

**UNIVERSIDADE ESTADUAL DE PONTA GROSSA  
SETOR DE CIÊNCIAS AGRÁRIAS E TECNOLOGIAS  
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO APLICADA**

**DIEGO HENRIQUE PRESNER**

**IDENTIFICAÇÃO DE ESPÉCIES DE PLANTAS UTILIZANDO AUTOENCODER  
CONVOLUCIONAL E APRENDIZAGEM NÃO SUPERVISIONADA**

**PONTA GROSSA**

**2022**

**DIEGO HENRIQUE PRESNER**

**IDENTIFICAÇÃO DE ESPÉCIES DE PLANTAS UTILIZANDO AUTOENCODER  
CONVOLUCIONAL E APRENDIZAGEM NÃO SUPERVISIONADA**

Projeto de Dissertação apresentado ao Programa de Pós-Graduação em Computação Aplicada, curso de Mestrado em Computação Aplicada da Universidade Estadual de Ponta Grossa, como requisito parcial para obtenção do título de Mestre.

Orientador: Prof. Dr. Alceu de Souza Britto Júnior

**PONTA GROSSA**

**2022**

P934 Presner, Diego Henrique  
Identificação de espécies de plantas utilizando autoencoder convolucional e  
aprendizagem não supervisionada / Diego Henrique Presner. Ponta Grossa, 2022.  
94 f.

Dissertação (Mestrado em Computação Aplicada - Área de Concentração:  
Computação para Tecnologias em Agricultura), Universidade Estadual de Ponta  
Grossa.

Orientador: Prof. Dr. Alceu de Souza Britto Júnior.

1. Identificação de plantas. 2. Auto-enconder. 3. Auto-encoder  
convolucional. 4. Representações de características. 5. Combinação de  
classificadores. I. Britto Júnior, Alceu de Souza. II. Universidade Estadual de  
Ponta Grossa. Computação para Tecnologias em Agricultura. III.T.

CDD: 004



UNIVERSIDADE ESTADUAL DE PONTA GROSSA

Av. General Carlos Cavalcanti, 4748 - Bairro Uvaranas - CEP 84030-900 - Ponta Grossa - PR - <https://uepg.br>**TERMO****TERMO DE APROVAÇÃO****Diego Henrique Presner****IDENTIFICAÇÃO DE ESPÉCIES DE PLANTAS UTILIZANDO AUTOENCODER CONVOLUCIONAL E APRENDIZAGEM NÃO SUPERVISIONAD**

Dissertação aprovada como requisito parcial para obtenção do grau de Mestre no Programa de Pós-Graduação em Computação Aplicada da Universidade Estadual de Ponta Grossa, pela seguinte banca examinadora:

Ponta Grossa, 09 de fevereiro de 2022.

Prof. Dr. Alceu de Souza Britto Junior, (UEPG, Presidente)

Prof. Dr. Arion de Campos Junior (UEPG)

Prof. Dr. Yandre Maldonado e Costa (UEM/PR)



Documento assinado eletronicamente por **Alceu de Souza Britto Junior, Professor(a)**, em 09/02/2022, às 11:41, conforme Resolução UEPG CA 114/2018 e art. 1º, III, "b", da Lei 11.419/2006.



Documento assinado eletronicamente por **Arion de Campos Junior, Professor(a)**, em 09/02/2022, às 11:41, conforme Resolução UEPG CA 114/2018 e art. 1º, III, "b", da Lei 11.419/2006.



Documento assinado eletronicamente por **Yandre Maldonado e Gomes da Costa, Usuário Externo**, em 14/07/2022, às 14:28, conforme Resolução UEPG CA 114/2018 e art. 1º, III, "b", da Lei 11.419/2006.



A autenticidade do documento pode ser conferida no site <https://sei.uepg.br/autenticidade> informando o código verificador **0865751** e o código CRC **37E4A119**.

## **AGRADECIMENTOS**

*Agradeço primeiramente a Deus pela minha vida e oportunidades, me concedendo sabedoria e força para estar concluindo este curso e obtendo mais essa conquista.*

*Aos meus pais, Eny e Paulo que nunca me deixaram desistir dos meus sonhos, e me apoiaram do início ao fim do mestrado, assim como, aos demais familiares que contribuíram de forma positiva durante essa etapa da minha vida.*

*À minha namorada Karyn, por sempre ser paciente, me ajudar e ser companheira nos momentos mais incertos, para que fosse possível seguir em frente e não desistir.*

*Ao meu orientador, Prof. Dr. Alceu de Souza Britto Jr, que de forma muito profissional e paciente me apoiou durante a trajetória na condução deste trabalho.*

*Aos professores e colegas que colaboraram significativamente no progresso da minha formação.*

*À UEPG, por disponibilizar toda a sua estrutura, tanto para o programa de mestrado quanto para esta pesquisa, bem como pela excelente qualidade e dedicação dos professores.*

*E por fim, agradeço a todos que contribuíram direta ou indiretamente para a realização desse trabalho.*

## RESUMO

As plantas possuem um papel fundamental para a existência de vida no planeta Terra, pois elas convertem gás carbônico (CO<sub>2</sub>) em oxigênio (O<sub>2</sub>) e servem de alimento para grande parte dos seres vivos, além de serem utilizadas por diversos segmentos industriais. A importância de trabalhos na linha de pesquisa de identificação/classificação de espécies de plantas, deve-se à vasta biodiversidade, em que muitas destas sofrem risco de extinção ou até mesmo não foram catalogadas/descobertas cientificamente. Ainda, existe a dificuldade de realizar as tarefas de classificação de forma manual (humana). Estudos apontam que a forma automatizada de classificação tem sido eficiente, seus processos demandam menor tempo e quantidade de trabalho ao pesquisador, obtendo assim bons resultados na classificação e rotulação de espécies botânicas. Neste trabalho foi utilizada uma rede neural artificial conhecida como auto-encoder, em específico o auto-encoder convolucional, que emprega o método de aprendizagem não supervisionada/autodidata, utilizando as bases não rotuladas, pois estas são mais fáceis de serem encontradas digitalmente, para realizar o treinamento dos modelos computacionais com imagens de um domínio diferente e pertencentes ao mesmo domínio. Posteriormente os modelos treinados foram utilizados para gerar representações de características diferentes das bases *Flavia*, *Leafsnap* e *PlantCLEF2015*, estas sendo utilizadas para treinar classificadores do tipo SVM, individualmente alcançando taxas de acerto de até 95,00 %. Métodos de combinação de classificadores também foram utilizados, mostrando-se capazes de atingir resultados competitivos com os apresentados no estado da arte.

**Palavras-chave:** Identificação de plantas, Auto-encoder, Auto-encoder convolucional, Representações de características, Combinação de classificadores

## ABSTRACT

Plants play a fundamental role in the existence of life on planet Earth, as they convert carbon dioxide (CO<sub>2</sub>) into oxygen (O<sub>2</sub>) and serve as food for most living beings, in addition to being used by various industrial segments. The importance of works in the research line of identification/classification of plant species is due to the vast biodiversity, in which many of these are at risk of extinction or even have not been scientifically cataloged/discovered. Still, there is the difficulty of performing the classification tasks manually. Studies show that the automated form of classification has been efficient, its processes demand less time and amount of work for the researcher, thus obtaining good results in the classification and labeling of botanical species. In this paper, an artificial neural network known as auto-encoder was used, specifically the convolutional auto-encoder, which employs the unsupervised/self-taught learning method, using unlabeled databases, as these are easier to be found digitally, to perform the training of computational models with images from a different domain and belonging to the same domain. Afterwards, the trained models were used to generate representations of different characteristics of the Flavia, Leafsnap and PlantCLEF2015 bases, which were used to train classifiers of the SVM type, individually reaching hit rates of up to 95,00%. Combination methods of classifiers were also used, showing themselves capable of achieving results that are competitive with those presented in the state of the art.

**Keywords:** Plant identification, Auto-encoder, Convolutional auto-encoder, Feature representations, Combination of classifiers

## LISTA DE FIGURAS

Figura 1- Exemplos de variações da mesma espécie de planta (intra-espécies).....	14
Figura 2 - Exemplos de semelhanças entre diferentes espécies de plantas (inter-espécies).....	15
Figura 3 - Tipos de aprendizagem de máquina para classificar imagens de elefantes e rinocerontes.....	22
Figura 4 - Estrutura básica de um auto-encoder. ....	23
Figura 5 - Camadas de convolução ( <b>L2 e L3</b> ) e a camada oculta ( <b>L4</b> ).....	26
Figura 6 - Arquitetura de fusão das respostas de N classificadores.....	28
Figura 7 - Etapas da metodologia: A) Divisão das amostras em $n \times n$ ; B) Extração das características com a CNN; C) Análise dos principais componentes, D) Concatenação das características, E) Normalização e F) Classificação com a SVM. ....	41
Figura 8 - Fluxograma do método proposto para a classificação de espécies de plantas.....	47
Figura 9 - Amostras de imagens de folhas encontradas na base <i>Flavia Data Base</i> . ....	50
Figura 10 - Amostras de imagens naturais encontradas na base <i>Kyoto Natural Images</i> . ....	51
Figura 11 - Amostras de <i>scan</i> e <i>scan-like</i> das folhas encontradas na base Plant CLEF 2015. ....	52
Figura 12 - Exemplo de descritor XML das imagens da base de dados. ....	53
Figura 13 - Amostras de cada uma das espécies existentes na base de dados <i>LeafSnap</i> . ....	54
Figura 14- Resultados da segmentação de algumas amostras que fazem parte da base <i>LeafSnap</i> . ....	55
Figura 15 - Amostras de laboratório da espécie <i>Betula lenta</i> . ....	55
Figura 16 - Arquitetura CAENN 25 camadas.....	56
Figura 17 - Codificador do modelo CAENN com arquitetura de 25 camadas. ....	57
Figura 18 - Decodificador do modelo CAENN com arquitetura de 25 camadas.....	58
Figura 19 - Arquitetura CAENN 20 camadas.....	59
Figura 20 - Codificador do modelo CAENN com arquitetura de 20 camadas. ....	60
Figura 21 - Decodificador do modelo CAENN com arquitetura de 20 camadas.....	61
Figura 22 - Reconstruções de uma imagem utilizando uma representação de 2.048 e 12.544. ....	62



Figura 23 - Relação entre as taxas de treinamento e teste das arquiteturas do modelo CAENN2: A) Extrator de 2.048 características. B) Extrator de 12.544 características. ....	66
Figura 24 - Relação entre as taxas de perda e a quantidade de épocas das arquiteturas do modelo CAENN2: A) Extrator de 2.048 características. B) Extrator de 12.544 características. ....	67
Figura 25 - Matriz de confusão apresentando os resultados obtidos pela SVM treinada com as características extraídas pelo extrator CAENN2_12544. ....	68
Figura 26 - Relação entre as taxas de treinamento e teste das arquiteturas do modelo CAENN1: A) Extrator de 2.048 características. B) Extrator de 12.544 características. ....	69
Figura 27 - Relação entre as taxas de perda e a quantidade de épocas das arquiteturas do modelo CAENN1: A) Extrator de 2.048 características. B) Extrator de 12.544 características. ....	70
Figura 28 - Matriz de confusão apresentando os resultados obtidos pela SVM treinada com as características extraídas pelo extrator CAENN1_2048. ....	71
Figura 29 – Gráfico em barras representando os resultados obtidos com as SVM's e os ensembles – Base Alvo Flavia. ....	74
Figura 30 - Gráfico em barras representando os resultados obtidos com as SVM's e os ensembles – Base Alvo Leafsnap. ....	75
Figura 31 - Gráfico do teste de Nemenyi apresentando a distância crítica entre os classificadores treinados com as representações extraídas da base Flavia. ....	76
Figura 32 - Gráfico do teste de Nemenyi apresentando a distância crítica entre os classificadores treinados com as representações extraídas da base Leafsnap. ....	76
Figura 33 – Matriz de Confusão gerada pela SVM treinada com as características obtidas pela CAENN1 de 25 camadas. ....	93
Figura 34 - Matriz de Confusão gerada pela SVM treinada com as características obtidas pela CAENN1 de 20 camadas. ....	93

## LISTA DE TABELAS

Tabela 1 - Comparação dos métodos extratores de características. ....	33
Tabela 2 - Resultados da CNN e combinação de canais. ....	36
Tabela 3 - Sumarização das Metodologias. ....	43
Tabela 4 - Estrutura da base de dados Flavia Data Base. ....	48
Tabela 5 - Resultados individuais dos classificadores SVM treinados com as representações de 2.048 e 12.544 características extraídas - CAENN2. ....	67
Tabela 6 - Resultados individuais dos classificadores SVM treinados com as representações de 2.048 e 12.544 características - CAENN1. ....	70
Tabela 7 - Resultados das combinações dos resultados dos classificadores SVM utilizando a base de dados Flavia e a regra da soma. ....	73
Tabela 8 - Resultados das combinações dos resultados dos classificadores SVM utilizando a base de dados Leafsnap e a regra da soma. ....	75
Tabela 9 - Comparação dos resultados com outros autores. ....	78
Tabela 10- Resultados aplicando a camada de GAP nas duas topologias. ....	89
Tabela 11 - Resultados aplicando a camada de GAP nas duas topologias. ....	90
Tabela 12 - Resultados dos <i>ensembles</i> de classificadores utilizando F1 score. ....	91

## LISTA DE SIGLAS

AE - *Autoencoder*  
ANN - *Artificial Neural Network*  
CAE - *Convolutional Autoencoder*  
CAENN - *Convolutional Autoencoder Neural Network*  
CD – *Critical Distance*  
CEN - *Convolutional Encoder Networks*  
CNN - *Convolutional Neural Network*  
CSV - *Comma-separated Values*  
DAE - *Denoising Autoencoder*  
EA - *Ensemble Averaging*  
EFL - *Edge Filling Length*  
FC - *Fully Connected*  
FGOC - *Fine-grained Object Classification*  
FRL - *Aprendizagem por representação de florestas*  
GD - *Gradient Descent*  
H5 - *Hierarchical Data Format*  
JSON - *JavaScript Object Notation*  
KNN - *K-Nearest Neighbors*  
LR - *Logistic Regression*  
MCFL - *Aprendizagem de contornos em múltiplas escalas*  
MSD – *Mean Squared Error*  
NB - *Naive Bayes*  
RF - *Random Forest*  
RGB - *Red-Green-Blue*  
RS - *Random Sampling*  
ReLU - *Rectified Linear Unit*  
SAE - *Sparse Autoencoder*  
SGD - *Stochastic Gradient Descent*  
SSDAE - *Sparse Stacked Denoising Autoencoder*  
SVC - *Support Vector Classification*  
SVM - *Support Vector Machine*  
TA - *Taxa de acerto*  
XML - *Extensible Markup Language*

## SUMÁRIO

1	<b>INTRODUÇÃO</b> .....	13
1.1	DEFINIÇÃO DO PROBLEMA .....	14
1.2	OBJETIVOS .....	16
1.3	HIPÓTESES DA PESQUISA .....	16
1.4	CONTRIBUIÇÕES .....	17
1.5	ESTRUTURA DO TRABALHO.....	17
2	<b>FUNDAMENTAÇÃO TEÓRICA</b> .....	19
2.1	TIPOS DE APRENDIZAGEM.....	19
2.1.1	Aprendizagem Supervisionada .....	19
2.1.2	Aprendizagem Não Supervisionada .....	20
2.1.3	Aprendizagem Semi Supervisionada .....	20
2.1.4	Aprendizagem por Transferência.....	21
2.1.5	Aprendizagem Autodidata.....	21
2.2	AUTO-ENCODER .....	23
2.3	AUTO-ENCODER CONVOLUCIONAL.....	26
2.4	MÉTODOS DE FUSÃO DE CLASSIFICADORES .....	28
2.5	CONSIDERAÇÕES FINAIS .....	31
3	<b>ESTADO DA ARTE</b> .....	32
3.1	TRABALHO RELACIONADOS .....	32
3.2	SUMÁRIO DA REVISÃO DA LITERATURA .....	42
4	<b>METODOLOGIA</b> .....	46
4.1	MÉTODO PROPOSTO.....	46
4.2	BASES DE DADOS .....	48
4.2.1	Base Flavia.....	48
4.2.2	Base Kyoto .....	51
4.2.3	Base Plantclef 2015.....	52
4.2.4	Base LeafSnap .....	54

4.3	ARQUITETURAS DOS MODELOS .....	56
4.4	Arquitetura CAENN 25 camadas .....	56
4.5	Arquitetura CAENN 20 camadas .....	59
4.6	CLASSIFICAÇÃO .....	63
5	<b>RESULTADOS EXPERIMENTAIS</b> .....	65
5.1	Protocolo Experimental .....	65
5.2	Resultados Modelo CAENN2 .....	66
5.3	Resultados Modelo CAENN1 .....	69
5.4	Resultados das Combinações dos Classificadores .....	72
6	<b>COMPARAÇÃO DOS RESULTADOS OBTIDOS</b> .....	77
7	<b>CONCLUSÃO</b> .....	81
	<b>REFERÊNCIAS</b> .....	83
	<b>APÊNDICE A – EXPERIMENTOS UTILIZANDO <i>GLOBAL AVERAGE POOLING</i> E <i>GLOBAL MAX POOLING</i></b> .....	89
	<b>APÊNDICE B - EXPERIMENTOS UTILIZANDO AS COMBINAÇÕES DE CLASSIFICADORES MÉTRICA F1 SCORE</b> .....	91
	<b>APÊNDICE C – MATRIZES DE CONFUSÃO DOS EXPERIMENTOS COM A BASE LEAFSNAP</b> .....	92

## **IDENTIFICAÇÃO**

Discente Diego Henrique Presner, mestrando em Computação Aplicada à Agricultura, orientado pelo professor Alceu de Souza Britto Júnior, tendo como linha de pesquisa computação, automação e gestão de dados em agricultura, com o objetivo específico de encontrar soluções para as tarefas utilizadas na identificação de plantas, não participante do programa de bolsas.

## 1 INTRODUÇÃO

As plantas possuem um papel fundamental para a existência de vida no planeta Terra, pois elas convertem gás carbônico ( $\text{CO}_2$ ) em oxigênio ( $\text{O}_2$ ) e servem de alimento para grande parte dos seres vivos. Também são utilizadas por diversos segmentos industriais como, por exemplo, produção de alimentos, medicamentos, produtos de higiene, entre outros, possuindo um papel importante nas atividades sociais e econômicas dos países produtores e exportadores.

Nos últimos anos, diversas pesquisas abordam o uso de processamento de imagens, técnicas de visão computacional e reconhecimento de padrões (redes neurais profundas) para a criação de sistemas capazes de identificar plantas. A importância destes trabalhos na linha de pesquisa de identificação/classificação de espécies de plantas, deve-se à vasta biodiversidade, em que muitas destas sofrem risco de extinção ou até mesmo não foram catalogadas/descobertas cientificamente. Ainda, existe a dificuldade de realizar as tarefas de classificação de forma manual (humana). Estudos apontam que a forma automatizada de classificação tem sido eficiente, seus processos demandam menor tempo e quantidade de trabalho ao pesquisador, obtendo assim bons resultados na classificação e rotulação de espécies botânicas (GOËAU et al., 2018).

Para criar um modelo computacional ideal para a tarefa de classificação de forma supervisionada são exigidas bases de dados que possuam uma grande quantidade de imagens rotuladas na fase de treinamento (LECUN et al., 2015). A quantidade de bases com imagens de plantas rotuladas é pequena, devido à grande dificuldade e aos altos custos para se obter estes tipos de dados (LIN et al., 2010). Para contornar o problema da falta de bases com dados rotulados, podemos utilizar métodos de aprendizado de forma não supervisionada (CELEBI et al., 2016), semi-supervisionada (CHAPELLE et al., 2009), autodidata (*self taught learning*) (RAINA et al., 2007) e aprendizagem por transferência (ZHANG et al., 2017). Neste trabalho utilizaremos uma rede neural artificial conhecida como auto-encoder, que emprega o método de aprendizagem não supervisionada/autodidata, utilizando bases não rotuladas, pois estas são mais fáceis de serem encontradas digitalmente, para realizar o treinamento dos modelos computacionais.

## 1.1 DEFINIÇÃO DO PROBLEMA

Na tarefa de classificação de espécies de plantas utilizando sistemas de visão computacional (*machine learning*), encontram-se alguns problemas que podem dificultar e interferir diretamente no desempenho e na capacidade de generalização de modelos de aprendizagem.

Um dos problemas que podemos citar é a pequena quantidade de exemplares de imagens/espécies de plantas rotuladas em grande parte das bases de dados disponibilizadas para treinamento e teste. Tendo em vista a vasta biodiversidade de espécies de plantas existentes no planeta, estas bases possuem uma capacidade de generalização limitada, devido à existência de poucas amostras e variabilidades de espécies que conseguem ser obtidas pelos colaboradores que contribuem para a criação destes repositórios. Sendo assim, a maioria das bases criadas apresentam espécies de plantas encontradas em poucas regiões do planeta, dificultando a criação de um sistema de classificação que seja robusto o suficiente para classificar espécies de plantas em escala global.

A variação morfológica entre as espécies de plantas também é uma das barreiras encontradas. Esta variação pode ser atribuída às mudanças climáticas das regiões em que se encontram as plantas, ocasionando variações nos aspectos das mesmas, como no tamanho, formato, coloração, textura, etc. Estas variações são denominadas na literatura como *intra-espécies*. A Figura 1 apresenta alguns exemplos deste tipo de variação.

**Figura 1-** Exemplos de variações da mesma espécie de planta (*intra-espécies*).

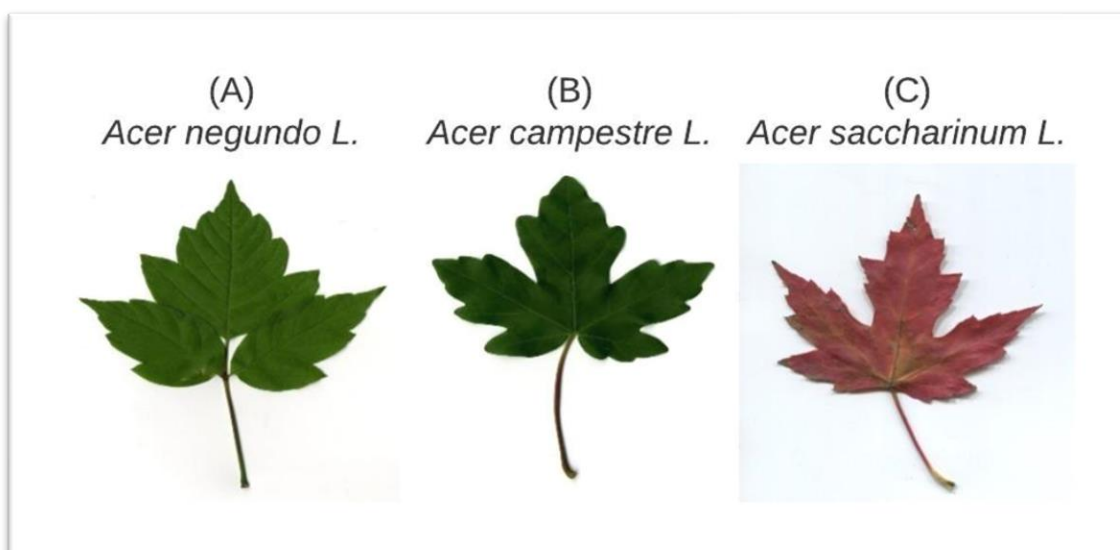


Fonte: O autor.



Existe também a similaridade de aspectos de plantas que pertencem a diferentes espécies, mesmo estas pertencendo a ecossistemas e localidades diferentes. Este é mais um fator que pode influenciar o resultado da classificação do modelo computacional, fazendo com que este atribua o rótulo de uma espécie à outra devido às características semelhantes encontradas entre elas. A Figura 2 apresenta exemplos de similaridade na morfologia de espécies diferentes, que caracteriza uma baixa variabilidade *inter-espécies*.

**Figura 2** - Exemplos de semelhanças entre diferentes espécies de plantas (inter-espécies).



Fonte: O autor.

Outro problema recorrente nas bases de dados é a presença de ruídos nas imagens das plantas. Estes ruídos podem ser ocasionados por sombras, excesso de luminosidade, objetos indesejáveis no plano de fundo da imagem como, por exemplo, fragmentos das próprias plantas que podem se soltar, pedaços de terra, pedras, dedos que sobrepõem as lentes dos sensores na hora de adquirir as imagens, etc. Alguns exemplos de ruídos citados anteriormente podem ser visualizados na Figura 11, apresentada na Seção 4.2.3 deste trabalho.

