimensionalidade dos dados, a perda de informação é minimizada. Os resultados obtidos indicam que o modelo CEGNN é eficiente e interpretável na tarefa de classificação de imagens em dez classes distintas, apresentando uma acurácia de 78,88% e uma precisão de 0,79 na classificação das imagens. Esses resultados destacam a eficácia da abordagem proposta em lidar com a complexidade das tarefas de classificação ressaltando sua viabilidade e relevância em diversas aplicações práticas, como análise de imagens do cérebro, radiológicas, de satélite, de robôs móveis e veículos autônomos, entre outras.

**Palavras-chave:** visão computacional; reconhecimento de imagens; sistemas inteligentes evolutivos; aprendizado profundo; computação granular.



## ABSTRACT

Recent advances in machine learning for computer vision and image classification emphasize two main aspects: (i) the explainability or interpretability of deep neural models for classification; and (ii) the ability for continuous online learning of the model after its deployment in a dynamic environment, as observed in a stream of images. In this work, we present a framework of Convolutional Evolving Granular Neural Network aimed at advancing the understanding and application of machine learning in computer vision, specifically in image recognition and classification. The network is equipped with an incremental algorithm, which addresses both issues (i) and (ii), providing a higher level of interpretability to the neural model and enabling lifelong continuous learning. The proposed modeling, named Convolutional Evolving Granular Neural Network (CEGNN), combines part of a Convolutional Neural Network (CNN) called VGG-16 with an evolving granular network (EGNN). The connectionist structure and the information granule parameters of EGNN are gradually developed and updated based on the analysis of principal components (PCAs) of latent variables that may represent features that are not directly observable, such as edges, textures, shapes, or objects, extracted from the stream of images. In particular, the VGG-16 CNN is exploited to generate a compact feature space, which refers to a representation of data features in a lower-dimensional space that preserves relevant information for a specific task, such as image classification, while the EGNN, composed of trapezoidal fuzzy granules and T-norm and S-norm aggregation functions, is used to capture patterns and classify images. The Principal Component Analysis (PCA) technique is implemented at the integration point between VGG-EGNN, aiming to represent the abstract features that influence the observed data, reducing data processing and online training time. This approach not only allows for efficient handling of images or video frames at relatively higher frequencies but also highlights that the accuracy and interpretability of the global model are enhanced by the re-configuration of connections resulting from PCA transformation in the latent space. This is possible because by reducing the dimensionality of the data, information loss is minimized. The results obtained indicate that the CEGNN model is efficient and interpretable in the task of classifying images into ten distinct classes, achieving an accuracy of 78.88% and a precision of 0,79 in image classification. These results highlight the effectiveness of the proposed approach in dealing with the complexity of classification tasks, emphasizing its viability and relevance in various practical applications, such as analysis of brain images, radiological images, satellite images, mobile robots, and autonomous vehicles, among others.

**Keywords:** computer vision; image recognition; evolving intelligent systems; Deep learning; granular computing.

## **INDICADORES DE IMPACTO**

O trabalho contribui para o desenvolvimento de tecnologias mais explicáveis e interpretáveis, aumentando a confiança em sistemas de IA em áreas como diagnósticos médicos, segurança, robótica, veículos autônomos e monitoramento ambiental. Nessas áreas, a rápida adaptação a novos dados é crucial. Economicamente, a melhoria na eficiência e precisão dos sistemas de classificação de imagens pode reduzir custos operacionais e aumentar a competitividade das empresas que adotam essas tecnologias. Culturalmente, ao promover métodos explicáveis de aprendizado de máquina, o trabalho facilita a aceitação e compreensão da IA na sociedade. A pesquisa envolve colaboração com áreas como comunicação, tecnologia e produção, alinhando-se com os Objetivos de Desenvolvimento Sustentável (ODS) da ONU, especialmente em Saúde e Bem-estar, e Indústria, Inovação e Infraestrutura. A divulgação dos resultados foi realizada por meio de resumos em anais de congressos científicos e produção de artigos. Os resultados foram apresentados no XXXI Congresso de Pós-Graduação e submetidos ao XXV Congresso Brasileiro de Automática, aguardando resposta.

## **IMPACT INDICATORS**

The work contributes to the development of more explainable and interpretable technologies, increasing trust in AI systems in areas such as medical diagnostics, security, robotics, autonomous vehicles, and environmental monitoring. In these areas, the rapid adaptation to new data is crucial. Economically, improving the efficiency and accuracy of image classification systems can reduce operational costs and increase the competitiveness of companies that adopt these technologies. Culturally, by promoting explainable machine learning methods, the work facilitates the acceptance and understanding of AI in society. The research involves collaboration with fields such as communication, technology, and production, aligning with the United Nations' Sustainable Development Goals (SDGs), especially in Health and Well-being, and Industry, Innovation, and Infrastructure. The dissemination of results was carried out through abstracts in conference proceedings and the production of articles. The results were presented at the XXXI Postgraduate Congress and submitted to the XXV Brazilian Congress of Automation, awaiting a response.

## LISTA DE FIGURAS

Figura 2.1 – Esquema de um sistema de visão computacional. . . . .	18
Figura 2.2 – Exemplo de CNN para classificação do dígito manuscrito zero. . . . .	19
Figura 2.3 – Diagrama da estrutura da VGG-16. . . . .	24
Figura 2.4 – Projeto da rede neural granular evolutiva EGNN ou GNN. . . . .	31
Figura 2.5 – EGNN: Evolução da arquitetura de rede neuro-fuzzy para classificação. . .	31
Figura 2.6 – Modelo de neurônio difuso. . . . .	32
Figura 3.1 – Exemplos de imagens da base de dados <i>Cats-vs-Dogs</i> . . . . .	40
Figura 3.2 – Exemplos de imagens da base de dados CODaN. . . . .	41
Figura 3.3 – Sistema de monitoramento convolucional granular evolutivo. . . . .	42
Figura 3.4 – Diagrama da estrutura do modelo CEGNN proposta. . . . .	45
Figura 3.5 – Operação de matrizes (imagem x kernel). . . . .	46
Figura 3.6 – Ilustração de uma curva ROC. . . . .	49
Figura 4.1 – Métricas de avaliação do modelo para duas classes. . . . .	51
Figura 4.2 – Matriz de confusão para a classificação de duas classes. . . . .	52
Figura 4.3 – Ilustração da curva ROC na classificação de duas classes. . . . .	52
Figura 4.4 – Métricas de avaliação do modelo para dez classes. . . . .	54
Figura 4.5 – Matriz de confusão para a classificação de dez classes. . . . .	55

## LISTA DE TABELAS

Tabela 2.1 – Estrutura de uma arquitetura VGG-16 . . . . .	25
Tabela 2.2 – Organização de um conjunto de dados com $n$ tratamentos, $p$ variáveis e $k$ componentes. . . . .	28
Tabela 2.3 – Resultante das componentes principais para $n$ tratamentos. . . . .	28
Tabela 4.1 – CEGNN: Desempenho de classificação de imagens para duas classes . . . .	50
Tabela 4.2 – CEGNN: Desempenho de classificação de imagens para dez classes . . . .	53
Tabela 4.3 – Desempenho de classificação de imagens para os modelos analisados. . . .	59

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	13
<b>1.1</b>	<b>Objetivos</b>	15
<b>1.2</b>	<b>Contribuições</b>	15
<b>1.3</b>	<b>Estrutura do Texto</b>	15
<b>2</b>	<b>INTELIGÊNCIA COMPUTACIONAL EM VISÃO COMPUTACIONAL</b>	17
<b>2.1</b>	<b>Redes Neurais Convolucionais</b>	19
<b>2.2</b>	<b>Explorando Abordagens Alternativas: Revisão da literatura</b>	21
<b>2.3</b>	<b>Rede VGG Convolutacional e Dados ImageNet</b>	23
<b>2.4</b>	<b>Análise de Componentes Principais</b>	25
<b>2.5</b>	<b>Sistemas Inteligentes Evolutivos</b>	28
<b>2.6</b>	<b>Rede Neural Granular Evolutiva EGNN para Classificação de Padrões</b>	30
<b>2.7</b>	<b>Síntese</b>	38
<b>3</b>	<b>REDE NEURAL GRANULAR CONVOLUCIONAL EVOLUTIVA</b>	39
<b>3.1</b>	<b>Detalhes sobre os Conjuntos de Dados a serem Aprendidos</b>	39
<b>3.2</b>	<b>Proposta para Classificação de Imagens</b>	41
<b>3.3</b>	<b>Evolução de Grânulos de Informação a partir do Fluxo de Imagens</b>	43
<b>3.4</b>	<b>Métricas para Avaliação dos Resultados</b>	48
<b>3.5</b>	<b>Síntese</b>	49
<b>4</b>	<b>RESULTADOS</b>	50
<b>4.1</b>	<b>Desempenho na Classificação de duas Classes</b>	50
<b>4.2</b>	<b>Desempenho na Classificação de dez Classes</b>	53
<b>4.3</b>	<b>Discussões sobre Interpretabilidade</b>	57
<b>4.4</b>	<b>Comparação com outros modelos</b>	58
<b>4.5</b>	<b>Síntese</b>	60
<b>5</b>	<b>CONCLUSÃO</b>	61
	<b>REFERÊNCIAS</b>	62

## 1 INTRODUÇÃO

Os recentes desenvolvimentos nas áreas de reconhecimento de padrões e análise de imagens têm desempenhado um papel importante em diversas aplicações, abrangendo setores e domínios do conhecimento. Esse cenário de progressos tecnológicos assume relevância significativa em várias áreas, incluindo indústrias, serviços, pesquisa científica e no suporte à economia global. A busca contínua pela melhoria da produtividade é uma demanda comum em diversos setores, impulsionada pela competitividade do mercado e pelas demandas dos consumidores por produtos de alta qualidade e eficiência. Técnicas fundamentadas em visão computacional e aprendizado de máquina têm sido cruciais em contextos que vão desde a detecção de problemas específicos até a análise de recursos naturais, meteorologia, imagens biomédicas e metalografia, entre outros (Li, 2023).

Esses avanços têm revolucionado a forma como são abordadas questões fundamentais em vários campos. Na agricultura, os algoritmos são capazes de identificar padrões em imagens de plantações, permitindo a detecção precoce de doenças ou infestações de pragas (Cardoso *et al.*, 2020). Já na meteorologia, a interpretação de imagens de satélite são realizadas por meio de algoritmos avançados que possibilitam uma compreensão mais precisa dos padrões climáticos e uma previsão mais confiável de eventos extremos (Bhardwaj *et al.*, 2019). Em imagens biomédicas, como exames de ressonância magnética e tomografia computadorizada, os algoritmos podem identificar padrões associados a condições médicas complexas, auxiliando no diagnóstico e no planejamento do tratamento (Litjens *et al.*, 2017). Além disso, na metalografia, a análise de imagens de microestruturas metálicas permite uma avaliação detalhada da qualidade dos materiais e a detecção de defeitos, garantindo a segurança e a confiabilidade dos produtos (Rodrigues *et al.*, 2017). Na agropecuária é comum desafios constantes na gestão de recursos, controle de pragas e otimização da produção para atender às demandas crescentes por alimentos de qualidade (Redivo *et al.*, 2012). Em resumo, a aplicação desses algoritmos está transformando profundamente a maneira como são exploradas e utilizadas informações visuais em uma ampla gama de áreas e setores.

Diante dessa evolução, é crucial o desenvolvimento contínuo de algoritmos destinados a aprimorar a visão computacional em várias aplicações. Esses algoritmos desempenham um papel essencial na interpretação de informações visuais e na extração de padrões relevantes em uma ampla gama de cenários. Muitos desses são compostos por técnicas de aprendizado de máquina, incluindo as redes neurais profundas. Dada sua complexidade, técnicas de otimização têm sido empregadas para melhorar a convergência paramétrica e estabilidade no treinamento dessas redes. Métodos como Mini-Batch SGD, Accelerated Nesterov, AdaGrad, AdaBoost e Adam têm desempenhado um papel essencial, permitindo a aplicação dessas técnicas em cenários que envolvem grandes e diversificadas bases de imagens (Koushik; Srinivasa, 2021; Jaraa; Bowenb, 2022). O uso crescente de aprendizado de máquina e redes neurais profundas, especialmente no âmbito do reconhecimento de padrões em imagens, voz e texto, tem se destacado especialmente diante da necessidade de lidar com volumes massivos de dados (Verne *et al.*, 2022).

No entanto, apesar dos progressos significativos na área de visão computacional e aprendizado de máquina, nota-se uma lacuna na literatura quanto à importância de dois aspectos fundamentais: a interpretabilidade dos modelos de aprendizado de máquina e sua capacidade de aprendizado contínuo em ambientes dinâmicos. Quando se referindo à interpretabilidade, trata-se da capacidade de compreender e explicar o funcionamento interno dos modelos de aprendizado de máquina, de entender o porquê e como eles tomam determinadas decisões (Leal-Lara; Barón-Velandia; Rocha-Calderón, 2021). Isso se torna especialmente relevante em aplicações críticas, onde a transparência nos processos decisórios é essencial. Ademais, a capacidade de

aprendizado contínuo *online* é vital em cenários dinâmicos, como fluxos de imagens em tempo real, onde os modelos precisam se adaptar a mudanças rápidas e frequentes.

Embora diversas estruturas redes tenham sido propostas para a classificação de imagens digitais, como as Redes Neurais Convolucionais (CNNs, *Convolutional Neural network*) (Godfellow, 2016; Calin, 2021), poucos foram projetados para lidar efetivamente com fluxos de imagens em tempo real. Mantendo a transparência nos processos decisórios, destacando-se então a necessidade urgente de desenvolver métodos inovadores que combinem interpretabilidade com capacidade de adaptação dinâmica, proporcionando confiabilidade e eficácia em ambientes em constante mudança.

Diante do contexto apresentado, este trabalho propõe uma metodologia inovadora denominada Rede Neural Granular Evolutiva Convolucional (CEGNN, *Convolutional Evolving Granular Neural Network*) para lidar com desafios relacionados à classificação de imagens digitais. A abordagem concentra-se na análise de imagens provenientes de diversas áreas do conhecimento, visando atingir uma classificação acurada e eficiente. Para atingir o objetivo proposto, adotou-se uma estratégia que incorpora elementos de uma Rede Neural Convolucional (CNN) VGG-16, uma arquitetura de rede neural convolucional amplamente reconhecida e utilizada, especialmente no contexto de visão computacional e classificação de imagens. Ela se destaca por sua profundidade, consistindo em 13 camadas de convolução e *pooling*, seguidas por três camadas totalmente conectadas para a classificação final. A utilização da VGG-16 permite a extração de atributos essenciais das imagens, contribuindo para a eficácia da classificação (Kong; Cheng, 2022). Essa arquitetura é então integrada a uma rede neural granular evolutiva (EGNN) (Leite; Jr; Gomide, 2013). A rede EGNN têm sido empregada com sucesso em uma variedade de tarefas de reconhecimento de padrões. A rede é caracterizada pela sua capacidade de aprendizado evolutivo e pela representação granular dos dados, permitindo uma adaptação dinâmica às mudanças no ambiente.

Na CEGNN, foram consideradas as primeiras camadas da rede VGG-16, até a camada denominada Dense1, com a remoção da camada final densa e estática. A camada Dense1, também conhecida como camada densa, é uma camada totalmente conectada em uma rede neural convolucional como a VGG-16. Essa camada recebe como entrada os recursos extraídos pelas camadas convolucionais anteriores e os processa por meio de unidades neurais totalmente conectadas. Essas unidades realizam operações de multiplicação de matrizes para mapear os dados que recebem em outro espaço, até que seja proferida a classificação final. A fim de conferir interpretabilidade ao modelo e possibilitar a adaptação temporal *online* por meio da EGNN, foi realizada uma Análise de Componentes Principais (PCA, *Principal Component Analysis*) nos recursos extraídos pelas 13 primeiras camadas da rede VGG-16. Essa análise permite uma compreensão mais profunda das características discriminativas aprendidas pelo modelo em diferentes níveis de abstração. A utilização do PCA entre a VGG-16 e a EGNN possibilita uma comparação direta das representações latentes geradas por ambas as redes, contribuindo para uma interpretação mais completa do processo de classificação e identificação em fluxos de imagens. A integração proposta não apenas busca alcançar resultados precisos de classificação, mas também visa conferir interpretabilidade ao modelo, já que EGNN particiona o espaço dos dados e estabelece regras linguísticas que associam grânulos de entrada e de saída. Além disso, EGNN permite a adaptação temporal *online*, possibilitando a preservação da acurácia e interpretabilidade do modelo em ambientes dinâmicos e contextos diversos.

## 1.1 Objetivos

O objetivo deste trabalho é apresentar uma modelagem inteligente convolucional evolutiva como solução para o problema de classificação e identificação em fluxos de imagens digitais, que seja adaptativa e capaz de lidar com diferentes contextos visuais. Para isso, os objetivos específicos são:

1. Desenvolver um modelo de Rede Neural Granular Evolutiva Convolucional (CEGNN, *Convolutional Evolving Granular Neural Network*) capaz de processar fluxos contínuos de imagens e realizar classificação em ambiente dinâmicos;
2. Implementar a classificação de imagens em tempo real, utilizando a integração de uma rede VGG-16 pré-treinada com uma eGNN de aprendizado incremental.

## 1.2 Contribuições

O reconhecimento de imagens é uma subárea crucial da visão computacional, um domínio que desempenha um papel fundamental em uma ampla gama de aplicações em diversos campos do conhecimento. Uma única imagem pode conter uma quantidade significativa de informação, representada pela organização e intensidade dos pixels. Para lidar com essa complexidade, modelos equipados com técnicas de aprendizado de máquina contínuo e de rápida adaptação têm despertado interesse nesta área. O processamento digital de imagens emergiu como um campo de estudo atrativo para pesquisadores ao longo das últimas décadas, e seu crescimento tem sido notável. No entanto, mesmo com os avanços alcançados, o desafio persiste em tornar as máquinas capazes de realizar tarefas de forma análoga ao córtex visual humano, especialmente no que diz respeito ao processamento e identificação de padrões em imagens.

Este trabalho demonstra a eficácia da abordagem da modelagem proposta, que combina uma rede convolucional com uma rede neural granular evolutiva. Com esse modelo evolutivo profundo é possível aprender de maneira contínua, adaptando-se a novos padrões espaço-temporais identificados em fluxos de imagens. O estudo busca provar a capacidade do modelo em lidar com a classificação de fluxos de imagens de forma eficiente e precisa, demonstrando sua utilidade em aplicações práticas de reconhecimento visual.

A proposta oferece contribuições para áreas como visão computacional, sistemas inteligentes evolutivos e classificação de imagens em múltiplas aplicações. Dessa forma, pode-se afirmar que a contribuição principal deste trabalho reside na introdução de uma abordagem inovadora para a classificação de fluxo de imagens, que combina uma rede convolucional com uma rede neural granular evolutiva. Trata-se de uma das primeiras investigações na literatura a integrar essas duas técnicas de modelagem.

## 1.3 Estrutura do Texto

O capítulo introdutório destaca os avanços recentes nas áreas de reconhecimento de padrões e análise de imagens, ressaltando sua importância em setores cruciais como a agropecuária e as indústrias em geral. O Capítulo 2 apresenta os conceitos básicos de visão computacional, redes neurais convolucionais, e sistemas inteligentes evolutivos. O estado da arte é apresentado.

No Capítulo 3 é apresentada a metodologia. Nesse capítulo há uma explicação sobre a abordagem para extração de atributos utilizando a VGG-16, sobre a composição da base de dados utilizada para os testes iniciais, e sobre o classificador evolutivo EGNN. É apresentada



a arquitetura proposta para processar dados pelas camadas da rede neural convolucional e pela rede neural granular evolutiva.

No Capítulo 4 são apresentados os experimentos e resultados obtidos a partir da combinação da rede VGG-16 e de EGNN para classificação imagens. Ele apresenta, ainda, métricas propostas para avaliações de desempenho, acurácia e complexidade, considerando o método proposto e outros métodos da literatura da área.

No Capítulo 5, são apresentadas as conclusões derivadas dos resultados específicos obtidos em relação à classificação de imagens. Além disso, são discutidas as percepções do trabalho e são apresentadas perspectivas futuras para possíveis desenvolvimentos ou experimentos adicionais.

## 2 INTELIGÊNCIA COMPUTACIONAL EM VISÃO COMPUTACIONAL

Em diversas tarefas do cotidiano, a visão humana é essencial para capturar informações importantes do ambiente. Apesar de ser uma habilidade básica para os seres humanos, o mesmo mecanismo de processamento de imagem torna-se complexo quando realizado por uma máquina (Rautaray; Agrawal, 2012). A visão computacional procura auxiliar na resolução de problemas complexos, simulando a visão humana, capturando, processando e analisando imagens. De modo geral, visão computacional pode ser definida como uma subárea do processamento de imagens que estuda o desenvolvimento de métodos e técnicas que possibilitam um sistema computacional interpretar imagens. Um sistema de visão computacional tem como objetivo dotar uma máquina com capacidade para descrever a essência de uma imagem (Gonzalez; Woods, 2000; Lima *et al.*, 2021).

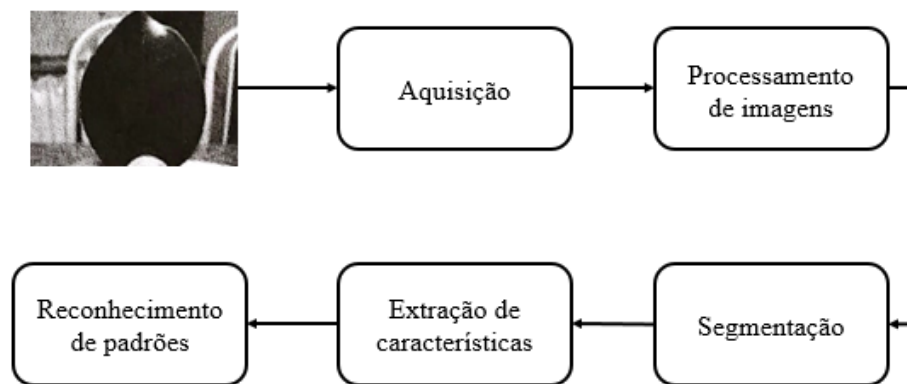
Conforme as tecnologias evoluem, diversas áreas têm se beneficiado da utilização de processamento de imagens via inteligência artificial. A visão computacional tem suas aplicações presentes em diversos segmentos que envolvem a análise de imagens, como inspeção industrial e controle de qualidade, segurança, reconhecimento facial, reconhecimento de gestos, aplicações espaciais, análise de imagens médicas, veículos autônomos e agro-negócio. Ademais, visão computacional já se encontra presente em áreas como agronomia, biologia, medicina, segurança e educação (Alvares; Souza; Castro, 2020; Barelli, 2018; Hashimoto *et al.*, 2019; Lima *et al.*, 2021; Neves; Neto; Gonzaga, 2021; Teixeira; Silva, 2020).

A pesquisa em visão computacional se caracteriza por uma grande variedade de algoritmos de alto desempenho designados para problemas específicos. Entretanto, se tratando de algoritmos de âmbito genérico, ainda há um longo caminho para que surjam propostas que se igualem à capacidade da visão biológica. Portanto, a visão computacional atua de maneira conjunta com o processamento de imagens. Pode-se pensar em um sistema de visão computacional como constituído de várias fases (Backes; Junior, 2016). São elas:

- a) **Aquisição:** a aquisição é o primeiro passo no processamento digital de imagens. Ela é geralmente composta por uma câmera digital fotográfica ou de vídeo que captura a imagem real e a transforma em uma imagem digital. Dependendo do dispositivo utilizado para a aquisição da imagem, esta pode variar entre bidimensional ou tridimensional;
- b) **Processamento de imagens:** o processamento da imagem visa melhorar a imagem, de maneira a atenuar ou suavizar algumas das características, como contraste ou ruídos. Essa etapa é realizada conforme a necessidade específica de cada aplicação;
- c) **Segmentação:** nessa etapa, divide-se a imagem nos objetos que a compõem, selecionando assim as partes que interessam nela (Gonzalez; Woods, 2000). A etapa da segmentação é considerada uma das mais importantes do processamento, pois é nela que são definidas quais serão as áreas e os objetos utilizados e analisados nas próximas etapas;
- d) **Extração de características/Análise de imagens:** etapa responsável por obter um conjunto de características do objeto de interesse que permita identificá-lo, possibilitando uma análise detalhada e eficiente para aplicações em campos como reconhecimento de padrões, visão computacional e aprendizado de máquina;
- e) **Reconhecimento de padrões:** a última etapa de processamento atribui sentido aos resultados. São analisadas algumas imagens, padrões perceptíveis ao olho humano, e as classes estimadas por um modelo computacional. É responsável por classificar ou agrupar as imagens com base em seus conjuntos de características.

