

**UNIVERSIDADE ESTADUAL PAULISTA
FACULDADE DE ENGENHARIA DE BAURU**

LUIZ CARLOS MARQUES JUNIOR

**CLASSIFICAÇÃO DE PLANTAS DANINHAS EM BANCO DE
IMAGENS UTILIZANDO REDES NEURAIAS
CONVOLUCIONAIS**

BAURU

2019

LUIZ CARLOS MARQUES JUNIOR

**CLASSIFICAÇÃO DE PLANTAS DANINHAS EM BANCO DE
IMAGENS UTILIZANDO REDES NEURAIS
CONVOLUCIONAIS**

Dissertação apresentada como requisito à obtenção do título de mestre em Engenharia Elétrica, pelo Programa de Pós-Graduação em Engenharia Elétrica, da Faculdade de Engenharia de Bauru, da Universidade Estadual Paulista.

Orientador: Prof. Dr. José Alfredo Covolan Ulson

BAURU

2019

Marques Junior, Luiz Carlos.

Classificação de Plantas Daninhas em Banco de
Imagens Utilizando Redes Neurais Convolucionais /
Luiz Carlos Marques Junior, 2019

64 f. : il.

Orientador: José Alfredo Covolan Ulson

Dissertação (Mestrado)-Universidade Estadual
Paulista. Faculdade de Engenharia, Bauru, 2019

1. Agricultura de Precisão. 2. Redes Neurais
Convolucionais. 3. Aprendizado Profundo. 4. Plantas
Daninhas. I. Universidade Estadual Paulista. Faculdade
de Engenharia. II. Título.

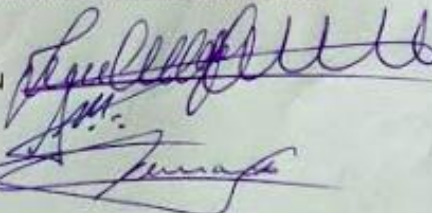
ATA DA DEFESA PÚBLICA DA DISSERTAÇÃO DE MESTRADO DE LUIZ CARLOS MARQUES JUNIOR, DISCENTE DO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA, DA FACULDADE DE ENGENHARIA - CÂMPUS DE BAURU.

Aos 28 dias do mês de maio do ano de 2019, às 10:00 horas, no(a) Anfiteatro da Seção Técnica de Pós-graduação da FEB, reuniu-se a Comissão Examinadora da Defesa Pública, composta pelos seguintes membros: Prof. Dr. JOSÉ ALFREDO COVOLAN ULSON - Orientador(a) do(a) Departamento de Engenharia Elétrica / Faculdade de Engenharia de Bauru - UNESP, Prof. Dr. ADRIANO DE SOUZA MARQUES do(a) COMPUTAÇÃO / INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SÃO PAULO, Prof. Dr. FERNANDO DE SOUZA CAMPOS do(a) Departamento de Engenharia Elétrica / Faculdade de Engenharia de Bauru - UNESP, sob a presidência do primeiro, a fim de proceder a arguição pública da DISSERTAÇÃO DE MESTRADO de LUIZ CARLOS MARQUES JUNIOR, intitulada **CLASSIFICAÇÃO DE PLANTAS DANINHAS EM BANCO DE DADOS DE IMAGEM UTILIZANDO REDES NEURAS CONVOLUCIONAIS**. Após a exposição, o discente foi arguido oralmente pelos membros da Comissão Examinadora, tendo recebido o conceito final: aprovado. Nada mais havendo, foi lavrada a presente ata, que após lida e aprovada, foi assinada pelos membros da Comissão Examinadora.

Prof. Dr. JOSÉ ALFREDO COVOLAN ULSON

Prof. Dr. ADRIANO DE SOUZA MARQUES

Prof. Dr. FERNANDO DE SOUZA CAMPOS



AGRADECIMENTOS

Gostaria de agradecer primeiramente a Deus, por ter me capacitado e permitido a execução desta pesquisa que culminou nesta tese.