## 1.2 OBJETIVOS

O objetivo geral deste trabalho é avaliar o uso de representação (extração de características) obtida por meio de aprendizagem autodidata, essa fazendo o uso de dados não rotulados de diferentes domínios, na classificação de espécies de plantas a partir da imagem do componente folha. Para tal, pretende-se atingir os seguintes objetivos específicos:

- Compreender o funcionamento do auto-encoder convolucional e as técnicas de aprendizado, em específico a aprendizagem não supervisionada e a autodidata.
- Treinar os auto-encoders convolucionais em imagens não rotuladas pertencentes a diferentes contextos.
- Aprender diferentes representações usando imagens encontradas em bases de plantas e pertencentes a outro domínio.
- Utilizar as representações de características extraídas pelos auto-encoders treinados na tarefa de classificação de plantas, bem como a fusão destas.
- Avaliar a viabilidade do uso da metodologia proposta com base nos resultados obtidos e os apresentados por outras metodologias levantadas no estado da arte.

## 1.3 HIPÓTESES DA PESQUISA

As hipóteses de pesquisa a serem investigadas nesse trabalho são apresentadas a seguir:

- Uma representação do problema aprendida segundo a abordagem autodidata considerando uma base de imagens do mesmo domínio (folhas de plantas não rotuladas) permite gerar modelos competitivos quando comparados a modelos treinados usando abordagem supervisionada.
- Uma representação do problema aprendida usando abordagem autodidata considerando uma base de imagens de outro domínio (imagens não rotuladas) fornece resultados competitivos quando comparados a modelos treinados usando abordagem supervisionada.

- Um ensemble de classificadores treinados segundo a abordagem autodidata pode contribuir no desempenho da classificação de espécies de plantas.

## 1.4 CONTRIBUIÇÕES

As contribuições geradas por meio deste trabalho podem ser relacionadas aos contextos científico e social. Sob o contexto científico, as principais contribuições são:

- A criação de um modelo treinado com diferentes representações de características.
- A avaliação da viabilidade do uso de representações aprendidas de forma não supervisionada utilizando bases de dentro e fora do contexto das imagens das folhas das plantas.
- A exploração da diversidade das representações aprendidas por meio da combinação de classificadores.

Sob o contexto social, as principais contribuições esperadas são:

- Redução dos custos empregados ao criar bases com uma alta capacidade de generalização ao utilizar os dados não rotulados e de fora do domínio.
- A classificação das plantas, especificamente suas espécies, de forma eficaz, fornecendo informações sobre cada uma delas, contribuindo para a preservação de espécies ameaçadas de extinção.

## 1.5 ESTRUTURA DO TRABALHO

Este trabalho está organizado em seis capítulos. O Capítulo 2 trata dos principais conceitos relacionados ao auto-encoder e alguns dos tipos existentes. No Capítulo 3 é discutido o estado da arte do reconhecimento automático de plantas/doenças utilizando abordagens rasas e profundas de aprendizagem de máquina. No Capítulo

4 é descrita a abordagem proposta para a detecção automática de folhas de plantas utilizando-se imagens e os modelos de auto-encoder treinados para a extração de características da base alvo. No Capítulo 5 são apresentados os resultados obtidos por meio da tarefa de classificação e, por fim, no Capítulo 6, são apresentadas as conclusões.

## 2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os conceitos referentes aos tipos de aprendizagem existentes, o conceito base de um auto-encoder e especificamente o auto-encoder convolucional, e os métodos/regras utilizadas para a fusão de classificadores.

### 2.1 TIPOS DE APRENDIZAGEM

Para desenvolver um modelo para visão computacional ou inteligência artificial, é necessário adotar um método de aprendizagem adequado levando em conta a aplicação a ser criada e a disponibilidade dos dados. Nesta seção, são apresentados alguns dos tipos de aprendizagem mais utilizados.

#### 2.1.1 Aprendizagem Supervisionada

A aprendizagem supervisionada consiste em construir um modelo treinado utilizando as imagens de uma determinada base de dados e os seus rótulos (fase de treinamento). Após treinarmos o modelo, este será capaz de atribuir rótulos de classes a instâncias onde estes são desconhecidos (fase de teste) (KOTSIANTIS, 2007). A aprendizagem supervisionada pode ser formalizada como um problema de inferir a função  $y = f(x)$ , baseado em um conjunto de treinamento  $D = \{(x_1, y_1), \dots, (x_n, y_n)\}$ , onde  $x_n$  são os dados de entrada e  $y_n$  os rótulos. Quando a tarefa supervisionada se trata de uma regressão,  $y$  é um valor contínuo (ou seja,  $y \in \mathbb{R}$ ). Quando se trata de uma classificação,  $y$  é categórico (ou discreto) e seu valor pertence a um conjunto finito de classes  $W = \{w_1, \dots, w_n\}$ . O poder de generalização da função aprendida é dado pela precisão executada em novos dados (conjunto de teste), assumindo que estes venham da mesma distribuição que a utilizada na fase de treinamento do modelo (FIGUEIREDO, 2003).

### 2.1.2 Aprendizagem Não Supervisionada

A aprendizagem não supervisionada consiste na utilização de bases de dados não rotulados, ou que possuam rótulos, mas que estes não sejam utilizados durante a fase de treinamento de um modelo. Esta abordagem permite a utilização de bases de dados construídas por imagens encontradas em diversas fontes, principalmente na internet, essa sendo uma fonte abundante e de fácil acesso, porém as imagens encontradas podem ter uma certa limitação em seu uso. Esta abordagem é utilizada para descobrir padrões relevantes entre os dados vindos de diferentes domínios (HASTIE et al., 2009). Uma abordagem típica desta aprendizagem é o agrupamento (por exemplo, utilizando o *K-means*), porém, outras tarefas como estimação de densidade, redução de dimensionalidade, geração de novas amostras e extração de características, essa sendo a utilizada neste trabalho, podem ser aplicadas.

### 2.1.3 Aprendizagem Semi Supervisionada

Pode-se dizer que a abordagem utilizada pela aprendizagem semi supervisionada consiste em uma fusão dos métodos utilizados pela abordagem supervisionada (Seção 2.1.1) e a não supervisionada (Seção 2.1.5), pois esta utiliza uma grande quantidade de dados não rotulados, juntamente com dados rotulados, esses oriundos da mesma distribuição, na fase de treinamento para aumentar a capacidade de generalização dos modelos treinados. Este método destaca-se por requerer menor esforço humano para rotular os dados das bases utilizadas, gerando uma maior precisão dos modelos treinados por meio dele (ZHU, 2005).

A aprendizagem semi-supervisionada utiliza um conjunto de amostras rotuladas  $D_n$  e não rotulados  $U$ . O objetivo é melhorar o classificador treinado com os exemplos rotulados ( $D_n$ ) utilizando os não rotulados ( $U$ ) (ZHU, GOLDEBERG, 2009). Para tal, uma estratégia é utilizada para rotular de forma automática os exemplos não rotulados a partir dos rotulados existentes.

#### 2.1.4 Aprendizagem por Transferência

A aprendizagem por transferência permite que domínios, tarefas e distribuições utilizadas no treinamento e teste dos modelos sejam diferentes, porém, estes devem possuir uma relação e os dados devem ser rotulados (PAN, YANG, 2009). Essa abordagem também pode ser citada na literatura como: aprendendo para aprender, transferência de conhecimento, transferência indutiva, entre outros nomes (THRUN, PRATT, 2012).

Dado um domínio de origem  $D_s$  e uma tarefa de aprendizagem  $T_s$ , um domínio alvo  $D_t$  e uma tarefa de aprendizagem  $T_t$ , o objetivo da aprendizagem por transferência é melhorar o aprendizado da função alvo preditiva  $f_t(x)$  no domínio de origem ( $D_t$ ) usando o conhecimento aprendido em  $D_s$  e  $T_s$ , onde  $D_s \neq D_t$  ou  $T_s \neq T_t$ , porém possuem uma relação como, por exemplo, a utilização de imagens de espécies de animais diferentes para treinar um modelo que tem por objetivo classificar animais de uma, ou mais classes, especificamente.

#### 2.1.5 Aprendizagem Autodidata

A aprendizagem autodidata (RAINA et al., 2007) é considerada um tipo especial de aprendizagem por transferência (Seção 2.1.3). Nesta abordagem, utiliza-se dados não rotulados de diferentes domínios na fase de treinamento do modelo, por meio da metodologia proposta pela aprendizagem não supervisionada (Seção 2.1.5) para extração de características de alto nível da(s) base(s) alvo. Posteriormente estas representações de características são aplicadas aos dados rotulados, gerando uma nova maneira de representar estes dados e, por fim, um algoritmo padrão de classificação supervisionada como, por exemplo, um classificador do tipo SVM, é utilizado para aprender em função das características obtidas. Por ser um tipo de aprendizado que possui poucas restrições em sua utilização, pode-se aplicar esta abordagem em diversas aplicações onde os dados, rotulados e não rotulados, são considerados escassos.

Na Figura 3, são apresentados alguns exemplos dos tipos de aprendizado citados anteriormente, aplicados na classificação de imagens de elefantes e rinocerontes. As imagens que possuem o fundo laranja, são rotuladas, as demais não.

**Figura 3** - Tipos de aprendizagem de máquina para classificar imagens de elefantes e rinocerontes. As imagens que possuem o fundo laranja, são rotuladas, as demais não.



Fonte: Adaptado de Raina et al., 2007.

De cima para baixo temos: A) Aprendizagem supervisionada (Seção 2.1.1), onde as imagens de ambas as classes são rotuladas; B) Aprendizagem semi-supervisionada (Seção 2.1.3) onde utiliza-se imagens rotuladas e não rotuladas, porém pertencendo às mesmas classes; C) Aprendizagem por transferência (Seção 2.1.4), onde utiliza-se imagens adicionais rotuladas pertencentes a diferentes classes, porém a um mesmo domínio (ambos são animais); D) Aprendizagem autodidata

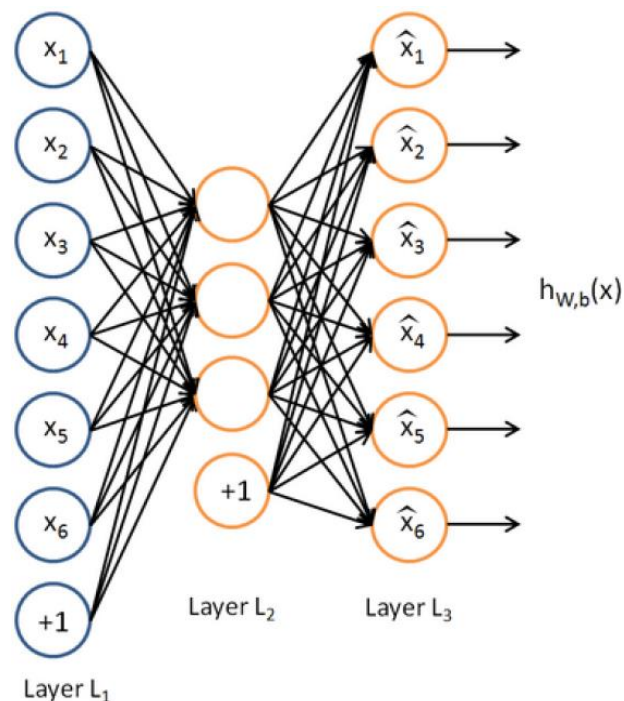


(Seção 2.1.5), onde utiliza-se imagens adicionais pertencentes a um diferente domínio sem serem rotuladas (imagens dos rios) para a extração de características de alto nível da(s) base(s) alvo (imagens dos elefantes e rinocerontes).

## 2.2 AUTO-ENCODER

O auto-encoder (AE) é uma rede neural artificial, que viabiliza o método de aprendizagem não supervisionada e autodidata, recebendo como entrada os dados não rotulados  $\{x_1, x_2, x_3, \dots, x_n\}$ , onde  $x_n \in \mathbb{R}_n$ , em sua primeira camada, extraíndo as principais características da camada de entrada ( $L_1$ ) e reconstruindo os dados ( $\hat{x}_i$ ) da camada de entrada em sua camada final ( $L_3$ ), com base nas principais características extraídas pela camada intermediária ( $L_2$ ). A Figura 4 mostra a arquitetura básica de um auto-encoder tradicional.

**Figura 4** - Estrutura básica de um auto-encoder.



Fonte: Ng et al., 2011.

Um AE é composto por duas estruturas principais: a primeira é denominada codificador (*encoder*) (VINCENT et al., 2010), onde transforma-se os vetores de

características  $x$  em uma representação oculta  $z$ , por meio de uma função  $f_\theta$ , a qual é apresentada a seguir:

$$z = f_\theta(x) = S_f(\mathbf{W}x + b), \quad (1)$$

onde  $f_\theta$  é uma função parametrizada por  $\theta = \{\mathbf{W}, b\}$ ,  $\mathbf{W}$  é uma matriz de pesos entre cada neurônio da camada de entrada e da camada oculta e  $b$  um vetor de viés, este incluído juntamente com os vetores de características.

A segunda estrutura é denominada decodificador (*decoder*) (VINCENT et al., 2010), a qual utiliza a representação  $z$  (principais características), gerada pelo *encoder*, como a entrada de uma função  $g_{\theta'}$ , e por meio desta reconstrói os vetores de características  $\hat{x}$  (Equação 2):

$$\hat{x} = g_{\theta'}(z) = S_g(\mathbf{W}'z + b'), \quad (2)$$

onde  $\mathbf{W}'$  é uma matriz (inversa de  $\mathbf{W}$ ) de pesos entre cada neurônio da camada oculta e da camada de saída do AE.

Ao treinar o AE, tem-se como objetivos, obter um conjunto de pesos dos processos de codificação  $\mathbf{W}$  e decodificação  $\mathbf{W}'$ , minimizando os erros (encontrando os melhores valores para os parâmetros:  $\theta, \theta' = \{\mathbf{W}, \mathbf{W}', b, b'\}$ ) entre os vetores de características de entrada (dados) e os de saída utilizados para a reconstrução, e obter a melhor expressão da camada oculta.

A função utilizada para a minimização de erros é apresentada a seguir (Equação 3) (VINCENT et al., 2010):

$$\theta = \min_{\theta, \theta'} \frac{1}{n} \sum_{i=1}^n L(x^{(i)}, g_{\theta'}(f_\theta(x^{(i)}))), \quad (3)$$

onde  $L$  é uma função de perda (geralmente) do erro quadrático (*Mean Squared Error* - MSD), a qual é representada como  $L(x, \hat{x}) = \|x - \hat{x}\|^2$ . Outra função

alternativa que pode ser utilizada para a minimização de erros é a da entropia cruzada (*cross-entropy*) (RAMOS et al., 2019), que é apresentada a seguir (Equação 4):

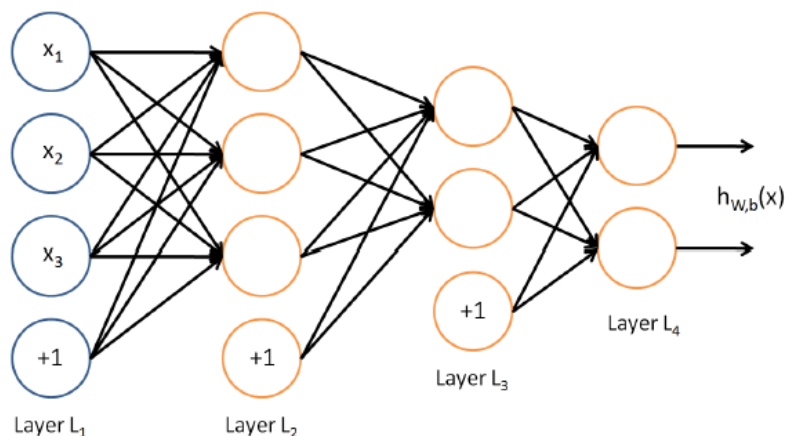
$$L(x, \hat{x}) = - \sum_{k=1}^d x_k \log \hat{x}_k + (1 - x_k) \log(1 - \hat{x}_k) \quad (4)$$

Ao criar um AE deve-se atentar para que ele não acabe gerando uma saída idêntica à sua entrada, pois a sua utilidade se encontra na capacidade de extrair características importantes dos dados de entrada ( $x_n$ ) e sua saída deve ser uma representação baseada nas principais características que se encontram na camada oculta ( $x_i \approx \hat{x}_i$ ).

Existem duas maneiras para garantir que o AE extraia somente as características mais importantes. A primeira consiste em reduzir o tamanho da camada oculta para uma dimensão inferior à da camada de entrada. A segunda maneira é escolher a dimensão do código e a capacidade do codificador e decodificador a partir da complexidade da distribuição a ser modelada. O auto-encoder, quando regularizado, tem a capacidade de utilizar a função de perda para incentivar o modelo a capturar outras propriedades como, por exemplo, a dispersão da representação (*Sparse Autoencoder*), robustez ao ruído (*Denosing Autoencoder*) e insignificância da derivada da representação (GOODFELLOW; BENGIO; COURVILLE., 2016). Esses modelos aprendem codificações de alta capacidade pois são treinados para maximizar a probabilidade dos dados ao invés de copiá-los, extraíndo características úteis de dados não rotulados (FAYYAZ et al., 2015).

Para realizar as tarefas relacionadas às extrações de características profundas, comumente utiliza-se as camadas de codificação (*encoder*) e a camada oculta (*hidden layer*) do AE, sendo assim, inutiliza-se as camadas de saída do modelo (*decoder*), como pode ser visto na Figura 5.

**Figura 5** - Camadas de convolução ( $L_2$  e  $L_3$ ) e a camada oculta ( $L_4$ ).



Fonte: Ng et al., 2011.

Os AE, suas variações e combinações de suas variações são utilizados em diversos estudos como, por exemplo, predição de deslizamentos de terra utilizando o *sparse autoencoder* (SAE) (HUANG et al., 2020), detecção de anomalias em dados de sistemas de monitoramento de saúde com *stacked autoencoder* (BAO et al., 2018), detecção de propriedades moleculares de líquidos com *denoising autoencoder* (MORADZADEH; ALURU., 2018) e diagnóstico de falhas de rolamentos fazendo o uso de uma estrutura formada pela fusão de variações do AE, denominada *sparse stacked denoising autoencoder* (SSDAE) (SUN et al., 2019). Neste trabalho utilizaremos o auto-encoder convolucional (CAE), o qual será apresentado na seção a seguir (Seção 2.3).

### 2.3 AUTO-ENCODER CONVOLUCIONAL

Os AEs totalmente conectados (Figura 4), ignoram a estrutura bidimensional ao processar dados da imagem, convertendo-os em uma estrutura unidimensional e consequentemente inserindo redundâncias nos parâmetros ao forçar a globalização das características. Para resolver este problema, pode-se instanciar as funções de codificação (*encoder*) e decodificação (*decoder*) utilizando CNNs, consequentemente criando um auto-encoder convolucional (CAE). As estruturas básicas das CNNs são as camadas de convolução e de *pooling* (ZHAI et al., 2018).

As camadas de convolução compreendem múltiplos nós convolucionais, os quais permitem receber como entrada mapas de características bidimensionais,

preservando a localidade espacial, onde os parâmetros de aprendizagem são definidos pela matriz de filtragem, a qual realiza a operação de convolução. A função de convolução (RAMOS et. al, 2019) é dada por (Equação 5):

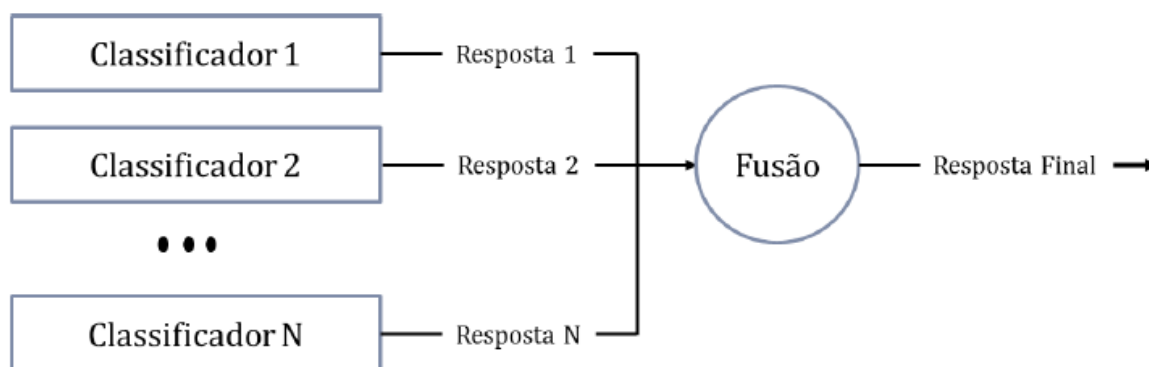
$$O = f(\sum_{i=1}^d K_i * x_i + b \times \mathbf{1}), \quad (5)$$

onde  $\mathbf{1}$  é uma matriz onde todos os elementos são 1,  $b$  é o viés escalar,  $x_i$  ( $i = 1, 2, 3, \dots, n$ ) são os mapas de características bidimensionais,  $K_i$  ( $i = 1, 2, 3, \dots, n$ ) são as matrizes de filtragem,  $d$  são matrizes com dimensões menores (3x3 por exemplo, os chamados *strides*) geradas pela camada de *pooling*,  $f$  é uma função de ativação, geralmente do tipo ReLU (*Rectified Linear Units*), " $*$ " é o operador convolucional e  $O$  é o mapa de características bidimensionais resultante da operação de convolução. Os CAEs podem ser treinados utilizando os algoritmos *Gradient Descente* (GD) ou *Stochastic Gradient Descente* (SGD).

## 2.4 MÉTODOS DE FUSÃO DE CLASSIFICADORES

Os métodos de fusão classificam os objetos por meio da combinação das predições de diversos classificadores, aplicando uma função de fusão em suas predições, como ilustrado na Figura 6.

**Figura 6** - Arquitetura de fusão das respostas de N classificadores.



Fonte: Araújo, Voncarlos M., 2016.

Nesta seção são apresentadas algumas das regras de fusão mais comuns, presentes no trabalho de (KITTLER et al., 1998).

### Voto Majoritário

Uma das estratégias mais simples de unir as respostas dos classificadores é por meio do voto majoritário das respostas. Essa regra estabelece que a classe de determinado objeto é a classe que mais recebeu votos (resposta final) dentre todos os classificadores. Caso haja empates no método de votação, a classe final pode ser predita por meio de uma seleção aleatória entre as classes empatadas (KUNCHEVA, 2004).

Em um problema contendo  $z$  classes possíveis e  $w$  classificadores, a classe de um objeto  $x$ , dada pela votação majoritária, pode ser definida como:

$$\text{classe}(x) = \underset{k=1}{\text{arg max}} \sum_{i=1}^w g(y_i, k), \quad (6)$$

onde  $y_i$  é o rótulo do objeto pelo  $i$ -ésimo classificador, e  $g(y, k)$  é uma função indicativa definida como:

$$g(y, k) = \begin{cases} 1 & \text{se } y = k \\ 0 & \text{se } y \neq k \end{cases} \quad (7)$$

Às vezes, a resposta dos classificadores pode ser um número contínuo que representa probabilidade (*score*) de um objeto  $x$  pertencente à classe  $k$ . Considere  $d_{y, k}$  o *score* dado pelo classificador  $y$  que  $x$  pertence a classe  $k$ , em um problema contendo  $n$  classificadores e  $c$  classes. Os métodos de fusão, apresentados a seguir, baseiam-se nesse *score* para definir a classe de um objeto.