É importante notar que as fases de um sistema de visão computacional não possuem uma descrição universal. Autores podem utilizar terminologias distintas, como chamar a etapa que foi denominada "processamento de imagens" de "pré-processamento" e agrupar todas as cinco fases sob o termo "processamento de imagens". A quantidade de fases também varia conforme a abordagem dos pesquisadores. Algumas fases podem ser eliminadas de acordo com a natureza do problema. Por exemplo, uma imagem pode seguir diretamente para a extração de características sem passar pela segmentação, seja por uma segmentação manual em casos mais desafiadores ou porque a imagem original já representa completamente a região de interesse para análise (Backes; Junior, 2016). Na Figura 2.1 é apresentado o esquema de um sistema de visão computacional. Note que as 5 etapas descritas acima estão representadas em blocos no diagrama.

Figura 2.1 – Esquema de um sistema de visão computacional.



Fonte: Backes e Junior (2016).

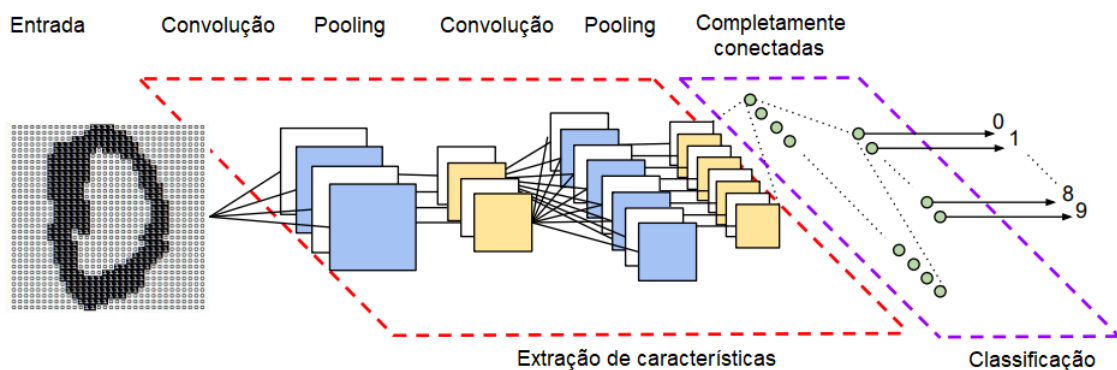
## 2.1 Redes Neurais Convolucionais

A crescente popularidade das abordagens que empregam técnicas de aprendizado profundo tem sido notável ao longo dos anos. Essa tendência resultou no surgimento de aplicações significativas em diversas subáreas do processamento de imagens e visão computacional (Bejnordi *et al.*, 2018; Ronneberger; Fischer; Brox, 2015). Em meio à vasta gama de técnicas de aprendizado de máquina, as redes neurais convolucionais (CNNs) destacam-se como o modelo mais bem-sucedido. Suas camadas convolucionais, dispostas de forma sequencial, aplicam, uma variedade de filtros, produzindo, assim, mapas característicos em espaços abstratos (Lei; Chen; Zhao, 2018).

No entanto, ao abordar a complexidade do panorama atual de técnicas de aprendizado de máquina, é crucial expandir a discussão além das CNNs. Uma análise aprofundada envolve não apenas as redes neurais convolucionais, mas também outras abordagens inovadoras, como autoencoders (AEs), autoencoders variacionais, autoencoders desacoplados, técnicas de desruído e *Generative Adversarial Networks* (GANs) (Cremona; Uzal; Pire, 2022; Ye; Bors, 2022). Essas técnicas, muitas vezes negligenciadas, desempenham papéis fundamentais em diversos contextos, justificando uma compreensão abrangente do cenário atual para embasar a escolha e avaliação de modelos em aplicações específicas.

Uma CNN é uma variação da rede *feed-forward* de múltiplas camadas, tendo sido inspirada no processamento de imagens do córtex visual (Vargas; Paes; Vasconcelos, 2016). De maneira semelhante aos processos tradicionais de visão computacional, uma CNN aplica filtros convolucionais em pixels, mantendo a relação de vizinhança entre os pixels. As CNNs constituem um dos primeiros casos de sucesso de aprendizado profundo, ganhando bastante visibilidade em competições de classificação de imagens que acontecem desde 2011 (Aggarwal, 2019). A Figura 2.2 mostra a arquitetura de uma CNN simples, com apenas duas camadas de filtros convolucionais para classificação do dígito manuscrito 0 de uma base de imagens MNIST.

Figura 2.2 – Exemplo de CNN para classificação do dígito manuscrito zero.



Fonte: Vargas, Paes e Vasconcelos (2016).

Em seguida são detalhados os elementos de uma CNN típica.

- a) **Camadas convolucionais** - As camadas convolucionais contém uma série de filtros (também chamados de *kernels*) que realizam a etapa de convolução em uma dada entrada e geram um conjunto de mapas de características (do inglês, *feature maps*) ou mapas de ativação (do inglês, *activation maps*) (Khan *et al.*, 2018). Um filtro de uma camada de convolução é um *grid* de números discretos (pesos), os quais são aprendidos durante a etapa de treinamento da CNN. A operação de convolução geralmente produz uma redu-

ção do tamanho da imagem na camada  $q + 1$ , se comparado com o tamanho da mesma na camada  $q$  anterior;

- b) Camada de pooling (*Pooling layer*)** - A operação de *pooling* é realizada em *grids* de tamanho  $P_q \times P_q$  em cada camada, e produz uma camada de mesma profundidade, diferentemente dos filtros. Para cada região quadrada de dimensões  $P_q \times P_q$  de cada *activation map*  $d_q$  é retornado o valor máximo desse *grid*. A esse tipo de operação de *pooling* dá-se o nome de *maxpooling*. Assim, por meio dessa operação é possível reduzir significativamente as dimensões espaciais (largura e altura) de cada *activation map* (Agarwal; Terrail; Jurie, 2019);
- c) Função de ativação ReLU** - Geralmente, a operação de convolução é intercalada com as operações de *pooling* e *ReLU*. A função de ativação *ReLU* é aplicada para cada  $L_q \times B_q \times d_q$  valores de uma camada a fim de limitá-los, passando os valores resultantes para a próxima camada. Assim, como trata-se apenas de um mapeamento de um para um dos valores de ativação, não há mudanças nas dimensões da camada (Agarwal; Terrail; Jurie, 2019);
- d) Camada totalmente conectada (*fully connected layer*)** - As camadas *fully connected* são constituídas por neurônios os quais possuem um peso associado a cada elemento do vetor de entrada. Essas camadas são responsáveis por traçar um caminho de decisão a partir de respostas dos filtros das camadas anteriores, para cada classe de resposta. Após as camadas completamente conectadas, a última etapa do processamento de uma imagem de entrada trata-se do uso de uma função de ativação do tipo *softmax*. Esta função é a de classificação, sendo a função *softmax* amplamente adotada devido à sua capacidade única de transformar a saída da rede neural em uma distribuição de probabilidade, atribuindo probabilidades a cada classe possível. Essa característica é essencial em tarefas de classificação, tornando a *softmax* uma escolha frequente para interpretar a saída da rede neural de maneira probabilística e identificar a classe mais provável.

Sem dúvidas, as CNNs têm demonstrado sua eficácia em tarefas de reconhecimento visual, incluindo o reconhecimento de obstáculos móveis e sinais de trânsito, segmentação de imagens biológicas e classificação de imagens médicas. Elas também têm contribuído significativamente para o avanço de abordagens de aprendizado de máquina em tarefas de detecção de objetos, como a aprendizagem por transferência baseada em CNNs pré-treinadas para desempenhar outras tarefas. Desde que a rede LetNet-5 foi proposta em 1998 (Shao *et al.*, 2020), a qual é considerada a primeira CNN treinada por algoritmo de gradiente a partir da retro-propagação de erros de estimação, várias variantes de CNNs foram propostas, tais como Alexnet, VGGNet, Googlenet, ResNet, DenseNet, MobileNet, EfficientNet (Bi *et al.*, 2020; Feng *et al.*, 2023; Huang *et al.*, 2018; Marques; Agarwa; Díez, 2020; Xie *et al.*, 2018; Wu; Shen; Hengel, 2019). Essas variantes geralmente superam a precisão da classificação em comparação com CNNs anteriores, cronologicamente, nas tarefas de classificação de imagens em contextos específicos. Diversas variantes das CNNs se diferem em suas arquiteturas e conexões de peso (Sun *et al.*, 2020a). Muitas variações de CNN não possuem um nome específico. Tratam-se de CNNs, colocado de uma maneira geral, estruturalmente projetadas para obterem bom resultado em uma aplicação.

Ademais, uma variedade de algoritmos de aprendizado e a introdução de mecanismos como *dropout* que consiste na aleatória desativação de unidades neurais durante o treinamento e *Batch Normalization* que normaliza as entradas de cada camada, desempenharam papéis cruciais na otimização dessas redes, sendo fundamentais para a melhoria das CNNs ao longo dos

anos (Wu *et al.*, 2019). A introdução de algoritmos de otimização, como o *Adaptive Moment Estimation* (Adam), utilizado no treinamento de redes neurais, também contribuiu substancialmente para o aprimoramento do desempenho global das CNNs. Ele é responsável por adaptar dinamicamente as taxas de aprendizado para cada parâmetro. E ao fazer isso, o algoritmo Adam proporciona uma convergência (minimização da função de perda) mais eficiente durante o treinamento (Tan *et al.*, 2022).

Em suma, as CNNs são modelos avançados de aprendizado profundo projetados para processar entradas no formato de imagens. Elas são capazes de atribuir pesos a diferentes aspectos das imagens e, posteriormente, distinguir entre elas. O propósito fundamental das CNNs reside na exploração do espaço de informação entre os pixels de uma imagem, empregando a técnica de convolução discreta (Anamika; Verma, 2020). É importante ressaltar que, ao se analisar os índices de acurácia alcançados pelas CNNs, é necessário apresentar evidências concretas e contexto específico que fundamentem essa afirmação. Resultados obtidos em conjuntos de dados amplamente reconhecidos, como CIFAR-10, CIFAR-100, ImageNet, MNIST e Fashion MNIST, Caltech-101 e Caltech-256, Common Objects in Context (MS COCO), Visual Object Classes (Pascal VOC), Horses vs Humans, são exemplos emblemáticos que fornecem uma base sólida para avaliar o desempenho desses modelos (Baldominos; ; Isasi, 2019; Li, 2023; Mishkina; Sergievskiy; Matas, 2017). A diversidade desses conjuntos de dados, abrangendo desde categorias complexas de imagens naturais até representações mais simplificadas, permite uma análise abrangente da adaptabilidade das CNNs em diferentes domínios de visão computacional.

## 2.2 Explorando Abordagens Alternativas: Revisão da literatura

Esta seção proporciona uma revisão da literatura sobre técnicas diversificadas no campo do reconhecimento e processamento de dados. Ao considerar diferentes metodologias propostas por pesquisadores, uma ampla gama de abordagens é discutida, oferecendo uma compreensão valiosa sobre as contribuições individuais dessas técnicas para o contexto teórico e metodológico do trabalho em questão.

A técnica proposta por Park *et al.* (2016) para reconhecimento de atividade humana utilizou uma rede neural recorrente (RNN) e abre caminho para a aplicação de modelos sequenciais no estudo em questão, que envolve análise de dados temporais. Uma RNN é um tipo de arquitetura de rede neural projetada para processar sequências de dados, onde a informação é passada de um estado anterior para o próximo, permitindo o aprendizado com dados sequenciais. No estudo, uma matriz de características é inicialmente criada a partir do conjunto de dados de atividades MSRC-12 e, em seguida, a RNN é treinada. Posteriormente, o desempenho da RNN treinada é avaliado utilizando conjuntos de dados de teste, resultando em uma precisão média de reconhecimento de ações de 99,55%.

Zhiqiang e Jun (2017) demonstraram o uso de CNNs para detecção de regiões de interesse em imagens, o que pode ser aplicável à investigação de características específicas em conjuntos de dados visuais. Os autores desenvolveram uma técnica para extrair informações de uma CNN e propuseram um algoritmo de detecção que se baseia na identificação de regiões de interesse na imagem e na otimização dos resultados por meio de regressão. Enquanto os métodos convencionais dependem de características manuais, que são extraídas por especialistas e podem ser limitadas, essa abordagem busca contornar essa dependência. No entanto, a técnica enfrenta desafios como oclusões na imagem (quando partes estão bloqueadas) e baixa resolução, que podem comprometer sua precisão.

Becker (2017) apresentou arquiteturas de CNNs para lidar com dados multi-linguais, o que poderia informar a abordagem de processamento de dados em diferentes idiomas. O autor propôs três arquiteturas de CNNs que lidam com dados multi-linguais extraídos do *Twitter* contendo quatro línguas. Os dois primeiros modelos propostos caracterizam-se pelo fato de possuírem um total de parâmetros muito menor que os demais *baselines* considerados, e ainda assim, obtêm resultados superiores com uma boa margem de diferença. O último modelo proposto é capaz de realizar uma classificação multitarefa, identificando a polaridade das sentenças e também a língua.

Os autores Thanh *et al.* (2018) exploraram CNNs para diagnóstico médico, destacando a aplicabilidade dessas redes na pesquisa de classificação de padrões. Eles descrevem uma arquitetura para o diagnóstico de leucemia. A CNN proposta possui 7 camadas, 5 são convolucionais e 2 são totalmente conectadas. Foi utilizado o aumento de dados na base ALL-IDB 1 com 108 imagens. Foram aplicadas operações de rotação, translação, desfoque e equalização de histograma. Como resultado, o aumento de dados proporcionou 1188 a serem utilizadas nos experimentos.

Os autores Shafique e Tehsin (2018) propuseram um modelo de CNN para diagnosticar diferentes subtipos de leucemia linfóide aguda. A arquitetura proposta foi pré-treinada e continha 7 camadas, sendo 4 convolucionais e 3 totalmente conectadas. A base de dados utilizada nos experimentos foi a ALL-IDB 2 com 260 imagens. Os autores avaliaram entradas de diferentes sistemas de cor e aplicaram o aumento de dados para realizar o treino da CNN. Sendo assim, a base foi aumentada para 760 imagens. Os resultados apresentados foram significativos para o problema. A CNN proposta obteve 99,50% de acurácia para o diagnóstico entre imagens normais e doentes 96,06% para o diagnóstico dos subtipos da leucemia.

Na metodologia proposta por Vogado *et al.* (2018), foi utilizada a técnica de transferência de aprendizado para extrair características da última camada totalmente conectada de três CNNs. Os autores analisaram as características extraídas e devido a alta dimensionalidade do vetor, realizaram a seleção de atributos utilizando a técnica razão de ganho, usada para selecionar as características mais importantes, basicamente avalia quão útil cada característica é para o problema em questão. O tamanho final do vetor de características foi estabelecido empiricamente, demonstrando que uma excelente acurácia já é alcançada com apenas 5% das características. Para classificar o vetor extraído, foi utilizado o *Support Vector Machine* (SVM), uma técnica de aprendizado de máquina amplamente empregada para tarefas de classificação e regressão com núcleo *Radial Basis Function* (RBF). Nesse trabalho, a metodologia de validação empregada foi o *k-fold cross-validation*. Contudo, os autores não buscaram uma otimização do classificador para melhorar seus resultados, optando pela utilização da configuração padrão do SVM.

Ranjan *et al.* (2018) propôs uma técnica de aprendizagem profunda para detectar rostos destacando a aplicabilidade dessas redes na pesquisa de classificação de padrões. O documento apresenta técnicas de detecção de faces baseadas em redes neurais profundas e convolutivas (DCNN), e pode ser dividido em duas categorias; abordagem de janela deslizante e a abordagem baseada em região. As principais questões que são abordadas nesse documento são minimizar a dependência de uma enorme base de dados de treinamento, controle de viés de dados e mortificação nos dados de treinamento e reduzindo o tempo de treinamento quando a rede é mais ampla e mais profunda.

Sun *et al.* (2020b) propuseram um novo método que utiliza algoritmos genéticos para evoluir as arquiteturas e valores iniciais dos pesos de uma rede neural convolucional profunda para resolver problemas de classificação de imagens. Nele, um novo esquema de representação é desenvolvido para inicializar efetivamente os pesos de conexão de redes neurais convolucio-

nais profundas, para evitar que as redes fiquem presas ao mínimo local que normalmente é uma questão importante na otimização retroativa baseada no gradiente.

Luchi e Adami (2020) propuseram a utilização de aprendizado profundo para a identificação da presença ou falta de um anel de retenção montado na ponta de um eixo veicular a partir de imagens. Uma rede neural convolucional foi utilizada para aprender as características das imagens e realizar a classificação. A arquitetura da rede foi otimizada (redução de 90% do número de parâmetros) a fim de aumentar a eficiência computacional, uma abordagem que pode ser considerada na análise visual.

Secchi e Carro (2022) propôs a combinação entre técnicas de aprendizado profundo e redes neurais convolucionais com o objetivo de gerar um algoritmo que fosse capaz de identificar e rastrear indivíduos em vídeos, além disso, monitorar suas ações com o propósito de identificar movimentos que possam significar um ato criminoso, utilizando o algoritmo do YOLO para a identificação, filtro de Kalman para o rastreamento e BlazePose para a identificação dos movimentos, o que pode inspirar a investigação de reconhecimento de padrões em dados de vídeo.

Esses estudos exemplificam a diversidade de abordagens existentes e resultados obtidos no campo do processamento de imagens, oferecendo percepções sobre o potencial e as limitações de diferentes técnicas utilizadas para contornar parcialmente os desafios específicos no reconhecimento e na representação de informações em uma variedade de contextos práticos.

### 2.3 Rede VGG Convolucional e Dados ImageNet

A rede neural convolucional VGG-16 representa uma evolução da AlexNet, um marco no campo de visão computacional. Desenvolvida por Krizhevsky, Sutskever e Hinton (2012), a AlexNet trouxe inovações importantes, incluindo o uso de unidades de ativação ReLU e a introdução bem-sucedida de técnicas de treinamento em Unidade de Processamento Gráfico (GPU) — um componente especializado usado para acelerar o treinamento de modelos de aprendizado de máquina, incluindo redes neurais. Essas inovações foram fundamentais para o sucesso da AlexNet na competição ImageNet. A VGG-16, por sua vez, incorpora esses princípios, apresentando entradas de  $224 \times 224 \times 3$  e 13 camadas, que alternam entre operações de convolução e *pooling*. A camada de *maxpooling* é frequentemente aplicada para reduzir a dimensão espacial dos mapas ao longo das camadas da rede. Essa redução é crucial para a eficiência no treinamento, embora essa operação não afete a profundidade da entrada, ela apenas reduz a altura e a largura dos mapas (Vargas; Paes; Vasconcelos, 2016).

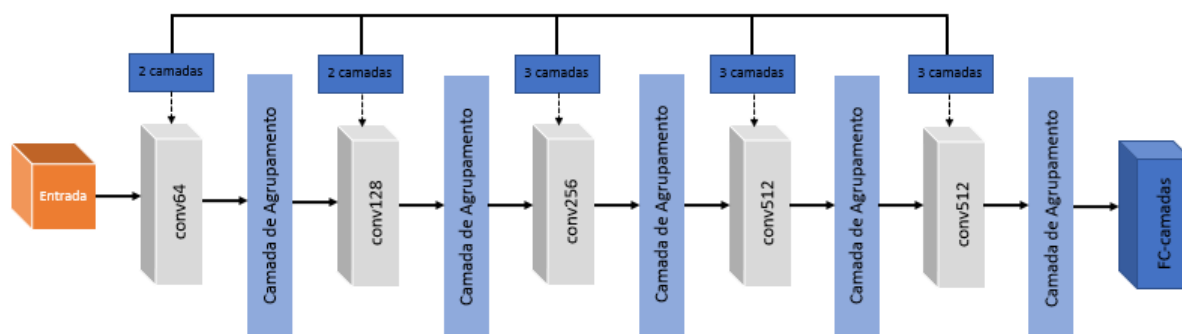
A criação da rede teve como objetivo principal investigar os efeitos da adição de profundidade em uma arquitetura convolucional, avaliando sua precisão no reconhecimento de imagens em larga escala, conforme abordado por Rezende *et al.* (2020). A arquitetura da VGG-16 é notável, envolvendo cerca de 138 milhões de parâmetros. Além de simplesmente aumentar a profundidade da rede, a VGG-16 demonstra uma melhoria eficaz no desempenho. Cada módulo básico é composto por um pequeno núcleo de convolução, um pequeno núcleo de *pooling* e a função de ativação *ReLU*.

A Figura 2.3 apresenta o diagrama da estrutura da VGG-16, a qual é constituída por cinco camadas convolutivas, três camadas totalmente conectadas e uma camada de saída *softmax*. O *maxpooling* é usado para separar as camadas e a função ReLU é usada para as unidades de ativação de todas as camadas ocultas. A camada de saída *softmax* desempenha um papel crucial na finalização do modelo, pois converte as pontuações de saída em probabilidades, permitindo a classificação multiclasse eficiente. Essa função é especialmente útil em tarefas em que o modelo precisa atribuir uma probabilidade para cada classe possível (Kong; Cheng, 2022).



A VGG-16 foi a primeira rede da categoria a utilizar cadeias de filtros pequenos ( $3 \times 3$ ) em cada camada convolucional. Isso era contrário aos princípios das suas antecessoras, LeNet e AlexNet, onde filtros grandes ( $9 \times 9$  e  $11 \times 11$ ) eram utilizados para capturar características similares na imagem. A grande contribuição da VGG foi a ideia de que múltiplas convoluções em sequência poderiam substituir os efeitos de filtros de máscaras maiores que resultavam em maior custo computacional.

Figura 2.3 – Diagrama da estrutura da VGG-16.



Fonte: Kong e Cheng (2022).

A arquitetura padrão, mencionada anteriormente é treinada com milhares de imagens, o que comprova sua capacidade de realizar classificações em diversas classes (Rodrigues; Sousa; Santos, 2021). Uma das vantagens desse modelo de rede convolucional é sua uniformidade, com os pesos pré-treinados na base de dados ImageNet (Rezende *et al.*, 2020), que estão disponíveis publicamente. Essa arquitetura padronizada tem sido empregada em várias aplicações e desafios, destacando-se como um extrator de características eficiente. Nesse cenário, uma arquitetura padrão refere-se a um modelo específico de rede neural que é reconhecido e usado como referência em determinado campo. Portanto, trata-se de uma rede neural amplamente utilizada, que, mencionada anteriormente, foi treinada com sucesso para classificar imagens com base nos dados do ImageNet.

O ImageNet, onde a VGG-16 foi treinada, é um extenso conjunto de dados que desempenha um papel crucial no treinamento de modelos de redes neurais, como a VGG-16. Este repositório abrange mais de 15 milhões de imagens de alta resolução, as quais foram minuciosamente classificadas em aproximadamente 22.000 categorias distintas. Essas imagens foram obtidas da internet e submetidas a um processo de rotulação, realizado por meio da ferramenta de *crowdsourcing* chamada *Mechanical Turk*, fornecida pela *Amazon*. O *crowdsourcing* é uma abordagem colaborativa que envolve a contribuição de um grande número de indivíduos para realizar tarefas específicas, nesse caso, rotular as imagens.