A minha esposa pelo apoio, compreensão e companheirismo em todos os momentos.

A família, em especial aos meus pais, por terem me dado a possibilidade e por ter me apoiado nos estudos.

Ao meu orientador Prof. Dr. José Alfredo Covolan Ulson, pelo apoio, tempo dedicado a este trabalho e confiança depositada em mim para elaboração desta pesquisa.

A Coordenação de Aperfeiçoamento de Pessoal de Nível Superior pela bolsa de mestrado possibilitando a oportunidade de aprofundar meus conhecimentos e contribuir para o cenário da ciência e tecnologia do Brasil.

RESUMO

As espécies exóticas invasoras, também conhecidas como plantas daninhas, competem por recursos, como sol, água e nutrientes paralelamente a cultura plantada, impondo prejuízos econômicos ao agricultor. Para minimizar este problema, atualmente os agricultores fazem uso de herbicidas para a eliminação e/ou controle das plantas daninhas. O uso de herbicidas depara-se com problemas: i) algumas plantas daninhas são resistentes a aplicação de herbicidas e, ii) quando aplicados em demasia pode-se ter a contaminação da cultura plantada, do lençol freático e dos mananciais como rios e lagos. Nesse contexto, visando o desenvolvimento de ferramentas que permitam a minimização do emprego de herbicidas, novas abordagens que fazem uso de visão computacional e inteligência artificial aparecem como soluções promissoras, agregando novas ferramentas a agricultura de precisão. Dentre essas soluções destaca-se o aprendizado profundo (do inglês *Deep Learning*), que utiliza as redes neurais convolucionais para extrair características relevantes, principalmente em imagens, dessa maneira, permite por exemplo a identificação e a classificação de plantas daninhas, o que possibilita ao agricultor optar tanto pela eliminação mecânica da planta daninha quanto a aplicação localizada de herbicidas e em quantidades adequadas. A partir deste desafio que é a correta classificação de diferentes espécies de plantas daninhas, especialmente plantas resistentes aos herbicidas comerciais, o objetivo deste trabalho foi aplicar e comparar a performance de quatro arquiteturas de redes neurais convolucionais para a classificação de plantas daninhas de cinco espécies contidas em um banco de imagens desenvolvido para este trabalho. Para isso foi realizado o treinamento e a classificação das espécies nas seguintes arquiteturas de redes neurais convolucionais: *VGG16*, *ResNet50*, *InceptionV3* e *InceptionResNetV2* com 20 épocas de treinamento. Os resultados indicam que a arquitetura *InceptionV3* apresenta o melhor desempenho, com 84,73% de exatidão na classificação nas cinco espécies, seguida pela arquitetura *InceptionResNetV2* com 82,87%, *VGG16* com 80,60%. A arquitetura *ResNet50* obteve o pior resultado com 20,00% de exatidão, a rede *InceptionV3* foi treinada novamente com 40 épocas, obtendo 88,50% de exatidão.

Palavras-chave: Agricultura de Precisão, Redes Neurais Convolucionais, Aprendizado Profundo, Plantas Daninhas.

ABSTRACT

Exotic invasive species, also known as weeds, compete for resources such as sun, water and nutrients in parallel with the planted crop, imposing economic losses to the farmer. To minimize this problem, farmers are currently using herbicides for the elimination and / or control of weeds. The use of herbicides has problems: i) some weeds are resistant to the application of herbicides and ii) when applied too much can contaminate the planted crop, groundwater and springs such as rivers and lakes. In this context, aiming at developing tools to minimize the use of herbicides, new approaches that make use of computer vision and artificial intelligence appear as promising solutions, adding new tools to precision agriculture. Among these solutions are the Deep Learning, which uses the convolutional neural networks to extract relevant features, mainly in images, thus, allows for example the identification and classification of weeds, which enables the farmer to opt for the mechanical elimination of the weeds as well as the localized application of herbicides and in adequate quantities. From this challenge, which is the correct classification of different weed species, especially plants resistant to commercial herbicides, the objective of this study was to apply and compare the performance of four architectures of convolutional neural networks for classification of weed five species contained in an image bank developed for this work. The training and classification of the species were carried out in the following convolutional neural network architectures: VGG16, ResNet50, InceptionV3 and InceptionResNetV2 with 20 training epochs. The results indicated that the InceptionV3 architecture presented the best performance, with 84.73% accuracy in the classification of the five species, followed by the InceptionResNetV2 architecture with 82.87%, VGG16 with 80.60%. The ResNet50 architecture obtained the worst result with 20.00% accuracy, the InceptionV3 network was trained again with 40 epochs, obtaining 88.50% accuracy.