### Regra do Produto

A regra do produto faz a multiplicação dos *scores* gerados pelos classificadores, tornando esta regra sensível aos baixos *scores*, pois um *score* próximo a 0 dado por um dos classificadores para determinada classe pode anular a chance de ela ser escolhida como a classe objeto predita. A regra do produto é definida por:

$$\text{classe}(x) = \underset{k=1}{\overset{z}{\text{arg max}}} \prod_{i=1}^w d_{y_i, k} \quad (8)$$

### Regra da Soma

A regra da soma faz a somatória dos *scores* gerados pelos classificadores e é definida na equação 9.

$$\text{classe}(x) = \underset{k=1}{\overset{z}{\text{arg max}}} \sum_{i=1}^w d_{y_i, k} \quad (9)$$

### Regra da Média

A regra da média simplesmente realiza uma média dos scores atribuídos pelos classificadores, como definido na equação 10.

$$\text{classe}(x) = \underset{k=1}{\overset{z}{\text{arg max}}} \frac{1}{w} \sum_{i=1}^w dy_{i,k} \quad (10)$$

### Regra da Mediana

A regra da mediana consiste em fazer a mediana dos scores gerados pelos classificadores, como resultado da classificação combinada, e é definida pela equação 11.

$$\text{classe}(x) = \underset{k=1}{\overset{z}{\text{arg max}}} \underset{i=1}{\overset{w}{\text{mediana}}} dy_{i,k} \quad (11)$$

### Regra do Máximo

A regra do máximo é realizada assumindo como resultado da combinação o maior score dado dentre os classificadores, como visto na equação 12.

$$\text{classe}(x) = \underset{k=1}{\overset{z}{\text{arg max}}} \underset{i=1}{\overset{w}{\text{max}}} dy_{i,k} \quad (12)$$

### Regra do Mínimo

Na regra do mínimo, o resultado da combinação é o valor mínimo encontrado dentre os scores de cada classificador atribuídos às classes. Essa regra, como a regra do produto, é sensível a baixos scores e é definida por:



$$\text{classe}(x) = \underset{k=1}{\overset{z}{\text{arg max}}} \underset{i=1}{\overset{w}{\text{min}}} dy_i, k \quad (13)$$

## 2.5 CONSIDERAÇÕES FINAIS

Como apresentado anteriormente, podemos ver que os auto-encoders possuem variações, que podem ser aplicadas em diferentes contextos de estudo, assim como, as combinações de suas variações. Para sistemas de visão computacional que utilizam imagens como entrada o uso do auto-encoder convolucional é o mais indicado, devido à sua capacidade de trabalhar com estruturas bidimensionais. Por este motivo iremos utilizar o auto-encoder convolucional para o desenvolvimento deste trabalho.

No Capítulo 3 será apresentado um levantamento do estado da arte, mostrando diferentes aplicações e abordagens utilizadas pelos autores, para o reconhecimento de padrões em imagens de plantas.

### 3 ESTADO DA ARTE

Neste capítulo são apresentados as abordagens e os sistemas de identificação de padrões utilizados por diversos autores que utilizam redes neurais (rasas e profundas). Também serão apresentados diferentes tipos de aprendizagem para executar a tarefa de classificação de plantas.

#### 3.1 TRABALHO RELACIONADOS

A utilização da extração de características das cores e formatos das imagens de folhas foi abordada por Caglayan et al. (2013). Os autores realizaram experimentos utilizando os seguintes algoritmos de classificação: *k-Nearest Neighbors* (KNN), *Support Vector Machine* (SVM), *Naive Bayes* (NB) e *Random Forest* (RF), e as imagens utilizadas para treinamento e classificação foram retiradas da base de dados *Flavia Data Base*, a qual é composta por 1908 imagens de folhas, divididas em 33 espécies/classes de plantas. As características foram divididas em dois grupos, o primeiro denominado SF (*Shape Features*) contendo cinco descritores fundamentais, os quais são: a distância média entre quaisquer dois pontos nas bordas das folhas (L), o comprimento da nervura principal (longitudinal -  $L_v$ ), a maior distância de uma folha (transversal -  $W$ ), a área da folha ( $A$ ) e o perímetro foliar ( $P$ ). O segundo grupo, denominado CF (*Color Features*), é dividido em dois subconjuntos: o primeiro contém a média ( $\mu$ ) e o desvio padrão ( $\sigma$ ) dos valores de intensidade dos canais RGB (as cores vermelho, verde e azul do inglês *Red-Green-Blue*), e a média desses canais, enquanto o segundo subconjunto contém os histogramas dos canais RGB. Foram realizados três experimentos: o primeiro utilizando apenas as características do grupo SF, o segundo utilizando as características SF em conjunto com as CF do primeiro subconjunto e o terceiro experimento utilizando todos os conjuntos de características (SF + CF1 + CF2). Os testes foram executados utilizando duas abordagens de amostragem: *Random Sampling* (RS), com um protocolo de 80% para treinamento e 20% para testes, e *Cross Validation* (CV), onde a base foi particionada em 10 diretórios (*folds*) de mesmo tamanho.

As melhores taxas de acerto, para todos os experimentos, foram de 87,61% CV, 93,95% RS e 96,32% RS, respectivamente, todas pertencendo ao classificador

RF. Como trabalhos futuros, os autores sugerem o uso de características de textura, em conjunto das características morfológicas e de cores, assim como amostras de folhas coletadas durante as quatro estações do ano.

Kumar et al. (2016), utilizaram o algoritmo *k-Nearest Neighbors* (KNN) e as características extraídas das bordas das imagens das folhas, usando características baseadas nas arestas e no formato das folhas. Para execução do experimento, foi utilizada a base de dados *Flavia Data Base* (mesma citada anteriormente). Os autores selecionaram 6 características manualmente (*handcrafted*), a saber: espessura, arredondamento, dispersão, *Edge Filling Length* (EFL), contagem de ramais e contagem de *loops*. Após a etapa de extração de características, o algoritmo KNN é utilizado para calcular a distância euclidiana que existe entre os valores (características) do vetor de testes e os valores de todas as amostras computadas.

Foram executados testes utilizando como valores de K (vizinhos): K=1, K=3 e K=5, e as taxas de acerto/acurácia obtidas foram de 91,56%, 94,37% e 92,69%, respectivamente. O melhor resultado obtido foi com o parâmetro K = 2, utilizando 1100 amostras de teste, obtendo assim, uma acurácia de 94,37%, resultado que ultrapassa as taxas de acerto das pesquisas correlatas citadas pelos mesmos.

Yang et al. (2017), propõe um método para classificação de plantas utilizando representações obtidas por meio de um Stacked Auto-encoder (SAE). Foram coletadas 90 imagens de folhas de diferentes espécies e em diferentes estações do ano, as quais passaram pelas etapas de pré-processamento e aumento de data, este consistindo em rotacionar cada amostra da base de dados em ângulos 0°, 30°, 60° e 90°. O SAE utilizado no experimento possui uma arquitetura composta por: uma camada de entrada, três camadas ocultas codificadoras e um classificador do tipo *Softmax* em sua saída. Na Tabela 1 são apresentados os resultados obtidos por diferentes arquiteturas utilizadas, assim como, o resultado obtido pelo método proposto:

**Tabela 1** - Comparação dos métodos extratores de características.

<b>Método</b>	<b>Taxa de acerto (%)</b>
SAE 1 camada + Classificador Softmax	85,40
SAE 2 camadas + Classificador Softmax	88,60
SAE 3 camada + Classificador Softmax (método dos autores)	93,30

Fonte: Adaptado de Yang et al. (2017).

Como pode-se observar na Tabela 1, os autores obtiveram a taxa de acerto de 93,30%, para a estrutura codificadora do SAE composta por três camadas e um classificador do tipo *Softmax* em sua saída.

Fuentes et al. (2018), desenvolveram um sistema automatizado para realizar a classificação de folhas de videiras. Foram coletadas três amostras de cada um dos dezesseis cultivares utilizados para a realização dos experimentos. Os autores criaram dois modelos utilizando redes neurais artificiais (ANN). O primeiro modelo foi treinado utilizando 12 parâmetros/características morfológicas e colorimétricas das amostras de folhas de cada um dos 16 cultivares em que o experimento foi realizado, como, por exemplo, o perímetro, a área, a escala de cor RGB (*Red-Green-Blue*), a dimensão fractal, entre outros. O segundo modelo foi treinado utilizando, como entrada, 2049 valores de comprimentos de onda, que foram obtidos por meio de espectroscopia no infravermelho próximo (NIR). Ambos os modelos foram treinados utilizando um protocolo de 70% para treinamento, 15% para validação e 15% para testes.

A taxa de acerto/acurácia obtida no primeiro modelo foi de 94,20%, com acurácia de 100% para fase de treino (84 amostras), 90,50% na validação (27 amostras) e 71,40% nos testes (27 amostras). Para o segundo modelo, utilizando os comprimentos de onda, foi obtido 91,70% de taxa de acerto/acurácia, com 100% para treino (88 amostras), 81,80% na validação (28 amostras) e 63,6% nos testes (28 amostras).

Araújo et al. (2018), propuseram uma abordagem de classificação hierárquica de gêneros e espécies de plantas fazendo o uso de classificação de objetos refinados (FGOC - *Fine-grained object classification*) e fusão de modelos de aprendizagem profunda. Segundo os autores, uma das maiores barreiras na classificação de plantas é a alta variação entre as espécies e a baixa entre o gênero. Foram utilizadas duas CNNs em cada nível de hierarquia (gênero - espécie), a primeira rede convolucional é treinada utilizando como entrada as imagens da folha inteira, e a segunda é treinada com fragmentos das imagens da folha, que são obtidos por meio de técnicas de aumento de dados. Após a etapa de aprendizado de características profundas, as saídas/resultados obtidas em cada rede (gêneros das plantas) são fundidas para definir qual é o resultado final da classificação (gênero decidido). Após a etapa de fusão, é verificada a quantidade de classes que o gênero escolhido possui, se o número de classes for igual a 1, a espécie já pode ser definida, caso o número de

classes seja maior que 1, a etapa de aprendizado de características finas será executada, ou seja, as imagens serão avaliadas por mais duas CNNs e suas saídas passarão por uma etapa de fusão das espécies previstas, sendo assim obtido o resultado que mostra a qual espécie de planta as imagens pertencem. Cada uma das CNNs utilizadas são um modelo da *GoogLeNet*, que foram pré-treinadas na base de dados da *ImageNet*, que é uma base amplamente utilizada em desafios/competições de aplicações de classificação de plantas (RUSSAKOVSKY, 2015).

Os resultados de acurácia obtidos foram de 81,01%, 86,97% e 77,96% para família, gênero e espécie, respectivamente. Como os autores não utilizaram a estrutura hierárquica da família, a média de classificação final (gênero/espécie) é de 86,44%. O resultado obtido ultrapassa os resultados de outros autores participantes do desafio/competição *ImageCLEF2015*.

Um método de aprendizagem profunda de extração de características foi proposto por ZHU et al. (2018), utilizando uma SVM para executar a tarefa de classificação de diferentes partes das plantas. A base de dados utilizada por eles é a *LifeCLEF2015*, onde foram selecionadas algumas amostras de cada parte das plantas, especificamente das flores, folhas, frutos, galhos e hastes. Posteriormente as imagens foram divididas em subconjuntos, separadas de acordo com o seu conteúdo. A estrutura da CNN utilizada por eles é compreendida por 16 camadas convolucionais, acompanhadas por 3 camadas totalmente conectadas (FC) e uma camada final *softmax*. As duas primeiras FCs possuem 4096 canais e a terceira possui 1000 canais. Foram também utilizadas 5 camadas de *max-pooling* com o parâmetro *stride* igual a 2. Para a realização dos experimentos, foram utilizados dois conjuntos de vetores de características, o primeiro denominado de "*Feature1*", foi extraído da última camada totalmente conectada, este sendo composto pelos dois canais de 4.096 características cada (C1 e C2), e o canal de 1000 características (C3), este sem a aplicação do *softmax* e com a função ReLU (*Rectified Linear Units*) desativada. Já o segundo conjunto foi gerado com os canais C1 e C2 com a função ReLU ativada e o canal C3 com a *softmax aplicada*. Após as características serem extraídas pela CNN, uma SVM linear é alimentada com estas. Os resultados obtidos pelos autores são apresentados na Tabela 2:

**Tabela 2** - Resultados da CNN e combinação de canais.

Base de dados	Feature1		Feature2	
Flor	88,2%(C1)	88.5%(C1+C3)	88,8%(C1)	88,8%(C1+C3)
Folha	65,4%(C1)	64.2%(C1+C3)	67,1%(C1)	67,1%(C1+C3)
Fruta	87,5%(C1)	87.9%(C1+C2)	89,6%(C2)	90,2%(C2+C3)
Ramo	69,5%(C2)	70.3%(C2+C3)	71,2%(C2)	71,2%(C2+C3)
Haste	60,3%(C1)	60.6%(C1+C3)	64,9%(C2)	65,2%(C2+C3)

Fonte: Adaptado de Zhu et al. (2018).

Como pode-se observar na Tabela 2, os experimentos com o conjunto de características “*Feature2*”, aplicando *softmax* e a função ReLU ativada, obtiveram as melhores taxas de acerto, combinando os canais na maioria dos subconjuntos. No subconjunto das flores a taxa de acerto foi a mesma utilizando a fusão dos canais C1+C3 e C1, onde pode-se concluir que existe a possibilidade de se obter o mesmo resultado utilizando apenas um canal, com menos características, diminuindo o custo computacional empregado na classificação do subconjunto das flores.

Khamparia et al. (2020), propuseram uma abordagem híbrida, fundindo redes neurais convolucionais e auto-encoders. A estrutura (rede neural) híbrida criada por eles foi denominada de rede convolucional codificadora (*Convolutional Encoder Networks* - CEN). O modelo gerado foi utilizado na tarefa de classificação de doenças nas folhas das plantas (considerando 3 plantações/cultivares e 5 tipos de doença), utilizando a base de dados *PlantVillage*, a qual é composta por 50.000 imagens de folhas saudáveis e doentes. Os autores selecionaram 900 imagens da base de dados para desenvolver os experimentos, utilizando 600 imagens para treinamento e 300 imagens para teste, as quais foram pré-processadas e redimensionadas para 32x32x3. A CEN criada é composta por 8 camadas de convolução, sendo paralelamente realizada uma normalização em lote de cada uma destas camadas. Também são utilizadas duas camadas de *max-pooling*, a primeira após a segunda camada de convolução e a segunda após a quarta camada, com o objetivo de reduzir a dimensão da matriz de convolução (quantidade de características).

Foram realizados experimentos alterando o tamanho dos filtros das camadas convolucionais (2x2 e 3x3 *pixels*) e o número de épocas da etapa de treinamento do *encoder* (3, 25, 50 e 100), e como melhores resultados foram obtidas acurácias

médias de 83,00% e 91,00%, para as configurações de filtro 2×2 e 3×3 *pixels*, respectivamente.

Na pesquisa de Tran et al. (2019), os autores fizeram o uso de redes neurais convolucionais (CNNs) para analisar sintomas causados pela falta de nutrientes (Cálcio – Ca, Potássio – K e Nitrogênio - N) durante as fases de folhagem e frutificação do tomate, com o objetivo de prevenir/mitigar as possíveis deficiências/doenças que possam afetar as plantas e, conseqüentemente, os níveis de produção. Um total de 571 imagens foram capturadas contendo a espécie e o estágio de crescimento das plantas, utilizando um protocolo de 80% (461 imagens) para treinamento e 20% (110 imagens) para validação. Foram utilizadas as arquiteturas de aprendizado profundo *Inception-ResNet v2*, essa utilizando aprendizagem supervisionada (treinada com dados rotulados), e auto-encoder, utilizando a aprendizagem não supervisionada. Posteriormente foi utilizada *Ensemble Averaging* (EA) (métodos de combinação) para combinar os dois modelos gerados, citados anteriormente.

As taxas de acerto obtidas pelas redes *Inception-ResNet v2* e o Autoencoder foram de 87,27% e 79,09%, respectivamente. Após combinar os resultados das redes por meio do método *Ensemble Averaging*, os autores obtiveram uma acurácia de 91,00%, provando a eficiência da utilização de sistemas criados com base em redes neurais profundas para identificação, com antecedência, de sintomas causados pela falta de nutrientes nas plantações de tomate, sugerindo como trabalhos futuros, a realização de pesquisas utilizando outros tipos de culturas.

Ramos et al. (2019), apresentam uma abordagem baseada no uso do auto-encoder convolucional (CAE) para a etapa de treinamento e Máquina de Suporte Vetorial (SVM) para classificar as espécies de plantas por meio das imagens das folhas. Para a realização dos experimentos, eles utilizaram a base de dados *Flavia Data Base*. O CAE possui uma estrutura formada por sete camadas de convolução, três camadas de *pooling*, sendo duas camadas de *max-pooling* e uma de *average-pooling*, e três camadas de *unpooling*. O treinamento da rede (CAE) foi realizado com as imagens da base e os vetores de características (*deep features*) foram gerados de forma não supervisionada. Os autores utilizaram a técnica de validação cruzada (*10 folds*) para chegar aos melhores valores a serem utilizados nos parâmetros durante a etapa de treinamento, os quais são: 103 épocas, tamanho do lote (*batch*): 60, função de ativação *ReLU* (*Rectified Linear Units*) e otimizador *RMSprop*, com uma taxa de

aprendizagem de 0,001%. Posteriormente a SVM foi treinada utilizando um protocolo (*holdout*) de 64% para treinamento, 16% para validação e 20% para teste.

A taxa de acurácia obtida pela SVM foi de 94,74%, comprovando que o CAE possui capacidade de extrair características significativas das imagens das folhas. Como trabalhos futuros, eles propõem o uso de outros modelos de AEs, a saber: *Sparse Autoencoders*, *Denosing Autoencoders*, *Contractive Autoencoders*, entre outros.

No artigo de Wang et al. (2019), uma rede neural convolucional siamesa é proposta para resolver o problema de classificação de folhas, utilizando pequenas amostras retiradas das bases de dados Flavia, Swedish e LeafSnap, buscando resolver os problemas de amostras esparsas nas tarefas de classificação. A chave para este método é extrair características das imagens usando uma rede neural convolucional e construir o espaço métrico usando o conceito de semelhança entre as diferentes características da imagem. A qualidade do espaço métrico e a seleção das amostras supervisionadas determinam a precisão da classificação do vizinho mais próximo.

Por meio da curva de função de perda e precisão, provou-se que um dispositivo de otimização de estrutura que atua no espaço métrico é eficaz. Os resultados experimentais também mostram que quando o número de amostras de treinamento é 20, a precisão de classificação deste método é a mais alta, 95,32%, 91,37% e 91,75% de acurácia são obtidas dos conjuntos de dados Flavia, Swedish e Leafsnap, respectivamente. Segundo os autores, esses resultados são competitivos no campo de classificação de aprendizagem profunda. Eles ressaltam que embora o método tenha alcançado bons resultados nas bases de dados utilizadas, mais pesquisas são necessárias para melhorar a capacidade de generalização do modelo.

Segundo Zheng et al. (2019), redes neurais profundas apresentam dificuldades em extrair características profundas dos contornos das folhas de plantas/cultivares, sendo assim, propuseram um método composto por aprendizagem de contornos em múltiplas escalas (MCFL) e aprendizagem por representação de florestas (FRL). A base de dados utilizada para treinamento foi a *SoyCultivarVein*, sendo composta por 600 imagens de folhas coletadas em 100 cultivares de soja localizados na província de Jilin, na China.

A primeira etapa (MCFL) do método consiste na extração das características, utilizando o modelo de *Richer Convolutional Features* (RCF), este sendo desenvolvido



para extrair características especificamente das bordas dos objetos encontrados nas imagens (LIU et al., 2017). Posteriormente os vetores e características passam pelas etapas/camadas do método FRL, primeiramente, sofrendo um processo de aumento de dados (*data augmentation*) gerando um vetor com 800 características. Em seguida, a dimensionalidade do vetor é reduzida utilizando um *encoder*, desta forma gerando um vetor com 400 características. O algoritmo de floresta em cascata, denominado *gcForest* (ZHOU et al., 2017) recebe o vetor, este sendo processado pelo modelo, gerando um vetor com apenas 8 características. Em seguida este vetor passa por uma etapa de *decoder*, obtendo como saída um vetor possuindo 16 características, este sendo utilizado para treinar uma KNN para classificar as imagens.

Os autores obtiveram uma taxa de acerto, quase perfeita, de 99,45%, ultrapassando os resultados obtidos por pesquisas mencionadas no estado da arte levantado por eles.

Na pesquisa de Chen et al. (2020), os autores utilizaram a aprendizagem por transferência entre redes neurais profundas para a classificação e identificação de doenças em imagens de folhas de arroz e milho. Foram utilizados a rede *VGGNet*, pré-treinada na base de dados *ImageNet*, e o módulo *Inception*, pois os pesos utilizados no modelo criado não foram gerados por meio de treinamento do zero, mas sim pela rede pré-treinada na base de dados citada anteriormente. A base de dados utilizada para treinar o modelo criado pelos autores (camadas inferiores da rede *VGGNet-19* com alguns ajustes) foi disponibilizada pela Fundação de Botânica Subtropical de Fujian, localizada na província de Xiamen, na China, sendo composta por 500 imagens de plantas doentes de arroz e 466 de milho que foram pré-processadas utilizando a ferramenta *Photoshop* e redimensionadas para 224 x 224 x 3 (imagens coloridas) *pixels*.

Como resultado foram obtidas taxas de acerto acima de 91,83%, e mesmo sob condições desfavoráveis como, por exemplo, a variação na luminosidade das imagens, obtiveram uma média de 92,00% na tarefa de classificação de plantas de arroz e 95,00% em taxa de acerto.

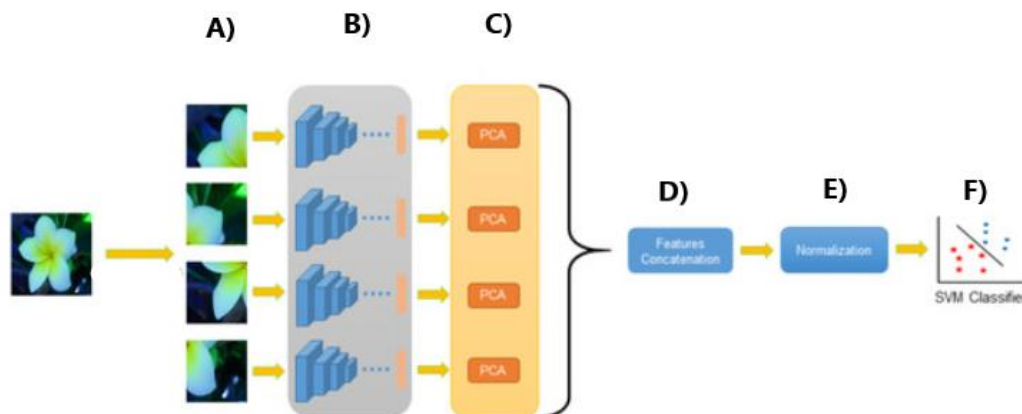
Trang et al. (2020), propõe um método para identificação de doenças de plantas através de um conjunto de imagens das folhas. Este método é desenvolvido com base em um autoencoder convolucional profundo com diferentes arquiteturas. As imagens originais são inicialmente reconstruídas pela estrutura codificadora do CAE e, em seguida, usando apenas parte do codificador, as representações de características

das imagens são extraídas. As representações de características alimentam um classificador do tipo SVM e um conjunto de imagens de plantas da base *Plant Village* é utilizado para avaliar o desempenho de reconstrução das imagens por meio do autoencoder convolucional. Posteriormente na etapa de classificação, os resultados da simulação obtidos mostraram que a diferença da arquitetura do autoencoder convolucional e os núcleos do classificador fornecem uma precisão média diferente.

Como resultado foram obtidas taxas de acerto de 98,12%, para a estrutura codificadora com profundidade igual a três e com o classificador do tipo SVM, este utilizando um kernel do tipo RBF, e 98,80%, para a estrutura codificadora com profundidade igual a três e com o classificador do tipo SVM utilizando um kernel do tipo Poly, esta última sendo o melhor resultado para a classificação de doenças de plantas.

Turkoglu et al. (2021), desenvolveram um sistema de reconhecimento de plantas baseado em Rede Neural Convolucional Multi-Divisional (MD-CNN) abordando a classificação de espécies de plantas. No sistema proposto, as imagens das plantas, retiradas das bases Flavia, Swedish, ICL, Foliage, Folio, Flower17, Flower102, e LeafSnap, foram divididas igualmente em  $n \times n$  e, em seguida, as representações de características foram extraídas para cada parte usando uma Rede Neural Convolucional (CNN). Para cada parte das características profundas obtidas, as características efetivas são selecionadas usando o algoritmo *Principal Component Analysis* (PCA). Posteriormente as principais características obtidas são combinadas e a classificação é realizada usando o método Support Vector Machine (SVM). A Figura 7 mostra as etapas da metodologia proposta pelos autores.