Desde 2010, como parte do *Pascal Visual Object Challenge*, uma competição anual que se concentra em desafios relacionados à visão computacional, especialmente no reconhecimento de objetos em imagens, ocorre anualmente o *ImageNet Large-Scale Visual Recognition Challenge* (ILSVRC). Essa competição desempenha um papel crucial no avanço da pesquisa em visão computacional. Durante o ILSVRC, é utilizado um subconjunto do ImageNet, consistindo de cerca de 1000 imagens em cada uma das 1000 categorias. Este desafio proporciona um ambiente competitivo para avaliar e comparar os avanços em algoritmos de reconhecimento visual, promovendo o desenvolvimento de técnicas mais eficientes e precisas na classificação de imagens (Krizhevsky; Sutskever; Hinton, 2017).

Em suma, a principal característica da VGG-16 é explorar filtros com dimensões ( $3 \times 3$ ) com pequenos campos receptivos para uma arquitetura profunda e mostrar que filtros ( $7 \times 7$ )

podem ser substituídos por uma sequência de três filtros ( $3 \times 3$ ), reduzindo assim a quantidade de parâmetros. A VGG-16 foi a abordagem vencedora da competição ILSVRC 2014 em localização, e obteve o segundo lugar em classificação (Anjos, 2019). A Tabela 2.1 mostra a estrutura e quantidade de parâmetros por camada da rede VGG-16.

Tabela 2.1 – Estrutura de uma arquitetura VGG-16

<b>Tipo de camada</b>	<b>Saída</b>	<b># Parâmetro</b>
1ª bloco e 1ª camada convolucional	(48, 48, 64)	1792
1ª bloco e 2ª camada convolucional	(48, 48, 64)	36928
1ª bloco e camada de max-pooling	(24, 24, 64)	0
2ª bloco e 1ª camada convolucional	(24, 24, 128)	73856
2ª bloco e 2ª camada convolucional	(24, 24, 128)	147584
2ª bloco e camada de max-pooling	(12, 12, 128)	0
3ª bloco e 1ª camada convolucional	(12, 12, 256)	295168
3ª bloco e 2ª camada convolucional	(12, 12, 256)	590080
3ª bloco e 3ª camada convolucional	(12, 12, 256)	590080
3ª bloco e camada de max-pooling	(6, 6, 256)	0
4ª bloco e 1ª camada convolucional	(6, 6, 512)	1180160
4ª bloco e 2ª camada convolucional	(6, 6, 512)	2359808
4ª bloco e 3ª camada convolucional	(6, 6, 512)	2359808
4ª bloco e camada de max-pooling	(3, 3, 512)	0
5ª bloco e 1ª camada convolucional	(3, 3, 512)	2359808
5ª bloco e 2ª camada convolucional	(3, 3, 512)	2359808
5ª bloco e 3ª camada convolucional	(3, 3, 512)	2359808
5ª bloco e camada de max-pooling	(1, 1, 512)	0
Camada flatten	(512)	0
1ª camada totalmente conectada	(4096)	102764544
2ª camada totalmente conectada	(4096)	16781312
3ª camada totalmente conectada	(1000)	409700

Fonte: Rodrigues, Sousa e Santos (2021).

## 2.4 Análise de Componentes Principais

A análise de componentes principais (PCAs, *principal component analysis*) é uma técnica da estatística multivariada que consiste em transformar um conjunto de variáveis originais em outro conjunto de variáveis de mesma dimensão denominadas de componentes principais. Os componentes principais apresentam propriedades importantes: cada componente principal é uma combinação linear de todas as variáveis originais. As componentes são independentes entre si, isto é, elas são ortogonais. Elas são estimadas com o propósito de reter, em ordem de estimação, o máximo de informação, em termos da variação total contida nos dados.

A análise de componentes principais é associada à ideia de redução de massa de dados, com menor perda possível da informação. Procura-se redistribuir a variação observada nos eixos originais de forma a se obter um conjunto de eixos ortogonais não correlacionados. Essa técnica pode ser utilizada para geração de índices e agrupamento de indivíduos. A análise agrupa os indivíduos de acordo com sua variação, ou seja, segundo seu comportamento dentro da população, representado pela variação do conjunto de características que define o indivíduo.

Segundo Regazzi (2000), apesar das técnicas de análise multivariada terem sido desenvolvidas para resolver problemas específicos, principalmente de Biologia e Psicologia, podem ser também utilizadas para resolver outros tipos de problemas em diversas áreas do conhecimento. A análise de componentes principais é a técnica mais conhecida, contudo é importante ter uma visão conjunta de todas ou quase todas as técnicas da estatística multivariada para resolver a maioria dos problemas práticos (Varella, 2008).

### 2.4.1 Matriz de dados $X$

Considere a situação em que são observadas  $p$  características de  $n$  indivíduos de uma população  $\pi$ . As características observadas são representadas pelas variáveis  $x_1, x_2, x_3, \dots, x_p$ . A matriz de dados é de ordem  $n \times p$  e normalmente denominada de matriz  $X$ :

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & \cdots & x_{1p} \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2p} \\ x_{31} & x_{32} & x_{33} & \cdots & x_{3p} \\ \cdots & \cdots & \cdots & \ddots & \cdots \\ x_{n1} & x_{n2} & x_{n3} & \cdots & x_{np} \end{bmatrix} \quad (2.1)$$

A estrutura de interdependência entre as variáveis da matriz de dados é representada pela matriz de covariância  $S$  ou pela matriz de correlação  $R$ . A compreensão dessa estrutura através das variáveis  $x_1, x_2, x_3, \dots, x_p$  pode ser complexa na prática. Assim, o objetivo da análise de componentes principais é transformar essa estrutura complicada, representada pelas variáveis  $x_1, x_2, x_3, \dots, x_p$ , em uma outra estrutura representada pelas variáveis  $y_1, y_2, y_3, \dots, y_p$  não correlacionadas e com variâncias ordenadas, para que seja possível comparar os indivíduos usando apenas as variáveis  $y_{is}$  que apresentam maior variância. A solução é dada a partir da matriz de covariância  $S$  ou da matriz de correlação  $R$ .

### 2.4.2 Matriz de covariância $S$

A partir da matriz  $X$  de dados de ordem  $n \times p$  é feita uma estimativa da matriz de covariância  $\Sigma$  da população  $\pi$  que é representada por  $S$ . A matriz  $S$  é simétrica e de ordem  $p \times p$ .

$$S = \begin{bmatrix} \widehat{Var}(x_1) & \widehat{Cov}(x_1x_2) & \widehat{Cov}(x_1x_3) & \cdots & \widehat{Cov}(x_1x_p) \\ \widehat{Cov}(x_2x_1) & \widehat{Var}(x_2) & \widehat{Cov}(x_2x_3) & \cdots & \widehat{Cov}(x_2x_p) \\ \widehat{Cov}(x_3x_1) & \widehat{Cov}(x_3x_2) & \widehat{Var}(x_3) & \cdots & \widehat{Cov}(x_3x_p) \\ \cdots & \cdots & \cdots & \ddots & \vdots \\ \widehat{Cov}(x_px_1) & \widehat{Cov}(x_px_2) & \widehat{Cov}(x_px_3) & \cdots & \widehat{Var}(x_p) \end{bmatrix} \quad (2.2)$$

Normalmente, as características são observadas em unidades de medidas diferentes entre si, e nesse caso, segundo Regazzi (2000), é conveniente padronizar as variáveis  $X_j (j = 1, 2, 3, \dots, p)$ . A padronização pode ser feita com média zero e variância igual a 1, ou com variância 1 e média qualquer.

### 2.4.3 Determinação dos componentes principais

Os componentes principais são determinados resolvendo-se a equação característica da matriz  $S$  ou  $R$ , dada por:

$$\det [R - \lambda I] = 0 \quad \text{ou} \quad |R - \lambda I| = 0 \quad (2.3)$$

$$R = \begin{bmatrix} 1 & r(x_1x_2) & r(x_1x_3) & \cdots & r(x_1x_p) \\ r(x_2x_1) & 1 & r(x_2x_3) & \cdots & r(x_2x_p) \\ r(x_3x_1) & r(x_3x_2) & 1 & \cdots & r(x_3x_p) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r(x_px_1) & r(x_px_2) & r(x_px_3) & \cdots & 1 \end{bmatrix} \quad (2.4)$$

Se a matriz  $R$  for de posto completo, igual a  $p$ , ou seja, não apresentar nenhuma coluna que seja combinação linear de outra, a equação  $|R - \lambda I| = 0$  terá  $p$  raízes, chamadas de autovalores ou raízes características da matriz  $R$ . Na montagem da matriz de dados  $X$ , é importante observar que o valor de  $n$  (indivíduos, tratamentos, genótipos, etc.) deve ser pelo menos igual a  $p + 1$ . Assim, para analisar o comportamento de  $p$  características de indivíduos em uma população, é recomendado que o delineamento estatístico apresente pelo menos  $p + 1$  tratamentos.

Sejam  $\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_p$  as raízes da equação característica da matriz  $R$  ou  $S$ , então:

$$\lambda_1 > \lambda_2 > \lambda_3 > \dots, \lambda_p \quad (2.5)$$

Para cada autovalor  $\lambda_i$  existe um autovetor  $\tilde{a}_i$ :

$$\tilde{a}_i = \begin{bmatrix} a_{i1} \\ a_{i2} \\ \vdots \\ a_{ip} \end{bmatrix} \quad (2.6)$$

Os autovetores  $\tilde{a}_i$  passam por dois processos importantes: primeiro, são normalizados, o que significa que a soma dos quadrados dos seus coeficientes é igual a 1. Segundo, eles são ortogonais entre si, ou seja, são perpendiculares uns aos outros. Essas características proporcionam propriedades fundamentais, permitindo a análise e compreensão do comportamento desses autovetores:

$$\sum_{j=1}^p a_{ij}^2 = 1 \quad (\tilde{a}_i \cdot \tilde{a}_i = 1) \quad (2.7)$$

$$\sum_{j=1}^p a_{ij}^2 \cdot a_{kj} = 0 \quad (\tilde{a}_i \cdot \tilde{a}_k = 0, i \neq k) \quad (2.8)$$

Sendo  $\tilde{a}_i$  o autovetor correspondente ao autovalor  $\lambda_i$ , então o variável-ésimo componente principal é dado por:

$$y_i = a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{ip}x_p \quad (2.9)$$

Os componentes principais apresentam as seguintes propriedades:

1. A variância do componente principal  $y_i$  é igual ao valor do autovalor  $\lambda_i$ .

$$\widehat{Var}(y_i) = \lambda_i \quad (2.10)$$

2. O primeiro componente é o que apresenta maior variância e assim por diante:

$$\widehat{Var}(y_1) > \widehat{Var}(y_2) > \widehat{Var}(y_p) \quad (2.11)$$

3. O total de variância das variáveis originais é igual ao somatório dos autovalores que é igual ao total de variância dos componentes principais:

$$\sum \widehat{Var}(x_i) = \sum \lambda_i = \sum \widehat{Var}(y_1) \quad (2.12)$$

4. Os componentes principais não são correlacionados entre si:

$$\widehat{Cov}(y_i, y_j) = 0 \quad (2.13)$$

#### 2.4.4 Valores dos Componentes Principais (Escore dos Componentes Principais)

Os escores representam os valores associados aos componentes principais. Após a redução de  $p$  para  $k$ -dimensões, os  $k$ -componentes principais serão os novos indivíduos e toda análise é feita utilizando os escores desses componentes. Na Tabela 2.2 é exemplificado a organização de um conjunto de dados composto por  $n$  tratamentos,  $p$  variáveis e  $k$  componentes principais.

Tabela 2.2 – Organização de um conjunto de dados com  $n$  tratamentos,  $p$  variáveis e  $k$  componentes.

Tratamento (indivíduos)	Variáveis				Escores dos componentes principais			
	$x_1$	$x_2$	$\cdots$	$x_p$	$y_1$	$y_2$	$\cdots$	$y_k$
1	$x_{11}$	$x_{12}$	$\cdots$	$x_{1p}$	$y_{11}$	$y_{12}$	$\cdots$	$y_{1k}$
2	$x_{21}$	$x_{22}$	$\cdots$	$x_{2p}$	$y_{21}$	$y_{22}$	$\cdots$	$y_{2k}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$n$	$x_{n1}$	$x_{n2}$	$\cdots$	$x_{np}$	$y_{n1}$	$y_{n2}$	$\cdots$	$y_{nk}$

Fonte: Varella (2008).

Assim, tem-se que os escores do primeiro componente para os  $n$  tratamentos são como mostra a Tabela 2.3.

Tabela 2.3 – Resultante das componentes principais para  $n$  tratamentos.

Tratamentos	Primeiro componente principal
1	$y_{11} = a_{11}x_{11} + a_{12}x_{12} + \cdots + a_{1p}x_{1p}$
2	$y_{21} = a_{11}x_{21} + a_{12}x_{22} + \cdots + a_{1p}x_{2p}$
$\vdots$	$\vdots$
$n$	$y_{n1} = a_{11}x_{n1} + a_{12}x_{n2} + \cdots + a_{1p}x_{np}$

Fonte: Varella (2008).

## 2.5 Sistemas Inteligentes Evolutivos

Sistemas inteligentes evolutivos são sistemas, baseados em redes neurais artificiais, regras fuzzy ou redes neuro-fuzzy, capazes de, a partir de dados de entrada adquiridos em modo *online* e, muitas vezes, em tempo real, determinar gradualmente tanto a sua estrutura quanto seus parâmetros (Angelov; Kasabov, 2006). A teoria do conjunto difuso surgiu a contar de uma necessidade de modelos matemáticos mais completos e inclusivos de incerteza (Zadeh, 1965). Uma característica típica dos sistemas fuzzy é que eles são baseados no conceito de participação fuzzy das informações (Neves *et al.*, 2022), (Barros; Bassanezi; Lodwick, 2017). Os sistemas

fuzzy adotam abordagens simplificadas para enfrentar desafios de natureza complexa. Em vez de controlar e/ou modelar um processo por meio de relações complexas, um sistema fuzzy procura modelar o problema mediante um conjunto simples de regras (Barros; Bassanezi; Lodwick, 2017). O processo de geração de regras é gradual e as regras não são fixas ou pré-determinadas. Além disso, regras podem ser revisadas para melhor se adaptarem às experiências vivenciadas. O interesse por essa nova abordagem para o problema de modelagem fuzzy vem crescendo nos últimos anos e diversas pesquisas tem apontado como uma abordagem eficiente na solução de diversos problemas.

Em meados de 2002, os sistemas fuzzy evolutivos surgiram como uma versão de sistemas com ajuste adaptativo de parâmetros e estrutura. Desde então, esse tipo de sistema avançado tem despertado grande interesse da academia e indústria com aplicações em diversas áreas do conhecimento (Pires; Serra, 2019). Os fluxos de dados *online* têm origem em uma variedade de fontes, tais como entretenimento de mídia, sistemas de vigilância, dispositivos móveis, multimídia, monitoramento industrial e controle, sistemas oceanográficos e atmosféricos, cuidados com a saúde, mercado de ações, satélites, sistemas financeiros e meteorológicos (Leite, 2012).

Leite, Jr e Gomide (2013) introduziram uma estrutura de rede neural granular para a modelagem de sistemas fuzzy em evolução a partir de fluxos de dados difusos. A rede neural granular em evolução (EGNN) é capaz de lidar com fluxos de dados graduais e abruptos. Os resultados das simulações sugerem que a abordagem de modelagem fuzzy EGNN pode lidar com dados difusos com sucesso e superam as abordagens alternativas de última geração em termos de precisão, transparência e compacidade.

Lughofer *et al.* (2015) propuseram uma nova metodologia para aprendizado de sistemas nebulosos evolutivos a partir de fluxo de dados em regressão/identificação de sistemas *online*. Ela vem com etapas de redução de complexidade dinâmica aprimorada, atuando sobre os componentes do modelo e sobre a estrutura de entrada e empregando regras difusas e generalizadas em posição de rotação arbitrária.

Alizadeh *et al.* (2016) propuseram um novo sistema nebuloso evolutivo denominado eHFIS (do inglês, *evolving Heterogeneous Fuzzy Inference System*), capaz de simultaneamente realizar seleção de entrada local e identificação de sistema de forma evolutiva e integrativa. O termo evolutivo é utilizado para definir tais sistemas adaptativos, uma vez que estes sistemas são capazes de ajustar seus parâmetros a partir de dados (o que geralmente é atribuído ao termo adaptativo), contudo, também adaptam sua estrutura, ou seja, a estrutura do sistema não permanece fixa.

Silva *et al.* (2018) propuseram um método de aprendizagem neuro-fuzzy baseado em fluxos de dados para detecção em linhas de energia de média tensão de falha de alta impedância (HIF). Foi utilizado um método de extração de características baseado em *wavelet* combinado com um variação da rede neuro-fuzzy em evolução com limiares flutuantes para o reconhecimento de padrões nos dados. O modelo de classificação neuro-fuzzy proposto, mostrou-se particularmente adequado para o problema porque o ambiente HIF está sujeito a mudanças. Os resultados mostraram que o sistema neuro-fuzzy é eficaz e robusto às mudanças, o que o torna capaz de manter sua detecção e precisão de classificação mesmo em situações em que outros classificadores exibem uma queda significativa na precisão devido a mudanças graduais e abruptas dos padrões de falhas.

Fortunato *et al.* (2020) propuseram a utilização das redes neuro-fuzzy granulares evolutivas, as quais são capazes de adaptar continuamente sua estrutura e atualizar seus parâmetros de acordo com um fluxo de dados para a classificação de distúrbios de qualidade de energia elétrica. Esses distúrbios podem ocorrer em várias partes de um sistema de potência e podem causar prejuízos financeiros a todos que estão a ele conectado. Na etapa de pré-processamento

dos dados para extração de atributos é considerado o valor eficaz das tensões de fase, o filtro de Hodrick e Prescott e a rede granular evolutiva para posterior classificação dos dados. A classificação de quatro distúrbios e da operação normal do sistema (problema de 5 classes) foi alcançada com acurácia média de 98%.

Leite, Jr. e Medeiros (2021) propuseram uma modelagem baseada em um Classificador Fuzzy Gaussiano evolutivo (eGFC), que é apoiado por um algoritmo de aprendizagem *online* semi-supervisionado para reconhecer padrões emocionais a partir de fluxos de dados de eletroencefalograma (EEG). Foram extraídas características do espectro de Fourier dos dados do EEG. Os dados são fornecidos por 28 indivíduos jogando os jogos *Train Sim World*, *Unravel*, *Slender The Arrival*, e *Goat Simulator* – um conjunto de dados públicos. A proposta se mostrou muito eficaz no aprendizado em tempo real dos dados EEG, atingindo um percentual de 72,2% de precisão usando uma base de regras variável em um problema de classificação.

Siminski (2021) propôs um novo algoritmo baseado em regra fuzzy que agrupa objetos e atributos em aglomerados difusos. Dessa forma, criaram uma base de regras fuzzy subespacial para um sistema fuzzy subespacial. O papel é acompanhado de exemplos numéricos que mostram que esta abordagem pode levar a uma melhor generalização (e, portanto, a uma menor previsão de erros nos dados) com interpretação preservada de modelos difusos.

Zhang, Zhan e Wu (2021) propuseram uma nova metodologia de um operador fuzzy de  $\alpha$  - vizinhança com reflexividade, construído a partir de e um novo modelo de conjunto difuso. O foco é a tomada de decisões em sistemas de informação, com sistemas de informação de valor real (RVISs). Os autores adotaram três métodos diferentes para a tomada de decisões de classificação, que consistem na subtração de números fuzzy intuicionistas, funções de classificação e coeficientes de intimidade. Experimentalmente, os resultados comprovam a estabilidade do desempenho da estratégia.

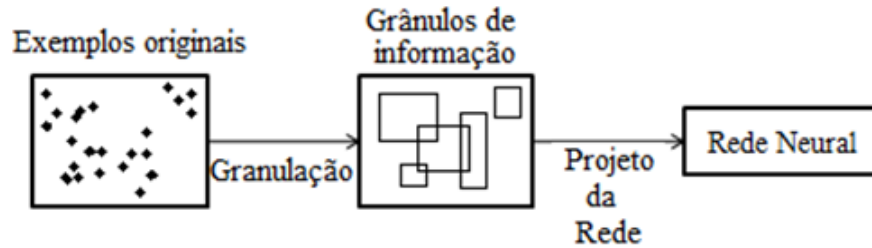
## 2.6 Rede Neural Granular Evolutiva EGNN para Classificação de Padrões

O conceito de redes neurais granulares (GNN) foi inicialmente estabelecido por Pedrycz e Vukovich (2001) para designar redes neurais com algum aspecto que envolva incerteza. Por exemplo: (i) os dados processados por uma rede granular não precisam ser necessariamente numéricos, mas podem ser intervalos, números fuzzy, intervalos fuzzy, distribuições de probabilidade e palavras; (ii) as operações das conexões entre neurônios ou do corpo neural podem não ser pontuais, mas levar em conta incerteza; (iii) as estimações providas pela rede podem ser invólucros em torno do valor-alvo; e/ou (iv) a representação dos dados no modelo neural é feita de forma modular, a partir do desenvolvimento de modelos locais (grânulos de informação encontrados no fluxo de dados *online*). A partir do ambiente GNN original, foram propostas várias outras abordagens neurais granulares. Dentre elas, se destaca a rede *evolving* GNN, eGNN (Leite; Jr; Gomide, 2010), (Leite, 2019), pois ela foi equipada com algoritmo de aprendizado de máquina incremental *online* e capacidade de adaptação estrutural do modelo, além da capacidade de atualização paramétrica usual. Além disso, eGNN contempla os quatro aspectos (i)-(iv) mencionados.

O aprendizado em GNN e eGNN segue um princípio comum que, geralmente, envolve dois estágios, conforme mostra a Figura 2.4. Primeiro, grânulos de informação – intervalos ou conjuntos fuzzy – são construídos a partir de uma base de representação numérica. Note que, algumas vezes, os dados são originalmente granulares. Em seguida, a aprendizagem – construção e refinamento – da rede neural é baseada nos grânulos de informação ao invés de ser baseada nos dados originais. Assim, a rede eGNN não é exposta a todos os dados de treinamento, muito mais numerosos que os grânulos formados. Por exemplo, quando amostras de dados não trans-

portam novas informações, os exemplos são incorporados pelos grânulos já conhecidos (Leite, 2019).

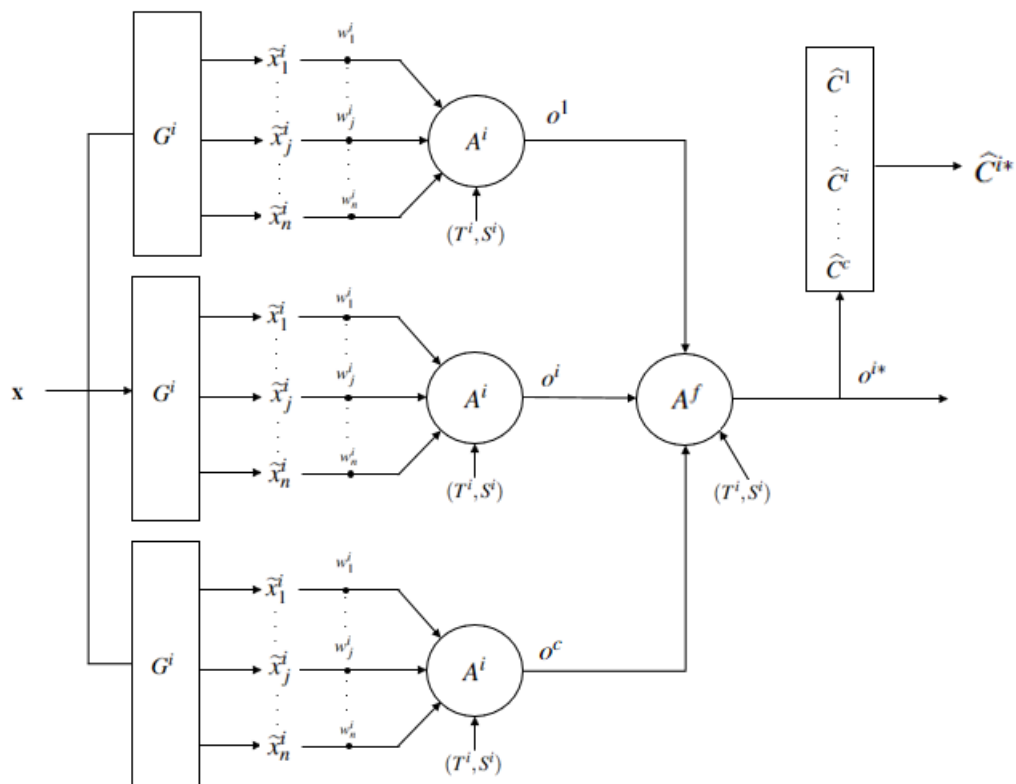
Figura 2.4 – Projeto da rede neural granular evolutiva EGNN ou GNN.