Keywords: Precision Agriculture, Convolutional Neural Networks, Deep Learning, Weeds.

LISTA DE FIGURAS

Figura 1 - Uso de Agrotóxicos Por Tipo no Brasil	16
Figura 2 - Principais Lavouras no Uso de Agrotóxicos no Brasil	17
Figura 3 - Aumento de Resistência de Plantas Daninhas ao Mesmo Herbicida	20
Figura 4 - Espécies de Plantas Daninhas Resistentes ao Glifosato,(a) Buva,(b) Capim Amargoso,(c) Capim-Azevém, (d) Capim-Pé-de-Galinha,(e) Caruru	21
Figura 5 - Representação do Neurônio Biológico.....	22
Figura 6 - Representação do Neurônio Matemático	23
Figura 7 - Arquitetura de Rede Neural com Propagação para Frente e Retro propagação..	25
Figura 8 - Representação da Inteligência Artificial e suas subáreas	26
Figura 9 - Rede Neural Convolutacional proposta por Lecun, Y.....	28
Figura 10 - Arquitetura de Aprendizagem Profunda Para Classificação de Plantas Daninhas	29
Figura 11 - Imagens Segmentadas Utilizadas para Classificação de Plantas Daninhas	31
Figura 12 - Plataforma de Coleta de Imagens BoniRob e Plantas Analisadas.....	32
Figura 13 - Fluxo metodológico de etapas para a aplicação das arquiteturas de aprendizado profundo e geração dos resultados	34
Figura 14 - Arquitetura VGG16.....	37
Figura 15 - Bloco Residual Rede Resnet	39
Figura 16 - Arquitetura InceptionV3.....	40
Figura 17 - Diferença entre Agrupamento Médio e Máximo.....	40
Figura 18 - Arquitetura InceptionResnetV2	41
Figura 19 - Comparação entre uma imagem da espécie Capim Azevém sem ruído (a), e com ruído inserido (b).....	42
Figura 20 - Comparação entre uma imagem da espécie Capim Pé de Galinha sem ruído (a), e com ruído inserido (b).....	43
Figura 21 - Gráfico de Perda de Treinamento das Arquiteturas	46
Figura 22 - Gráfico de Exatidão de Treinamento das Arquiteturas	46
Figura 23 - Gráfico de Perda de Validação das Arquiteturas	47
Figura 24 - Gráfico de Exatidão de Validação das Arquiteturas	48
Figura 25 - Gráfico de Perda de Treinamento e Validação VGG16.....	49
Figura 26 - Gráfico de Exatidão de Treinamento e Validação VGG16.....	49
Figura 27 - Gráfico de Perda de Treinamento e Validação ResNet50.....	50
Figura 28 - Gráfico de Exatidão de Treinamento e Validação ResNet50.....	50
Figura 29 - Gráfico de Perda de Treinamento e Validação InceptionV3	51

Figura 30 - Gráfico de Exatidão de Treinamento e Validação InceptionV3	51
Figura 31 - Gráfico de Perda de Treinamento e Validação InceptionResNetV2	52
Figura 32 - Gráfico de Exatidão de Treinamento e Validação InceptionResNetV2	52
Figura 33 - Matriz de Confusão Arquitetura VGG16.....	54
Figura 34 - Matriz de Confusão Arquitetura ResNet50	54
Figura 35 - Matriz de Confusão Arquitetura InceptionV3.....	55
Figura 36 - Matriz de Confusão Arquitetura InceptionResNetV2.....	55
Figura 37 - Matriz de Confusão Arquitetura InceptionV3 40 Épocas	57