**Figura 7** - Etapas da metodologia: A) Divisão das amostras em  $n \times n$ ; B) Extração das características com a CNN; C) Análise dos principais componentes, D) Concatenação das características, E) Normalização e F) Classificação com a SVM.



Fonte: Adaptado de Turkoglu et al. (2021).

Como resultado as bases Flavia, Swedish, ICL, Foliage, Folio, Flower17, Flower102, e LeafSnap, obtiveram taxas de acerto de 100%, 100%, 99,77%, 99,93%, 100%, 97,87%, 98,03% e 97,80%, respectivamente.

### 3.2 SUMÁRIO DA REVISÃO DA LITERATURA

No estado da arte (Seção 3.1) foram apresentados alguns trabalhos em que os autores fazem o uso de modelos de aprendizagem profunda e rasa para realizar a tarefa de classificação de plantas e doenças que podem afetá-las. Alguns autores fazem o uso de características extraídas manualmente (Kumar et al., 2016) e outros utilizam as representações geradas pelos modelos de aprendizagem profunda (Khamparia et al., 2020). Pode-se constatar também que os autores utilizaram as mesmas bases para treinar os seus modelos, ou fizeram o uso de modelos pré treinados para realizar a classificação das plantas, o que difere do proposto neste trabalho. Algumas bases de dados possuem poucas amostras, o que pode dificultar a capacidade de generalização dos modelos gerados, tornando importante a avaliação do uso de abordagens de aprendizagem não supervisionada e autodidata.

Nesta seção é apresentada uma breve descrição sobre as metodologias aplicadas para a classificação de espécies de plantas utilizadas pelos autores citados na seção 3.1, além de apresentar os resultados obtidos pelos mesmos ao testarem suas metodologias. Na Tabela 2, é apresentado um resumo das metodologias empregadas, o tipo de extrator de características utilizado, bem como o método de classificação, a base de dados utilizada para os testes, e os resultados obtidos pelos autores.

Tabela 3 - Sumarização das Metodologias.

(contínua)

<b>Autores</b>	<b>Metodologia Empregada</b>	<b>Extrator de Características</b>	<b>Método de Classificação</b>	<b>Base de Testes</b>	<b>Melhores TA (%)</b>	
(Caglayan et al., 2013)	A utilização da extração de características das cores e formatos das imagens de folhas.	Grupos de características SF e CF.	KNN, SVM, RF e NB	Flavia Data Base	96,32% (utilizando RS)	
(Kumar et al., 2016)	A utilização das características extraídas das bordas das imagens das folhas, usando características baseadas nas arestas e no formato das folhas.	Espessura, arredondamento, dispersão, <i>Edge Filling Length</i> (EFL), contagem de ramais e contagem de <i>loops</i> .	KNN	Flavia Data Base	94,37% (utilizando K = 2)	
(Yang et al. 2017)	Classificação de espécies de plantas utilizando SAE.	Vetores de características profundas extraídos por diferentes arquiteturas de modelos SAE.	Softmax	Imagens extraídas da área de estudos e rotacionadas em diferentes ângulos.	93,30%	
(Fuentes et al., 2018)	Desenvolveram um sistema automatizado para realizar a classificação de folhas de videiras por meio de redes neurais.	1ª ANN – treinada com características de cor e forma das plantas. 2ª ANN – treinada com os valores de comprimento de onda NIR.	ANN	Imagens das folhas e comprimentos de onda da área de estudo.	Modelo (1ªANN): 94,20%	Modelo (2ªANN): 91,70%
(Araújo et al., 2018)	Classificação hierárquica de gêneros e espécies de plantas fazendo o uso de classificação de objetos refinados e fusão de modelos de aprendizagem profunda.	Características profundas extraídas pelas CNNs das imagens inteiras e fragmentadas.	CNN	LifeCLEF2015	86,44% (média de classificação final gênero/espécie)	
(ZHU et. al, 2018)	Método de aprendizagem profunda de extração de características de multicomponentes.	Vetores de características profundas aprendidas pelas camadas da CNN.	SVM	LifeCLEF2015	<i>Feature1</i> : 88,50% (flor)	<i>Feature2</i> : 90,20% (fruta)

Tabela 3 - Sumarização das Metodologias.

(contínua)

Autores	Metodologia Empregada	Extrator de Características	Método de Classificação	Base de Testes	Melhores TA (%)	
(Khamparia et. al, 2020)	Fusão de redes neurais convolucionais e auto-encoders para a identificação de doenças nas folhas das plantas.	Vetores de características profundas aprendidas pelas camadas da CEN.	CEN	PlantVillage	Stride 2x2: 83,00%	Stride 3x3: 91,00%
(Tran et al., 2019)	Uso de CNNs para analisar sintomas causados pela falta de nutrientes durante as fases de desenvolvimento do tomate.	Vetores de características profundas aprendidas pelas camadas da CNN + AE.	<i>Inception-ResNetv2</i> + AE	Imagens extraídas da área de estudos.	<i>Inception-ResNetv2</i> : 87,27%	AE: 79,09%
(Ramos et al., 2019)	Classificação de espécies de plantas utilizando CAE.	Vetores de características profundas aprendidas pelo CAE.	SVM	Flavia Data Base	94,74%	
(Zheng et al., 2019)	Utilizaram um método composto por aprendizagem de contornos em múltiplas escalas (MCFL) e aprendizagem por representação de florestas (FRL) utilizando imagens de folhas de soja.	MCFL e FRL	KNN	SoyCultivarVein	99,45%	
(Wang et al., 2019)	Classificação de espécies de plantas utilizando redes neurais siamesas.	Vetores de características profundas aprendidas pela CNN.	SVM	Flavia, Swedish e LeafSnap	95,32%, 91,37% e 91,75%, respectivamente.	

**Tabela 3 - Sumarização das Metodologias.**

(conclusão)

<b>Autores</b>	<b>Metodologia Empregada</b>	<b>Extrator de Características</b>	<b>Método de Classificação</b>	<b>Base de Testes</b>	<b>Melhores TA (%)</b>
(Chen et al., 2020)	Classificação e identificação de doenças em imagens de folhas de arroz e milho.	Vetores de características profundas aprendidas pelas camadas inferiores da <i>VGGNet-19</i> .	VGGNet-19	Base disponibilizada pela Fundação de Botânica Subtropical de Fujian.	91,83%
(Trang et al., 2020)	Classificação e identificação de doenças em imagens de folhas de plantas.	Vetores de características profundas extraídos por diferentes arquiteturas de modelos CAENN.	SVM	PlantVillage	98,80%
(Turkoglu et al., 2021)	Sistema de reconhecimento de plantas baseado em Rede Neural Convolutacional Multi-Divisional (MD-CNN) abordando a classificação de espécies de plantas.	Vetores de características profundas extraídas pela MD-CNN.	SVM	Flavia, Swedish, ICL, Foliage, Folio, Flower17, Flower102, e LeafSnap	100%, 100%, 99,77%, 99,93%, 100%, 97,87%, 98,03% e 97,80%, respectivamente.

Fonte: O autor.

## 4 METODOLOGIA

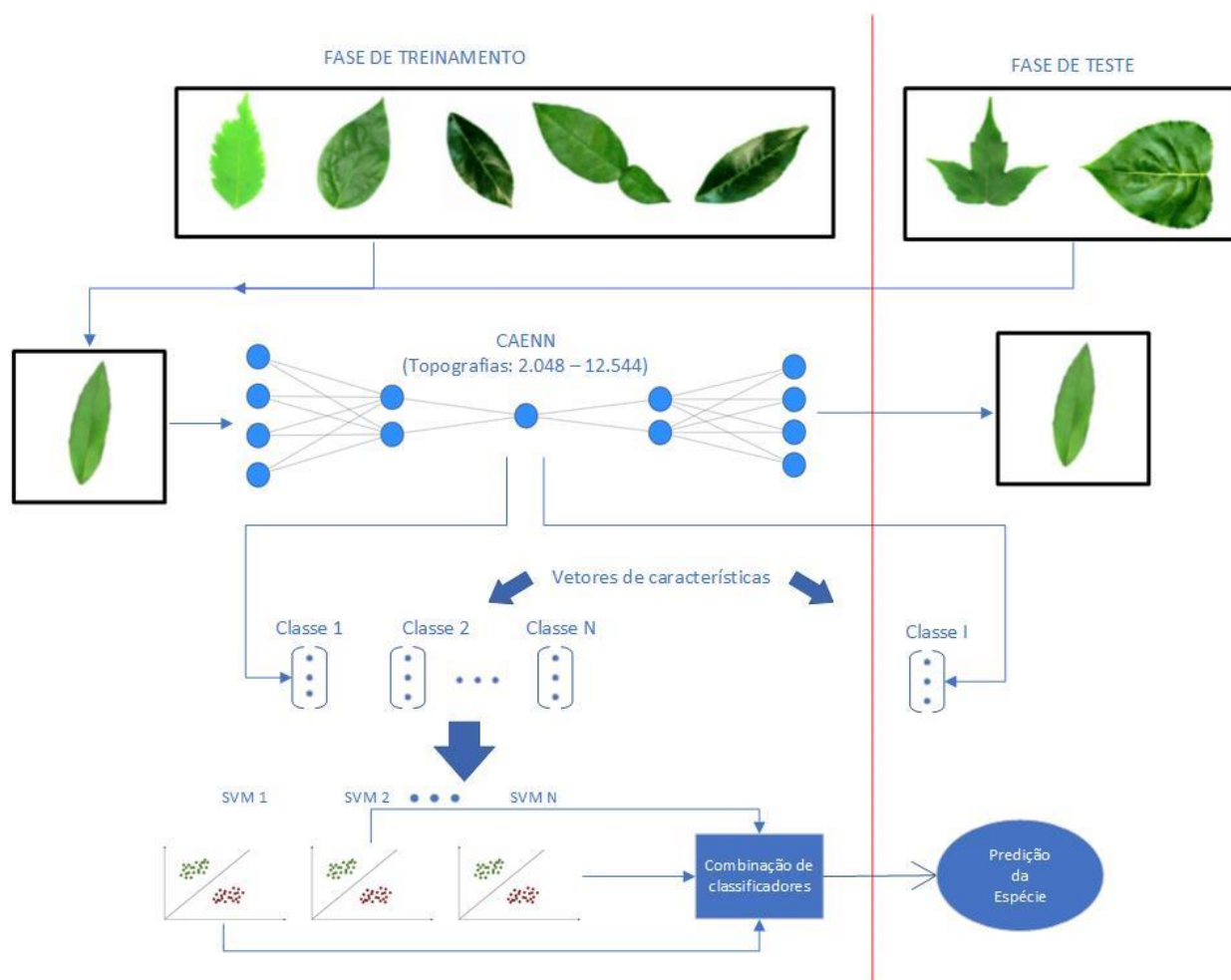
Neste capítulo são apresentados o método proposto, as bases de dados utilizadas para treinamento e classificação, a estrutura de criação do modelo a ser treinado e utilizado na tarefa de extração de características (CAE), os classificadores utilizados e as métricas utilizadas até o momento.

### 4.1 MÉTODO PROPOSTO

Este trabalho apresenta a classificação de espécies de plantas utilizando representações geradas por modelos CAENN (*Convolutional Auto-Encoder Neural Network*) treinados com as bases de dados não rotuladas, pois deste modo aplica-se os princípios da aprendizagem autodidata. Os modelos gerados foram utilizados para a extração de características das bases de dados alvo. Posteriormente os vetores de representações de características obtidos por meio dos modelos extratores de características foram utilizados para o treinamento de classificadores rasos. Um *ensemble* de classificadores foi utilizado para realizar a classificação das espécies e avaliação das representações geradas pelos CAENNs por meio das taxas de acerto obtidas. Posteriormente, foram utilizadas regras de fusão de classificadores para combinar os resultados obtidos por cada representação de características extraídas por cada um dos CAENN's. O fluxograma (Figura 8) a seguir ilustra as etapas executadas durante esta pesquisa:



**Figura 8** - Fluxograma do método proposto para a classificação de espécies de plantas.



Fonte: O autor.

Como pode ser visto na Figura 8, a primeira etapa consiste no preparo das bases de dados para treinamento dos modelos e extração de características. Posteriormente são treinados os modelos extratores de características por meio do auto-encoder convolucional, utilizando as bases de dados apresentadas nas Seções 4.2.1 e 4.2.2. Tais modelos são utilizados para extrair as características das bases de dados alvo apresentadas nas seções 4.2.1, 4.2.3 e 4.2.4, respectivamente. Após os vetores de características serem extraídos das bases alvo, esses são submetidos aos classificadores, obtendo-se as taxas de acerto individuais de cada representação. Por fim, é realizada a fusão das predições obtidas por cada representação, por meio da combinação dos classificadores.

## 4.2 BASES DE DADOS

Nesta seção são apresentadas, em detalhes, as bases de dados utilizadas nesse trabalho para o treinamento dos modelos extratores de características, assim como, as bases alvo, que tiveram suas características extraídas para o treinamento e teste dos classificadores.

### 4.2.1 Base Flavia

A base de dados *Flavia Data Base* (Wu et al., 2007) é uma das bases utilizadas para o treinamento dos CAENNs, e pertence ao mesmo domínio do problema alvo. A base possui 1 GB de tamanho e é disponibilizada gratuitamente na *internet*. A base é composta por um repositório de 1908 imagens de plantas divididas em 33 espécies/classes, conforme é mostrado na Tabela 4:

**Tabela 4** - Estrutura da base de dados Flavia Data Base.

(contínua)

INDICE	Nome Científico	Nome Popular (em inglês)	Qtd. Arq./Classe
1	<i>Phyllostachys edulis</i> (Carr.) Houz.	Pubescent Bamboo	58
2	<i>Aesculus chinensis</i>	Chinese Horse Chestnut	62
3	<i>Berberis anhweiensis</i> Ahrendt	Anhui Barberry	64
4	<i>Cercis chinensis</i>	Chinese Redbud	71
5	<i>Indigofera tinctoria</i> L.	True Indigo	69
6	<i>Acer palmatum</i>	Japanese Maple	55
7	<i>Phoebe nanmu</i> (Oliv.) Gamble	Nanmu	61
8	<i>Kalopanax septemlobus</i> (Thunb. ex A.Murr.) Koidz.	Castor Aralia	51
9	<i>Cinnamomum japonicum</i> Sieb.	Chinese Cinnamon	54
10	<i>Koelreuteria paniculata</i> Laxm.	Goldenrain Tree	58
11	<i>Ilex macrocarpa</i> Oliv.	Big-fruited Holly	49
12	<i>Pittosporum tobira</i> (Thunb.) Ait. f.	Japanese Cheesewood	62
14	<i>Chimonanthus praecox</i> L.	Wintersweet	51
15	<i>Cinnamomum camphora</i> (L.) J. Presl	Camphortree	64

**Tabela 4 - Estrutura da base de dados Flavia Data Base.**

(conclusão)

INDICE	Nome Científico	Nome Popular (em inglês)	Qtd. Arq./Classe
16	<i>Viburnum awabuki</i> K.Koch	Japan Arrowwood	59
17	<i>Osmanthus fragrans</i> Lour.	Sweet Osmanthus	55
18	<i>Cedrus deodara</i> (Roxb.) G. Don	Deodar	76
19	<i>Ginkgo biloba</i> L.	Ginkgo, Maidenhair Tree	61
20	<i>Lagerstroemia indica</i> (L.) Pers.	Crape myrtle, Crepe myrtle	60
21	<i>Nerium oleander</i> L.	Oleander	65
22	<i>Podocarpus macrophyllus</i> (Thunb.) Sweet	Yew Plum Pine	59
23	<i>Prunus serrulata</i> Lindl. var. <i>lannesiana</i> auct.	Japanese Flowering Cherry	54
24	<i>Ligustrum lucidum</i> Ait. f.	Glossy Privet	54
25	<i>Tonna sinensis</i> M. Roem.	Chinese Toon	64
26	<i>Prunus persica</i> (L.) Batsch	Peach	53
27	<i>Manglietia fordiana</i> Oliv.	Ford Woodlotus	51
28	<i>Acer buergerianum</i> Miq.	Trident Maple	52
29	<i>Mahonia bealei</i> (Fortune) Carr.	Beale's Barberry	54
30	<i>Magnolia grandiflora</i> L.	Southern Magnolia	56
31	<i>Populus xcanadensis</i> Moench	Canadian Poplar	63
32	<i>Liriodendron chinense</i> (Hemsl.) Sarg.	Chinese Tulip Tree	52
33	<i>Citrus reticulata</i> Blanco	Tangerine	55

Fonte: Wu et al., 2007.

Cada um dos arquivos de imagem possui resolução de 1600x1200 pixels e um tamanho médio de 550 KB. A Figura 9 mostra alguns exemplos de imagens extraídas da base de dados.

**Figura 9** - Amostras de imagens de folhas encontradas na base *Flavia Data Base*.



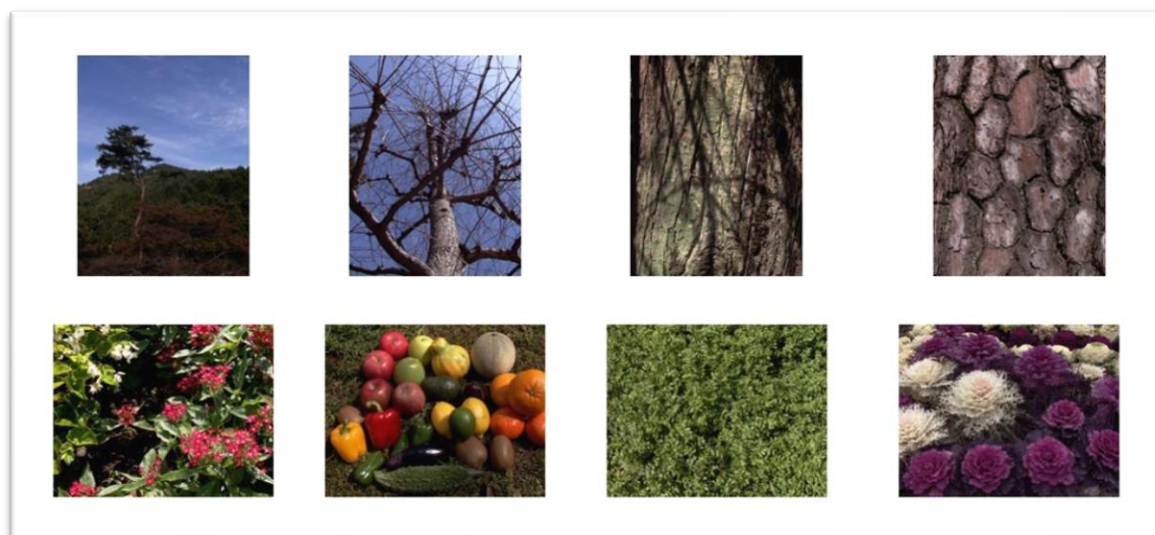
Fonte: O autor.

As imagens da base já se encontram padronizadas, ou seja, suas amostras estão uniformizadas e não possuem ruídos, tornando desnecessária a aplicação de um método de pré-processamento para eliminar objetos indesejados.

#### 4.2.2 Base Kyoto

A base de dados *Kyoto Natural Images* (DOI et al., 2003), essa pertencendo a um domínio diferente do problema alvo, é mais uma das bases utilizadas para treinar os CAENNs. Ela é uma base de dados não rotulada, que contém 62 imagens naturais com resolução de 1000x1280 pixels. As imagens da base foram corrigidas pelos seus criadores, com calibração gama, efeito de tamanho da íris e tempo de exposição. A Figura 10 apresenta alguns exemplos das imagens da base de dados.

**Figura 10** - Amostras de imagens naturais encontradas na base *Kyoto Natural Images*.



Fonte: O autor.

#### 4.2.3 Base Plantclef 2015

A base de dados PlantCLEF 2015, essa pertencendo ao mesmo domínio do problema alvo, é comumente utilizada para fins acadêmicos e competições realizadas pela *ImageCLEF/LifeCLEF* (GOËAU et al., 2015). A base contém 113.204 imagens no total, sendo 91.758 utilizadas para o treino e 21.446 para teste, pertencentes a 41.794 vistas de 1.000 espécies de árvores, ervas e samambaias encontradas em uma região central da europa, próxima à França e outros países vizinhos. A base foi criada com o apoio de diversos colaboradores e cada imagem pertence a apenas uma das categorias em que ela é dividida (planta inteira, fruta, flor, folha, caule, ramo e *scan/scan-like* da folha). A Figura 11, mostra alguns exemplos de imagens extraídas da base de dados em diversos ambientes, em específico as imagens das folhas, as quais foram utilizadas neste trabalho, pertencentes às categorias *scan* e *scan-like*.

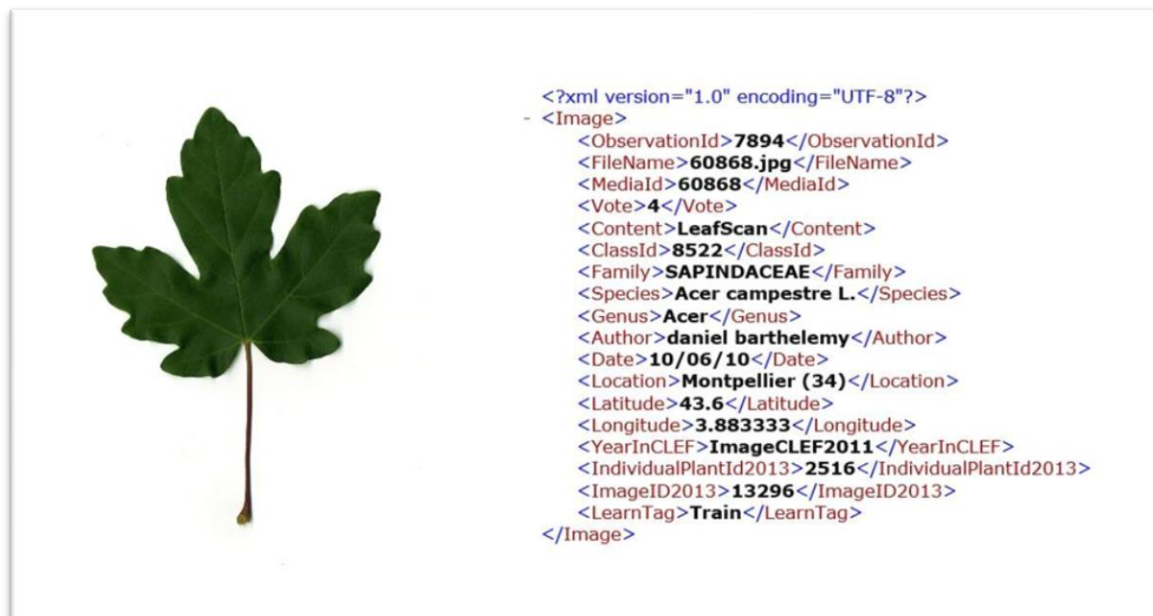
**Figura 11** - Amostras de *scan* e *scan-like* das folhas encontradas na base Plant CLEF 2015.



Fonte: O autor.

Cada imagem da base de dados possui um descritor no formato *Extensible Markup Language* (XML) associado, este possuindo informações sobre a imagem, como a espécie, conteúdo, id, entre outras. A Figura 12, mostra um exemplo de um scan de folha da espécie *Acer campestre L.* e o conteúdo de seu arquivo XML, respectivamente.

**Figura 12** - Exemplo de descritor XML das imagens da base de dados.



Fonte: O autor.

A base de dados *PlantCLEF2015* foi a escolhida para ser uma das bases alvo deste projeto. Posteriormente os vetores de características (profundas), extraídas pelos modelos treinados com as bases *Flavia* (Seção 4.2.1) e *Kyoto* (Seção 4.2.2) serão submetidos à etapa de classificação.

#### 4.2.4 Base LeafSnap

A base de dados *LeafSnap* (KUMAR et al., 2012), essa pertencendo ao mesmo domínio do problema alvo, é composta por imagens de folhas de 185 espécies de plantas coletadas no nordeste dos Estados Unidos. A Figura 13, apresenta exemplos de cada uma das espécies das amostras que compõem a base de dados.