Fonte: Leite (2019).

Considere que o fluxo de dados  $(\mathbf{x}, y)^{[h]}$ ,  $h = 1, \dots$ , é medido a partir de uma função desconhecida  $f$ . Entradas  $x_j$  e saída  $y$  são dados numéricos e uma classe. A Figura 2.5 mostra um modelo EGNN de quatro camadas. A camada de entrada recebe  $x^{[h]}$ . A camada granular é um conjunto de grânulos  $G^i$ ,  $i = 1, \dots, c$ , estratificado a partir dos dados de entrada, formando uma partição fuzzy da  $j$ -ésima entrada domínio. Um grânulo  $G^i = G_1^i \times \dots \times G_n^i$  é uma relação fuzzy, ou seja, um conjunto fuzzy multidimensional em  $X_1 \times \dots \times X_n$ . Assim,  $G^i$  tem a função de pertinência  $G^i(x) = \min(G_1^i(x_1), \dots, G_n^i(x_n))$  em  $X_1 \times \dots \times X_n$  (Decker *et al.*, 2020).

Figura 2.5 – EGNN: Evolução da arquitetura de rede neuro-fuzzy para classificação.



Fonte: Decker *et al.* (2020).

De modo geral, é possível modelar tanto dados quanto grânulos como objetos trapezoidais. Essa característica amplia a flexibilidade do modelo, permitindo a representação de informações de maneira mais abrangente. A medida de semelhança, utilizada para quantificar a



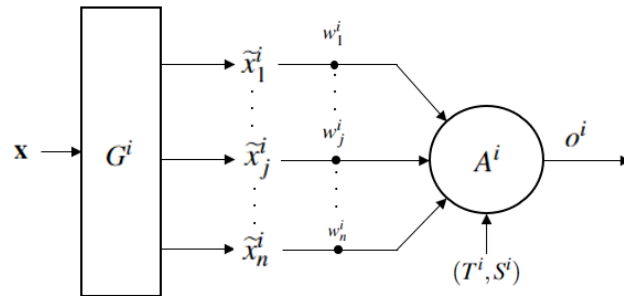
correspondência entre uma instância numérica e o conhecimento existente, é formalmente apresentada na Equação 2.14 (Leite; Jr; Gomide, 2013). Essa escolha de representação trapezoidal é estratégica, pois oferece uma adaptação eficaz a diferentes perfis de dados, proporcionando uma análise mais robusta e abrangente.

$$\tilde{x}_j^i = 1 - \frac{\left| \underline{g}_j^i - x_j \right| + \left| \underline{g}_j^i - x_j \right| + \left| \bar{g}_j^i - x_j \right| + \left| \bar{g}_j^i - x_j \right|}{4 \left( \max \left( \bar{g}_j^i, x_j \right) - \min \left( \underline{g}_j^i, x_j \right) \right)} \quad (2.14)$$

A camada de agregação é composta pelos neurônios  $A^i$ . Um neurônio fuzzy  $A^i$  combina graus de similaridade ponderados  $(\tilde{x}_1^i \omega_1^i, \dots, \tilde{x}_n^i \omega_n^i)$  em um único valor  $o^i$  que se refere ao nível de ativação do  $R^i$ . A camada de saída processa  $(o^1, \dots, o^c)$  usando um neurônio  $A^f$  que realiza a  $S$ -norma máxima. A classe  $C^{i*}$  da regra mais ativa  $R^{i*}$  é a saída. Sob suposição de pesos e neurônios específicos fuzzy as regras extraídas do eGNN são do tipo  $R^i(x) : \text{se } (x_1 \text{ é } G_1^i) \text{ e } \dots \text{ e } (x_n \text{ é } G_n^i)$  então  $(\hat{y} \text{ é } \hat{C}^i)$  (Decker *et al.*, 2020).

O modelo de neurônio da EGNN são do tipo nebulosos. Neurônios nebulosos são modelos de neurônios baseados em operadores de agregação. Operadores de agregação  $A : [0, 1]^n \rightarrow [0, 1], n > 1$  combinam valores de entrada na unidade hipercubo  $[0, 1]^n$  em um único valor em  $[0, 1]$ . Eles devem satisfazer o seguinte: monotonicidade em todos os argumentos e condições limites (Leite; Jr; Gomide, 2013). A Figura 2.6 mostra um exemplo de neurônio no qual o processamento sináptico é dado pelo produto  $T$ -norma e o operador de agregação  $A^i$  é usado para combinar entradas individuais. A saída  $o^i$  é  $A^i(\tilde{x}_1^i \omega_1^i, \dots, \tilde{x}_n^i \omega_n^i)$ .

Figura 2.6 – Modelo de neurônio difuso.



Fonte: Decker *et al.* (2020).

Como  $G_j^i$  é uma associação trapezoidal, sua função, seu suporte, seu núcleo, seu ponto médio e sua largura são apresentados nas seguintes equações:

$$\text{supp}(G_j^i) = \left[ \underline{g}_j^i, \bar{g}_j^i \right] \quad (2.15)$$

$$\text{core}(G_j^i) = \left[ \underline{g}_j^i, \bar{g}_j^i \right] \quad (2.16)$$

$$\text{mp}(G_j^i) = \frac{\underline{g}_j^i + \bar{g}_j^i}{2} \quad (2.17)$$

$$\text{wdt}(G_j^i) = \bar{g}_j^i - \underline{g}_j^i \quad (2.18)$$

Um  $\rho$  inicial diferente produz classificadores diferentes. No entanto,  $\rho$  muda autonomamente ao longo do tempo, conforme:

$$\rho^{[h]} = \begin{cases} (1 + \frac{r}{h_r})\rho^{[h-h_r]}, r > \eta \\ (1 - \frac{\eta-r}{h_r})\rho^{[h-h_r]}, r \leq \eta \end{cases} \quad (2.19)$$

É possível expandir a largura dos conjuntos fuzzy  $G_j^i$  dentro da área  $E_j^i$  delimitado por  $\rho$ , i.e.,  $wdt(G_j^i) \leq \rho$ .  $E_j^i$  é dada por  $[mp(G_j^i) - \frac{\rho}{2}, mp(G_j^i) + \frac{\rho}{2}]$ . Claramente,  $wdt(G_j^i) \leq wdt(E_j^i)$ . O valor de  $\rho$  afeta a granularidade das informações e, conseqüentemente, a precisão do modelo. Nesse sentido,  $\rho \in [0, 1]$  é usado para controlar o tamanho das regiões de expansão.

A EGNN começa com uma base de regras vazia,  $\rho^{[0]} = 0,5$  é usado como padrão. Seja  $r$  o número de regras criadas em etapas  $h_r$  e  $\eta$  uma taxa de referência, se o número de regras crescer mais rapidamente do que a taxa  $\eta$ , então  $\rho$  é aumentada, caso contrário  $\rho$  é reduzida, pois na Equação 2.19 os valores apropriados para  $\rho$  são encontrados de forma autônoma. Se  $\rho = 1$ , então a eGNN é estruturalmente estável, mas incapaz de capturar mudanças abruptas. Por outro lado, se  $\rho = 0$ , então a eGNN se ajusta aos dados, causando complexidade excessiva do modelo. A adaptabilidade é alcançada a partir de valores intermediários. Reduzir  $\rho$  requer redução de grânulos grandes, de acordo com:

$$\begin{aligned} \text{Se } mp(G_j^i) - \frac{\rho(\text{nov})}{2} > \underline{g}_j^i \text{ então } \underline{g}_j^i(\text{nov}) &= mp(G_j^i) - \frac{\rho(\text{nov})}{2} \\ \text{Se } mp(G_j^i) + \frac{\rho(\text{nov})}{2} > \bar{g}_j^i \text{ então } \bar{g}_j^i(\text{nov}) &= mp(G_j^i) + \frac{\rho(\text{nov})}{2} \end{aligned}$$

Quando se tratando do desenvolvimento de grânulos, se o suporte de ao menos uma entrada de  $x$  está fora de regiões de expansão ( $E_1^i, \dots, E_n^i$ ), então a EGNN gera um novo grânulo,  $G^{c+1}$ . Esse novo grânulo é constituído por conjuntos fuzzy cujos parâmetros são:

$$G_j^{c+1} = \left( \underline{g}_j^{c+1}, \underline{g}_j^{c+1}, \bar{g}_j^{c+1}, \bar{g}_j^{c+1}, x_j, x_j, x_j, x_j \right) \quad (2.20)$$

A atualização dos grânulos consiste em expandir ou contrair o núcleo dos conjuntos fuzzy  $G_j^i$ . Em particular,  $G^i$  é escolhido de  $\text{argmax}(o^1, \dots, o^c)$ . A adaptação continua dependendo de onde  $x_j$  está posicionado em relação ao  $G_j^i$ :

$$\begin{aligned} \text{Se } x_j \in \left[ mp(G_j^i) - \frac{\rho}{2}, \underline{g}_j^i \right] \text{ então } \underline{g}_j^i(\text{nov}) &= x_j \\ \text{Se } x_j \in \left[ mp(G_j^i) - \frac{\rho}{2}, mp(G_j^i) \right] \text{ então } \underline{g}_j^i(\text{nov}) &= x_j \\ \text{Se } x_j \in \left[ mp(G_j^i), mp(G_j^i) + \frac{\rho}{2} \right] \text{ então } \underline{g}_j^i(\text{nov}) &= mp(G_j^i) \\ \text{Se } x_j \in \left[ mp(G_j^i) - \frac{\rho}{2}, mp(G_j^i) \right] \text{ então } \bar{g}_j^i(\text{nov}) &= mp(G_j^i) \\ \text{Se } x_j \in \left[ mp(G_j^i), mp(G_j^i) + \frac{\rho}{2} \right] \text{ então } \bar{g}_j^i(\text{nov}) &= x_j \\ \text{Se } x_j \in \left[ \bar{g}_j^i, mp(G_j^i) + \frac{\rho}{2} \right] \text{ então } \bar{g}_j^i(\text{nov}) &= x_j \end{aligned}$$

Operações sobre parâmetros centrais,  $\underline{g}_j^i$  e  $\bar{g}_j^i$ , provocam a adaptação do ponto médio, como mostra a equação:

$$mp(G_j^i)(\text{nov}) = \frac{\underline{g}_j^i(\text{nov}) + \bar{g}_j^i(\text{nov})}{2} \quad (2.21)$$

Portanto, contrações de suporte podem ser necessárias:

$$\text{Se } mp(G_j^i)(\text{nov}) - \frac{\rho}{2} > \underline{g}_j^i \text{ então } \underline{g}_j^i(\text{nov}) = mp(G_j^i)(\text{nov}) - \frac{\rho}{2}$$

$$\text{Se } mp(G_j^i)(\text{nov}) + \frac{\rho}{2} < \bar{g}_j^i \text{ então } \bar{g}_j^i(\text{nov}) = mp(G_j^i)(\text{nov}) + \frac{\rho}{2}$$

Para a atualização dos pesos da rede neural,  $\omega_j^i \in [0, 1]$  é proporcional à importância do  $j$ -ésimo atributo de  $G_j^i$  à saída da rede neural. Quando um novo grânulo  $G^{c+1}$  é gerado, os pesos são definidos como  $\omega_j^{c+1} = 1, \forall j$ .

O  $\omega_j^i$  atualizado é associado ao grânulo mais ativo  $G^i, i = \text{argmax}(o^1, \dots, o^c)$ , por:

$$\omega_j^i(\text{nov}) = \omega_j^i(\text{velho}) - \beta^i \tilde{x}_j^i \in | \quad (2.22)$$

no qual  $\tilde{x}_j^i$  é a semelhança com  $G_j^i$ ;  $\beta^i$  depende do número de acerto ( $\text{acerto}^i$ ) e erro ( $\text{erro}^i$ ) de classificações, como mostrado em:

$$\beta^i = \frac{\text{erro}^i}{\text{acerto}^i + \text{erro}^i} \quad (2.23)$$

$$\in^{[h]} = C^{[h]} - \widehat{C}^{[h]} \quad (2.24)$$

onde  $\in^{[h]}$  é o erro de estimativa atual,  $\widehat{C}^{[h]}$  é a classe estimada e  $C^{[h]}$  é a classe real.

O procedimento de aprendizagem da eGNN é sumarizado no Algoritmo 1.

---

### Algoritmo 1: Aprendizado de rede neurais EGNN

---

- 1 : **selecione** um tipo de neurônio para a camada de agregação;
  - 2: **defina** os parâmetros  $\rho^{[0]}, h_r, \eta$ ;
  - 3: **leia** a instância  $x^{[h]}, h = 1$ ;
  - 4: **use** o gráfico de controle para rotular  $\mathbf{x}^{[h]}$  com  $C^{[h]}$ ;
  - 5: **crie** o grânulo  $G^{c+1}$ , neurônios  $A^{c+1}, A^f$ , e conexões;
  - 6: **para**  $h = 2, \dots$  faça
  - 7: **leia e avance**  $\mathbf{x}^{[h]}$  através da rede;
  - 8: **calcule** os níveis de ativação das regras ( $o^1, \dots, o^c$ );
  - 9: **agregue** a ativação usando  $A^f$  para obter a estimativa  $C^{[h]}$ ;
  - 10: // a classe  $C^{[h]}$  torna-se disponível;
  - 11: **calcule** o erro de saída  $\in^h = C^h - \widehat{C}^h$ ;
  - 12: **se**  $x^{[h]}$  **não é**  $E^i \forall i$  **ou**  $\in^h \neq 0$  **então**
  - 13: **crie** grânulos  $G^{c+1}$ , neurônios  $A^{c+1}$ , conexões;
  - 14: **associe**  $G^{c+1}$  a  $C^{[h]}$ ;
  - 15: **se não**
  - 16: **atualização**  $G^{i^*}, i^* = \text{argmax}(o^1, \dots, o^c)$ ;
  - 17: **adaptar pesos**  $\omega_j^{i^*} \forall j$ ;
  - 18: **finalizar se**
  - 19: **se**  $h = \beta h_r, \beta = 1, \dots$  **então**
  - 20: **adaptar** a granularidade do modelo  $\rho$ ;
  - 21: **fim para**
  - 22: **fim para**
-

### 2.6.1 Interpretabilidade e Explicabilidade em Modelos de Aprendizado de Máquina

A interpretabilidade e a explicabilidade são conceitos intimamente relacionados no campo do aprendizado de máquina, ambos desempenhando um papel fundamental na compreensão e confiança nos modelos gerados. Enquanto a interpretabilidade se refere à capacidade de um ser humano compreender o funcionamento interno de um modelo e suas decisões, a explicabilidade vai além, enfocando a capacidade do modelo de fornecer explicações claras e compreensíveis sobre suas previsões ou decisões. Ambos os conceitos são essenciais para garantir que os modelos de aprendizado de máquina sejam confiáveis, transparentes e facilmente compreensíveis para os usuários finais (Arrieta *et al.*, 2020).

Na prática, para explicar as previsões de um modelo de aprendizado de máquina, pode-se utilizar um algoritmo que gera explicações. Uma explicação geralmente relaciona os valores dos recursos de uma instância à previsão do modelo de uma maneira humanamente compreensível. A explicabilidade no aprendizado de máquina significa que é possível explicar o que acontece em seu modelo da entrada à saída, tornando os modelos mais transparentes e resolvendo ou minimizando o problema da “caixa preta” (Burkart; Huber, 2021).

O *Local Interpretable Model-Agnostic Explanations* (LIME) é um exemplo desses algoritmos e foi desenvolvido por pesquisadores da Universidade de Washington para entender o que acontece dentro de um algoritmo, capturando interações de recursos. O LIME é uma estrutura mais geral que visa tornar as previsões de qualquer modelo de aprendizado de máquina mais interpretáveis. O LIME é independente de modelo, o que significa que pode ser aplicado a qualquer modelo de aprendizado de máquina.

A técnica tenta entender o modelo perturbando a entrada de amostras de dados e entendendo como as previsões mudam. O algoritmo executa várias perturbações de vários recursos em torno de uma previsão específica e mede os resultados; ele também lida com entradas irregulares. Isso acaba sendo um benefício em termos de interpretabilidade, porque pode-se perturbar a entrada alterando componentes que fazem sentido para humanos, por exemplo, palavras ou partes de uma imagem, mesmo que o modelo esteja usando componentes muito mais complicados como recursos, como a incorporação de palavras (Ribeiro; Singh; Guestrin, 2016). O LIME é utilizado para explicar uma infinidade de classificadores, como florestas aleatórias, máquinas de vetor de suporte (SVM) e redes neurais, nos domínios de texto e imagem. Implementações dessas técnicas tornaram-se disponíveis recentemente à medida que bibliotecas de programação específicas foram desenvolvidas (Ribeiro; Singh; Guestrin, 2016; Alber *et al.*, 2019).

Para explicações de imagem, uma determinada imagem é segmentada algoritmicamente em superpixels, e a relevância de cada superpixel para uma determinada classificação é determinada usando um modelo linear. O algoritmo é independente de modelo, pois requer apenas as saídas do classificador para imagens diferentes. De fato, o LIME pode ser usado para qualquer sistema de classificação de imagens, não apenas redes neurais, pois não emprega procedimentos específicos de retropropagação ou etapas específicas para qualquer tipo de modelo individual (Ribeiro; Singh; Guestrin, 2016).

Outro modelo de algoritmo de explicabilidade é o *Shapley Additive exPlanations* (SHAP), desenvolvido por Lundberg e Lee (2017), para explicar previsões individuais, baseando-se nos valores de Shapley da teoria dos jogos. O método tenta interpretar de maneira mais direta as decisões dos modelos de inteligência artificial, diferente das soluções caixa preta. Para isso, o SHAP explica uma saída específica calculando a contribuição de cada característica (atributo) da entrada para a previsão (instância) (Burkart; Huber, 2021). Assim, o SHAP indica as influências negativas ou positivas das características para um resultado do modelo. O intuito do SHAP é explicar uma previsão feita a partir de uma instância, descobrindo o valor de Shapley de cada característica que resultou na previsão.

Essas características podem ser um grupo de valores de característica para uma imagem. Além disso, o SHAP cumpre três propriedades, já presentes no método clássico de valores de Shapley, desejáveis em métodos de atribuição aditiva de atributos (Lundberg; Lee, 2017): a precisão local, que determina a mesma saída para o modelo aproximado e o modelo original; a ausência, que garante nenhum impacto na saída para recursos ausentes; e a consistência, que afirma que, se um recurso do modelo aumenta ou permanece o mesmo independentemente dos outros recursos, o impacto desse recurso não deve diminuir.

Porém, encontrar os valores de Shapley para um grande número de atributos pode ser um trabalho pesado. Para evitar isso, Lundberg e Lee (2017) propuseram o Kernel SHAP, um método adaptado do LIME linear para calcular os valores de Shapley. O Kernel SHAP consegue obter esses valores com algumas amostras de conjuntos de características, utilizando dados de treinamento do modelo e uma regressão linear ponderada onde os coeficientes da solução são os valores de Shapley. É importante acrescentar que o Kernel SHAP é independente do modelo para o cálculo dos valores de Shapley, podendo ser aplicado a qualquer modelo de Aprendizado de Máquina. Contudo, existem outras versões de SHAP que possuem otimizações específicas para certos modelos, como os métodos Tree SHAP (McFall *et al.*, 2023), Deep SHAP (Aldughayfiq *et al.*, 2023), Low-Order SHAP (Nair *et al.*, 2019), Linear SHAP e Max SHAP (Burgerhof *et al.*, 2017).

Atualmente, existem inúmeras técnicas para explicabilidade em aprendizado de máquina. Entre essas técnicas, destacam-se Anchors (Ribeiro; Singh; Guestrin, 2018), que identificam condições simples para explicar as previsões do modelo. Os Partial Dependence Plots (PDP) (Johnson *et al.*, 2022) fornecem uma visão global do relacionamento entre uma variável específica e a saída do modelo, enquanto os Individual Conditional Expectations (ICE) plots (Goldstein *et al.*, 2015) oferecem uma perspectiva mais granular ao examinar o efeito de uma variável em diferentes instâncias de dados. O Leave One Covariate Out (LOCO) (Katsevich; Ramdas, 2020) é uma abordagem útil para avaliar a importância de características individuais, removendo-as uma de cada vez. O Accumulated Local Effects (ALE) (Galkin *et al.*, 2020) destaca a contribuição de cada característica ao longo de diferentes intervalos de seus valores. Enquanto isso, o Deep Learning Important Features (DeepLIFT) (Shrikumar; Greenside; Kundaje, 2017) explora a importância de características em redes neurais profundas. A Layer-wise Relevance Propagation (LRP) (Montavon *et al.*, 2019) atribui relevância a cada neurônio de uma rede neural, permitindo a interpretação de suas decisões. O Contrastive Explanations Method (CEM) (Dhurandhar *et al.*, 2018a) destaca diferenças entre duas instâncias para explicar as previsões do modelo, enquanto o ProfWeight (Dhurandhar *et al.*, 2018b) avalia a importância de características considerando suas interações. Essas são apenas algumas das técnicas disponíveis, evidenciando a crescente preocupação em tornar os modelos de aprendizado de máquina mais transparentes e compreensíveis.

Segundo Alonso, Magdalena e González-Rodríguez (2009) os modelos de regra fuzzy são uma abordagem de modelagem que utilizam lógica fuzzy para capturar a incerteza e a imprecisão inerentes aos sistemas do mundo real. Esses modelos são especialmente úteis em cenários onde as relações entre as variáveis são complexas e difíceis de expressar de maneira precisa. Ao aplicar modelos de regra fuzzy, é possível criar conjuntos de regras que descrevem as relações entre as variáveis de entrada e saída de uma maneira intuitiva e compreensível. Cada regra é composta por uma condição (ou conjunto de condições) que descreve as circunstâncias em que ela se aplica, e uma conclusão que especifica a ação a ser tomada com base nessas circunstâncias. Por exemplo, em um sistema de controle de temperatura para um forno, uma regra fuzzy poderia ser expressa da seguinte forma: (i) se a temperatura do forno está muito alta ou (ii) a temperatura ambiente está muito alta, então reduza a potência do aquecedor. Aqui,

a lógica fuzzy permite que os termos muito alta e muito baixa sejam interpretados de maneira flexível, levando em consideração a incerteza associada às medições de temperatura (Alonso; Magdalena; González-Rodríguez, 2009). Ao aplicar esses modelos de regras à interpretação de modelos de aprendizado de máquina, é possível criar regras que descrevem as decisões tomadas pelo modelo de uma maneira compreensível para os usuários.

Essas técnicas oferecem uma variedade de abordagens para entender e explicar o comportamento dos modelos de aprendizado de máquina. Num contexto mais amplo, representam opções interessantes para análises futuras, fornecendo uma gama de abordagens para interpretar o comportamento dos modelos de aprendizado de máquina.

## 2.7 Síntese

Este capítulo apresentou o contexto de visão computacional e aprendizagem de máquina baseado em redes neurais convolucionais e sistemas inteligentes evolutivos. Ele também introduziu uma estrutura de modelagem de fluxo de dados fuzzy baseada em uma abordagem de rede neural granular fuzzy evolutiva. A estrutura eGNN processa fluxos de dados fuzzy usando modelos granulares fuzzy, neurônios de agregação fuzzy e um algoritmo de aprendizado incremental *online*. Sua estrutura codifica um conjunto de regras fuzzy e um sistema de inferência fuzzy que estabelece um compromisso entre precisão e interpretabilidade combinando modelos fuzzy funcionais e linguísticos. A EGNN fornece aproximação de valor único, bem como aproximação granular de funções. Alguns métodos de explicabilidade foram abordados, incluindo o LIME e o SHAP, que oferecem maneiras de entender as previsões dos modelos de forma mais transparente e intuitiva. O LIME, por exemplo, perturba a entrada de dados para entender como as previsões mudam, enquanto o SHAP calcula a contribuição de cada característica para uma predição específica, destacando influências positivas e negativas. Estudos comparativos entre diferentes métodos de interpretação podem ajudar a identificar a abordagem mais eficaz em diferentes contextos e aplicações.