LISTA DE QUADROS

Quadro 1 - Estados Líderes em Vendas e Consumo de Agrotóxicos no Brasil	18
Quadro 2 - Espécies de Plantas Daninhas Seleccionadas para este Trabalho	21
Quadro 3 - Arquiteturas de Redes Neurais Profundas Armazenadas no Keras e usadas neste trabalho	36
Quadro 4 - Performance das Arquiteturas com base nas matrizes de confusão	56
Quadro 5 - Performance da Arquitetura InceptionV3 com 40 épocas de treinamento, com base nas matrizes de confusão.....	57

LISTA DE SIGLAS

ALS	Acetolactate Synthase
ANVISA	Agência Nacional de Vigilância Sanitária
API	Application Programming Interface
EMBRAPA	Empresa Brasileira de Pesquisa Agropecuária
IBAMA	Instituto Brasileiro do Meio Ambiente e dos Recursos Naturais Renováveis
ILSVRC	Imagenet Large Scale Visual Recognition Challenge
LMR	Limite Máximo de Resíduos
NDVI	Normalized Difference Vegetation Index
OMS	Organização Mundial da Saúde
ONU	Organização das Nações Unidas
PIB	Produto Interno Bruto
RGB	Red Green Blue
SINDIVEG	Sindicato Nacional da Indústria de Produtos para a Defesa Vegetal
SLIC	Simple Linear Iterative Clustering
VANT	Veículo Aéreo Não Tripulado

SUMÁRIO

1	INTRODUÇÃO	12
1.1	Contextualização do Tema	12
1.2	Objetivos	14
1.3	Estrutura da Dissertação	14
2	REVISÃO BIBLIOGRÁFICA	16
2.1	A Agricultura no Brasil e o Uso de Pesticidas	16
2.2.1	Plantas Daninhas do Brasil Resistentes ao Herbicida Glifosato	20
2.3	Redes Neurais Artificiais	22
2.4	Aprendizado Profundo	26
2.5	Redes Neurais Convolucionais	27
2.6	Redes Neurais Convolucionais na Agroindústria	28
2.6.1	Detecção Automática e Classificação de Plantas Daninhas sob Condições Naturais de Iluminação	29
2.6.2	Detecção de Plantas Daninhas em Plantações de Soja usando Redes Convolucionais	30
2.6.3	Segmentação Semântica em Tempo Real de Plantações e Plantas Daninhas para Robôs de Agricultura de Precisão com o Conhecimento de Plano de Fundo em CNNs	31
3	MATERIAIS E MÉTODOS	34
3.1	Biblioteca Tensorflow	34
3.2	Biblioteca Keras	35
3.3	Arquiteturas de Redes Neurais Convolucionais	37
3.3.1	VGG16: Redes Convolucionais Muito Profundas para Reconhecimento de Imagem em Grande Escala	37
3.3.2	ResNet-50: Aprendizagem Residual Profunda para Reconhecimento de Imagem	38
3.3.3	InceptionV3: Repensando a Arquitetura Inception para Visão Computacional	39
3.2.4	Inception-v4, Inception-ResNet e o Impacto das Conexões Residuais no Aprendizado	41
3.4	Obtenção do Banco de Imagens	41
3.5	Inserção de Ruídos	42
3.6	Parametrização do Algoritmo	43
4	RESULTADOS E DISCUSSÕES	45
5	CONCLUSÕES	58
	REFERÊNCIAS	59

1 INTRODUÇÃO

Este capítulo apresenta a contextualização e importância deste projeto de pesquisa, assim como a motivação, o objetivo e metodologia aplicada a este trabalho de pesquisa.