**Figura 13** - Amostras de cada uma das espécies existentes na base de dados *LeafSnap*.

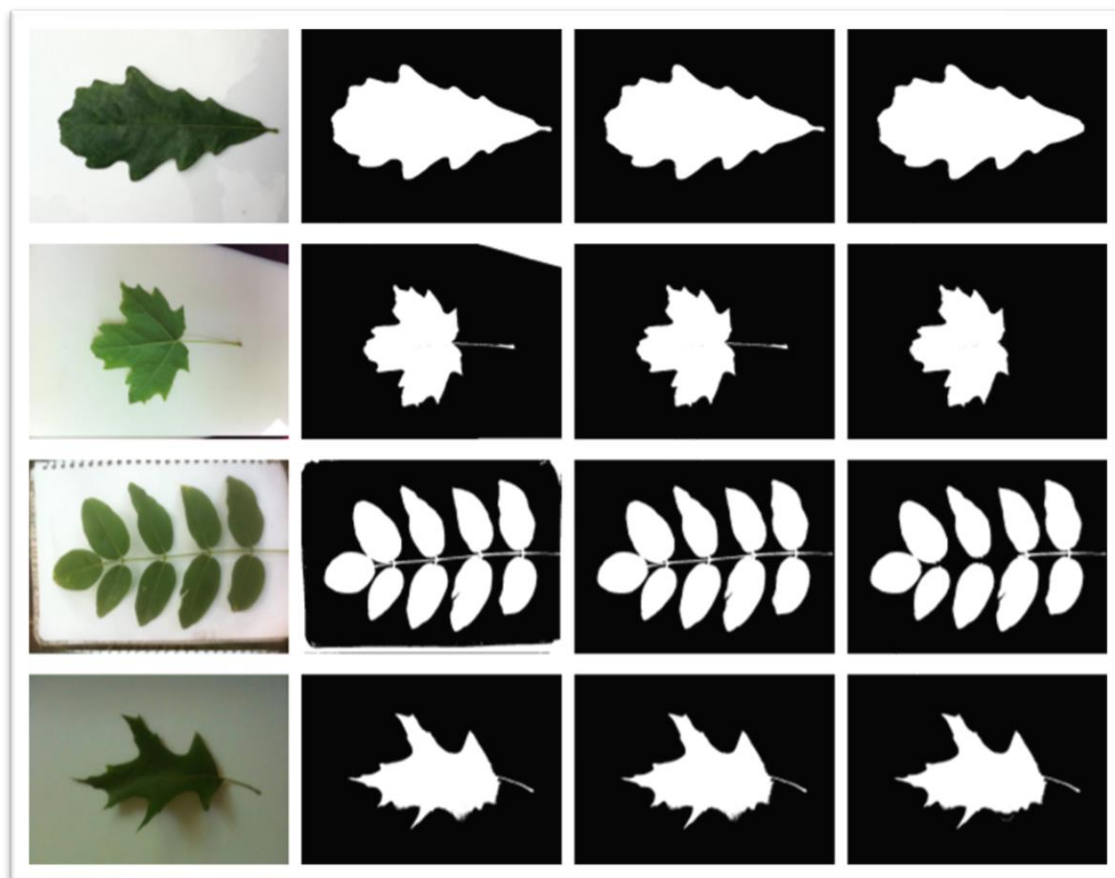


Fonte: Kumar et al., 2012.

A base de dados é dividida em dois grupos, sendo o primeiro composto de imagens coletadas em campo, totalizando 7.719 amostras, e o segundo grupo composto por imagens coletadas em laboratório, contendo 23.147 amostras. As amostras de imagens coletadas, foram utilizadas no sistema automatizado de segmentação desenvolvido pelos autores/criadores da base e suas respectivas segmentações também são disponibilizadas para uso. Alguns exemplos dos resultados da segmentação realizada pelo sistema podem ser vistos na Figura 14.



**Figura 14-** Resultados da segmentação de algumas amostras que fazem parte da base *LeafSnap*.



Fonte: Kumar et al., 2012.

O grupo das imagens de laboratório, não segmentadas, foi escolhido para ser uma das bases alvo deste projeto. Posteriormente os vetores de representação de características foram gerados pelos modelos treinados com as bases citadas nas Seções 4.2.1 e 4.2.2, e foram utilizados para o treinamento dos classificadores. A Figura 15 apresenta alguns exemplos de amostras das imagens utilizadas desta base para a avaliação dos modelos gerados.

**Figura 15 -** Amostras de laboratório da espécie *Betula lenta*.



Fonte: O autor.

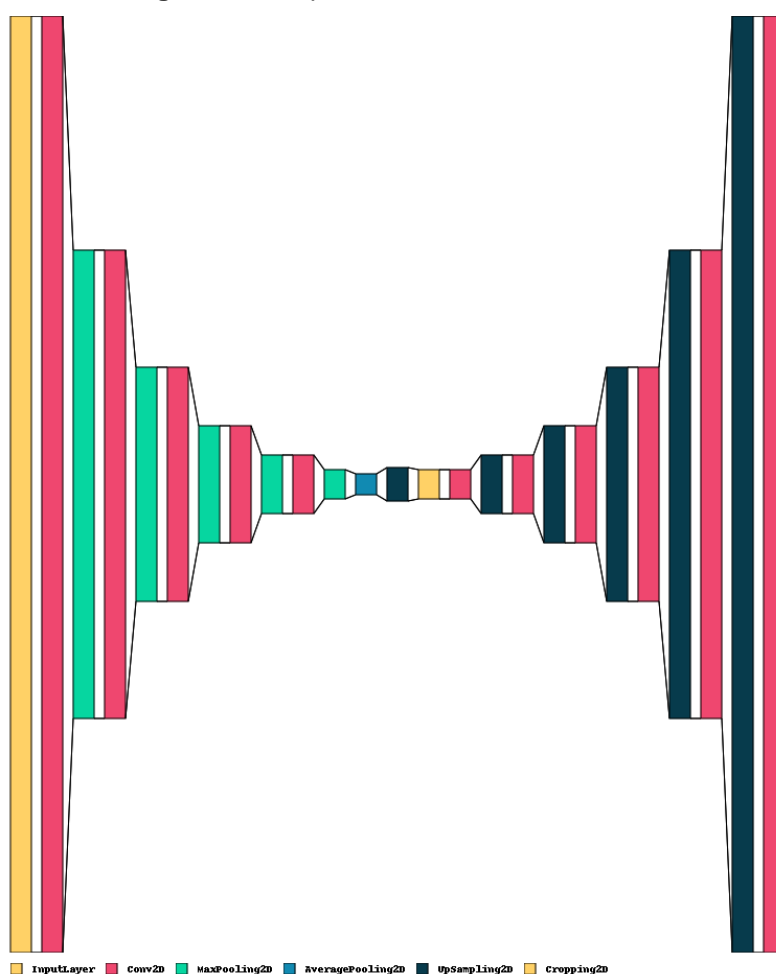
### 4.3 ARQUITETURAS DOS MODELOS

Foram definidas, experimentalmente, duas topologias, as quais são utilizadas para a geração dos modelos extratores de características, utilizando-se dos conceitos e métodos propostos pela aplicação dos CAENN's na confecção de suas estruturas.

#### 4.4 Arquitetura CAENN 25 camadas

A primeira estrutura é composta por uma camada de entrada, onze camadas de convolução, cinco camadas de *max-pooling*, uma camada de *average-pooling*, uma camada de “*cropping*” e seis camadas de *unpooling*. A arquitetura da primeira, onde o somatório do número de camadas é igual a 25, pode ser visualizada na Figura 16.

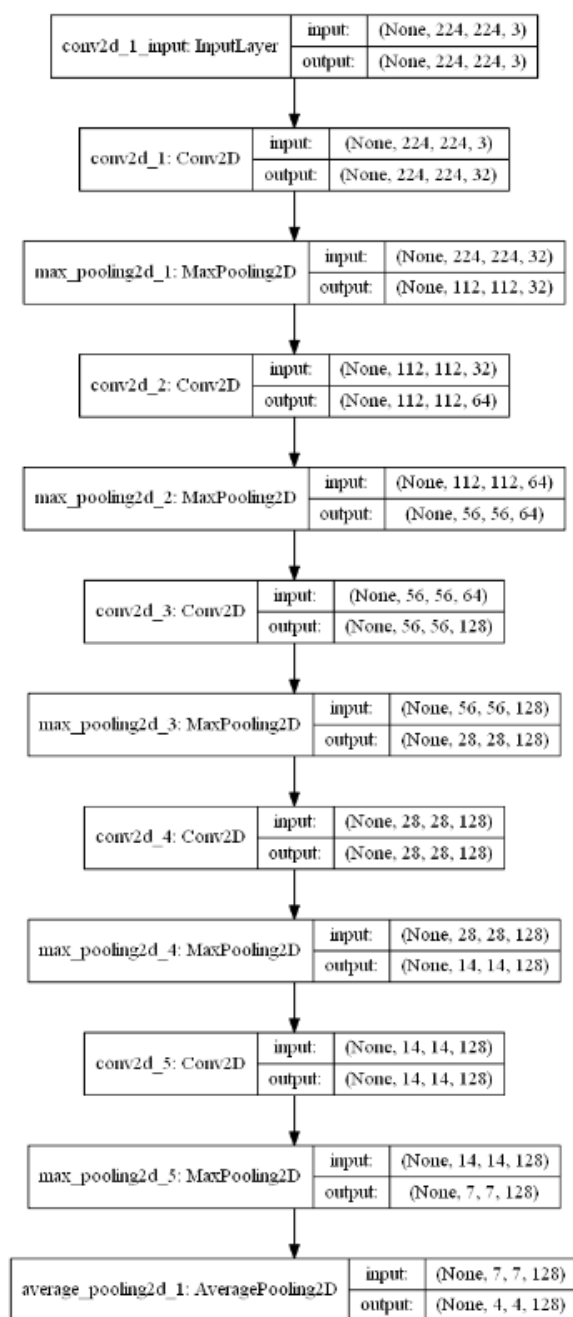
**Figura 16** - Arquitetura CAENN 25 camadas.



Fonte: O autor.

A estrutura codificadora (*encoder*) desta arquitetura, começa na camada de entrada, primeira camada da estrutura completa, e vai até a camada de *average pooling*, décima segunda camada, essa sendo definida como a camada oculta da CAENN. A estrutura de codificação do auto-encoder, apresentando as entradas e saídas de cada uma das camadas é apresentada na Figura 17.

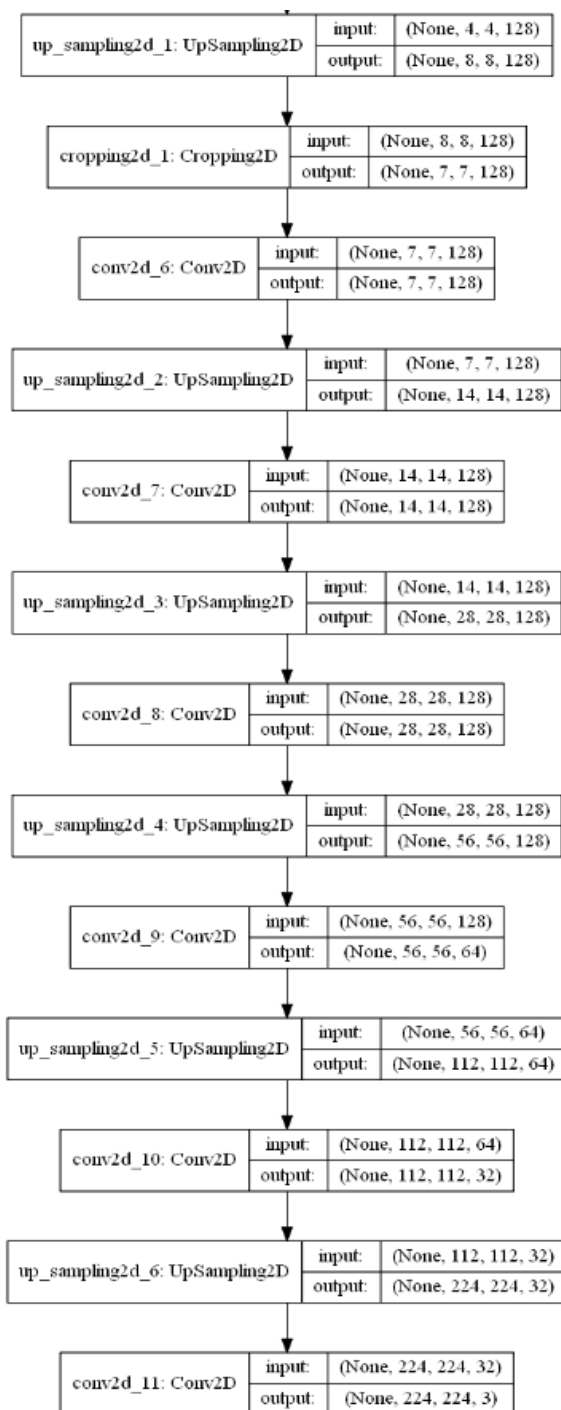
**Figura 17** - Codificador do modelo CAENN com arquitetura de 25 camadas.



Fonte: O autor.

As camadas que se encontram após a camada oculta, que vão da décima terceira até a vigésima quinta camada, fazem parte da estrutura de decodificação desta arquitetura (*decoder*). A estrutura do decodificador da primeira estrutura, apresentando as entradas e saídas de cada uma de suas camadas, é apresentada na Figura 18.

**Figura 18** - Decodificador do modelo CAENN com arquitetura de 25 camadas.

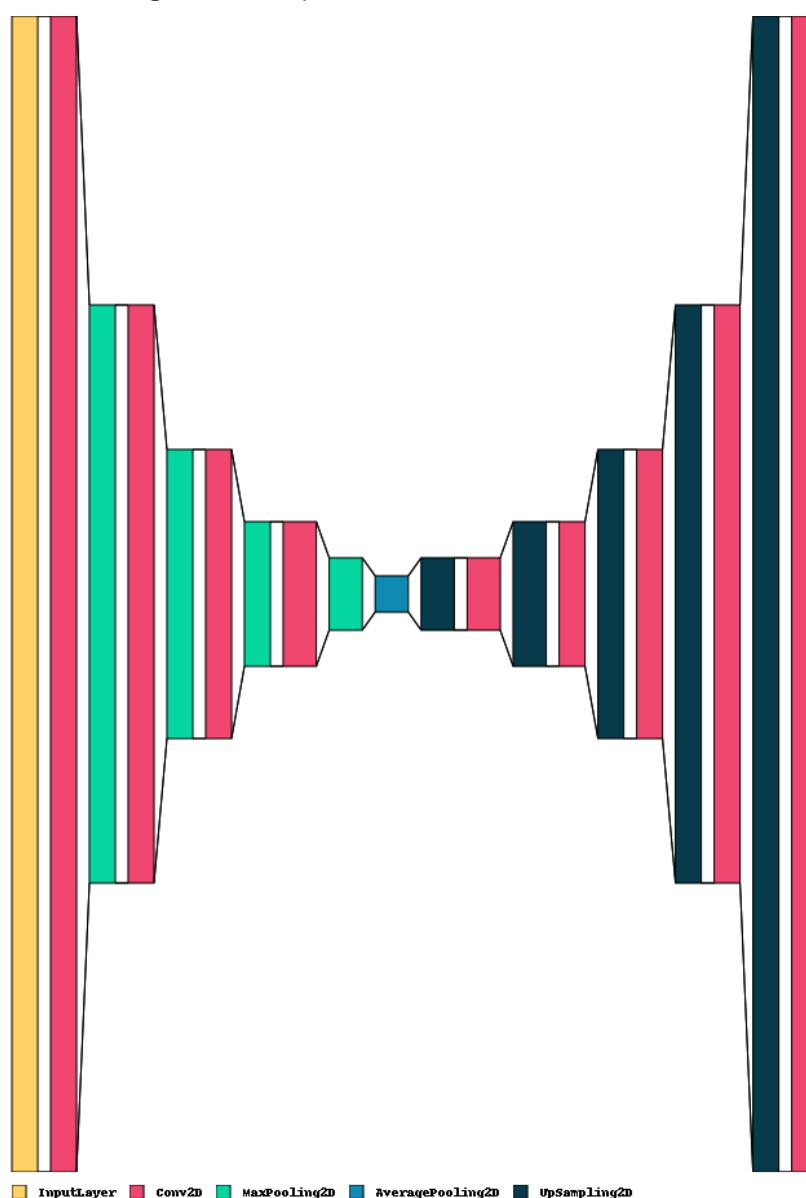


Fonte: O autor.

#### 4.5 Arquitetura CAENN 20 camadas

A segunda estrutura é composta por uma camada de entrada, nove camadas de convolução, quatro camadas de *max-pooling*, uma camada de *average-pooling* e cinco camadas de *unpooling*, totalizando 20 camadas. A arquitetura da segunda estrutura pode ser visualizada na Figura 19.

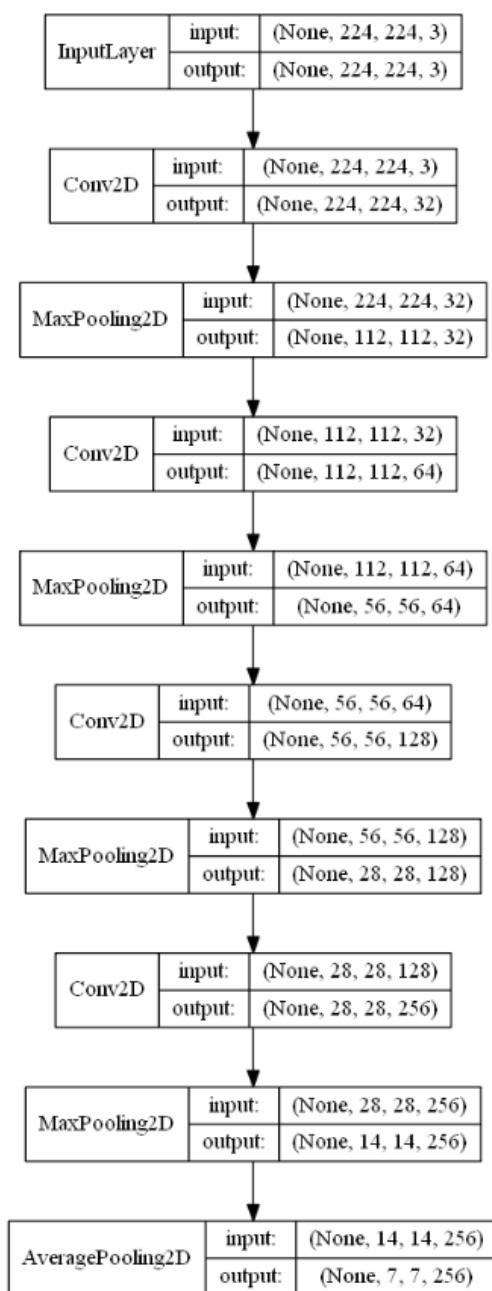
Figura 19 - Arquitetura CAENN 20 camadas.



Fonte: O autor.

A estrutura codificadora (*encoder*) desta arquitetura, começa na camada de entrada, primeira camada da estrutura completa, e vai até a camada de *average-pooling*, décima camada, essa sendo definida como a camada oculta da CAENN. A estrutura de codificação do auto-encoder, apresentando as entradas e saídas de cada uma das camadas é apresentada na Figura 20.

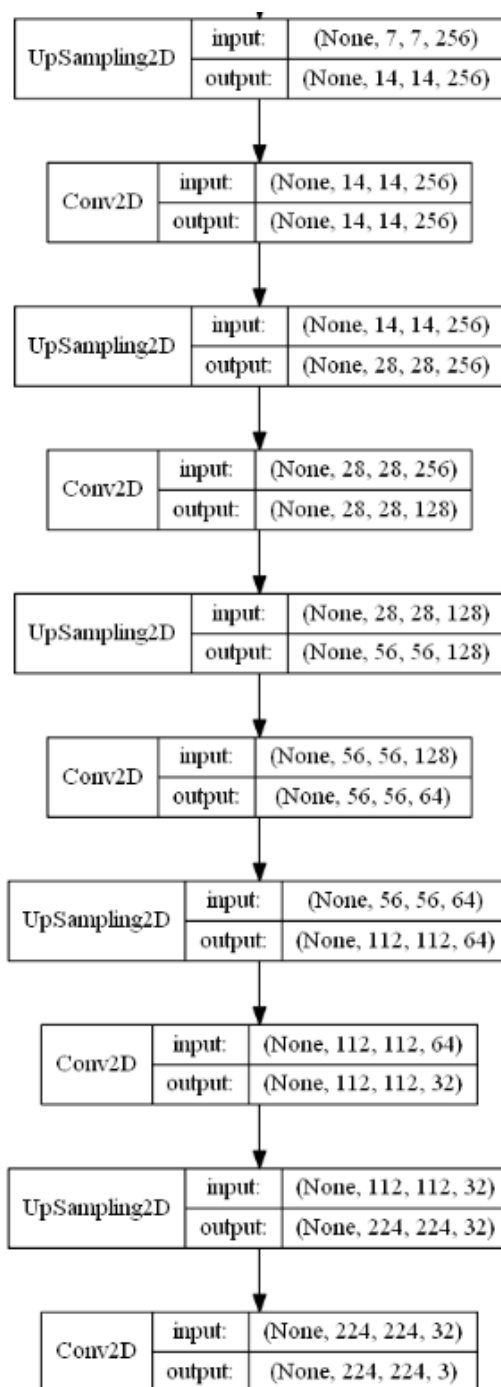
**Figura 20** - Codificador do modelo CAENN com arquitetura de 20 camadas.



Fonte: O autor.

As camadas que se encontram após a camada oculta, que vão da décima primeira até a vigésima camada, fazem parte da estrutura de decodificação desta arquitetura (*decoder*). A estrutura do decodificador da primeira estrutura, apresentando as entradas e saídas de cada uma de suas camadas, é apresentada na Figura 21.

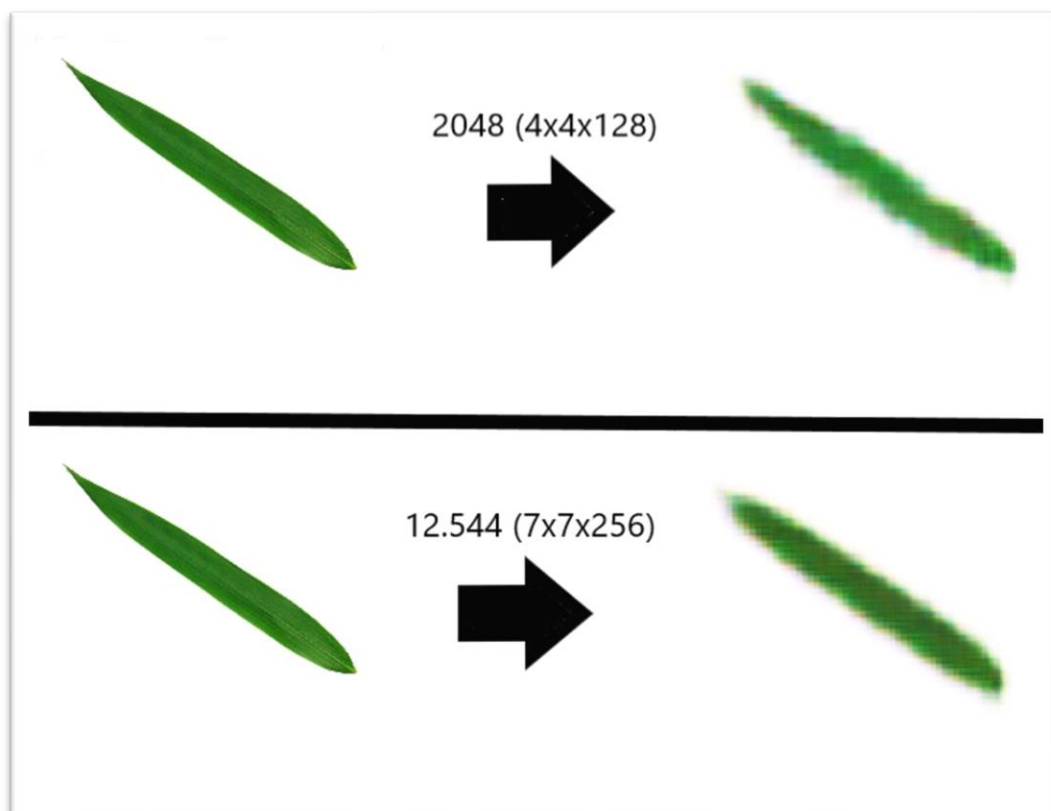
**Figura 21** - Decodificador do modelo CAENN com arquitetura de 20 camadas.



Fonte: O autor.