### 3 REDE NEURAL GRANULAR CONVOLUCIONAL EVOLUTIVA

Este capítulo propõe uma abordagem de modelagem que possibilita a adaptação de uma rede destinada ao processamento de fluxo de dados numéricos para lidar com fluxo de imagens. A combinação escolhida envolve a utilização da rede convolucional VGG-16 em conjunto com a rede neuro-fuzzy granular evolutiva (EGNN), conforme apresentado no Capítulo 2.

Camadas específicas da rede VGG-16 são empregadas para a extração de atributos das imagens digitais, simultaneamente. Já a rede EGNN é aplicada para proporcionar adaptação temporal *online* à VGG-16. A fusão dessas redes é realizada para transformar a rede granular evolutiva, com camadas de CNN, em uma versão de CNN evolutiva.

Adicionalmente, a técnica de redução de dimensionalidade denominada Análise de Componentes Principais (PCA) é aplicada de maneira criteriosa no ponto em que as redes neurais convergem. Esse ponto de convergência representa o estágio em que a extração de características das imagens digitais pela VGG-16 e a adaptação temporal *online* pela EGNN são integradas para formar uma única rede. A redução de dimensionalidade por meio de PCA é aplicada neste ponto para comprimir os dados enquanto preserva as características mais relevantes. Isso simplifica a transição entre as diferentes etapas do processamento e ajuda a reduzir os desafios associados à alta dimensionalidade dos dados, tornando os resultados da análise das imagens mais claros e precisos.

Dessa forma, as características extraídas pela VGG-16 e reduzidas em dimensionalidade pelo PCA são processadas pela EGNN de forma dinâmica, permitindo que a rede convolucional evolutiva se adapte continuamente aos novos padrões identificados nos fluxos de imagens. Essa fusão de redes transforma a rede granular evolutiva, com camadas de CNN, em uma versão evolutiva da CNN.

#### 3.1 Detalhes sobre os Conjuntos de Dados a serem Aprendidos

Para avaliar o desempenho do modelo proposto na classificação de fluxo de imagens, dois conjuntos de dados distintos foram empregados. O primeiro conjunto, denominado *Cats-vs-Dogs*, foi utilizado para testes iniciais com duas classes: gato e cachorro. O segundo conjunto, denominado *Common Objects Day and Night (CODaN)*, possibilitou a expansão da avaliação para um ambiente mais complexo, com 10 classes, sendo elas: bicicleta, carro, motocicleta, ônibus, barco, gato, cachorro, garrafa, xícara e cadeira.

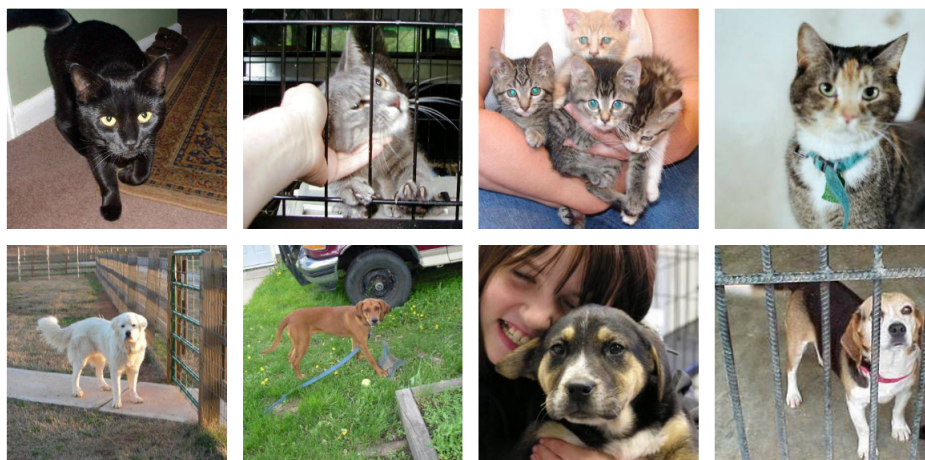
##### 3.1.1 Conjunto de Dados *Cats-vs-Dogs*

O conjunto de dados *Cats-vs-Dogs* ou Gato versus Cachorro foi obtido do *Kaggle Repository* e é dedicado à classificação de imagens de dois tipos de animais: gatos e cachorros. Estes dados foram disponibilizados em 2018 pelo repositório Kaggle (2018). As principais características desse conjunto são as seguintes: 5011 imagens de gatos e 5017 imagens de cachorros. As imagens neste conjunto apresentavam diferentes tamanhos, variando de  $(327 \times 500)$ ,  $(499 \times 375)$ , entre outros.

Na Figura 3.1 são apresentados alguns exemplos de imagens do conjunto de dados *Cats-vs-Dogs* com as duas classes utilizadas. Essa escolha foi feita para destacar a diversidade de amostras presentes no conjunto de dados e fornecer uma visão geral das características das imagens que o modelo precisou aprender a distinguir durante os testes. A inclusão desses exemplos visuais ajuda a ilustrar a natureza do problema de classificação binária e a variedade de poses, raças e ambientes em que os animais podem estar presentes nas imagens.



Figura 3.1 – Exemplos de imagens da base de dados *Cats-vs-Dogs*.



Fonte: Kaggle (2018).

Inicialmente, utilizou-se este conjunto de dados para testar a abordagem proposta e fornecer conhecimento sobre o desempenho do modelo em uma tarefa de classificação binária. O conjunto de dados *Cats-vs-Dogs* foi escolhido devido à sua simplicidade e natureza binária, contendo apenas duas classes: gatos e cachorros. Essa escolha permitiu uma avaliação preliminar do desempenho do modelo proposto em uma tarefa de classificação binária antes de lidar com um ambiente mais complexo com 10 classes. Além disso, a disponibilidade e o tamanho adequado desse conjunto de dados o tornaram uma opção viável para testes iniciais e ajustes de parâmetros do modelo.

### 3.1.2 Conjunto de Dados de Objetos Comuns Dia e Noite (CODaN)

O conjunto de dados *Common Objects Day and Night (CODaN)* foi selecionado para avaliação mais abrangente, considerando um ambiente de classificação de imagens em 10 classes no domínio diurno. Essas classes abrangem uma variedade de objetos cotidianos e foram extraídas de três conjuntos de dados distintos:

- (a) o COCO que é um conjunto de dados aberto que se concentra em objetos comuns em contextos complexos. Ele contém imagens com anotações detalhadas para objetos em 80 categorias (Lengyel *et al.*, 2021; Lin *et al.*, 2014);
- (b) o ImageNet que é um conjunto com mais de 15 milhões de imagens que estavam classificadas em mais de 22 mil categorias ou classes distintas. Cada classe representa um tipo específico de objeto, animal, pessoa, cenário, entre outros conceitos que podem ser encontrados em imagens (Rezende *et al.*, 2020);
- (c) o ExDark que é um conjunto de dados específico para condições de iluminação adversas, como ambientes noturnos. Ele é projetado para avaliação de algoritmos em situações desafiadoras de baixa luminosidade (Tan, 2008).

As classes incluídas no conjunto de dados CODaN são: bicicleta, carro, motocicleta, ônibus, barco, gato, cachorro, garrafa, xícara e cadeira. As principais características desse conjunto incluem 10.000 imagens de treinamento para a CNN e 2.500 imagens de teste, com 250 imagens disponíveis para cada classe. É importante ressaltar que para a avaliação foram

utilizadas exclusivamente imagens do domínio diurno do conjunto de dados CODaN. Apenas as imagens de treinamento e teste diurnas foram disponibilizadas, sendo que as imagens noturnas foram utilizadas somente para teste posteriormente. A base de dados CODaN utilizada neste estudo foi fornecida por Lengyel *et al.* (2021) e já estava previamente organizada, filtrada e recortada em dimensões de  $(224 \times 224)$  para garantir dimensões uniformes e exclusividade entre as classes. Todas as imagens foram processadas de acordo com os padrões estabelecidos pela fonte, o que possibilitou sua utilização no treinamento e teste do modelo proposto.

Na Figura 3.2 são apresentados alguns exemplos de imagens do conjunto de dados CODaN, utilizado com dez classes.

Figura 3.2 – Exemplos de imagens da base de dados CODaN.



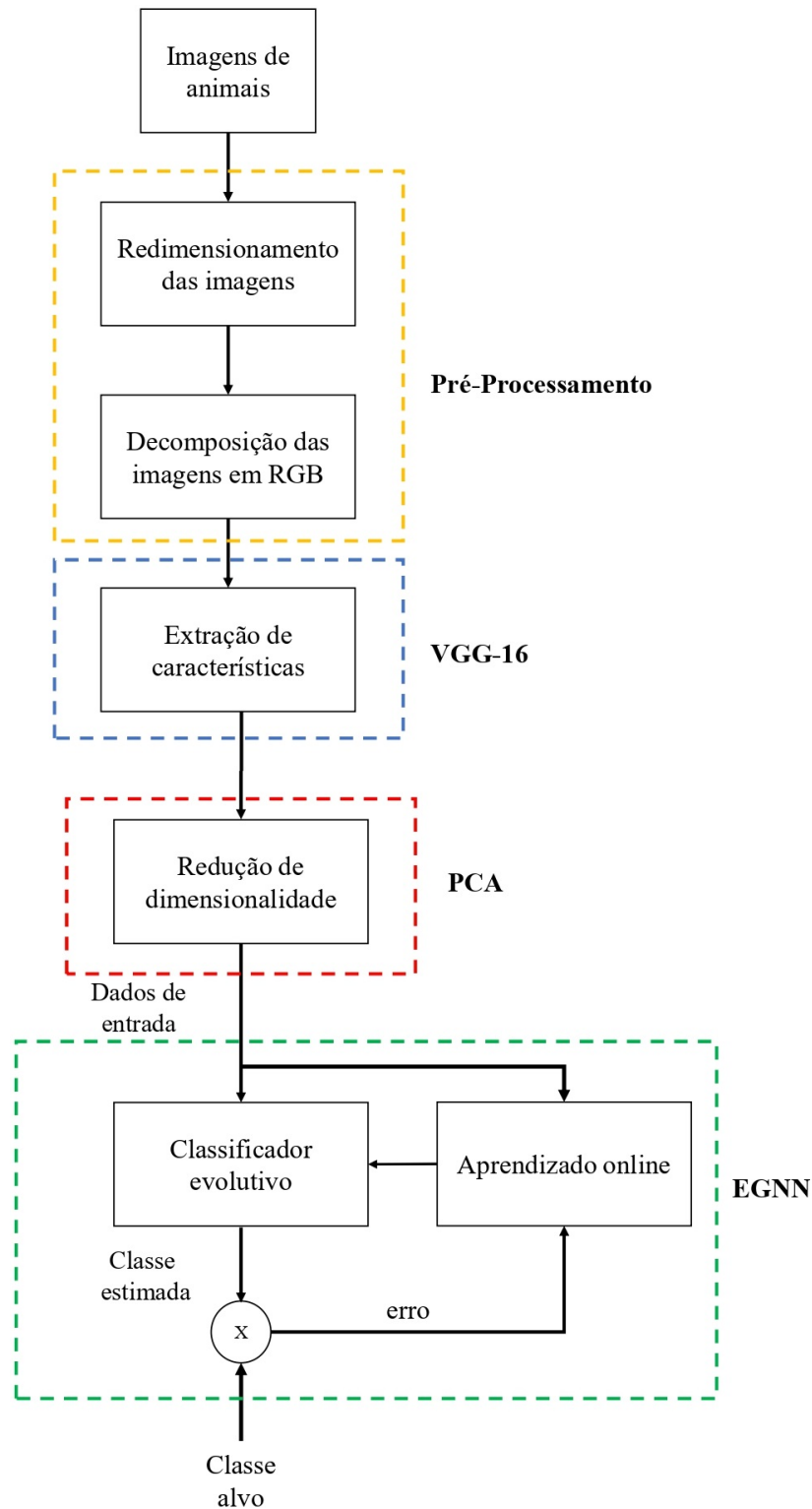
Fonte: Lengyel *et al.* (2021).

### 3.2 Proposta para Classificação de Imagens

O sistema de monitoramento evolutivo proposto para o procedimento de rastreamento e análise de imagens digitais é representado no diagrama da Figura 3.3. As seguintes etapas são destacadas: o pré-processamento, a rede neural convolucional VGG-16, a Análise de Componentes Principais (PCA) e a rede neuro-fuzzy granular evolutiva (EGNN).

Inicialmente, o fluxo de dados passa por uma fase de pré-processamento que inclui o redimensionamento das imagens garantindo uniformidade e otimização para análise subsequente. Durante essa etapa, ocorre a decomposição dos canais RGB, na qual cada imagem é separada em seus três componentes de cor principais: vermelho (*Red*), verde (*Green*) e azul (*Blue*). Essa decomposição é fundamental para a extração de características, pois permite que o modelo analise padrões complexos e representações discriminativas diretamente das imagens originais. Cada canal de cor é tratado individualmente, criando uma imagem separada para cada um deles. Essas imagens resultantes são representadas por matrizes bidimensionais, onde cada valor de pixel corresponde à intensidade da cor naquela posição da imagem original (Backes; Junior, 2016). Essa abordagem possibilita que as características específicas de cada canal de cor sejam analisadas separadamente, ou combinadas de diferentes maneiras para atingir efeitos visuais desejados. Portanto, ao explorar os diferentes canais de cores, o modelo é capaz de compreender e interpretar de forma mais abrangente as combinações visuais das imagens, contribuindo para uma análise mais precisa e detalhada.

Figura 3.3 – Sistema de monitoramento convolucional granular evolutivo.



Fonte: Da autora (2024).

A etapa seguinte envolve a aplicação da rede neural convolucional VGG-16 para extração de atributos. Utilizando transferência de aprendizado, a VGG-16 desempenha um papel importante na geração de representações discriminativas dimensionais para cada imagem, denotada por  $x$  de cada imagem (Xiaowei; Miqing, 2021). Essa abordagem não apenas acelerou

o processo de treinamento da rede, mas também melhorou a generalização e a capacidade de classificação do sistema, especialmente em cenários com conjuntos de dados limitados. Esse processo é fundamental para a construção de uma base sólida de características relevantes à tarefa de classificação. Durante o processo de extração de características pela VGG-16, esses canais de cores são analisados para identificar padrões complexos e representações discriminativas diretamente das imagens originais. O que consiste em extrair cada canal de cor e criar uma imagem separada para cada um deles. Essas imagens resultantes são matrizes bidimensionais, onde cada valor de pixel representa a intensidade da cor correspondente naquela posição da imagem original. Por exemplo, em uma imagem RGB, um pixel na posição  $(x, y)$  terá três valores associados a ele: um para o canal vermelho, um para o canal verde e um para o canal azul.

Posteriormente, a análise de componentes principais é empregada para reduzir a dimensionalidade dos dados que serão fornecidos para a rede granular evolutiva. Esse passo tem como objetivo simplificar a complexidade do conjunto de características, preservando ao mesmo tempo as informações mais relevantes. A redução de dimensionalidade otimiza a eficiência computacional e facilita a transição entre as diferentes etapas do processamento.

Por fim, as componentes principais são fornecidas como entrada para uma Rede Neuro-Fuzzy Granular Evolutiva (EGNN), que atua como um classificador evolutivo. Esta rede dinâmica e adaptativa é alimentada com as características processadas pela VGG-16 e reduzidas pelo PCA. A EGNN realiza a classificação das imagens e continua a aprender de forma incremental mesmo após a implementação inicial. O modelo é construído de maneira *online*, utilizando os atributos extraídos de cada imagem de animal. A estimativa de classe é realizada após a apresentação de uma imagem no sistema. Para aprimorar a precisão, um passo incremental de aprendizado é aplicado exclusivamente na EGNN, visando minimizar erros de estimação para imagens semelhantes que possam ser apresentadas no futuro.

### 3.3 Evolução de Grânulos de Informação a partir do Fluxo de Imagens

As camadas convolucionais da rede VGG-16 desempenham o papel de extrair características da imagem, enquanto as últimas camadas utilizam essas características para classificar a imagem de entrada. Em muitas arquiteturas de rede neural convolucional, a última camada com pesos ajustáveis é uma camada totalmente conectada. O presente trabalho propôs substituir as camadas totalmente conectadas da VGG-16 por uma EGNN para a aprendizagem *online* de novas imagens, configurando-a com um número de saídas igual ao número de classes no novo conjunto de dados.

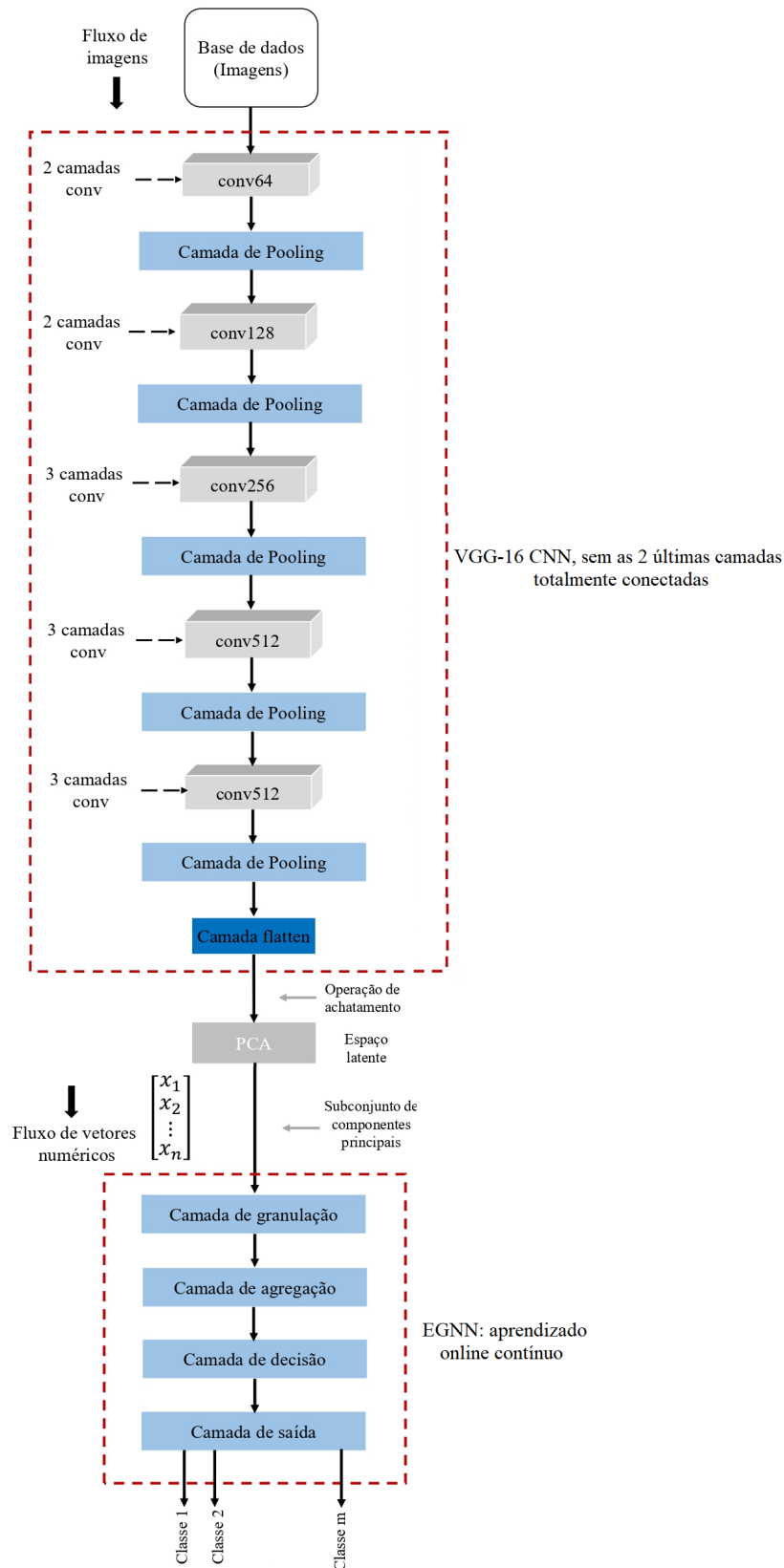
É relevante ressaltar que todo esse processo de substituição e adaptação ocorreu em um ambiente computacional significativamente avançado. O *hardware* utilizado, caracterizado por um *Inspiron 24 All in One* 13ª geração Intel® Core™ i7-1355U, Windows 11 Home, Intel® Iris® Xe Graphics, 16 GB DDR4, 512 GB SSD, desempenhou um papel fundamental na eficiência do processamento. Essas características do computador garantiram a capacidade de lidar com a complexidade da EGNN e a adaptabilidade da VGG-16, proporcionando uma implementação eficaz e rápida dessa metodologia. O algoritmo foi desenvolvido na linguagem *Python*, utilizando o ambiente *Spyder* e o *Anaconda*, fornece informações importantes sobre a linguagem de programação e as ferramentas utilizadas para a implementação, o que é relevante para a comunidade científica e para aqueles que desejam replicar ou entender melhor o trabalho.

O modelo VGG-16 foi adaptado de modo a preservar suas camadas iniciais pré-treinadas, mas substituindo a camada totalmente conectada final pela arquitetura de uma eGNN. Os pesos do modelo VGG-16, inicialmente obtidos a partir da classificação no conjunto de dados Image-

Net, foram mantidos, com apenas as camadas finais sendo modificadas para se adequarem ao novo contexto. A Figura 3.4 apresenta o funcionamento da arquitetura proposta, destacando a substituição da camada de classificação da VGG-16 pela eGNN.

Como mencionado, a rede VGG-16 requer imagens de entrada com dimensões de  $(224 \times 224)$  pixels. Portanto, antes de alimentar a VGG-16 com o fluxo de imagens, foi realizado um pré-processamento, que incluiu redimensionar as imagens para atender a essas dimensões específicas, pois as imagens originais poderiam ter tamanhos diferentes. Esse processo garantiu que todas as imagens de entrada tivessem as mesmas dimensões e fossem processadas pela rede. Logo após foi realizada a normalização das imagens entre 0 e 1. Essa etapa é fundamental para garantir uma consistência nas características estatísticas das imagens e proporcionar uma melhor estabilidade durante o treinamento da rede neural convolucional. Posteriormente, o conjunto de dados passou a ter um tamanho fixo de  $(224 \times 224)$  pixels e canais RGB (*red*, *green* e *blue*). Portanto, a entrada da rede foi uma matriz tridimensional de tamanho  $(224 \times 224 \times 3)$ .

Figura 3.4 – Diagrama da estrutura do modelo CEGNN proposta.



Fonte: Da autora (2024).

A VGG-16 é composta por várias camadas convolucionais empilhadas uma após a outra. Cada camada convolucional aplica uma operação de convolução nas imagens de entrada usando

um conjunto de filtros, chamados kernel, aprendidos durante o treinamento. Esses filtros são pequenas matrizes geralmente de tamanho  $(3 \times 3)$  que são deslizadas sobre a imagem de entrada, multiplicando e somando os valores dos pixels correspondentes. Essa operação de convolução pode ser expressa matematicamente pela seguinte equação:

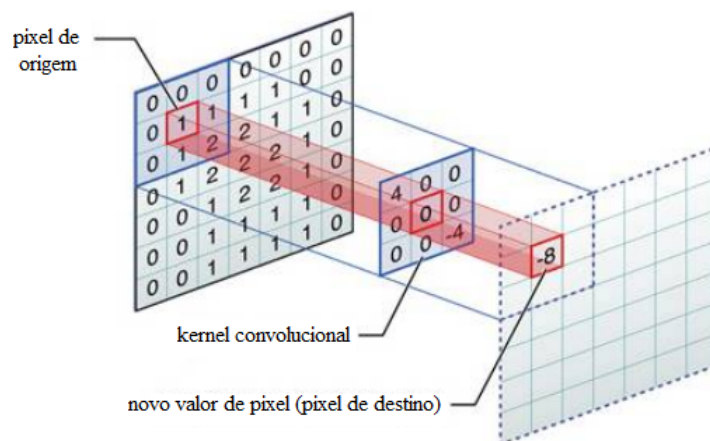
$$S(i, j) = \sum_{m=1}^3 \sum_{n=1}^3 I(i+m-1, j+n-1) \cdot F(m, n) \quad (3.1)$$

Onde:

- $S(i, j)$  representa o valor do pixel na posição  $(i, j)$  da imagem resultante;
- $I(i+m-1, j+n-1)$  é o valor do pixel na posição deslocada  $(i+m-1, j+n-1)$  na imagem de entrada;
- $F(m, n)$  é o valor do filtro na posição  $(m, n)$ .