1.1 Contextualização do Tema

O agronegócio desempenha função de destaque no cenário econômico nacional, sendo responsável por 25% do Produto Interno Bruto (PIB) do Brasil e 20% dos postos de trabalho. Sendo ainda um dos líderes mundiais na produção e exportação de grãos como soja e milho, de carnes, como a bovina e de frango e cana de açúcar para a produção de etanol (Vasconcelos, 2018).

Entretanto, se por um lado ano após ano o Brasil vem aumentando a sua produção e exportação agrícola, em contrapartida vem aumentando também o uso de agrotóxicos nas plantações. Sendo que as plantações de soja, cana-de-açúcar e milho juntas respondem por 75% dos defensivos agrícolas consumidos no Brasil. Atualmente o Brasil gasta cerca de US\$ 10 bilhões por ano em defensivos agrícolas, representando 20% do mercado mundial estimado em US\$ 50 bilhões (Vasconcelos, 2018).

Uma das explicações para o uso em demasia dos defensivos agrícolas no Brasil deve-se pelo fato do clima tropical do país, assim, não tendo na maior parte de sua extensão um período de inverno severo para interromper o desenvolvimento das pragas. Deve-se também a expansão da produção agrícola, por exemplo, a safra de grãos no ano de 2010 era de 149 milhões de toneladas, já em 2017 foi de 238 milhões, e principalmente a produção de monoculturas na mesma área como milho e soja.

Dados de uma relatório de 2017 feito pela Organização das Nações Unidas (ONU) estimou que cerca de 200 mil pessoas morrem anualmente no mundo vítimas de envenenamento agudo por pesticidas, principalmente trabalhadores rurais e moradores do campo (ESTADOS UNIDOS, 2017). No Brasil, 84,2 mil pessoas sofreram intoxicação após exposição a defensivos agrícolas entre 2007 e 2015, uma média de 25 intoxicações por dia, conforme dados do Relatório Nacional de Vigilância em Saúde de Populações Expostas a Agrotóxicos 2018, elaborado pelo Ministério da Saúde (BRASIL, 2018).

Dentre os defensivos agrícolas, o herbicida é o mais utilizado nacionalmente, principalmente na lavoura de soja em plantas daninhas que competem pela luz e nutrientes com a cultura. Um estudo realizado pelo Sindicato Nacional da Indústria de Produtos para a Defesa Vegetal (SINDIVEG) indica que 60% dos defensivos consumidos em 2017 no Brasil foram herbicidas. O estudo ainda indica que as culturas de soja, cana de açúcar e milho, consumiram respectivamente 52,2%, 11,7% e 10,6% de todos os defensivos agrícolas consumidos, representando um total de agrotóxicos consumidos de 74,5% (SINDIVEG, 2017).

Com base nestas informações, surge a necessidade da exploração e desenvolvimento de novas técnicas que possibilitem a identificação de plantas daninhas para a correta dosagem e aplicação de agrotóxicos ou eliminação mecânica. Nesse contexto, a agricultura de precisão e as ferramentas que fazem parte dela aparecem como possíveis soluções promissoras para este desafio. Dentre as ferramentas que podem ser utilizadas para fornecer soluções para esta problemática foi selecionado para este trabalho a aplicação do aprendizado profundo (*Deep Learning*), associado à visão computacional para a classificação de diferentes espécies de plantas daninhas.

O aprendizado profundo é uma técnica de inteligência artificial, que vem obtendo excelentes resultados, principalmente na área de visão computacional, por meio das redes neurais convolucionais (*Convolutional Neural Networks*). Dentre suas aplicações destacam-se a classificação de imagens (Krizhevsky et al., 2012), (Simonyan, K; Zisserman, A, 2015), (Szegedy, C. et al., 2015), (He, K. et al., 2016) a classificação de imagens com localização (UIJLINGS, J. R. R. et al., 2013), (Girshick, R et al., 2014), (Girshick, R, 2015) detecção de objetos (Sermanet, P et al., 2014), (Redmon, J et al., 2016) e segmentação de objetos (BadrinarayanaN, V et al., 2017), (HE, K et al., 2017) .