Ambas as estruturas foram definidas experimentalmente, de acordo com estruturas encontradas no estado da arte recebendo modificações na quantidade de camadas utilizadas, e recebem como entrada as imagens das bases *Flavia Data Base* (Seção 4.2.1) e *Kyoto Natural Images* (Seção 4.2.2). As imagens de ambas as bases são redimensionadas para resolução de  $224 \times 224 \times 3$  e transformadas em vetores de representações de características. A primeira estrutura de codificação consegue gerar vetores de representação com 2.048 ( $4 \times 4 \times 128$ ) características, enquanto a segunda produz vetores com 12.544 ( $7 \times 7 \times 256$ ) características. Na Figura 22 é apresentado um exemplo de imagem sendo reconstruída, utilizando as representações de 2.048 e 12.544 características, respectivamente, passando pelas etapas de codificação e decodificação dos modelos gerados.

**Figura 22** - Reconstruções de uma imagem utilizando uma representação de 2.048 e 12.544.



Fonte: O autor.

As camadas de ambos os CAENN receberam como valores de filtro: [16, 32, 64, 128, 256], número de dimensões (*strides*) igual a 2 para as camadas de *max-pooling*



e 3 para as camadas de convolução, estas fazendo o uso de uma função de ativação do tipo ReLU (*Rectified Linear Unit*). A ordem dos valores do filtro é invertida na hora de executar a decodificação da camada oculta. Posteriormente os modelos foram compilados utilizando um otimizador do tipo RMSprop, este utilizando como parâmetro uma taxa de aprendizado de  $10^{-3}$ , e uma função de perda do tipo MSE (*Mean Squared Error*), sendo treinados utilizando 103 épocas, quantidade definida pelo monitoramento da minimização da perda ('*loss*').

Após o treinamento dos modelos, foram gerados dois arquivos para cada um, o primeiro no formato *JavaScript Object Notation* (.json) que possui metadados como, o nome das camadas e suas configurações, e o segundo no formato *Hierarchical Data Format* (.h5), contendo os pesos para cada um dos modelos gerados por meio das estruturas apresentadas nas Figuras 16 e 19.

Posteriormente os modelos gerados são utilizados para a extração das características (profundas) das imagens da base de dados alvo. Os modelos são carregados separadamente à cada execução e os codificadores são selecionados/carregados, sendo submetidos posteriormente à tarefa de extração de características das imagens das bases alvo.

Após o CAENN extrair as características, são gerados dois arquivos no formato *Comma-separated values* (.csv) para cada base alvo, o primeiro contendo os vetores de representação de características aprendidos (X\_deep.csv) e o segundo os rótulos das imagens (y.csv). Foram utilizados como rótulo de cada amostra de imagem os nomes das espécies encontradas em cada arquivo XML, arquivo de texto ou o próprio nome do arquivo de imagem de acordo com o que é disponibilizado juntamente com as bases utilizadas.

#### 4.6 CLASSIFICAÇÃO

Após as características serem extraídas por meio dos modelos CAENN gerados, os vetores de representações de características são utilizados para realizar o treinamento dos classificadores. Foram utilizados classificadores do tipo SVM, especificamente o SVC (*Support Vector Classification*), que é um classificador próprio para ser aplicado em tarefas de classificação. Os classificadores SVC, receberam como parâmetros: *kernel* não estacionário do tipo linear (Regressão Linear

Bayesiana), regularização igual a 10 e probabilidade igual a “verdadeiro”. Os experimentos foram realizados utilizando uma divisão (*holdout*) de 70% para treinamento e 30% para testes das bases Flavia (Seção 4.2.1), PlantCLEF2015 (Seção 4.2.3) e LeafSnap (Seção 4.2.4).

Posteriormente, os SVM's foram combinados utilizando a regra da soma, pois essa apresentou os melhores resultados durante os experimentos, a qual é apresentada na seção 2.4. Primeiramente foram combinadas as SVM's treinadas com as representações de dimensões diferentes extraídas pelos mesmos auto-encoders convolucionais, separadamente (CAENN1 2.048 + 12.544 e CAENN2 2.048 + 12.544) e, por fim, foi realizada a combinação de todos as SVM's treinadas com ambas as representações gerados pelos dois auto-encoders convolucionais ((CAENN1 2.048 + 12.544) + (CAENN 2.048 + 12.544)).

## 5 RESULTADOS EXPERIMENTAIS

Neste capítulo apresentamos os resultados obtidos nos experimentos realizados, dividindo-os em três seções: Na Seção 5.1 descrevemos o protocolo experimental utilizado neste trabalho. Na Seção 5.2 são apresentados os resultados dos experimentos utilizando o modelo extrator de características, denominado CAENN2, no qual foi utilizada a base Flavia (Seção 4.2.1) para o seu treinamento. Na Seção 5.3 são apresentados os resultados dos experimentos utilizando o modelo extrator de características, denominado CAENN1, nos quais foi utilizada a base Kyoto (Seção 4.2.2) para o seu treinamento. Por fim, na Seção 5.4 são apresentados os resultados das combinações dos classificadores treinados com as diferentes representações de características geradas por cada modelo extrator de características treinado, ou seja, os auto-encoders.

### 5.1 Protocolo Experimental

O protocolo experimental definido neste projeto pode ser apresentado da seguinte maneira:

1º - Primeiramente foi realizada a preparação das imagens das bases de dados, dividindo-as em treino e teste, para o treinamento das CAENNs (Kyoto e Flavia) e posteriormente das bases de dados utilizadas como base alvo nos experimentos (Flavia, Leafsnap e PlantCLEF2015);

2º - Posteriormente, foi definida a topologia a ser utilizada para o treinamento dos modelos extratores de características (CAENN1 e CAENN2) resultando nas arquiteturas apresentadas na seção 4.3, a primeira sendo capaz de extrair representações de 2.048 características e a segunda representações de 12.544 características das bases alvo.

3º - Após a definição das topologias e o desenvolvimento do código, os modelos foram treinados com as bases de dados Kyoto e Flavia, gerando dois modelos com topologias diferentes os quais são utilizados para extrair/gerar as representações de

características de cada base alvo, que posteriormente são salvas em um arquivo no formato .csv.

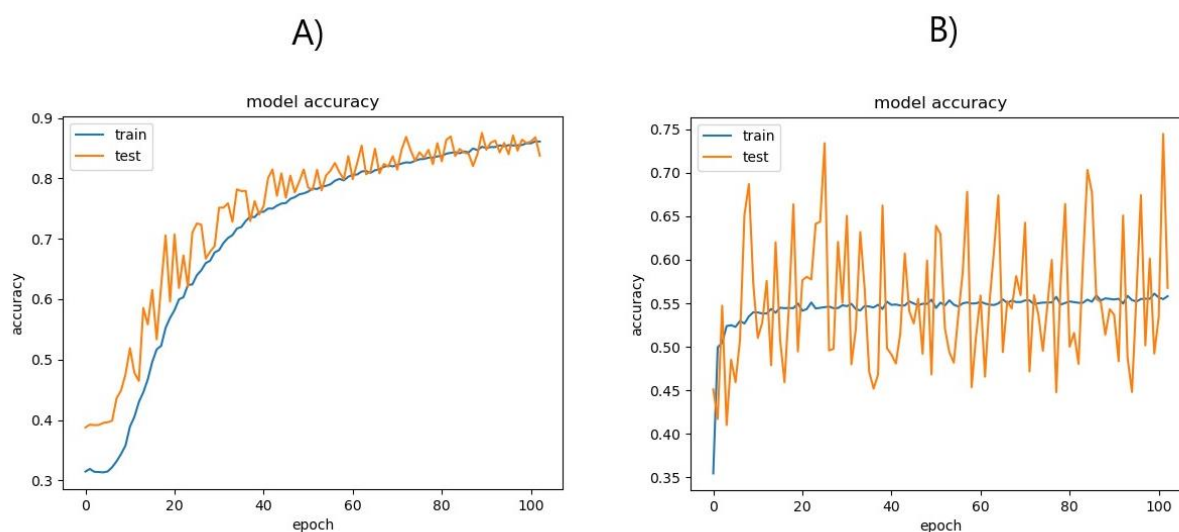
4º - Na tarefa de classificação, as diferentes representações geradas por meio das CAENNs são divididas em treino e teste, e são utilizadas para o treinamento dos indutores/classificadores do tipo SVM, e por meio destes as taxas de acerto são obtidas individualmente por cada classificador, estas sendo apresentadas nas Seções 5.2 e 5.3.

5º - Por fim, é realizada a combinação dos resultados obtidos por cada uma das diferentes representações geradas e as análises estatísticas as quais são apresentados na seção 5.4.

## 5.2 Resultados Modelo CAENN2

Os gráficos apresentados na Figura 23, apresentam as relações entre as taxas de acerto do treinamento e teste do modelo CAENN2, treinado na base Flavia citada na Seção 4.2.2 deste trabalho, utilizando ambas as representações e classificadores descritos na Seção 4.4.

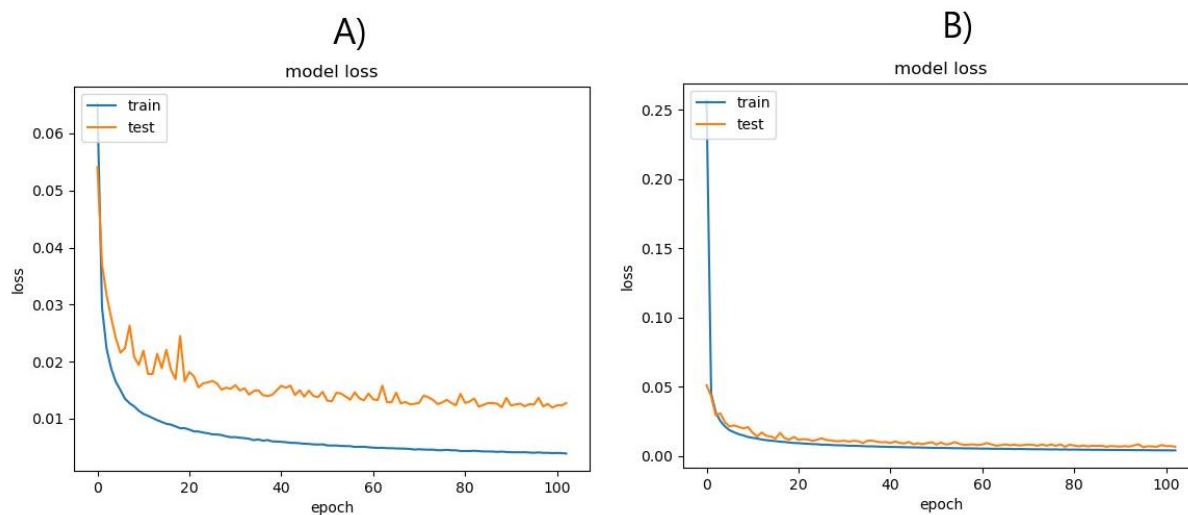
**Figura 23** - Relação entre as taxas de treinamento e teste das arquiteturas do modelo CAENN2: A) Extrator de 2.048 características. B) Extrator de 12.544 características.



Fonte: O autor.

Os gráficos mostrados na Figura 24, apresentam as relações entre as taxas de perda e quantidade de épocas do modelo CAENN2, utilizando ambas as estruturas citadas na Seção 4.4:

**Figura 24** - Relação entre as taxas de perda e a quantidade de épocas das arquiteturas do modelo CAENN2: A) Extrator de 2.048 características. B) Extrator de 12.544 características.



Fonte: O autor.

Na Tabela 5 são apresentadas as taxas de acerto dos classificadores treinados com os vetores de representação de características extraídos pelo modelo CAENN2, este pertencendo ao mesmo domínio das bases alvos (imagens de folhas de plantas).

**Tabela 5** - Resultados individuais dos classificadores SVM treinados com as representações de 2.048 e 12.544 características extraídas - CAENN2.

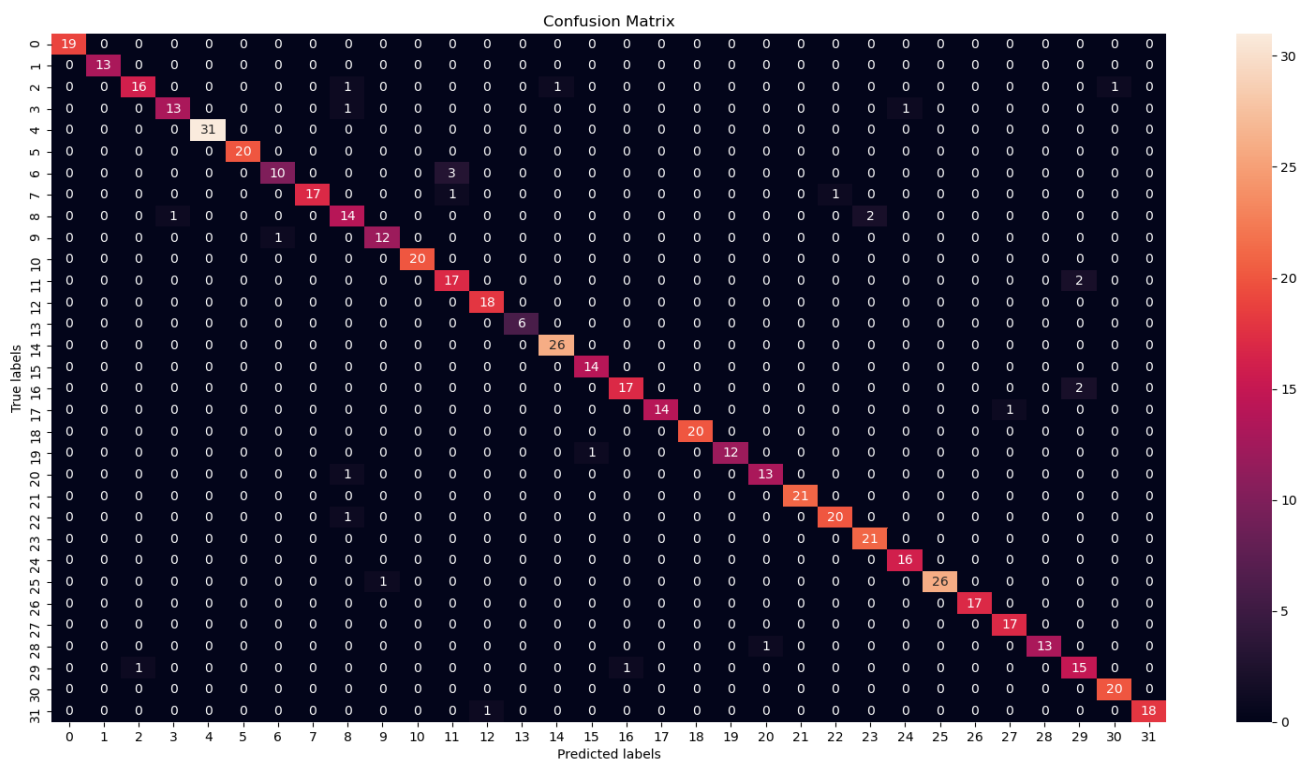
Extrator	Base Alvo	Taxa de acerto (%)
CAENN2_2048	Flavia	92,32
CAENN2_2048	LeafSnap	91,31
CAENN2_2048	PlantCLEF2015	79,41
CAENN2_12544	Flavia	<b>95,28</b>
CAENN2_12544	LeafSnap	89,51
CAENN2_12544	PlantCLEF2015	79,65

Fonte: O autor.

Como pode-se observar na Tabela 5, o melhor resultado obtido para o modelo treinado com uma base pertencente ao mesmo domínio da base alvo (CAENN2\_12544), com uma arquitetura configurada para gerar vetores de características com um tamanho de 12.544, foi utilizando a base de dados Flavia, obtendo como resultado uma acurácia de 95,28 pontos percentuais. Com este resultado, mostra-se que a primeira hipótese levantada neste trabalho, de que representações de características aprendidas por meio da aprendizagem autodidata e pertencentes a um mesmo domínio são competitivas com os resultados apresentando no estado da arte. Vale ressaltar que a base utilizada, tanto para o treinamento do modelo, quanto para o treinamento do classificador com as representações geradas pelo modelo, são as mesmas.

Na Figura 25 apresentamos a matriz de confusão gerada pela SVM treinada utilizando o extrator CAENN2\_12544, apresentando a quantidade de imagens da amostra de teste, que foram classificadas corretamente, assim como, incorretamente.

**Figura 25** - Matriz de confusão apresentando os resultados obtidos pela SVM treinada com as características extraídas pelo extrator CAENN2\_12544.



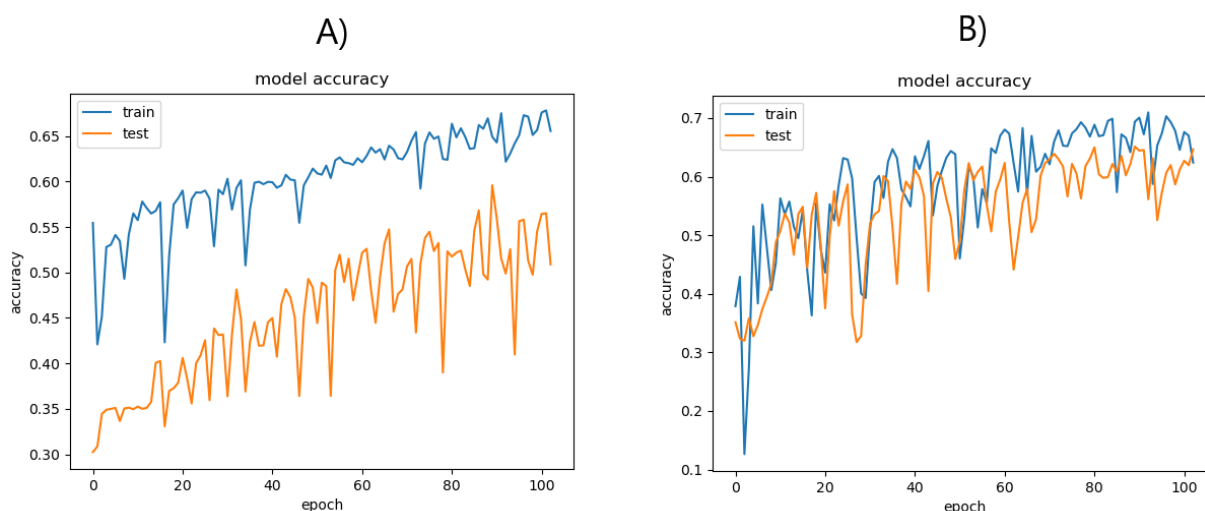
Fonte: O autor.

Como pode-se observar na matriz de confusão (Figura 25), possuímos alguns casos de amostras classificadas incorretamente como, por exemplo, três imagens pertencentes à classe 6 que corresponde à espécie *Indigofera tinctoria L.*, foram classificadas incorretamente, sendo atribuídas à classe 11, correspondente à espécie *Koelreuteria paniculata Laxm.* Pode-se observar também que todas as amostras de teste da classe 4, correspondente à espécie *Berberis anhwaiensis Ahrendt*, foram classificadas corretamente.

### 5.3 Resultados Modelo CAENN1

Os gráficos apresentados na Figura 26, apresentam as relações entre as taxas de acerto do treinamento e teste do modelo CAENN1, treinado na base Kyoto citada na Seção 4.2.1 deste trabalho, utilizando ambas as estruturas descritas na Seção 4.3.

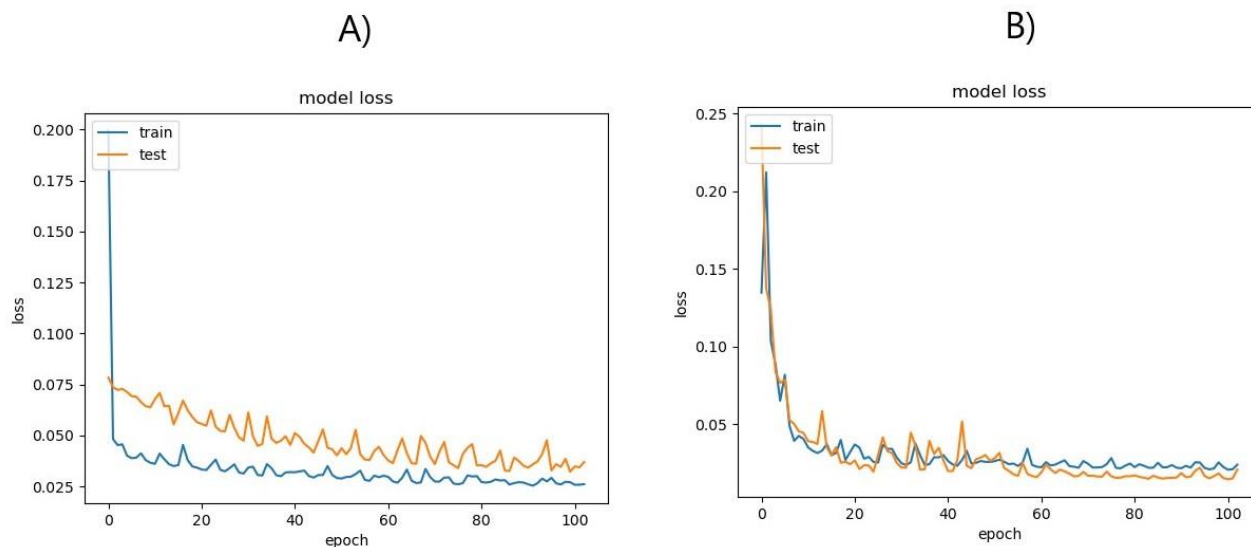
**Figura 26** - Relação entre as taxas de treinamento e teste das arquiteturas do modelo CAENN1: A) Extrator de 2.048 características. B) Extrator de 12.544 características.



Fonte: O autor.

Os gráficos mostrados na Figura 26, apresentam as relações entre as taxas de perda e quantidade de épocas do modelo CAENN1, utilizando ambas as estruturas citadas na Seção 4.4:

**Figura 27** - Relação entre as taxas de perda e a quantidade de épocas das arquiteturas do modelo CAENN1: A) Extrator de 2.048 características. B) Extrator de 12.544 características.



Fonte: O autor.

Na Tabela 6 são apresentadas as taxas de acerto dos classificadores treinados com os vetores de representação de características extraídos pelo modelo CAENN1:

**Tabela 6** - Resultados individuais dos classificadores SVM treinados com as representações de 2.048 e 12.544 características - CAENN1.

Extrator	Base Alvo	Taxa de acerto (%)
CAENN1_2048	Flavia	93,71
CAENN1_2048	LeafSnap	91,92
CAENN1_2048	PlantCLEF2015	78,69
CAENN1_12544	Flavia	93,36
CAENN1_12544	LeafSnap	91,31
CAENN1_12544	PlantCLEF2015	79,01

Fonte: O autor.