Essa equação descreve como cada *pixel* na imagem resultante é calculado pela soma ponderada dos pixels na região correspondente da imagem de entrada, ponderados pelos valores do filtro, capturando características visuais locais, como bordas, texturas e padrões simples. A convolução é concluída quando o filtro percorre toda a imagem. Durante esse processo, cada posição da imagem é visitada, e em cada posição, é calculado um valor resultante da convolução. O resultado final é uma nova matriz, chamada de mapa de características. Essa matriz tem dimensões menores que a imagem original, já que o filtro é menor. Considerando o tamanho da imagem de entrada de  $(224 \times 224)$  pixels, o último mapa de características terá dimensões  $(7 \times 7)$  pixels para cada filtro utilizado. A convolução da rede preserva a relação espacial entre os pixels, aprendendo as características da imagem a partir de suas diferentes áreas. A Figura 3.5 mostra o mecanismo de convolução de uma imagem.

Figura 3.5 – Operação de matrizes (imagem x kernel).



Fonte: Hochuli e Oliveira (2016).

Após a realização das convoluções em cada camada, a VGG-16 possui camadas *max pooling* para reduzir a dimensionalidade dos mapas de características. A VGG-16 utiliza a camada *max pooling* com uma janela de  $(2 \times 2)$  e um passo de 2. Isso reduz pela metade a resolução espacial dos mapas de características. Após as camadas convolucionais e de *max pooling*, a VGG-16 tem uma camada *flatten*. Essa camada transforma os mapas de características 2D em um vetor 1D. Por exemplo, se o mapa de características tiver dimensões  $(7 \times 7 \times 512)$ , a camada



*flatten* o transformará em um vetor de tamanho  $(7 \times 7 \times 512) = 25.088$ . O modelo processou cada imagem do fluxo de entrada em forma matricial e em seguida as empilhou em um vetor 1D, sendo a quantidade de vetores  $n$  correspondente à quantidade de imagens de entrada.

Ao invés de utilizar diretamente o vetor obtido da camada *flatten*, a implementação específica deste trabalho optou pelo vetor de 4096 características proveniente da camada *dense1*. Essa camada, com 4096 neurônios, representa uma etapa significativa de redução de dimensionalidade e captura informações de alto nível sobre as características extraídas nas camadas convolucionais anteriores. A escolha desse vetor de 4096 características visou preservar informações relevantes e compactas, as quais em uma etapa subsequente, foi realizada a redução de dimensionalidade do vetor extraído da rede convolucional VGG-16.

O método PCA foi aplicado para extrair um novo vetor, que é uma combinação linear dos atributos do vetor original. No processo de PCA, as etapas típicas incluem a centralização dos dados, ajustando-os de modo que a média coincida com o ponto central do sistema de coordenadas. Em seguida foi feito o cálculo da matriz de covariância para entender as relações entre diferentes características do conjunto de dados. A obtenção dos autovalores e autovetores da matriz de covariância é crucial no PCA, destacando as direções principais nos dados e a importância relativa dessas direções. Ao ordenar os autovalores em ordem decrescente, foram identificadas as direções associadas à maior variabilidade nos dados. Foram projetados então, os dados nos autovetores selecionados, determinando a nova dimensionalidade do conjunto de características. Essa abordagem metódica do PCA foi fundamental para preservar a essência das características extraídas pela VGG-16, mesmo em um espaço dimensional menor. A transformação resultante simplificou o conjunto de dados, maximizando a variabilidade e proporcionando um vetor mais compacto e representativo do conjunto de características para análises seguintes.

Inicialmente, o processo foi iniciado considerando as 5 componentes melhor rankeadas segundo seus auto-valores e a base de dados de treinamento. Essa abordagem permitiu uma exploração abrangente das diferentes configurações, avaliando o impacto da dimensionalidade do conjunto de características no desempenho do classificador. O vetor resultante da aplicação do PCA, representado por  $x = [x_1, x_2, \dots, x_n]'$ , foi fornecido como entrada da rede granular evolutiva. Em seguida, foi considerado o fluxo de vetores de dados numéricos e suas respectivas classes, expressos como  $(x, y)^{[h]}$ ,  $h = 1, \dots$ , então, a combinação PCA-eGNN foi aplicada de maneira supervisionada, isto é, usando os atributos  $x_j$  e a classe verdadeira  $y$ .

O modelo eGNN, usado no topo da rede VGG-16, foi responsável por processar dados numéricos usando granulos de informação, que são estruturas granulares baseadas em lógica fuzzy, operadores de agregação e um algoritmo de aprendizado incremental. Os grânulos de informação foram responsáveis por codificar e representar a incerteza inerente aos dados, oferecendo uma abordagem robusta para lidar com complexidades e variações nos padrões de imagem. Os operadores de agregação permitiram combinar eficientemente as informações dos grânulos, enquanto o algoritmo de aprendizado incremental possibilitou a adaptação contínua do modelo com a apresentação de novos dados. Essa abordagem dinâmica e evolutiva contribuiu para aprimorar a representação do sistema ao longo do tempo, permitindo uma capacidade de classificação robusta e adaptável. A eGNN não apenas processou os dados numéricos resultantes do PCA, mas também incorporou a incerteza e a capacidade de aprendizado contínuo, tornando-a uma peça fundamental para a eficácia do modelo na classificação de imagens.

Durante a fase experimental, além da variação da quantidade de PCAs outros parâmetros foram variados e ajustados para otimizar o desempenho do modelo como a granularidade ( $\rho$ ), o número de iterações para adaptação da granularidade ( $h_r$ ), a constante de redução de pesos ( $\beta$ ) e o número inicial de grânulos ( $c$ ). Após a adaptação das redes escolhidas para extração de



características e classificação em relação ao problema proposto, foi iniciada a etapa de utilização de métricas para avaliação do modelo.

### 3.4 Métricas para Avaliação dos Resultados

Nesta seção são apresentadas as métricas utilizadas para avaliação do modelo CEGNN para a tarefa de classificação de imagens. As métricas utilizadas incluem acurácia, precisão, *Recall*, *F1-Score*, matriz de confusão e a curva ROC (*Receiver Operating Characteristic*) (Géron, 2019).

A acurácia ( $A_{CC}$ ) avalia a proporção de previsões corretas, representadas pelos verdadeiros positivos e verdadeiros negativos, em relação ao total de entradas avaliadas.

$$A_{CC} = \frac{VP + VN}{VP + VN + FP + FN} \cdot 100\%, \quad (3.2)$$

onde VP e VN são os verdadeiros positivos e verdadeiros negativos, respectivamente; FP e FN são os falsos positivos e falsos negativos, respectivamente.

Os verdadeiros positivos representam as previsões corretas de ocorrência de uma classe, os verdadeiros negativos representam as previsões corretas de ausência da classe, os falsos positivos representam as previsões incorretas de ocorrência da classe e os falsos negativos representam as previsões incorretas de ausência da classe.

A precisão, métrica que avalia a quantidade de verdadeiros positivos sobre a soma de todos os valores positivos é dada por:

$$\text{Precisão} = \frac{\text{Verdadeiro Positivo}}{\text{Verdadeiro Positivo} + \text{Falso Positivo}} \quad (3.3)$$

O *Recall*, também conhecido como taxa ou sensibilidade de verdadeiros positivos, mede a proporção de exemplos positivos corretamente identificados pelo modelo em relação ao número total de exemplos que são verdadeiramente positivos, e é dado por:

$$\text{Recall} = \frac{\text{Verdadeiro Positivo}}{\text{Verdadeiro Positivo} + \text{Falso Negativo}} \quad (3.4)$$

O *F1-Score* é uma métrica que combina precisão e *Recall* em uma única medida. É usado quando se quer considerar os dois de forma equilibrada, e é dado por:

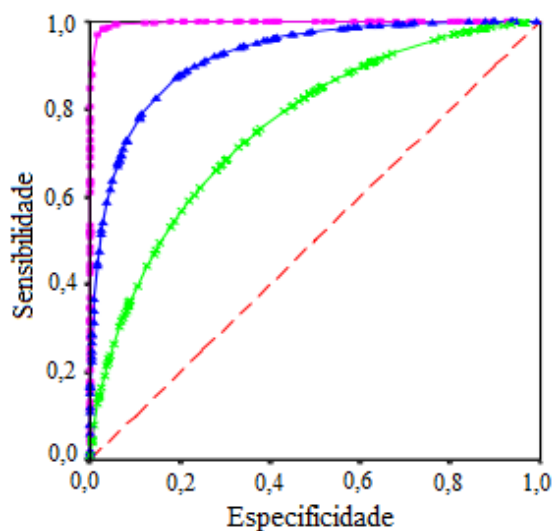
$$F1\text{-Score} = 2 \times \frac{\text{Precisão} \times \text{Recall}}{\text{Precisão} + \text{Recall}} \quad (3.5)$$

As métricas precisão, *Recall* e *F1-Score* variam entre 0 e 1, sendo que 1 representa o desempenho ideal e 0 o desempenho mais baixo. A matriz de confusão também foi obtida, pois oferece uma medida eficaz do modelo de classificação, mostrando o número de classificações corretas *versus* as classificações erradas para cada classe, em um conjunto de N exemplos. Cada coluna da matriz representa as amostras de uma classe prevista, enquanto as linhas representam os casos de uma classe estimada pelo modelo (Santana; Leite, 2020).

Por fim, com o objetivo de analisar graficamente o modelo, foi obtida a curva ROC (*Receiver Operating Characteristic*), capaz de demonstrar o desempenho de um modelo de *machine learning* pela relação entre a Taxa de Verdadeiros Positivos (Sensibilidade) e a Taxa de Falsos Positivos (Especificidade), variando o limiar (ponto de corte na probabilidade estimada).

A Figura 3.6 apresenta a ilustração de uma curva ROC, onde a cor rosa representa a curva para um classificador ideal e quanto mais a curva se aproxima do tracejado vermelho, inferior é o desempenho do classificador.

Figura 3.6 – Ilustração de uma curva ROC.



Fonte: Braga (2020).

A análise da curva ROC fornece ferramentas para selecionar modelos possivelmente ideais (modelos ótimos) e descartar modelos não tão ótimos, independentemente do contexto de custos ou da distribuição de classe (Braga, 2020).

### 3.5 Síntese

Este capítulo apresentou o diagrama do sistema proposto para o trabalho e a arquitetura da rede VGG convencional a ser utilizada, juntamente com o ImageNet, na qual foi treinada. Além disso, introduziu a estrutura da modelagem de fluxo de dados de imagens proposta, baseada na combinação da rede VGG-16 para extração de atributos das imagens digitais e na abordagem da rede neural granular fuzzy evolutiva para classificação das imagens. A estrutura proposta consiste na substituição das camadas totalmente conectadas da VGG-16 por uma EGNN para aprendizagem *online*. O capítulo também abordou a utilização de métricas para avaliação do modelo proposto.

## 4 RESULTADOS

Neste capítulo, são apresentados os resultados obtidos a partir da avaliação do modelo treinado utilizando o conjunto de dados *Cats-vs-Dogs* e CODaN. Métricas relevantes, como acurácia, precisão, *Recall*, *F1-Score*, matriz de confusão e a curva ROC, foram incluídas para avaliar o desempenho do modelo. Essas métricas são consideradas essenciais para uma análise abrangente do desempenho do modelo em diferentes aspectos. Os resultados apresentados ofereceram uma visão detalhada do desempenho do modelo, permitindo uma análise crítica de sua eficácia em relação aos objetivos propostos. Quaisquer ajustes ou otimizações adicionais necessários serão discutidos em relação à esses resultados, proporcionando reflexões para futuras iterações do modelo.

Alguns cenários também foram analisados como a média de regras que representa a quantidade média de regras geradas pelo modelo durante o processo de classificação. Essas regras são padrões identificados pelo algoritmo para diferenciar as duas classes, e a média de regras fornece uma medida da complexidade do modelo. Um número maior de regras pode indicar uma maior complexidade ou sobreajuste do modelo aos dados.

O tempo de processamento necessário para que o algoritmo atinja a convergência durante o treinamento também foi considerado. Geralmente, um tempo de convergência menor é desejável, pois indica uma eficiência maior do algoritmo. É importante notar que o tempo de convergência pode variar dependendo de vários fatores, incluindo a complexidade do modelo, o tamanho do conjunto de dados, a quantidade de épocas de treinamento e a capacidade computacional disponível. Portanto, é possível que o tempo de convergência para o conjunto de dados de duas classes seja superior ao do conjunto de dados de dez classes devido a diferenças na complexidade dos modelos ou nas características dos conjuntos de dados.

Os testes foram realizados em um sistema com especificações robusto, como mencionado no capítulo anterior. Essa configuração proporcionou um ambiente de teste confiável e eficiente para avaliar o desempenho do modelo, garantindo resultados precisos e consistentes.

### 4.1 Desempenho na Classificação de duas Classes

Os resultados relacionados ao conjunto de dados de duas classes, com foco na classificação entre Gato e Cachorro, são apresentados na Tabela 4.1. A eficácia do esquema de rede híbrida proposto, CEGNN, foi destacada por meio dos resultados de métricas de desempenho como acurácia, precisão, *Recall* e *F1-Score*. Essas métricas permitiram uma avaliação da capacidade do modelo CEGNN em distinguir entre as classes de Gato e Cachorro. A acurácia indicou a proporção de classificações corretas em relação ao total de classificações, enquanto a precisão mediu a proporção de instâncias positivas corretamente classificadas em relação ao total de instâncias classificadas como positivas. O *Recall*, por sua vez, avaliou a proporção de instâncias positivas corretamente identificadas em relação ao total de instâncias positivas. Por fim, o *F1-score* mediu equilíbrio entre precisão e *Recall*.

Tabela 4.1 – CEGNN: Desempenho de classificação de imagens para duas classes

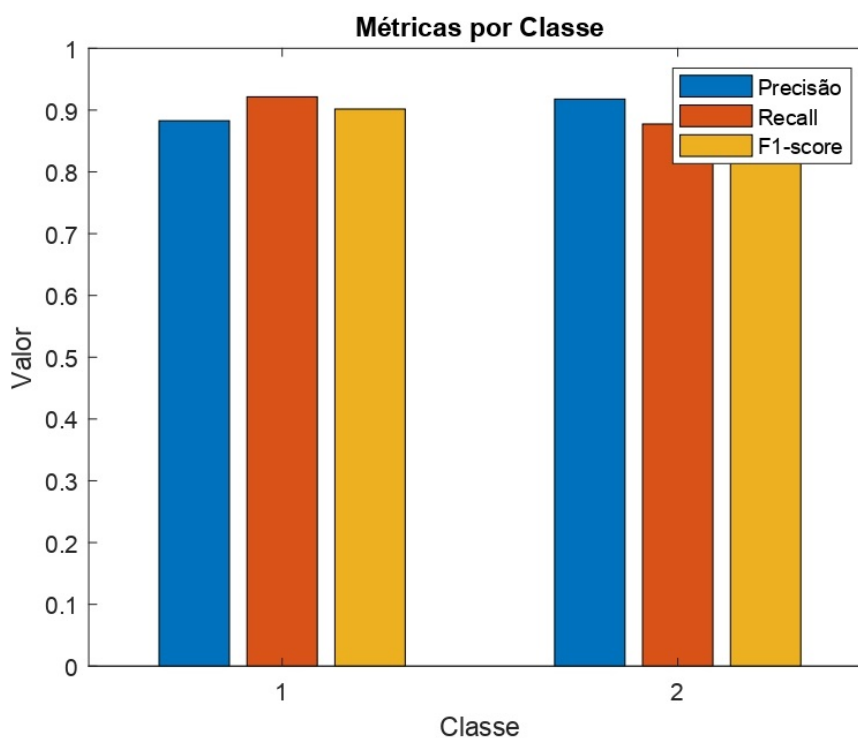
Características	$A_{CC}$ (%)	Média de regras	Tempo (s)	Precisão	Recall	F1
5	91,23	66,07	115,8	0,91	0,91	0,90
10	89,03	136,03	116,2	0,88	0,88	0,89
15	88,70	172,72	123,2	0,85	0,92	0,89
20	86,94	195,05	511,6	0,86	0,86	0,86

Fonte: Da autora (2024).

Notavelmente, o desempenho mais expressivo foi alcançado ao incorporar um conjunto de novas variáveis, nomeadamente os cinco primeiros componentes principais do conjunto de dados original. Nessa configuração, a performance total alcançou uma taxa de acerto de 91,23% para as classes consideradas com um tempo de processamento de 115,8 segundos. A precisão do modelo foi avaliada em 0,91, indicando sua habilidade em evitar classificar erroneamente instâncias negativas como positivas. Além disso, o *Recall* atingiu 0,91, evidenciando a capacidade do algoritmo em identificar corretamente todas as instâncias positivas.

Um aspecto crucial foi o valor do *F1-Score*, que atingiu 0,90. Este índice reflete um equilíbrio efetivo entre precisão e *Recall*, sublinhando o desempenho do modelo em termos de acurácia e capacidade de identificação correta. Esses resultados ressaltam a eficiência do esquema proposto, sugerindo sua aplicabilidade promissora para a tarefa de classificação, com especial destaque para a gestão de instâncias positivas e negativas. A Figura 4.1 exibe, de forma gráfica, os resultados provenientes das métricas utilizadas na avaliação do modelo para cada classe. Observou-se que o modelo demonstrou maior precisão na classificação da classe (1), ao passo que apresentou uma precisão menor na análise da classe (0).

Figura 4.1 – Métricas de avaliação do modelo para duas classes.



Fonte: Da autora (2024).

A Figura 4.2 exibe a matriz de confusão para CEGNN, destacando a contagem de classificações precisas em comparação com aquelas incorretas para as imagens de Gato e Cachorro. Para a análise seguinte, a matriz de confusão possui duas classes distintas, sendo a classe (0) referente a imagens de cachorros e a classe (1) a imagens de gatos. O modelo alcançou uma classificação de 91,23% para o fluxo de imagens usando cinco características extraídas por meio da análise de componentes principais. O tempo de processamento dos dados foi de 115,80 segundos para todas as amostras, utilizando granularidades iniciais pequenas, como  $\rho = 0,5$ , um total de 1000 iterações, sendo uma iteração por instância de entrada de teste e aproximadamente 86 regras.

Figura 4.2 – Matriz de confusão para a classificação de duas classes.

**Matriz de confusão**

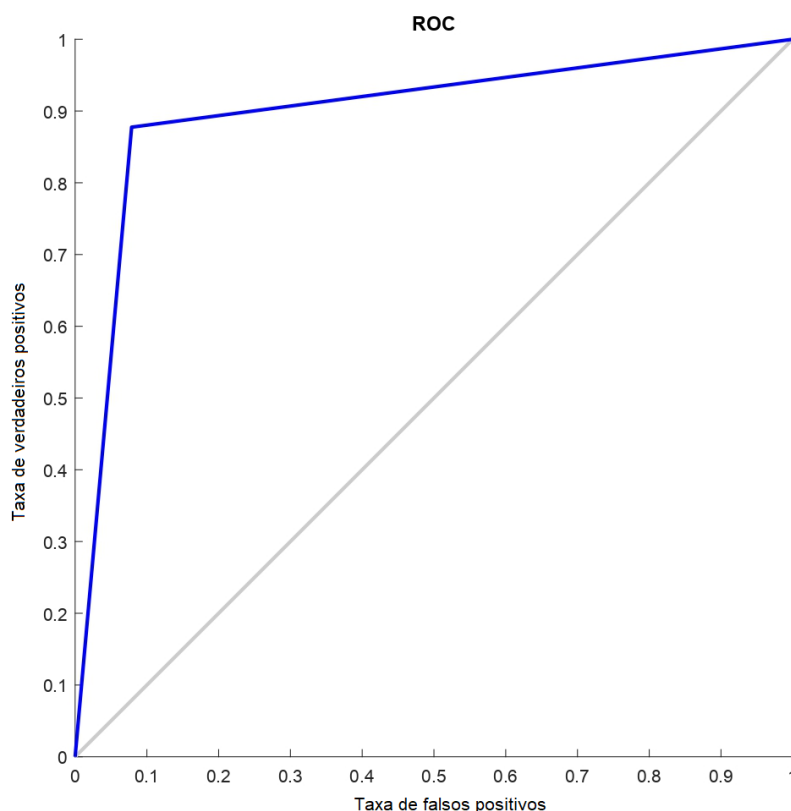
Classe de saída	0	4695	557	89,40%
	1	322	4454	93,30%
		93,60%	88,90%	<b>91,20%</b>
		0	1	

Fonte: Da autora (2024).

Observou-se que, entre as 5017 amostras de imagens contendo cachorros, 322 foram erroneamente classificadas como imagens contendo gatos, enquanto 4695 foram corretamente classificadas. Ao contrário, entre 5011 amostras contendo imagens de gato, 557 foram erroneamente classificadas como contendo cachorros, e 4454 foram corretamente classificadas.

A Figura 4.3 apresenta a curva ROC. Ela oferece uma representação visual da taxa de verdadeiros positivos em relação à taxa de falsos positivos para um modelo específico, no contexto de classificação binária. A análise da curva ROC atesta a capacidade discriminativa do modelo, proporcionando percepções sobre a sua eficácia em diferentes pontos de operação ao variar o limiar de decisão.

Figura 4.3 – Ilustração da curva ROC na classificação de duas classes.



Fonte: Da autora (2024).

Notou-se que a curva ROC inicialmente sobe com um ângulo relativamente suave. Notavelmente, ao atingir uma taxa de verdadeiros positivos (sensibilidade) de 0,89, foi observado que a curva está associada a um ponto específico no eixo x, onde a taxa de falsos positivos (especificidade) foi de aproximadamente 0,7. Esse ponto particular indica um limiar de decisão específico, ressaltando a capacidade do modelo de alcançar uma sensibilidade de 90% mantendo uma taxa de falsos positivos em torno de 70%. Essa análise foi importante para compreender como o modelo se comporta em termos de equilíbrio entre identificar corretamente instâncias positivas e limitar classificações incorretas.

À medida que a Curva ROC continua, ela assume uma forma de reta horizontal levemente inclinada, estendendo-se até o ponto em que a sensibilidade atingiu 90% ( $y = 0,9$ ), enquanto a taxa de falsos positivos cresceu moderadamente até atingir 99% ( $x = 0,99$ ) no eixo x. Essa seção da curva indica a manutenção de uma elevada sensibilidade, sugerindo que o modelo continua a identificar corretamente a grande maioria das instâncias positivas. No entanto, há um aumento moderado na taxa de Falsos Positivos, indicando uma propensão do modelo a classificar incorretamente algumas instâncias negativas.

A análise detalhada da curva ROC revela um ponto crítico e uma transição distintiva, onde o modelo alcança uma Sensibilidade de 90% com uma taxa de falsos positivos de 70%, seguida por uma fase de manutenção de Sensibilidade elevada, acompanhada de um aumento moderado na taxa de Falsos Positivos até atingir 99%. Essa compreensão foi essencial para otimizar e ajustar o modelo, adaptando-se às demandas específicas do contexto e fornecendo uma visão clara do equilíbrio entre verdadeiros positivos e falsos positivos em diferentes pontos de corte, especialmente valiosa em problemas de classificação binária, como a distinção entre Gato e Cachorro.