Para realizar a classificação das espécies de plantas daninhas criou-se um banco de imagens com cinco espécies que apresentam resistências aos principais herbicidas comercialmente utilizados no Brasil, assim, caso não seja feito o correto manejo dessas pragas o agricultor terá perdas financeiras significativas.

Diante deste desafio técnico científico, foi proposto para este trabalho utilizar quatro arquiteturas de redes Neurais Profundas (*Deep Neural Networks*), VGG16,

ResNet50, *InceptionV3* e *InceptionResNetV2*, para a identificação e classificação de cinco espécies de plantas daninhas, verificando assim a exatidão de cada arquitetura para realizar esta tarefa, determinando qual arquitetura obteve a melhor performance e assim, possibilitando verificar se esta abordagem foi eficaz para o problema proposto.

1.2 Objetivos

O objetivo principal desta dissertação é realizar a classificação de 5 espécies de plantas daninhas encontradas em todo o território nacional, utilizando redes neurais convolucionais em um banco de imagens. Para a criação do banco coletou-se imagens na *internet* das espécies de plantas daninhas que são encontradas em todo território nacional e apresentam resistência a aplicação dos principais herbicidas do mercado atual, posteriormente foi inserido diversos ruídos nestas imagens, criando um banco de imagens para o treinamento e validação das arquiteturas estudadas.

Também é objetivo deste trabalho verificar a exatidão das arquiteturas de aprendizado profundo na classificação, portanto, podendo-se determinar qual arquitetura obteve os melhores. Dessa maneira verificando a performance da abordagem proposta para o problema de classificação de imagens plantas daninhas.

Espera-se por meio desta dissertação contribuir para o estudo das técnicas de aprendizado profundo e a aplicação de inteligência artificial na resolução dos problemas e desafios que a agricultura brasileira vem enfrentando, e possa deparar-se no futuro. Dentre estes desafios o que foi abordado neste trabalho é a possibilidade da correta classificação de plantas daninhas o que permite ao agricultor dosar a quantidade ideal de herbicida por espécie de planta daninha ou realizar a eliminação mecânica.

1.3 Estrutura da Dissertação

No capítulo 2 é apresentado a revisão da literatura. Neste capítulo serão abordados os temas referentes à dissertação que se encontram na literatura desde dados referentes ao uso de agrotóxicos no Brasil, as principais plantas daninhas que causam prejuízos aos agricultores e apresentam alta resistência a herbicidas, também é abordado o histórico das redes neurais até chegar nas redes neurais convolucionais e trabalhos recentes que fizeram uso desta técnica na agricultura de

precisão para a classificação de plantas daninhas. O intuito deste capítulo é posicionar o leitor acerca do tema e da problemática na qual o trabalho se encontra.

No capítulo 3 são apresentados os materiais e métodos utilizados. Métodos envolvendo a criação do banco de imagens, inserção de ruído e classificação destas imagens por meio das quatro arquiteturas de redes neurais convolucionais estudadas.

No capítulo 4 aborda-se os resultados obtidos para cada arquitetura de rede neural convolucional por meio dos gráficos de exatidão e perdas, posteriormente apresentam-se as matrizes de confusão para cada arquitetura, onde poder-se-á verificar a performance de classificação para as diferentes espécies de plantas.