Como pode-se observar na Tabela 6, o melhor resultado obtido para o modelo treinado com uma base pertencente a um domínio diferente dos das bases alvos utilizadas neste trabalho (CAENN1\_Kyoto), com uma arquitetura configurada para gerar vetores de características com um tamanho de 2.048, foi utilizando a base de dados Flavia. Como citado anteriormente, a base em que este modelo foi treinado é

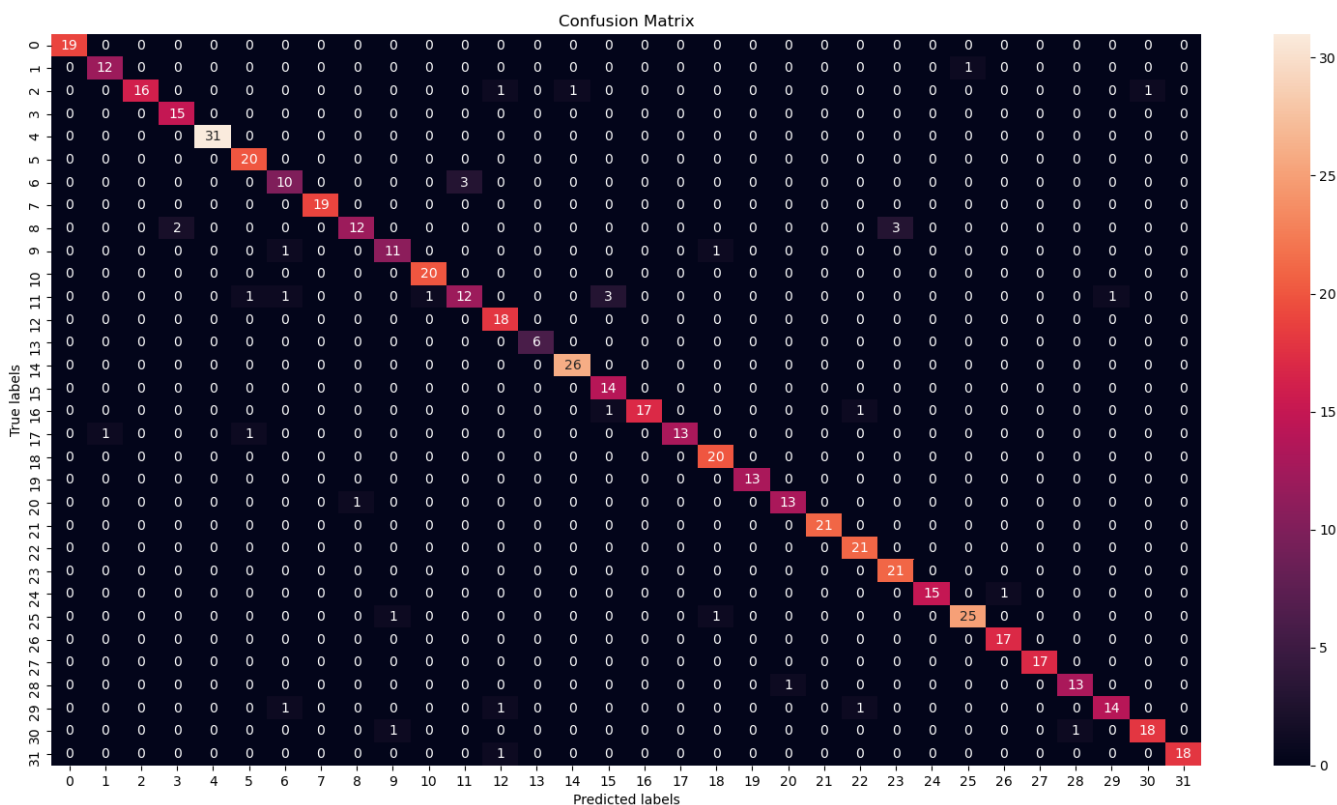


uma base fora do contexto da base alvo, obtendo como resultado uma acurácia de 93,71 pontos percentuais.

Neste experimento, as representações contendo 2.048 características obtiveram melhores taxas de acerto nas bases Flavia (93,71%) e LeafSnap (91,92%). Com uma arquitetura de 12.544, a base PlantCLEF2015, foi a única que apresentou uma pequena melhora na sua acurácia, aumentando de 78,69 para 79,01 pontos percentuais.

Na Figura 28 apresentamos a matriz de confusão gerada pela SVM treinada utilizando o extrator CAENN1\_2048, apresentando a quantidade de imagens da amostra de teste, que foram classificadas corretamente, assim como, incorretamente.

**Figura 28** - Matriz de confusão apresentando os resultados obtidos pela SVM treinada com as características extraídas pelo extrator CAENN1\_2048.



Fonte: O autor.

Como pode-se observar na matriz de confusão (Figura 28), possuímos alguns casos de amostras classificadas incorretamente como, por exemplo, três imagens pertencentes à classe 23 que corresponde à espécie *Phoebe nanmu (Oliv.) Gamble*

foram classificadas incorretamente, sendo atribuídas à classe 8, correspondente à espécie *Podocarpus macrophyllus* (Thunb.) Sweet.

No Apêndice C deste trabalho, pode-se visualizar as matrizes de confusões geradas utilizando a base alvo Leafsnap.

Ao compararmos os resultados obtidos por cada um dos modelos extratores treinados, nota-se que com uma representação de características gerada por um modelo pertencente ao mesmo domínio (CAENN2\_12.544 – Base alvo Flavia) foi alcançado o melhor resultado, este sendo de 95,28 pontos percentuais utilizando representações com um tamanho de 12.544 características. No entanto, com os resultados obtidos por meio dos experimentos utilizando o modelo extrator de características treinado em uma base de um domínio diferente e com uma topologia capaz de extrair representações com dimensões menores e consequentemente gerando um menor custo computacional de processamento (CAENN1\_2048 – Base alvo Flavia), obteve-se um resultado de 93,71 pontos percentuais. Por meio destes experimentos, verifica-se que a segunda hipótese levantada neste trabalho, de que um modelo extrator treinado com dados de fora do domínio da tarefa alvo, mostra-se capaz de atingir bons resultados, estes sendo competitivos se comparados com os apresentados por outros autores citados no estado da arte. No Apêndice A são apresentados resultados obtidos por meio de experimentos realizados utilizando GAP (*Global Average Pooling*) e GMP (*Global Max Pooling*).

#### 5.4 Resultados das Combinações dos Classificadores

Nesta seção são apresentados os resultados das combinações das predições dos classificadores treinados com as diferentes representações de características geradas por meio de cada arquitetura. É importante salientar que, embora tenham sido avaliados diferentes tipos de combinação, somente os resultados obtidos com a regra da soma são apresentados, por apresentarem as melhores acurácias, assim como, os experimentos foram realizados apenas nas bases Flavia e Leafsnap, por estas mostrarem os melhores resultados da técnica de combinação. A base PlantCLEF 2015 foi desconsiderada devido à dificuldade de trabalhar com as amostras disponibilizadas por ela, por conta do desbalanceamento existente em suas amostras.

Na Tabela 7, são apresentados os resultados das combinações utilizando a regra da soma, onde os valores máximos das predições de cada espécie, obtidas por cada classificador treinado com as diferentes representações de características, são selecionados para definir o resultado final da classificação das espécies:

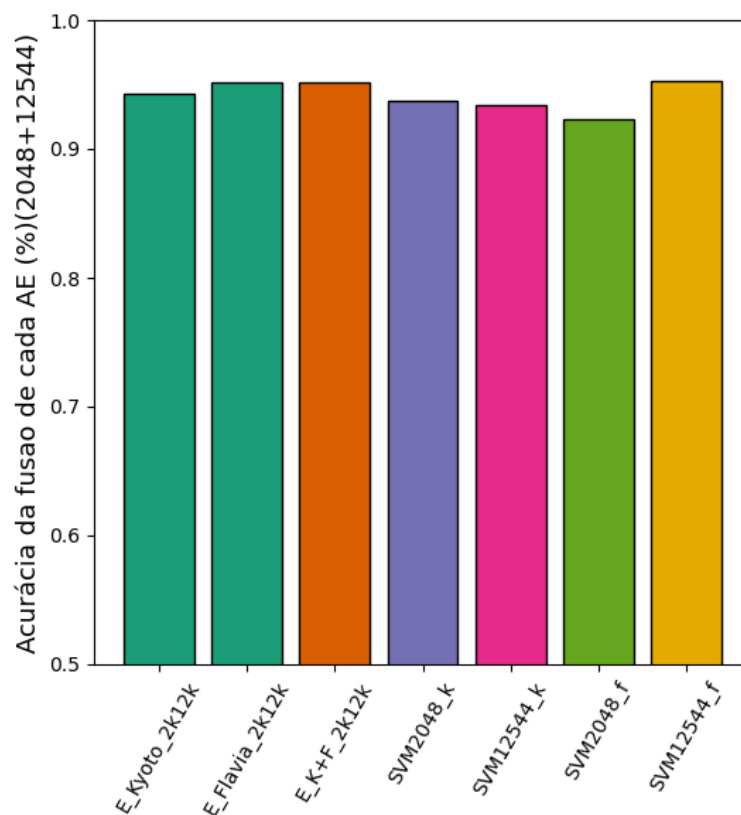
**Tabela 7** - Resultados das combinações dos resultados dos classificadores SVM utilizando a base de dados Flavia e a regra da soma.

CAENN	Base Alvo	Taxa de acerto (%)
Kyoto (2.048+12.544)	Flavia	94,58
Flavia (2.048+12.544)	Flavia	94,76
Kyoto + Flavia (2.048+12.544)	Flavia	95,11

Fonte: O autor.

Na Figura 29 é apresentado um gráfico em barras mostrando-nos um comparativo dos resultados obtidos com as SVM e os *ensembles* utilizando a base de dados alvo Flavia.

**Figura 29** – Gráfico em barras representando os resultados obtidos com as SVM's e os ensembles – Base Alvo Flavia.



Fonte: O autor.

Como pode-se observar na Tabela 7, a combinação das diferentes representações geradas pelos dois modelos (CAENN1 + CAENN2), por meio da regra da soma, alcançou o melhor resultado com a base alvo Flavia, alcançando uma taxa de acerto de 95,46 pontos percentuais. Nota-se que a combinação dos vetores de tamanhos diferentes obtidos pelos modelos extratores de características treinados com a base Flávia, alcançaram um resultado de 95,28 pontos percentuais, com uma diferença de apenas 0,18 pontos percentuais da abordagem combinando todas as diferentes representações de cada CAENN.

Na Tabela 8, são apresentados os resultados das combinações utilizando a regra da soma e, como base alvo, a base de dados *Leafsnap*, onde os valores máximos das predições de cada espécie, obtidas por cada classificador treinado com as diferentes representações de características, são selecionados para definir o resultado final da classificação das espécies:

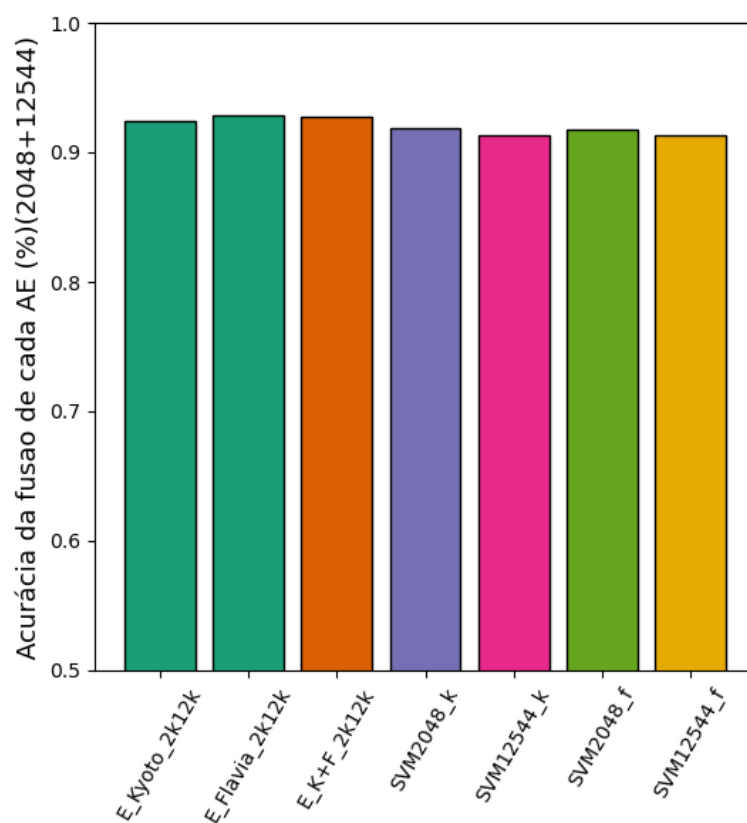
**Tabela 8** - Resultados das combinações dos resultados dos classificadores SVM utilizando a base de dados Leafsnap e a regra da soma.

CAENN	Base Alvo	Taxa de acerto (%)
Kyoto (2.048+12.544)	Leafsnap	92,33
Flavia (2.048+12.544)	Leafsnap	92,66
Kyoto + Flavia (2.048+12.544)	Leafsnap	92,88

Fonte: O autor.

Na Figura 30 é apresentado um gráfico em barras mostrando-nos um comparativo dos resultados obtidos com as SVM e os *ensembles* utilizando a base de dados alvo Leafsnap.

**Figura 30** - Gráfico em barras representando os resultados obtidos com as SVM's e os ensembles – Base Alvo Leafsnap.

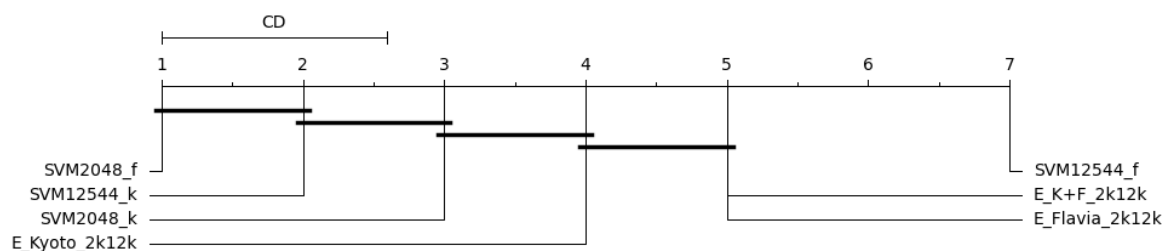


Fonte: O autor.

Novamente a combinação de todas as representações de características, extraídas pelos dois modelos, forneceu o melhor resultado com a base alvo *Leafsnap*, alcançando uma taxa de acerto de 92,88%. Nota-se que as fusões dos vetores de tamanhos diferentes de cada modelo (CAENN1 2.048 + 12.544 e CAENN2 2.048 + 12.544), alcançaram 92,33 e 92,66 pontos percentuais, respectivamente. No Apêndice B são apresentados os resultados obtidos pelo uso da métrica *F1 score* durante os experimentos realizados com a combinação de classificadores

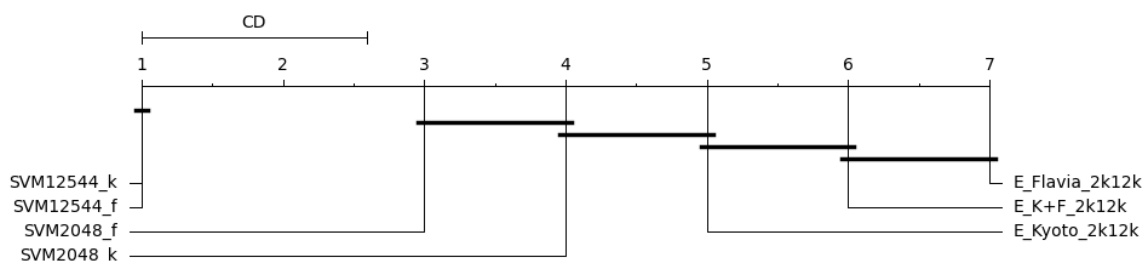
Na Figura 31 e 32 são apresentadas comparações estatísticas geradas utilizando o teste de Nemenyi (DEMŠAR, J., 2006) para a base de dados Flavia e Leafsnap. Este teste nos permite realizar uma análise das distâncias críticas (CD) de cada um dos resultados obtidos pelos classificadores, assim como, de suas combinações, mostrando se existe uma diferença estatística relevante entre eles ou se são considerados estatisticamente iguais. O valor do coeficiente CD é igual a 1,59 e por meio dele o gráfico nos mostra que se a diferença entre os resultados obtidos pelos classificadores for inferior ao valor do coeficiente, eles são considerados estatisticamente iguais.

**Figura 31** - Gráfico do teste de Nemenyi apresentando a distância crítica entre os classificadores treinados com as representações extraídas da base Flavia.



Fonte: O autor.

**Figura 32** - Gráfico do teste de Nemenyi apresentando a distância crítica entre os classificadores treinados com as representações extraídas da base Leafsnap.



Fonte: O autor.

Como pode-se observar nos gráficos (Figuras 31 e 32), quase todos os classificadores, apresentados em ordem crescente, possuem uma significância igual a de algum outro como, por exemplo, as SVM's (SVM2048\_f e SVM12544\_k) treinadas com as representações individuais de 2.048 e 12.544, extraídas pelos CAENN's treinados na base Flavia e Kyoto, exceto a SVM treinada com a representação de 12.544 da base Flavia na Figura 29 (SVM12544\_k). Também pode-se observar que a distância crítica entre a SVM2048\_f e a combinação de todas as representações E\_K+F\_2k12K são significativamente diferentes, estatisticamente, em ambos os gráficos.

Para finalizar esta seção e a discussão sobre combinação de classificadores, pode-se concluir que os testes com combinação de classificadores geram modelos com taxas de acerto superiores ou próximas aos melhores classificadores individuais, apresentados nas seções 5.2 e 5.3 deste trabalho. Os resultados obtidos comprovam que as técnicas de combinação podem aumentar os resultados ou até mesmo diminuir a possibilidade de se escolher um modelo ineficiente para classificar as amostras das plantas.

## **6 COMPARAÇÃO DOS RESULTADOS OBTIDOS**

A fim de prover uma análise comparativa do método proposto com trabalhos de outros autores, o método foi posto à prova com diversas metodologias que utilizaram as mesmas distribuições de base de dados. A Tabela 9 é uma versão reduzida da tabela da seção 3.1, pois apresenta apenas os trabalhos que utilizaram as mesmas bases de dados, especificamente as bases alvo, utilizadas neste trabalho.

Tabela 9 - Comparação dos resultados com outros autores.

Autores	Metodologia Empregada	Extrator de Características	Método de Classificação	Base de Testes	Melhores TA (%)		
(Caglayan et al., 2013)	A utilização da extração de características das cores e formatos das imagens de folhas.	Grupos de características SF e CF.	KNN, SVM, RF e NB	Flavia Data Base	96,32%		
(Kumar et al., 2016)	A utilização das características extraídas das bordas das imagens das folhas, usando características baseados nas arestas e no formato das folhas.	Espessura, arredondamento, dispersão, <i>Edge Filling Length</i> (EFL), contagem de ramais e contagem de <i>loops</i> .	KNN	Flavia Data Base	94,37%		
(Ramos et al., 2019)	Classificação de espécies de plantas utilizando CAE.	Vetores de características profundas aprendidas pelo CAE.	SVM	Flavia Data Base	94,74%		
(Wang et al., 2019)	Classificação de espécies de plantas utilizando redes neurais siamesas.	Vetores de características profundas aprendidas CNN.	SVM	Flavia e LeafSnap	95,32% e 91,75%, respectivamente.		
(Turkoglu et al., 2021)	Sistema de reconhecimento de plantas baseado em Rede Neural Convolutacional Multi-Divisional (MD-CNN) abordando a classificação de espécies de plantas.	Vetores de características profundas extraídas pela MD-CNN.	SVM	Flavia e LeafSnap	100% e 97,8%, respectivamente.		
(Araújo et al., 2018)	Classificação hierárquica de gêneros e espécies de plantas fazendo o uso de classificação de objetos refinados e fusão de modelos de aprendizagem profunda.	Características profundas extraídas pelas CNNs das imagens inteiras e fragmentadas.	CNN	LifeCLEF2015	86,44%		
(ZHU et al., 2018)	Método de aprendizagem profunda de extração de características de multicomponentes.	Vetores de características profundas aprendidas pelas camadas da CNN.	SVM	LifeCLEF2015	<i>Feature1</i> : 88,50% (flor)	<i>Feature2</i> : 90,20% (fruta)	
Método Proposto	Método de aprendizagem profunda de extração de características utilizando CAENN e aprendizagem autodidata.	Diferentes representações de características aprendidas pelos modelos extratores.	SVM	Flavia, PlantCLEF2015 e LeafSnap	Flavia (CAENN2-12544): 95,28%	PlantCLEF 2015: (CAENN2-12544): 79,65%	Leafsnap (Ensemble): 92,88%

Fonte: O autor.



Os cinco primeiros trabalhos apresentados na Tabela 9 utilizam a base de dados Flavia. O melhor resultado apresentado na tabela pertence ao trabalho de Turkoglu et al. (2021) alcançando uma taxa de acerto de 100 pontos percentuais, utilizando representações de características obtidas por meio de uma Rede Neural Convolucional Multi-Divisional (MD-CNN). O segundo melhor resultado fica para o trabalho de Caglayan et al. (2013), o qual atinge uma taxa de acerto de 96,32 pontos percentuais, utilizando características do formato e cor das folhas das plantas. O terceiro melhor resultado pertence ao trabalho de Wang et al. (2019), atingindo a taxa de acerto de 95,33 pontos percentuais, sendo este um valor bem próximo ao atingido pelo modelo CAENN2 com a arquitetura extratora de 12.544 características (95,28%). A metodologia proposta neste trabalho obteve resultados superiores, utilizando o modelo CAENN2 para extrair as representações de 12.544 características, assim como, com a combinação dos classificadores (95,11%), aos apresentados no trabalho de Ramos et al. (2019) (94,74%) e Kumar et al. (2016) (94,37%), sendo que o que o primeiro utiliza uma metodologia semelhante à utilizada neste trabalho, diferindo em sua arquitetura, a qual trabalha com vetores de características de tamanhos maiores, que consequentemente impactam no custo computacional, aumentando o tempo para realizar o treinamento dos auto-encoders, assim como dos classificadores utilizados para a identificação das espécies das plantas.

Nos trabalhos de Wang et. al (2019) e Turkoglu et al. (2021), também foi utilizada a base Leafsnap, obtendo taxas de acerto de 91,75 e 97,80 pontos percentuais, respectivamente. Os resultados alcançados neste trabalho com esta base, mostram-se superiores ao alcançado por Wang et. al (2019), utilizando a CAENN1 para extrair representações contendo 2.048 características (91,92%), assim como, todas as taxas de acerto utilizando a combinação por meio da regra da soma, apresentadas na Tabela 8. Com este experimento pode-se comprovar a viabilidade do uso de bases fora do contexto para o treinamento de modelos extratores de características, obtendo resultados competitivos e até mesmo superiores aos apresentados no estado da arte.

Os dois últimos trabalhos apresentados na Tabela 9, apresentam os resultados de Araújo et al. (2018) (86,44%) e Zhu et al. (2018) (90,20%), fazendo uso da base de dados PlantCLEF2015. Ambos os autores conseguiram resultados superiores aos atingidos pela metodologia empregada neste trabalho, porém suas metodologias são diferentes, sendo que o primeiro utilizou aumento dos dados e o segundo obteve o

maior resultado utilizando as imagens dos frutos das plantas. Alguns testes utilizando os vetores de características das imagens da base PlantCLEF2015 foram conduzidos, porém os resultados apresentados não corresponderam com o desejado, mesmo assim, mostram-se competitivos se compararmos o melhor resultado obtido com o modelo CAENN2 com arquitetura de 12.544 características, que foi de 79,65 pontos percentuais.

## 7 CONCLUSÃO

Neste trabalho apresentamos uma abordagem para a classificação de espécies de plantas utilizando métodos de aprendizagem não-supervisionada e autodidata, por meio do auto-encoder (AE), especificamente o auto-encoder convolucional (CAE). Esta abordagem nos permite trabalhar com bases de dados não rotuladas, pois estas podem ser encontradas com maior facilidade, tendo em vista a baixa quantidade de base de dados rotuladas de espécies de plantas disponíveis para a criação de modelos supervisionados.

Para a execução dos experimentos foram utilizadas quatro bases de dados. As bases de dados *Flavia* (Seção 4.2.1) e *Kyoto* (Seção 4.2.2) foram selecionadas para o treinamento dos modelos extratores de características (CAENN1 e CAENN2) de forma não supervisionada, utilizando duas arquiteturas diferentes, sendo a primeira capaz de gerar vetores de representação de 2.048 características e a segunda vetores de representação de 12.544 características. As bases de dados *Flavia*, *Leafsnap* e *PlantCLEF2015* foram selecionadas para serem as bases alvo, estas sendo submetidas às diferentes arquiteturas de cada modelo CAENN para a extração das representações de características, as quais posteriormente foram utilizadas para o treinamento dos classificadores.

Os resultados obtidos mostram-se competitivos com os encontrados no estado da arte e, em alguns casos, chegam a ser superiores. Os experimentos utilizando os modelos extratores atingiram resultados satisfatórios em todas as bases alvo, permitindo-nos comprovar a viabilidade do uso de bases pertencentes e não pertencentes ao mesmo domínio das bases alvo, para o treinamento de modelos extratores de características por meio da aprendizagem autodidata. Com a análise estatística empregada pelo teste de Nemenyi, foi possível identificar que os classificadores treinados com representações menores possuem um desempenho considerado estatisticamente semelhante aos treinados com representações maiores, conseqüentemente exigindo um menor custo computacional ao treiná-los e apresentando resultados satisfatórios.

A metodologia utilizada neste trabalho difere de muitas apresentadas no estado da arte, sendo que o trabalho que mais se assemelha com este é o de Ramos et al.

(2019), este tendo seu resultado ultrapassado pelo modelo CAENN2, extrator de 12.544 características, que conseqüentemente atingiu a mais alta taxa de acerto deste trabalho, como pode ser visto na Seção 5.2. Alguns autores utilizaram modelos pré-treinados e, em muitos casos, as mesmas bases de dados para treinamento e testes, o que se assemelha aos experimentos conduzidos utilizando a base Flavia, estes utilizando-a tanto para o treinamento do CAENN, quanto para o treinamento dos classificadores SVM com as representações de características extraídas da mesma.