## 4.2 Desempenho na Classificação de dez Classes

Os resultados relacionados a classificação do conjunto de dados de 10 classes são apresentados na Tabela 4.2. A eficácia do esquema de rede híbrida proposto, CEGNN, foi destacada por meio dos resultados obtidos por meio de métricas de desempenho como acurácia, precisão, *Recall* e *F1-score*. Essas métricas ofereceram uma avaliação do quão bem o modelo CEGNN foi capaz de distinguir entre as classes de bicicleta, carro, motocicleta, ônibus, barco, gato, cachorro, garrafa, xícara e cadeira.

Tabela 4.2 – CEGNN: Desempenho de classificação de imagens para dez classes

Características	ACC (%)	Média de regras	Tempo (s)	Precisão	Recall	F1
5	62,08	37,59	25,98	0,62	0,62	0,61
10	78,88	37,75	24,51	0,79	0,78	0,79
15	73,80	39,31	26,85	0,76	0,73	0,74
20	71,25	82,37	71,25	0,72	0,71	0,71

Fonte: Da autora (2024).

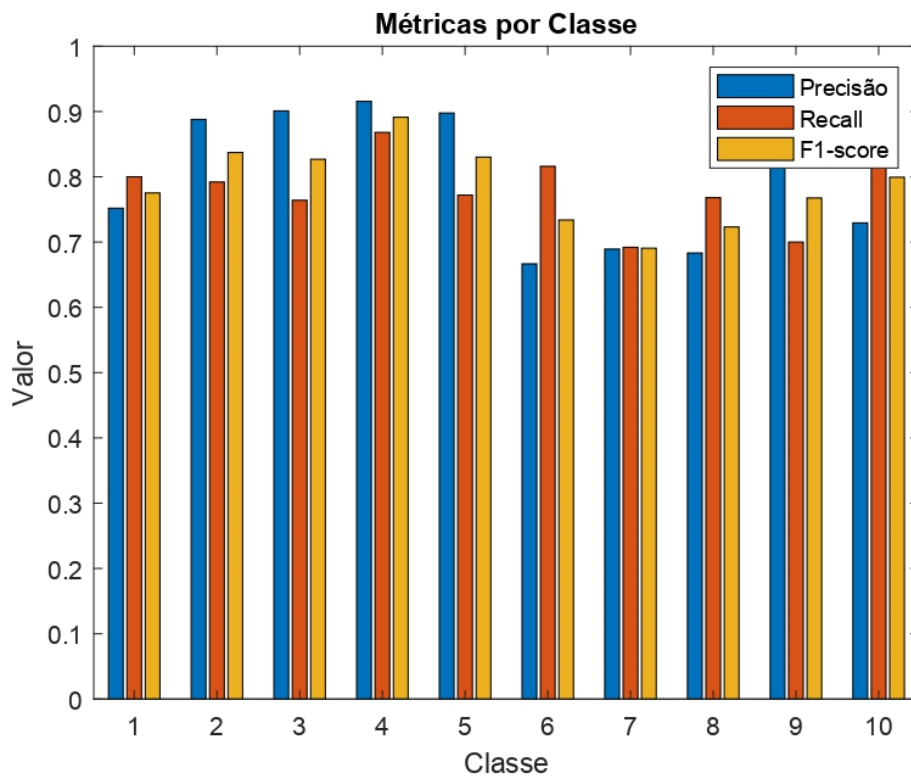
Notou-se que um aprimoramento de desempenho foi alcançado ao se utilizar 10 componentes principais na análise. Nessa configuração expandida, a acurácia global atingiu a marca de 78,88% com um tempo de processamento de 24,51 segundos. A precisão do modelo foi calculada em 0,79, indicando eficácia em evitar classificações equivocadas de instâncias negativas como positivas. Além disso, a taxa de *Recall* atingiu 0,78, demonstrando a capacidade do

algoritmo em identificar corretamente todas as instâncias positivas. O índice *F1-Score* atingiu 0,79, indicando um equilíbrio sólido entre precisão e *Recall*.

Quando se está lidando com um conjunto de dados de 10 classes, como é o caso mencionado, a complexidade e a diversidade das características presentes nas imagens podem ser significativamente maiores do que em um conjunto de dados com apenas 2 classes. Isso ocorre porque, ao lidar com mais classes, há uma ampla gama de objetos, formas, texturas e contextos visuais que precisam ser distintamente reconhecidos e classificados pelo modelo. Em contrapartida, com um número maior de PCAs, o modelo pode capturar mais variações nos dados do que com apenas 10 PCAs. Isso pode resultar em uma representação mais detalhada das características das imagens no espaço de características. No entanto, se a complexidade adicional não estiver alinhada com a estrutura dos dados ou se introduzir ruído desnecessário, isso pode afetar negativamente o desempenho do modelo. Nem sempre um número maior de componentes principais significa melhor desempenho. A seleção inadequada de PCAs pode levar a uma representação inadequada dos dados ou à inclusão de características irrelevantes, o que pode prejudicar o desempenho do modelo.

A Figura 4.4 exibe, de forma gráfica, os resultados provenientes das métricas utilizadas na avaliação do modelo para cada classe. Observou-se que o modelo demonstrou maior precisão na classificação da classe (2), seguido pela classe (4) ao passo que apresentou uma precisão menor na análise da classe (7). Essa disparidade nos resultados destacou a necessidade de uma investigação mais aprofundada para compreender as razões por trás dessa diferença de desempenho entre as classes específicas para refinamentos futuros do modelo, visando melhorar a precisão em ambas as classes.

Figura 4.4 – Métricas de avaliação do modelo para dez classes.



Fonte: Da autora (2024).

A Figura 4.5 exibe a matriz de confusão para CEGNN, destacando a contagem de classificações precisas em comparação com aquelas incorretas para as imagens de dez classes de animais e objetos. Na análise subsequente, a matriz de confusão contém dez classes distintas, onde a classe (0) refere-se a imagens de bicicletas, a classe (1) a imagens de barcos, a classe (2) a imagens de garrafas, a classe (3) a imagens de ônibus, a classe (4) a imagens de carros, a classe (5) a imagens de gatos, a classe (6) a imagens de cadeiras, a classe (7) a imagens de xícaras, a classe (8) a imagens de cachorros e a classe (9) a imagens de motocicletas.

Cada elemento na diagonal principal representa o número de amostras corretamente classificadas para as classes correspondentes. O modelo alcançou uma taxa de classificação de 78,88% para o fluxo de imagens, utilizando 10 características extraídas por meio da análise de componentes principais. O tempo de processamento dos dados foi de 24,51 segundos para todas as amostras, empregando granularidades iniciais pequenas, como  $\rho = 0,5$ , um total de 800 iterações e aproximadamente 46 regras.

Figura 4.5 – Matriz de confusão para a classificação de dez classes.

**Matriz de confusão**

Classe de saída	0	205		3		2	1	17	1	1	20
	1	2	201	5	4	4	6	17	3	1	7
	2	1	3	184		1	5	18	33		5
	3		2	2	209	21	2	13			1
	4			12	12	201	8	7	3	1	6
	5		1	2		4	192	10	5	35	1
	6	7	2	15	2	11	9	189	8	4	3
	7	2	3	21		9	6	19	182		8
	8	1		1		1	40	6	2	197	2
	9	16		6		6		9	1		212
		1	2	3	4	5	6	7	8	9	
		<b>Classe estimada</b>									

Fonte: Da autora (2024).



Observou-se que, a maior confusão do modelo ao classificar as amostras foi em relação a classe (7) referente as imagens de xícaras, onde das 250 amostras de imagens de xícaras, 182 foram adequadamente classificadas. Entretanto, foram identificados equívocos na predição, onde 1 amostra foi incorretamente associada a imagens de bicicletas, 3 amostras a imagens de barcos, 33 amostras a imagens de garrafas, 3 amostras a imagens de carros, 5 amostras a imagens de gatos, 8 amostras a imagens de cadeiras, 2 amostras a imagens de cachorros e 1 amostra a imagens de motocicletas. Essa análise detalhada destacou áreas específicas de desafio em relação à classe de imagens de garrafas, proporcionando uma compreensão mais aprofundada do desempenho do modelo em cada categoria. A análise detalhada enfoca especialmente a classe de imagens de garrafas, evidenciando que a forma mais alongada das xícaras pode ter contribuído para confusões com garrafas. A presença dessas confusões permitiu reflexões sobre possíveis melhorias e otimizações no processo de classificação, contribuindo para o refinamento do modelo em questão.

Destaca-se a classe (9), associada às imagens de motocicletas, como apresentando a menor taxa de confusão. Das 250 amostras de imagens de motocicletas avaliadas, 212 foram corretamente classificadas. No entanto, observou-se que o modelo incorreu em classificações equivocadas em relação a outras categorias. Foram registradas 20 amostras erroneamente associadas a imagens de bicicletas, 7 amostras a imagens de barcos, 5 amostras a imagens de garrafas, 1 amostra a imagem de ônibus, 6 amostras a imagem de carros, 1 amostra a imagens de gato, 3 amostras a imagem de cadeira, 8 amostras a imagem de xícara e 2 amostras a imagens de cachorros. Nesse caso, foi notável uma maior incidência de confusões ao analisar imagens de bicicletas. Esse cenário pode ser atribuído à semelhança visual entre os objetos em questão, visto que ambos compartilham características comuns, como rodas e estruturas alongadas. Essa complexidade na distinção entre bicicletas e motocicletas destacou um desafio significativo para o modelo, exigindo uma atenção especial na identificação de características distintivas para cada classe durante o treinamento. A compreensão dessas nuances foi importante para aprimorar a capacidade discriminativa do modelo em situações visualmente semelhantes, contribuindo para uma classificação mais precisa e robusta.

A matriz de confusão é uma ferramenta essencial para interpretar o desempenho de um modelo de classificação. Esta matriz mostra a relação entre as classificações reais e as predições feitas pelo modelo para cada classe. Por meio da matriz de confusão, foi possível identificar onde o modelo está acertando e onde está cometendo erros, permitindo uma análise detalhada do desempenho em diferentes cenários. Portanto, a interpretabilidade é fundamental para garantir a confiança e a compreensão dos resultados produzidos pelos modelos de aprendizado de máquina. Ao entender como os modelos tomaram decisões e quais fatores influenciaram suas predições, os usuários poderão tomar decisões mais informadas e identificar possíveis áreas de melhoria e otimização.

A análise do modelo proposto foi realizada por meio da interpretação das características discriminativas aprendidas durante o treinamento pela CEGNN. Essas características são representações abstratas das classes de interesse, capturando padrões visuais específicos presentes nas imagens. Por exemplo, o modelo pode aprender a identificar a presença de certos padrões de textura ou características morfológicas, como orelhas pontiagudas e focinho alongado em imagens de cães. Essas características discriminativas permitiram ao modelo fazer distinções entre diferentes classes, contribuindo para suas decisões de classificação. Ao reconhecer padrões específicos associados a cada classe, o modelo pode atribuir corretamente novas amostras às suas respectivas categorias com base nessas características aprendidas.

### 4.3 Discussões sobre Interpretabilidade

A interpretabilidade de modelos de aprendizado de máquina têm sido um tema de crescente importância, especialmente em aplicações onde a transparência e a capacidade de entender como as decisões são tomadas são essenciais. Isso se refere à capacidade de entender e explicar como os modelos operam internamente, incluindo os critérios utilizados para tomar decisões. A importância dessa interpretabilidade é evidenciada pelo crescente interesse em técnicas que permitam explicar o funcionamento dos modelos, como LIME e SHAP, que fornecem uma compreensão sobre a contribuição de cada característica para as previsões.

A escolha da arquitetura VGG-16 para este estudo é fundamentada no mapeamento suave que ela produz do espaço de entrada para o espaço de saída. Com seus 138 milhões de parâmetros, a VGG16 demonstra uma capacidade de generalização notável, refletida na suavidade da superfície de mapeamento. Essa característica é crucial em termos de interpretabilidade, pois facilita a extração de grânulos de informação dessa superfície. Em um espaço com maiores dimensões, das 4096 componentes que emergem na camada 13 da VGG-16, apenas uma fração, aproximadamente 25 a 30, são verdadeiramente significativas. Para maximizar a variância nesse espaço transformado de menor dimensão, é possível recorrer ao PCA. No entanto, a aplicação do PCA neste ponto enfrenta um desafio: requer acesso à base de dados offline completa, o que é inviável em um contexto de fluxo de dados. No entanto, dada a natureza experimental e o uso de conjuntos de dados conhecidos, essa limitação é contornada pela utilização de uma abordagem incremental combinada com PCA, permitindo assim a análise contínua dos dados.

Neste trabalho, foi explorada a interpretabilidade do modelo por meio da criação de regras utilizando as PCAs para representar as características originais de maneira mais concisa e informativa. A seleção dessas PCAs baseou-se em uma abordagem empírica, visando capturar a maior parte da variabilidade presente nos dados originais. Essa escolha possibilitou uma análise mais eficiente e simplificada da classificação realizada pela CEGNN.

Ao fundamentar o modelo em partições do espaço latente, em vez de uma rede totalmente conectada no topo, como é comum em muitas abordagens convencionais, a interpretabilidade do modelo foi aprimorada. Isso se deve ao fato de que o modelo divide o espaço de entrada em regiões distintas e associa cada região a uma classe ou categoria específica. Assim, torna-se mais fácil compreender como o modelo chegou a uma determinada decisão, uma vez que essa decisão está diretamente relacionada à região específica do espaço de entrada onde os dados estão localizados.

Portanto, a primeira variável da regra gerada pelo EGNN foi a primeira componente principal, e assim por diante. Essa abordagem proporcionou uma interpretação direta das características relevantes para a classificação das imagens, facilitando a compreensão das decisões do modelo, sendo a importância de cada componente medida pelos seus autovalores, o que facilitou a compreensão das decisões do modelo CEGNN. O uso de PCAs facilitou a identificação de padrões e relacionamentos nos dados, tornando-os mais acessíveis e compreensíveis. Diante disso, notou-se que a CEGNN foi capaz de processar dados *online* e lidar com uma variedade de valores possíveis, proporcionando transparência e interpretabilidade na construção resultante.

Essa estratégia de interpretabilidade justifica a preferência por uma rede fundamentada em partições do espaço latente, em vez de uma rede totalmente conectada no topo. Ao adotar essa estratégia, o modelo pôde capturar representações mais importantes e passíveis de interpretação das características visuais das imagens. Essa abordagem capacitou o modelo a formular regras mais claras e interpretáveis para explicar suas decisões. Assim, ao substituir uma rede totalmente conectada por uma estrutura baseada em partições do espaço latente, a interpretabili-

dade do modelo foi aprimorada, facilitando a compreensão de seu funcionamento e dos motivos fundamentais às previsões que produz.

Para futuros trabalhos, sugere-se explorar ainda mais a interpretabilidade do modelo proposto, investigando formas de visualizar e explicar as decisões do modelo de maneira mais intuitiva e compreensível. Além disso, foi possível realizar estudos comparativos entre diferentes métodos de interpretação para identificar a abordagem mais eficaz para explicar as decisões dos modelos de aprendizado de máquina em diferentes contextos e aplicações. Essa análise comparativa pode fornecer contribuições valiosas sobre quais técnicas são mais adequadas para diferentes tipos de problemas e conjuntos de dados.

#### 4.4 Comparação com outros modelos

Nesta seção, serão apresentados os resultados da abordagem proposta (CEGNN) em comparação com outros modelos, com foco na substituição da VGG-16 por diferentes arquiteturas, nomeadamente VGG-19 e InceptionV3, no contexto da integração com a rede EGNN. A análise comparativa visa avaliar o desempenho relativo das combinações VGG16-EGNN, VGG19-EGNN e InceptionV3-EGNN, proporcionando uma compreensão abrangente das capacidades de cada modelo em termos de aprendizado contínuo e interpretabilidade.

A comparação concentrou-se na complexidade do espaço latente gerado pela CNN, sendo a VGG-16, VGG-19 e InceptionV3 utilizadas como geradoras desse espaço. Os resultados foram contrastados para determinar a eficácia de cada abordagem na geração de representações latentes menos complicadas e mais interpretáveis. O desempenho dos modelos foi avaliado em termos de sua capacidade de aprendizado contínuo ao longo da vida e na interpretabilidade dos resultados obtidos. A escolha de diferentes arquiteturas teve como objetivo proporcionar esclarecimento sobre como a complexidade da CNN influencia esses aspectos. Essa análise comparativa contribuiu para uma compreensão mais aprofundada das vantagens e desvantagens de cada combinação de arquitetura CNN-EGNN, fornecendo informações relevantes para futuras pesquisas e desenvolvimentos na área de classificação de fluxo de imagens.

A arquitetura VGG-19, uma evolução da VGG-16, se destacou por sua profundidade e abordagem uniforme. Composta por 19 camadas de convolução, essa arquitetura utiliza filtros convolucionais pequenos ( $3 \times 3$ ) em todas as camadas, mantendo uma abordagem uniforme. A profundidade extra da VGG-19 permitiu que o modelo aprenda representações mais complexas e abrangentes de características visuais em diferentes níveis de abstração. Embora essa arquitetura seja mais intensiva em termos de recursos computacionais, sua simplicidade e estrutura homogênea facilitaram a compreensão e implementação (Awan *et al.*, 2021).

Já a InceptionV3, por outro lado, adota uma abordagem mais inovadora conhecida como "Inception" ou "GoogleNet". Essa arquitetura introduz módulos inception, que são conjuntos de convoluções paralelas com diferentes tamanhos de filtro. Essa estratégia permitiu que a rede neural extraísse informações em escalas espaciais variadas, capturando detalhes finos e características globais simultaneamente. Os módulos inception promoveram eficiência computacional ao evitar o uso excessivo de parâmetros e, assim, contribuíram para uma representação mais rica e adaptável das características visuais das imagens (Hidayat; Astuti; Yaqin, 2023).

Ambas as arquiteturas, VGG19 e InceptionV3, são amplamente utilizadas em tarefas de visão computacional, embora com abordagens diferentes. A VGG19 enfatiza a simplicidade e profundidade, enquanto a InceptionV3 destaca-se pela eficiência na extração de características em múltiplas escalas espaciais, resultando em modelos robustos e capazes de lidar com uma variedade de desafios em reconhecimento de imagens.

A comparação foi conduzida utilizando a base de dados de 10 classes, e foram consideradas 10 Componentes Principais (PCAs) na análise. Essa escolha visou proporcionar uma avaliação mais abrangente e representativa do desempenho dos modelos em um cenário multi-classe, considerando as diferentes categorias presentes na base de dados. A Tabela 4.3 apresenta o desempenho de classificação de imagens para os modelos analisados.

Tabela 4.3 – Desempenho de classificação de imagens para os modelos analisados.

<b>Modelo</b>	<b>CEGNN</b>	<b>VGG19-EGNN</b>	<b>InceptionV3-EGNN</b>
<b>Características</b>	10	10	10
<i>A<sub>CC</sub></i> (%)	78,88	76,2	82,24
<b>Média de regras</b>	37,75	37,86	38,95
<b>Tempo (s)</b>	24,51	24,42	25,12
<b>Precisão</b>	0,79	0,76	0,83
<b>Recall</b>	0,78	0,76	0,82
<b>F1</b>	0,79	0,76	0,82

Fonte: Da autora (2024).

Os resultados da análise comparativa revelam que, ao empregar a base de dados com 10 classes e considerando 10 Componentes Principais (PCAs), o modelo InceptionV3-EGNN alcançou a mais elevada acurácia, atingindo 82,24%. Na sequência, o modelo VGG16-EGNN registrou 78,88%, seguido pelo VGG19-EGNN com 76,2%. No contexto da média de regras, o VGG16-EGNN apresentou um valor de 37,75, seguido pelo VGG19-EGNN com 37,86 e o InceptionV3-EGNN com 38,95. Quanto ao tempo de processamento, não se observaram diferenças expressivas entre os modelos, variando de 24,42 a 25,12 segundos. No que se respeita às métricas de precisão, *Recall* e *F1-Score*, o modelo InceptionV3-EGNN também liderou, alcançando valores de 0,83, 0,82 e 0,82, respectivamente. Por outro lado, VGG19-EGNN e VGG16-EGNN apresentaram valores ligeiramente inferiores nessas métricas. Esses resultados indicam que o modelo InceptionV3-EGNN demonstrou um desempenho superior em acurácia e métricas de avaliação, evidenciando sua eficácia na classificação do fluxo de imagens nesse contexto específico.

É importante notar que os valores de precisão, *recall*, F1 e acurácia apresentados na tabela são iguais. Isso ocorre devido ao fato de que a base de dados utilizada para a avaliação dos modelos é balanceada. Em uma base de dados balanceada, o número de amostras em cada classe é aproximadamente o mesmo, o que significa que não há um viés em direção a nenhuma classe específica. Quando um conjunto de dados é balanceado, as métricas de avaliação do modelo tendem a refletir de forma equitativa o desempenho em todas as classes, resultando em valores idênticos de precisão, *recall*, F1 e acurácia para todas as classes.

A superioridade do modelo InceptionV3-EGNN em relação à acurácia e métricas de avaliação, quando comparado aos modelos VGG16-EGNN e VGG19-EGNN, pode ser atribuída às características intrínsecas da arquitetura InceptionV3. Ao incorporar módulos inception, que empregam convoluções paralelas com filtros de diferentes tamanhos, a InceptionV3 possibilita que a rede neural assimile informações em múltiplas escalas espaciais simultaneamente, capturando detalhes finos e características globais. Essa habilidade de extrair características em diversas escalas revela-se especialmente vantajosa em tarefas de classificação de imagens, onde objetos de interesse puderam variar em tamanho e complexidade.

Além disso, a eficiência computacional inerente aos módulos inception contribuiu para um equilíbrio aprimorado entre capacidade de aprendizado e recursos computacionais disponí-

veis. Ao evitar o uso excessivo de parâmetros, a InceptionV3 consegue representar de forma mais abrangente as características visuais das imagens, permitindo uma generalização mais eficaz para dados não vistos. Os resultados apresentados mostraram que a InceptionV3-EGNN conseguiu extrair informações mais relevantes e discriminativas da base de dados de 10 classes, culminando em uma acurácia superior e métricas de avaliação aprimoradas. Esses achados indicam que a arquitetura inovadora da InceptionV3, com seus módulos inception, revela-se particularmente eficiente na abordagem da complexidade e variabilidade nas imagens analisadas, resultando em um desempenho mais robusto na tarefa de classificação.

Ao analisar as diferentes arquiteturas de CNNs, a escolha da VGG-16 se justifica pela sua capacidade de produzir mapeamentos suaves no espaço de entrada para o espaço de saída. Os grânulos de informação gerados pela VGG-16 são mais amplos devido à suavidade da superfície de mapeamento, o que significa que todos os pontos dentro dessa região são atribuídos à mesma classe. Por outro lado, em arquiteturas como a VGG-19 e a InceptionV3, a superfície de mapeamento é mais fragmentada, levando a uma classificação mais granular, onde diferentes regiões podem ser associadas a diferentes classes. Embora outros modelos possam apresentar uma acurácia ligeiramente melhor, a VGG-16 se destaca em termos de capacidade de generalização devido à sua suavidade na superfície de mapeamento. Essa característica é crucial para garantir que o modelo seja capaz de classificar corretamente amostras de dados não vistos e desconhecidos. Portanto, apesar das vantagens apresentadas por outras arquiteturas em determinados aspectos, a escolha da VGG-16 se baseia na sua capacidade de produzir mapeamentos mais suaves e, conseqüentemente, em uma melhor capacidade de generalização.