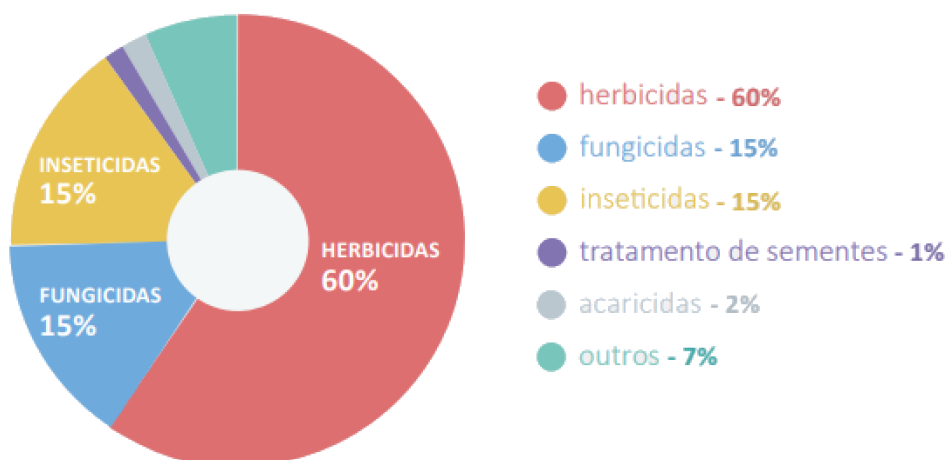
O capítulo 5 se concentra em apresentar as conclusões gerais do trabalho e propostas de trabalhos futuros.

2 REVISÃO BIBLIOGRÁFICA

2.1 A Agricultura no Brasil e o Uso de Pesticidas

Ao longo dos últimos 50 anos a agricultura brasileira passou por um período de expansão e transformação em relação a produção dos insumos agrícolas, principalmente em culturas de soja, milho, cana de açúcar, café e algodão . Um trabalho que até então dependia de mão de obra humana com auxílio animal para o preparo e produção dos insumos, atualmente possui alto nível de automação, seja pelo uso de máquinas agrícolas de colheita, de semeadura ou até mesmo pela aplicação de defensivos agrícolas por meio de aviões pulverizadores. Entretanto o aumento da produtividade destas culturas tem representado um aumento significativo no uso de defensivos agrícolas, em especial os defensivos de finalidade herbicida para o controle e extermínio de plantas daninhas. A figura 1 reforça esses dados mostrando para quais tipos de pragas mais se utilizou defensivos agrícola no Brasil no ano de 2017.

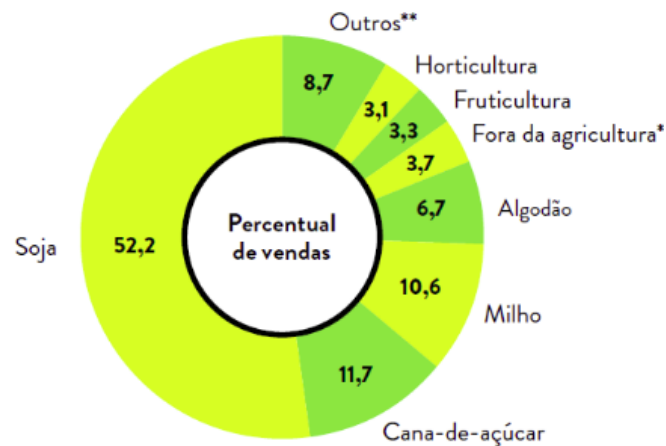
Figura 1- Uso de Agrotóxicos Por Tipo no Brasil



Fonte: SINDIVEG (2017)

Os defensivos com a função de herbicida representaram 60% de todo o consumo nacional do ano de 2017, sendo acompanhado pelos fungicidas e inseticidas, cada um com 15% respectivamente do consumo nacional. Para que se possa compreender melhor a aplicação destes defensivos, na figura 2 o consumo de defensivo agrícola por cultura plantada é apresentado.