Como trabalhos futuros pretende-se explorar o desempenho de outras arquiteturas de auto-encoders, capazes de gerar representações de características com dimensões diferentes das apresentadas neste trabalho, assim como combiná-las para criar um modelo robusto.

## REFERÊNCIAS

ARAÚJO, Voncarlos M. **Identificação de espécies de plantas utilizando combinação de classificadores**. 2016. Dissertação (Mestrado em Computação Aplicada) Universidade Estadual de Ponta Grossa, 2016.

ARAÚJO, V. et al. Fine-grained Hierarchical Classification of Plant Leaf Images using Fusion of Deep Models. *In: IEEE INTERNATIONAL CONFERENCE ON TOOLS WITH ARTIFICIAL INTELLIGENCE*, 2018. Volos: ICTAI, 2018. p 1-5. Disponível em: <https://ieeexplore.ieee.org/document/8576009>. Acesso em: 12 abr. 2020.

BAO, Yuequan et al. Computer vision and deep learning–based data anomaly detection method for structural health monitoring. **Structural Health Monitoring**, v. 18, n. 2, p. 401-421, 2019.

CAGLAYAN, Ali; GUCLU, Oguzhan; CAN, Ahmet Burak. A plant recognition approach using shape and color features in leaf images. *In: INTERNATIONAL CONFERENCE ON IMAGE ANALYSIS AND PROCESSING*, 2013. Berlin: ICIAP, 2013. p. 161-170. Disponível em: [https://link.springer.com/chapter/10.1007/978-3-642-41184-7\\_17](https://link.springer.com/chapter/10.1007/978-3-642-41184-7_17). Acesso em: 20 abr. 2020.

CELEBI, M. Emre; AYDIN, Kemal (Ed.). **Unsupervised learning algorithms**. Berlin: Springer International Publishing, 2016.

CHAPELLE, Olivier; SCHOLKOPF, Bernhard; ZIEN, Alexander. Semi-supervised learning (chappelle, o. et al., eds.; 2006) [book reviews]. **IEEE Transactions on Neural Networks**, v. 20, n. 3, p. 542-542, 2009.

CHEN, Junde et al. Using deep transfer learning for image-based plant disease identification. **Computers and Electronics in Agriculture**, v. 173, p. 105393, 2020.

DEMŠAR, Janez. Statistical comparisons of classifiers over multiple data sets. **The Journal of Machine Learning Research**, v. 7, p. 1-30, 2006.

DENG, Jia et al. Imagenet: A large-scale hierarchical image database. *In: IEEE conference on computer vision and pattern recognition*, 2009. MIAMI: IEE, 2009. p 1-5. Disponível em: <https://ieeexplore.ieee.org/document/5206848>. Acesso em: 20 abr. 2020.

DOI, Eizaburo et al. Spatiochromatic receptive field properties derived from information-theoretic analyses of cone mosaic responses to natural scenes. **Neural computation**, v. 15, n. 2, p. 397-417, 2003.

DUVENAUD, David. **Automatic model construction with Gaussian processes**. 2014. Tese de Doutorado (Doutorado em Filosofia). Filosofia, University of Cambridge, Cambridge, 2014.

FAYYAZ, Mohsen et al. A novel approach for Finger Vein verification based on self-taught learning. *In: Iranian Conference on Machine Vision and Image Processing (MVIP)*, 9., 2015. Tehran: IEEE, 2015. p. 88-91. Disponível em: <https://ieeexplore.ieee.org/document/7397511>. Acesso em: 02 jun. 2020.

FIGUEIREDO, Mário AT. Adaptive sparseness for supervised learning. **IEEE transactions on pattern analysis and machine intelligence**, v. 25, n. 9, p. 1150-1159, 2003.

FUENTES, Sigfredo et al. Automated grapevine cultivar classification based on machine learning using leaf morpho-colorimetry, fractal dimension and near-infrared spectroscopy parameters. **Computers and Electronics in Agriculture**, v. 151, p. 311-318, 2018.

GOËAU, Hervé et al. Lifeclef plant identification task. *In: PlantCLEF2015*, 2015. Toulouse: CLEF, 2015. Disponível em: <https://www.imageclef.org/lifeclef/2015/plant>. Acesso em: 02 mar. 2019.

GOËAU, Hervé et al. Deep learning for plant identification: how the web can compete with human experts. **Biodiversity Information Science and Standards**, v. 2, p. e25637, 2018.

GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Aaron. **Deep learning**. MIT press, 2016.

HASTIE, Trevor; TIBSHIRANI, Robert; FRIEDMAN, Jerome. Unsupervised learning. *In: THE ELEMENTS OF STATISTICAL LEARNING*. New York: Springer, 2009. p. 485-585. Disponível em: <https://link.springer.com/book/10.1007/978-0-387-84858-7>. Acesso em: 15 dez. 2020.

HUANG, Faming et al. A deep learning algorithm using a fully connected sparse autoencoder neural network for landslide susceptibility prediction. **Landslides**, v. 17, n. 1, p. 217-229, 2020.

KHAMPARIA, Aditya et al. Seasonal crops disease prediction and classification using deep convolutional encoder network. **Circuits, Systems, and Signal Processing**, v. 39, n. 2, p. 818-836, 2020.

KOTSIANTIS, Sotiris B. et al. Supervised machine learning: A review of classification techniques. **Emerging artificial intelligence applications in computer engineering**, v. 160, n. 1, p. 3-24, 2007.

KITTLER, J. et al. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998.

KUMAR, Pullela SVVSR et al. Leaf classification based on Shape and Edge feature with k-NN Classifier. *In: INTERNATIONAL CONFERENCE ON CONTEMPORARY COMPUTING AND INFORMATICS*, 2, 2016. Greater Noida: IC3I, 2016. p. 548-552. Disponível em: <https://ieeexplore.ieee.org/document/7918024>. Acesso em: 15 jul. 2020.

KUNCHEVA, L. I. *Combining Pattern Classifiers: Methods and Algorithms*. 2004.

LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. **nature**, v. 521, n. 7553, p. 436-444, 2015.

LIU, Yun et al. Richer convolutional features for edge detection. In: **PROCEEDINGS OF THE IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION**, 2017. Honolulu: IEEE, 2017. p. 3000-3009. Disponível em: <https://ieeexplore.ieee.org/document/8100105>. Acesso em: 15 jul. 2020.

MORADZADEH, Alireza; ALURU, Narayana R. Molecular dynamics properties without the full trajectory: A denoising autoencoder network for properties of simple liquids. **The journal of physical chemistry letters**, v. 10, n. 24, p. 7568-7576, 2019.

NG, Andrew et al. Sparse autoencoder. **CS294A Lecture notes**, v. 72, n. 2011, p. 1-19, 2011.

OTSU, Nobuyuki. A threshold selection method from gray-level histograms. **IEEE transactions on systems, man, and cybernetics**, v. 9, n. 1, p. 62-66, 1979.

PAN, Sinno Jialin; YANG, Qiang. A survey on transfer learning. **IEEE Transactions on knowledge and data engineering**, v. 22, n. 10, p. 1345-1359, 2009.

RAINA, Rajat et al. Self-taught learning: transfer learning from unlabeled data. In: **PROCEEDINGS OF THE 24TH INTERNATIONAL CONFERENCE ON MACHINE LEARNING**, 24, 2007. New York: ACM, 2007. p. 759-766. Disponível em: [https://www.researchgate.net/publication/200038912\\_Selftaught\\_learning\\_Transfer\\_learning\\_from\\_unlabeled\\_data](https://www.researchgate.net/publication/200038912_Selftaught_learning_Transfer_learning_from_unlabeled_data). Acesso em 19 mar. 2020.

RAMOS, Mery M. Paco et al. A Feature Extraction Method Based on Convolutional Autoencoder for Plant Leaves Classification. In: **IEEE COLOMBIAN CONFERENCE ON APPLICATIONS IN COMPUTATIONAL INTELLIGENCE**, 2019. Barranquilla: IEE, 2019. p. 143-154. Disponível em: <https://ieeexplore.ieee.org/abstract/document/8781985>. Acesso em: 20 jul. 2020

RUSSAKOVSKY, Olga et al. Imagenet large scale visual recognition challenge. **International journal of computer vision**, v. 115, n. 3, p. 211-252, 2015.



SUN, Meidi et al. A sparse stacked denoising autoencoder with optimized transfer learning applied to the fault diagnosis of rolling bearings. **Measurement**, v. 146, p. 305-314, 2019.

THRUN, Sebastian; PRATT, Lorien (Ed.). **Learning to learn**. Springer Science & Business Media, 2012.

TRAN, Trung-Tin et al. A comparative study of deep CNN in forecasting and classifying the macronutrient deficiencies on development of tomato plant. **Applied Sciences**, v. 9, n. 8, p. 1601, 2019.

TRANG, Kien; TONTHAT, Long; THAO, Nguyen Gia Minh. Plant leaf disease identification by deep convolutional autoencoder as a feature extraction approach. *In: INTERNATIONAL CONFERENCE ON ELECTRICAL ENGINEERING/ELECTRONICS, COMPUTER, TELECOMMUNICATIONS AND INFORMATION TECHNOLOGY*, 17, 2020. Phuket: IEEE, 2020. p. 522-526. Disponível em: <https://ieeexplore.ieee.org/document/9158218>. Acesso em 20 jul. 2020.

TURKOGLU, Muammer et al. A multi-division convolutional neural network-based plant identification system. **PeerJ Computer Science**, v. 7, p. e572, 2021.

VINCENT, Pascal et al. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. **Journal of machine learning research**, v. 11, n. 12, 2010.

WANG, Bin; WANG, Dian. Plant leaves classification: A few-shot learning method based on siamese network. **IEEE Access**, v. 7, p. 151754-151763, 2019.

WU, Stephen Gang et al. A leaf recognition algorithm for plant classification using probabilistic neural network. *In: IEEE INTERNATIONAL SYMPOSIUM ON SIGNAL PROCESSING AND INFORMATION TECHNOLOGY*, 2007. Giza: IEEE, 2007. p. 11-16. Disponível em: <https://ieeexplore.ieee.org/document/4458016>. Acesso em: 25 jul. 2020.

YANG, Meng-Meng; NAYEEM, Arifur; SHEN, Ling-Ling. Plant classification based on stacked autoencoder. *In: INFORMATION TECHNOLOGY, NETWORKING, ELECTRONIC AND AUTOMATION CONTROL CONFERENCE (ITNEC)*, 2, 2017. Chengdu: IEEE, 2017. p. 1082-1086. Disponível em: <https://ieeexplore.ieee.org/document/8284906>. Acesso em: 21 mai. 2020.

ZHAI, Junhai et al. Autoencoder and its various variants. *In: INTERNATIONAL CONFERENCE ON SYSTEMS, MAN, AND CYBERNETICS (SMC)*, 2018. Miyazaki: IEEE, 2018. p. 415-419. Disponível em: <https://ieeexplore.ieee.org/document/8616075>. Acesso em: 21 mai. 2020

ZHANG, Jing; LI, Wanqing; OGUNBONA, Philip. Transfer learning for cross-dataset recognition: a survey. **arXiv preprint arXiv:1705.04396**, 2017.

ZHENG, Wenbo; GOU, Chao; YAN, Lan. Forest Representation Learning with Multiscale Contour Feature Learning for Leaf Cultivar Classification. *In: INTERNATIONAL CONFERENCE ON BIOINFORMATICS AND BIOMEDICINE (BIBM)*, 2019. San Diego: IEEE, 2019. p. 774-777. Disponível em: <https://ieeexplore.ieee.org/document/8983276>. Acesso em: 07 jun. 2020.

ZHOU, Zhi-Hua; FENG, Ji. Deep forest. **arXiv preprint arXiv:1702.08835**, 2017.

ZHU, Heyan et al. Plant identification based on very deep convolutional neural networks. **Multimedia Tools and Applications**, v. 77, n. 22, p. 29779-29797, 2018.

ZHU, Xiaojin Jerry. Semi-supervised learning literature survey. 2005.

ZHU, Xiaojin; GOLDBERG, Andrew B. Introduction to semi-supervised learning. **Synthesis lectures on artificial intelligence and machine learning**, v. 3, n. 1, p. 1-130, 2009.

## APÊNDICE A – EXPERIMENTOS UTILIZANDO *GLOBAL AVERAGE POOLING* E *GLOBAL MAX POOLING*

Foram realizados experimentos utilizando camadas de *Global Average Pooling* (GAP) e *Global Max Pooling* (GMP), adicionadas aos CAENN's durante a extração de características das bases alvo e posteriormente classificadas individualmente pelas SVM's. Aplicando estas camadas durante a extração de características, são obtidos vetores contendo 128 características para a topologia que possui 25 camadas. Para a topologia que possui 20 camadas são geradas representações contendo 256 características. A Tabela 10 apresenta os resultados obtidos pelos experimentos utilizando GAP na extração de representações de características da base Flavia e Leafsnap.

**Tabela 10-** Resultados aplicando a camada de GAP nas duas topologias.

CAENN	Base Alvo	Taxa de acerto (%)
Kyoto_2048 (2.048 + GAP)	Flavia	87,95
Flavia_2048 (2.048 + GAP)	Flavia	90,57
Kyoto_12544 (12.544 + GAP)	Flavia	37,87
Flavia_12544 (12.544 + GAP)	Flavia	72,77
Kyoto_2048 (2.048 + GAP)	Leafsnap	80,99
Flavia_2048 (2.048 + GAP)	Leafsnap	88,59
Kyoto_12544 (12.544 + GAP)	Leafsnap	7,19
Flavia_12544 (12.544 + GAP)	Leafsnap	19,45

Fonte: O autor.

Na Tabela 11 são apresentados os resultados obtidos pelos experimentos utilizando GMP na extração de representações de características das bases Flavia e Leafsnap.

**Tabela 11** - Resultados aplicando a camada de GAP nas duas topologias.

<b>CAENN</b>	<b>Base Alvo</b>	<b>Taxa de acerto (%)</b>
Kyoto_2048 (2.048 + GMP)	Flavia	84,11
Flavia_2048 (2.048 + GMP)	Flavia	91,97
Kyoto_12544 (12.544 + GMP)	Flavia	70,15
Flavia_12544 (12.544 + GMP)	Flavia	94,76
Kyoto_2048 (2.048 + GMP)	Leafsnap	78,32
Flavia_2048 (2.048 + GMP)	Leafsnap	81,92
Kyoto_12544 (12.544 + GMP)	Leafsnap	63,06
Flavia_12544 (12.544 + GMP)	Leafsnap	79,15

Fonte: O autor.

Como pode-se observar em ambas as tabelas, os resultados obtidos utilizando GAP e GMP foram menores em todos os experimentos. Com a base Leafsnap o modelo CAENN1 utilizando a camada de GAP apresentou resultados bastante reduzidos em relação ao modelo CAENN2.

## APÊNDICE B - EXPERIMENTOS UTILIZANDO AS COMBINAÇÕES DE CLASSIFICADORES MÉTRICA F1 SCORE

Neste apêndice são apresentados os resultados dos experimentos utilizando os *ensembles* dos classificadores e a métrica *F1 score*, a qual pode ser definida como uma média harmônica da precisão e revocação que atinge seu melhor valor em 1 e o pior em 0. Na Tabela 12 são apresentados os resultados obtidos pelos experimentos utilizando os *ensembles* e a métrica *F1 score*.

**Tabela 12** - Resultados dos *ensembles* de classificadores utilizando *F1 score*.

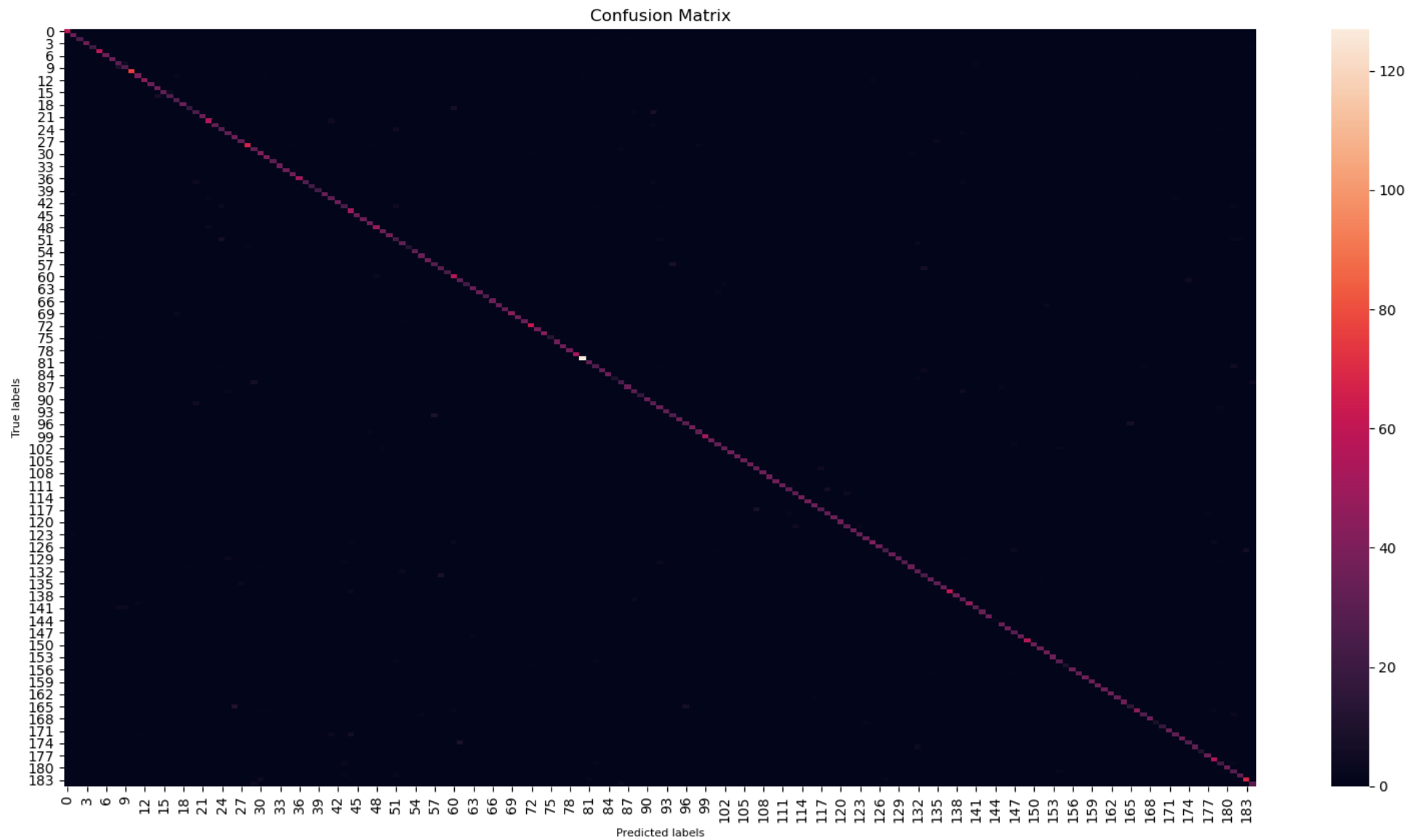
CAENN	Base Alvo	F1 score (%)
Kyoto (2.048+12.544)	Flavia	0,9418
Flavia (2.048+12.544)	Flavia	0,9507
Kyoto + Flavia (2.048+12.544)	Flavia	0,9489
Kyoto (2.048+12.544)	Leafsnap	0,9245
Flavia (2.048+12.544)	Leafsnap	0,9288
Kyoto + Flavia (2.048+12.544)	Leafsnap	0,9272

Fonte: O autor.

## **APÊNDICE C – MATRIZES DE CONFUSÃO DOS EXPERIMENTOS COM A BASE LEAFSNAP**

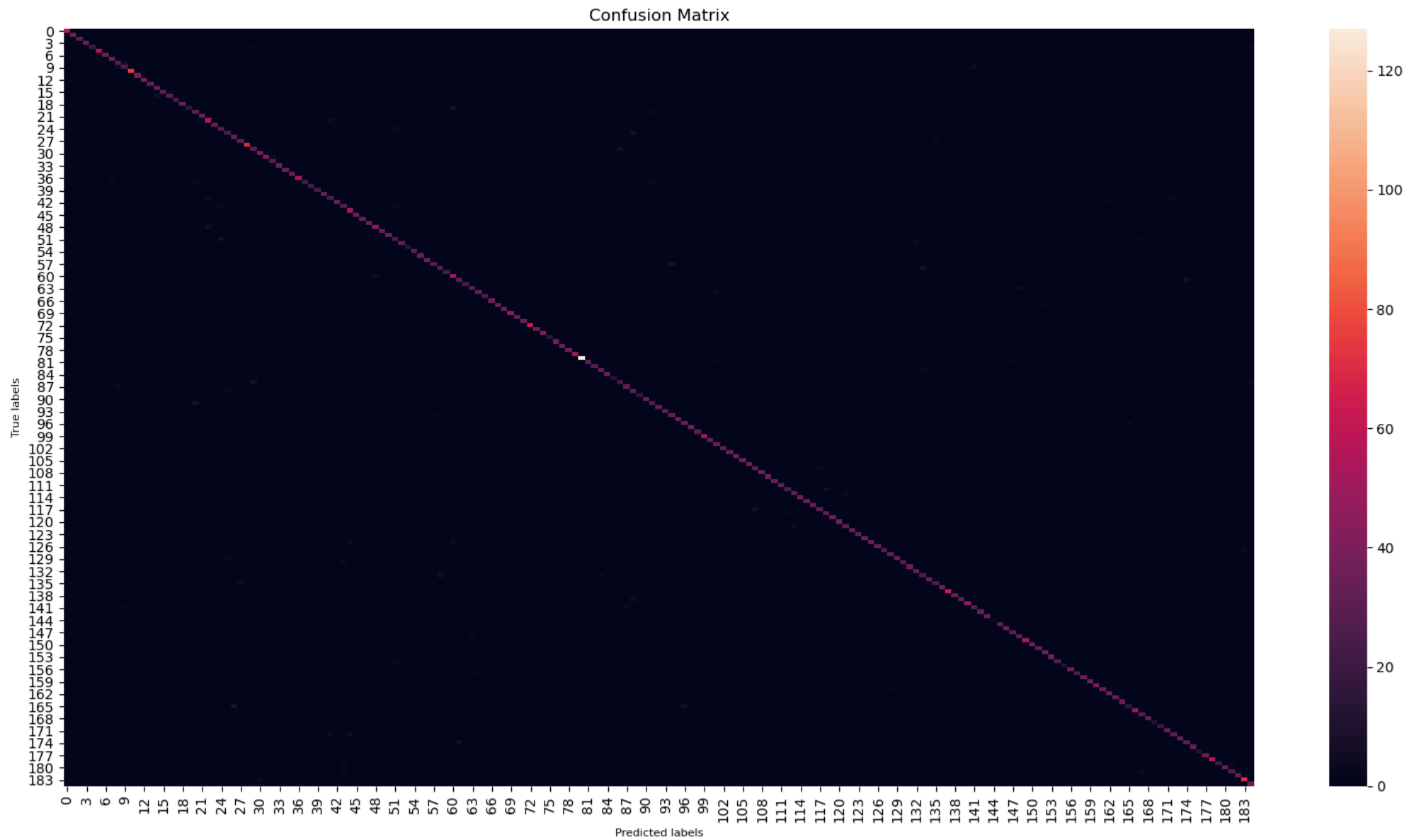
Neste apêndice são apresentadas as matrizes de confusão nas Figuras 33 e 34, geradas por meio dos experimentos utilizando a base Leafsnap e ambas as arquiteturas do modelo extrator CAENN1, treinado na base Kyoto, pois estes foram os melhores resultados obtidos individualmente com a base Leafsnap. Estas matrizes não possuem a quantidade de imagens classificadas explícita em cada bloco, devido à grande quantidade de classes, porém, pode-se analisar os resultados comparando a tonalidade de cada um com a barra que se encontra à direita de cada matriz.

**Figura 33** – Matriz de Confusão gerada pela SVM treinada com as características obtidas pela CAENN1 de 25 camadas.



Fonte: O autor.

**Figura 34** - Matriz de Confusão gerada pela SVM treinada com as características obtidas pela CAENN1 de 20 camadas.



Fonte: O autor.