#### 4.5 Síntese

Este capítulo apresentou os resultados do desempenho do modelo proposto CEGNN na classificação de imagens. Foram utilizadas diferentes arquiteturas de CNNs, incluindo VGG16, VGG19 e InceptionV3, visando uma análise comparativa de desempenho. Destaca-se que o modelo InceptionV3-EGNN demonstrou uma acurácia de 82,24%, superando a VGG16-EGNN com 78,88% e a VGG19-EGNN com 76,2%. No entanto, a escolha da VGG-16 foi motivada pela sua suavidade na superfície de mapeamento, favorecendo uma melhor generalização dos dados. Além disso, o capítulo explorou os desafios e as estratégias adotadas para abordar a interpretabilidade do CEGNN, dada a complexidade das CNNs. Uma das abordagens discutidas foi a análise da evolução contínua das conexões, juntamente com a identificação e análise de regras geradas pelo modelo. A aplicação de técnicas como a análise de componentes principais também foi considerada relevante nesse contexto. Embora o modelo tenha demonstrado potencial para adaptação a ambientes dinâmicos, ressalta-se que a interpretabilidade permanece como uma área desafiadora em constante desenvolvimento. A compreensão completa do funcionamento interno do modelo e a explicação transparente de suas decisões continuam a ser alvos de investigação e aprimoramento. Foi discutido como o modelo proposto, CEGNN, aborda a interpretabilidade por meio da análise das regras geradas e da visualização da matriz de confusão. A interpretação das regras revelou padrões específicos aprendidos pelo modelo durante o treinamento, proporcionando conhecimento sobre as características discriminativas de cada classe. Além disso, a matriz de confusão permitiu uma análise detalhada do desempenho do modelo em diferentes cenários.

## 5 CONCLUSÃO

Este trabalho apresentou uma proposta inovadora que combina as características da rede convolucional VGG-16 com a flexibilidade da rede neuro-fuzzy granular evolutiva para a classificação de fluxo de imagens de modo *online*. Essa abordagem estratégica permite explorar as vantagens distintas de ambas as arquiteturas, combinando a capacidade de extração de características da rede convolucional e a adaptabilidade proporcionada pelo algoritmo evolutivo na rede granular evolutiva. A aplicação da análise de componentes principais para redução de dimensionalidade também desempenhou um papel fundamental na otimização do desempenho do modelo, permitindo uma representação mais compacta e informativa das características das imagens. Essa combinação de técnicas resultou em um modelo robusto e eficaz, capaz de lidar com a complexidade e a variabilidade dos dados de maneira eficiente e precisa.

Destacou-se ainda a escolha estratégica da VGG-16, cuja superfície de mapeamento mais suave permite uma melhor generalização e interpretabilidade, mesmo que em comparação com outros modelos sua acurácia seja ligeiramente menor. Isso demonstra a importância de considerar não apenas métricas de desempenho isoladas, mas também características intrínsecas das arquiteturas utilizadas. A capacidade de evolução contínua da modelagem convolucional foi evidenciada, possibilitando o processamento em tempo real com baixo custo computacional. Além disso, é importante ressaltar o desempenho alcançado pelo modelo proposto, com uma acurácia de 78,88% e uma precisão de 0,79 na tarefa de classificação para dez classes distintas. A relação entre acurácia e precisão refletiu a capacidade do modelo de classificar corretamente as amostras positivas e negativas.

Portanto, a eficácia da abordagem desenvolvida em lidar com a complexidade das tarefas de classificação de imagens, reforçando sua viabilidade e relevância em aplicações práticas como análise de doenças radiológicas, detecção de doenças e lesões em imagens, avaliação de condição corporal, análise de comportamento do motorista em imagens de veículos autônomos, reconhecimento de sinais de trânsito, mapeamento e localização. Adicionalmente, ressalta-se que os resultados e descobertas deste trabalho serão submetidos para publicação em artigos científicos, contribuindo para a disseminação do conhecimento e compartilhamento das inovações propostas neste estudo. Essa iniciativa visa contribuir para o avanço da pesquisa e promover a aplicação prática da abordagem desenvolvida em diversos contextos.

Para as próximas etapas da pesquisa, estão planejado ajustes no algoritmo visando aprimorar a classificação de imagens em áreas específicas. Serão realizadas simulações e análises comparativas entre os resultados obtidos com imagens de teste e imagens reais de campo, adequadas para cada aplicação. Essas análises permitirão avaliar a robustez e a generalização do modelo em cenários do mundo real, fornecendo insights sobre sua eficácia e adaptabilidade em contextos específicos, como análise de imagens radiológicas e reconhecimento de objetos em imagens de veículos autônomos.

## REFERÊNCIAS

- AGARWAL, S.; TERRAIL, J. O. D.; JURIE, F. Recent advances in object detection in the age of deep convolutional neural networks. **Computer Vision and Pattern Recognition**, 2019.
- AGGARWAL, C. C. **Neural Networks and Deep Learning**. 1. ed. ISBN 978-3-030-06856-1: Springer Cham, 2019.
- ALBER, M. *et al.* investigate neural networks. **Journal of Machine Learning Research**, v. 20, p. 1–8, 2019.
- ALDUGHAYFIQ, B. *et al.* Explainable ai for retinoblastoma diagnosis: Interpreting deep learning models with lime and shap. **Diagnosics**, v. 13, p. 1–19, 2023.
- ALIZADEH, S. *et al.* Online local input selection through evolving heterogeneous fuzzy inference system. **IEEE Transactions on Fuzzy Systems**, v. 24, p. 1364 – 1377, 2016.
- ALONSO, J. M.; MAGDALENA, L.; GONZÁLEZ-RODRÍGUEZ, G. Looking for a good fuzzy system interpretability index: An experimental approach. **International Journal of Approximate Reasoning**, v. 51, p. 115–134, 2009.
- ALVARES, A. J.; SOUZA, A. C. A.; CASTRO, M. F. de. Implementação de um aplicativo móvel (app) para leitura de medidores de água e energia baseado em visão computacional. **Revista Brasileira de Computação Aplicada**, v. 12, p. 107–121, 2020.
- ANAMIKA, D.; Verma, G. K. Convolutional neural network: a review of models, methodologies and applications to object detection. **Progress in Artificial Intelligence**, v. 9, p. 85–112, 2020.
- ANGELOV, P.; KASABOV, N. Evolving intelligent systems - eis. **IEEE SMC e News Letter**, v. 15, p. 1–13, 2006.
- ANJOS, J. R. L. Inferencnn: Uma biblioteca para inferência de redes neurais convolucionais multiplataforma em opencl. **Dissertação de Mestrado. Programa de Pós-graduação em Ciência da Computação. Universidade Federal de Pernambuco**, 2019.
- ARRIETA, A. B. *et al.* Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. **Information Fusion**, v. 58, p. 82–115, 2020.
- AWAN, M. J. *et al.* Image-based malware classification using vgg19 network and spatial convolutional attention. **Electronics**, v. 10, p. 1–19, 2021.
- BACKES, A. R.; JUNIOR, J. J. d. m. S. **Introdução à Visão Computacional Usando MATLAB Manual Para Normalização de Publicações Técnico-científicas**. Rio de Janeiro: [s.n.], 2016.
- BALDOMINOS, A.; , Y. S. ; ISASI, P. A survey of handwritten character recognition with mnist and emnist. **Applied sciences**, v. 9, 2019.
- BARELLI, F. **Introdução à Visão Computacional: Uma abordagem prática com Python e OpenCV**. [S.l.]: Casa do Código, 2018.
- BARROS, L. C.; BASSANEZI, R. C.; LODWICK, W. A. A first course in fuzzy logic, fuzzy dynamical systems, and biomathematics: theory and applications. **Springer**, 2017.

- BECKER, W. E. Uma abordagem de redes neurais convolucionais para análise de sentimento multi-lingual. **Dissertação de Mestrado. Programa de Pós-graduação em Ciência da Computação. Pontifícia Universidade Católica do rio Grande do Sul**, 2017.
- BEJNORDI, B. E. *et al.* Using deep convolutional neural networks to identify and classify tumor-associated stroma in diagnostic breast biopsies. **Modern Pathology**, v. 31, p. 1502–1512, 2018.
- BHARDWAJ, A. *et al.* Uav imaging of a martian brine analogue environment in a fluvio-aeolian setting. **Remote Sens**, v. 11, p. 1–32, 2019.
- BI, C. *et al.* Mobilenet based apple leaf diseases identification. **Mobile Networks and Applications**, v. 27, p. 172–180, 2020.
- BRAGA, A. C. da S. Curvas roc: aspectos funcionais e aplicações. **tese de doutorado. Escola de Engenharia da Universidade do Minho. Braga, Portugal**, p. 1–267, 2020.
- BURGERHOF, J. G. *et al.* The southampton hand assessment procedure revisited: A transparent linear scoring system, applied to data of experienced prosthetic users. **Journal of hand therapy**, v. 30, p. 49–57, 2017.
- BURKART, N.; HUBER, M. F. A survey on the explainability of supervised machine learning. **Journal of Artificial Intelligence Research**, v. 70, p. 245–317, 2021.
- CALIN, O. **Deep Learning Architectures: A Mathematical Approach**. 1. ed. ISBN 978-3030367237: Springer, 2021.
- CARDOSO, J. R. F. *et al.* Development of a robotic structure for acquisition and classification of images (eraci) in sugarcane crops. **Revista Ciência Agronômica**, v. 51, p. 1–15, 2020.
- CREMONA, J.; UZAL, L.; PIRE, T. Wganvo: odometria visual monocular baseada em redes adversarias generativas. **Revista Iberoamericana de Automática e Informática Industrial**, v. 19, p. 144–153, 2022.
- DECKER, L. *et al.* Comparison of evolving granular classifiers applied to anomaly detection for predictive maintenance in computing centers. **IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS)**, 2020.
- DHURANDHAR, A. *et al.* Explanations based on the missing: Towards contrastive explanations with pertinent negatives. **Neural Information Processing Systems**, v. 21, p. 1–22, 2018a.
- DHURANDHAR, A. *et al.* Improving simple models with confidence profiles. **Neural Information Processing Systems**, v. 31, p. 1–16, 2018b.
- FENG, S. *et al.* One-dimensional vggnet for high-dimensional data. **Applied Soft Computing**, v. 135, p. 1–18, 2023.
- FORTUNATO, D. *et al.* Modelagem granular neuro-fuzzy evolutiva para classificação de distúrbios em sistemas de distribuição de potência. **Congresso Brasileiro de Automática (CBA)**, p. 1–7, 2020.
- GALKIN, F. *et al.* Human gut microbiome aging clock based on taxonomic profiling and deep learning. **iScience**, v. 23, p. 1–33, 2020.



- GÉRON, A. **Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow**. [S.l.]: "O'Reilly Media, Inc.", 2019.
- GOLDSTEIN, A. *et al.* Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. **Journal of Computational and Graphical Statistics**, v. 24, p. 1–22, 2015.
- GONZALEZ, R. C.; WOODS, R. E. **Processamento de Imagens Digitais**. [S.l.]: Blucher, 2000.
- GOODFELLOW, I. **Deep Learning**. 1. ed. ISBN 978-0262035613: The Mit Press, 2016.
- HASHIMOTO, D. A. *et al.* Computer vision analysis of intraoperative video: Automated recognition of operative steps in laparoscopic sleeve gastrectomy. p. 414–421, 2019.
- HIDAYAT, T.; ASTUTI, I. A.; YAQIN, A. Grouping of image patterns using inceptionv3 for face shape classification. **International Journal on Informatics Visualization**, v. 7, 2023.
- HOCHULI, A. G.; Oliveira, L. E. S. de. Redes neurais convolucionais. **Notas de aula. Programa de Pós-Graduação em Informática Departamento de Informática**, p. 1–49, 2016.
- HUANG, G. *et al.* Condensenet: An efficient densenet using learned group convolutions. **Conference on Computer Vision and Pattern Recognition**, p. 2752–2761, 2018.
- JARAA, J. D. Z.; BOWENB, S. Learning curve analysis on adam, sgd, and adagrad optimizers on a convolutional neural network model for cancer cells recognition. **ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal**, v. 11, p. 263–283, 2022.
- JOHNSON, P. M. *et al.* Using machine learning to examine freight network spatial vulnerabilities to disasters: A new take on partial dependence plots. **Transportation Research Interdisciplinary Perspectives**, v. 14, p. 1–10, 2022.
- KAGGLE. **DataSet: Cat and Dog (Cats and Dogs dataset to train a DL model)**. [S.l.], 2018. Disponível em: <<https://www.kaggle.com/datasets/shaunthesheep/microsoft-catsvsdogs-dataset?datasetId=550917>>.
- KATSEVICH, E.; RAMDAS, A. The leave-one-covariate-out conditional randomization test. **Journal of Computational and Graphical Statistics**, v. 13, 2020.
- KHAN, S. *et al.* **A Guide to Convolutional Neural Networks for Computer Vision**. 8(1):1–207, feb 2018, issn 2153-1056.. ed. [S.l.]: Morgan Claypool Publishers, 2018.
- KONG, L.; CHENG, J. Classification and detection of covid-19 x-ray images based on densenet and vgg16 feature fusion. **Biomedical Signal Processing and Control**, v. 77, p. 1–11, 2022.
- KOUSHIK, S. S.; SRINIVASA, K. Detection of respiratory diseases from chest x rays using nesterov accelerated adaptive moment estimation. **Measurement**, v. 176, p. 109–153, 2021.
- KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. Imagenet classification with deep convolutional neural networks. **Advances in neural information processing systems**, v. 25, p. 1097–1105, 2012.

- KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. **Communications of the ACM**, v. 60, p. 84–90, 2017.
- LEAL-LARA, D.-D.; BARÓN-VELANDIA, J.; ROCHA-CALDERÓN, C.-E. Interpretability in the field of plant disease detection: A review. **Revista Facultad de Ingeniería**, v. 30, p. 1–22, 2021.
- LEI, D.; CHEN, X.; ZHAO, J. Opening the black box of deep learning. **Computer Science**, V1, p. 1–27, 2018.
- LEITE, D. Evolving granular systems. **Doctoral Thesis. State University of Campinas, School of Electrical and Computer Engineering**, p. 188, 2012.
- LEITE, D. **Comparison of Genetic and Incremental Learning Methods for Neural Network-Based Electrical Machine Fault Detection. In: Predictive Maintenance in Dynamic Systems (Lughofer, E., Sayed-Mouchaweh, M. (eds)).** [S.l.]: Springer, Cham, 278-282, 2019.
- LEITE, D.; JR, P. C.; GOMIDE, F. Evolving granular neural networks from fuzzy data streams. **Neural Networks**, v. 38, p. 1–16, 2013.
- LEITE, D.; JR., V. F.; MEDEIROS, R. Adaptive gaussian fuzzy classifier for real-time emotion recognition in computer games. **IEEE Latin American Conference on Computational Intelligence (LA-CCI)**, 2021.
- LEITE, D. F.; JR, P. C.; GOMIDE, F. Redes neurais granulares para aprendizagem incremental semi-supervisionada. **XVIII Congresso Brasileiro de Automática**, 2010.
- LENGYEL, A. *et al.* Zero-shot domain adaptation with a physics prior. **International Conference on Computer Vision (ICCV)**, p. 4399–4409, 2021.
- LI, Z. Identification for red automobile based on cifar10 using machine learning models. **Journal of Physics: Conference Series**, v. 2428, p. 1–7, 2023.
- LIMA, G. A. *et al.* Sistema de visão computacional para identificação automática de potenciais focos do mosquito aedes aegypti usando drones. **Revista Ibérica de Sistemas e Tecnologias de Informação**, v. 43, p. 93–110, 2021.
- LIN, T.-Y. *et al.* Microsoft coco: Common objects in context. **In European conference on computer vision. Computer Vision – ECCV 2014**, v. 8693, p. 740–755, 2014.
- LITJENS, G. *et al.* A survey on deep learning in medical image analysis. **Medical Image Analysis**, v. 42, p. 60–88, 2017.
- LUCHI, L. F.; ADAMI, A. G. Deep learning aplicado a inspeção visual da presença de um componente de conjunto de eixo. **Scientia cum industria**, v. 8, p. 135–144, 2020.
- LUGHOFER, E. *et al.* Generalized smart evolving fuzzy systems. **Evolving Systems**, v. 6, p. 269–292, 2015.
- LUNDBERG, S.; Lee, S.-I. A unified approach to interpreting model predictions. **31st Conference on Neural Information Processing Systems (NIPS 2017)**, v. 25, 2017.

- MARQUES, G.; AGARWA, D.; Díez, I. de la T. Automated medical diagnosis of covid-19 through efficientnet convolutional neural network. **Applied Soft Computing**, v. 96, p. 1–11, 2020.
- MCFALL, G. P. *et al.* Identifying key multi-modal predictors of incipient dementia in parkinson's disease: a machine learning analysis and tree shap interpretation. **Frontiers in Aging Neuroscience**, v. 15, p. 1124232–1124232, 2023.
- MISHKINA, D.; SERGIEVSKIY, N.; MATAS, J. Systematic evaluation of convolution neural network advances on the imagenet. **Computer Vision and Image Understanding**, v. 161, p. 11–19, 2017.
- MONTAVON, G. *et al.* Layer-wise relevance propagation: An overview. **Explainable AI: Interpreting, Explaining and Visualizing Deep Learning**, p. 193–209, 2019.
- NAIR, V. N. *et al.* Explaining adverse actions in credit decisions using shapley decomposition. **Statistics - Machine Learning**, p. 1–20, 2019.
- NEVES, E. P. *et al.* Sistema audiometrico baseado em regras fuzzy. **Revista Eletrônica Paulista de Matemática**, V22, p. 1–19, 2022.
- NEVES, L. A. P.; NETO, H. V.; GONZAGA, A. **Avanços em Visão Computacional**. [S.l.]: Omnipax, 2021.
- PARK, S. *et al.* A depth camera-based human activity recognition via deep learning recurrent neural network for health and social care services. **Procedia Computer Science**, v. 100, p. 78–84, 2016.
- PIRES, D. S.; SERRA, G. L. O. Modelagem de filtro de kalman nebuloso baseado em agrupamento evolutivo de dados experimentais. **Anais do XXII Congresso Brasileiro de Automática**, v. 1, 2019.
- RANJAN, R. *et al.* Deep learning for understanding faces: machines may be just as good, or better, than humans. **IEEE Signal Processing Magazine**, v. 35, p. 66–83, 2018.
- RAUTARAY, S.; AGRAWAL, A. Vision based hand gesture recognition for human computer interaction: A survey. **Artificial Intelligence Review**, v. 3, p. 1–54, 2012.
- REDIVO, A. *et al.* O uso da rastreabilidade e certificação na produção de bovinos de corte: um estudo de caso. **XXXII Encontro Nacional de Engenharia de Produção. Desenvolvimento Sustentável e Responsabilidade Social: As Contribuições da Engenharia de Produção**, 2012.
- REGAZZI, A. Análise multivariada. **Departamento de Informática da Universidade Federal de Viçosa**, v. 2, p. 1–12, 2000.
- REZENDE, V. *et al.* Aplicação de redes neurais convolucionais na classificação de doenças de plantas. **Congresso Brasileiro de Inteligência Computacional**, 2020.
- RIBEIRO, M. T.; SINGH, S.; GUESTRIN, C. Model-agnostic interpretability of machine learning. **Workshop on Human Interpretability in Machine Learning (WHI 2016)**, p. 1–5, 2016.

- RIBEIRO, M. T.; SINGH, S.; GUESTRIN, I. C. Anchors: High-precision model-agnostic explanations. **Proceedings of the AAAI Conference on Artificial Intelligence**, v. 32, 2018.
- RODRIGUES, D. de A. *et al.* Classificação automática do tipo de ferro fundido utilizando reconhecimento de padrões em imagens de microscopia. **Revista Matéria**, v. 22, p. 1–16, 2017.
- RODRIGUES, J.; SOUSA, A.; SANTOS, A. Aplicação de redes neurais convolucionais e processamento digital de imagens para classificação do estado dos olhos e avaliação de sonolência. **Revista Brasileira de Computação Aplicada**, v. 13, p. 1–10, 2021.
- RONNEBERGER, O.; FISCHER, P.; BROX, T. U-net: Convolutional networks for biomedical image segmentation. **Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015**, p. 234–241, 2015.
- SANTANA, M. W.; LEITE, D. F. Aprendizado de máquina fuzzy incremental para classificação de faltas em sistemas de potência. **Abakós**, v. 8, p. 3–28, 2020.
- SECCHI, H. K.; CARRO, S. A. Uso de deep learning aplicado no reconhecimento de ações humanas a partir de vídeos em alta resolução visando identificar movimentos suspeitos. **Colloquium Exactarum**, v. 14, p. 67–75, 2022.
- SHAFIQUE, S.; TEHSIN, S. Acute lymphoblastic leukemia detection and classification of its subtypes using pretrained deep convolutional neural networks. **Technology in Cancer Research Treatment**, v. 17, p. 1–7, 2018.
- SHAO, X. *et al.* Deep convolutional neural networks combine raman spectral signature of serum for prostate cancer bone metastases screening. **Nanomedicine: Nanotechnology, Biology and Medicine**, v. 29, p. 1–7, 2020.
- SHRIKUMAR, A.; GREENSIDE, P.; KUNDAJE, A. Learning important features through propagating activation differences. **arXiv.org Free E- Journals**, v. 4, 2017.
- SILVA, S. *et al.* Evolving neuro-fuzzy network for real-time high impedance fault detection and classification. **Neural Computing and Applications**, v. 32, p. 7597–7610, 2018.
- SIMINSKI, K. Fubinfs – fuzzy biclustering neuro-fuzzy system. **fuzzy sets and systems**, v. 438, p. 84–106, 2021.
- SUN, Y. *et al.* Evolving deep convolutional neural networks for image classification. **IEEE Transactions on Evolutionary Computation**, v. 24, p. 1–14, 2020.
- SUN, Y. *et al.* Evolving deep convolutional neural networks for image classification. **IEEE Transactions on Evolutionary Computation**, v. 24, 2020.
- TAN, R. T. Visibility in bad weather from a single image. **IEEE Conference on Computer Vision and Pattern Recognition**, p. 1–8, 2008.
- TAN, Z. *et al.* Efficient semantic image synthesis via class-adaptive normalization. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 44, p. 4852–4866, 2022.
- TEIXEIRA, C. R. G.; SILVA, T. R. Comunicação, esportes e visão computacional: explorando a visualidade dos jogos olímpicos no instagram. **Rizoma**, v. 8, p. 110–127, 2020.

- THANH, T. T. P. *et al.* Leukemia blood cell image classification using convolutional neural network. **International Journal of Computer Theory and Engineering**, v. 10, p. 54–58, 2018.
- VARELLA, C. . A. Análise multivariada aplicada as ciências agrárias. **Pós-graduação em Agronomia. Universidade Federal Rural do Rio de Janeiro**, p. 1–12, 2008.
- VARGAS, A. C. G.; PAES, A.; VASCONCELOS, C. N. Um estudo sobre redes neurais convolucionais e sua aplicação em detecção de pedestres. **Proceedings of the XXIX Conference on Graphics, Patterns and Images**, p. 1–4, 2016.
- VERNE, T. S. *et al.* Detecção e reconhecimento de plantas de pequeno porte utilizando aprendizagem de máquina. **Colloquium Exactarum**, v. 14, p. 36–45, 2022.
- VOGADO, L. H. S. *et al.* Leukemia diagnosis in blood slides using transfer learning in cnns and svm for classification. **Engineering Applications of Artificial Intelligence**, v. 72, p. 415–422, 2018.
- WU, S. *et al.* L1-norm batch normalization for efficient training of deep neural networks. **IEEE Transactions on Neural Networks and Learning Systems**, v. 30, p. 1–9, 2019.
- WU, Z.; SHEN, C.; HENGEL, A. van den. Wider or deeper: Revisiting the resnet model for visual recognition. **Pattern Recognition**, v. 90, p. 119–133, 2019.
- XIAOWEI, G.; MIQING, L. A multi-granularity locally optimal prototype-based approach for classification. **Information Sciences**, v. 569, p. 157–183, 2021.
- XIE, S. *et al.* Artifact removal using improved googlenet for sparse-view ct reconstruction. **Scientific Reports**, p. 1–9, 2018.
- YE, F.; BORS, A. G. Deep mixture generative autoencoders. **IEEE Transactions on Neural Networks and learning Systems**, v. 33, p. 1–8, 2022.
- ZADEH, L. Fuzzy sets. **Information Control**, v. 8, p. 338–353, 1965. 21, 22.
- ZHANG, K.; ZHAN, J.; WU, W. Z. On multicriteria decision-making method based on a fuzzy rough set model with fuzzy -neighborhoods. **IEEE Transactions on Fuzzy Systems**, v. 29, 2021.
- ZHIQIANG, W.; JUN, L. A review of object detection based on convolutional neural network. In: **2017 36th Chinese Control Conference (CCC)**. [S.l.: s.n.], 2017. p. 11104–11109.