Figura 2 - Principais Lavouras no Uso de Agrotóxicos no Brasil



*Reflorestamento, pastagem, floricultura e fumo **Café, grãos, arroz e feijão

Fonte: SINDIVEG (2017)

A soja lidera o uso de defensivos agrícolas com 52,2% sendo seguida pela cana de açúcar com 11,7% e milho com 10,6%. Juntas as 3 culturas representam 74,5% do uso de defensivos agrícolas no Brasil em 2017. A título de comparação, os ingredientes ativos com ação herbicida que lideram a lista dos agrotóxicos mais comercializados, tiveram, em 2009, uma quantidade comercializada da ordem de 127 mil toneladas. Já em 2013, o glifosato sozinho, o herbicida mais vendido, teve mais de 185 mil toneladas comercializadas (IBAMA, 2013). Para que se possa verificar a distribuição da comercialização destes agrotóxicos no quadro 1 são apresentados os principais estados comercializadores e consumidores de agrotóxicos no Brasil no ano de 2017, adaptado com base nos dados dos Relatórios de Comercialização de Agrotóxicos Publicado pelo IBAMA em 2017.

Quadro 1 - Estados Líderes em Vendas e Consumo de Agrotóxicos no Brasil

Total das Vendas de Agrotóxicos e Afins nas Regiões e Estados Brasileiros – 2017			
Unidade de medida = toneladas de ingrediente ativo (IA)			
Estado	Quantidade (Toneladas)	Porcentagem Nacional	Total Nacional (Toneladas)
Mato Grosso (MT)	100.638,47	18,64%	539.944,95
São Paulo (SP)	77.232,66	14,30%	539.944,95
Rio Grande do Sul (RS)	70.143,64	13,00%	539.944,95
Paraná (PR)	61.130,01	11,32%	539.944,95
Total	309.144,78	57,26%	539.944,95

Fonte: IBAMA (2017), adaptado pelo autor

O estado do Mato Grosso lidera com 18,64%, sendo este estado o líder nacional na produção de soja, cultura esta que depende intensamente do uso de herbicidas. Em seguida o estado de São Paulo com 14,30%, líder na produção de cana de açúcar. O estado do Rio Grande do Sul que vem aumentando o cultivo de soja representa 13,00% da comercialização e o Paraná com 11,32% ocupa o terceiro lugar em produção de cana e segundo lugar em produção de soja, é o quarto estado em uso de agrotóxicos, juntos os quatro estados representam 57,26% da comercialização nacional de defensivos agrícolas. Tal consumo destes agrotóxicos, vem causando uma série de prejuízos sociais, ambientais e econômicos, como a contaminação do solo e lençóis freáticos, morte ou intoxicação de trabalhadores rurais e ainda aumento dos custos de produção das culturas agrícolas devido a resistência das plantas daninhas.

Nos itens seguintes, informações sobre as plantas daninhas, a resistência dessas plantas a herbicidas e quais espécies apresentam maior resistência e prejuízos ao agricultor são abordados, assim como as 5 espécies de plantas daninhas que são foco deste estudo.

2.2 Plantas Daninhas

Segundo dados da EMBRAPA (Empresa Brasileira de Pesquisa Agropecuária) as plantas daninhas aparecem como um dos fatores que mais afetam a produtividade. Prejudicando a cultura plantada seja pela competição por recursos como sol, água e nutrientes, ou pela alopatia, que é a capacidade das plantas produzirem substâncias prejudiciais para outras plantas e conseqüentemente causando a perda de rendimento da produção. Causam também efeitos indiretos, como o aumento do custo, a piora na colheita e depreciação da qualidade do produto. Em casos em que não é feito controle algum de plantas daninhas, as perdas podem chegar a mais de 90% da produção, com o controle tem-se em média de 13 a 15% de perda na produção de grãos.

A primeira ideia do conceito de plantas daninhas surgiu ainda nos tempos bíblicos, no momento em que o homem deu início às atividades agrícolas selecionando plantas consideradas úteis (cultivadas) daquelas consideradas inúteis (invasoras). Nos dias de hoje, plantas daninhas englobam todas as plantas que interferem no crescimento das cultivadas, mostrando-se persistentes, e que atuam de forma negativa nas atividades humanas, sendo consideradas como plantas indesejadas. Este tipo de planta costuma crescer em condições adversas, como ambientes secos ou úmidos, com temperaturas baixas ou elevadas e variados tipos de solos. Estas plantas apresentam capacidade de produzir sementes viáveis em abundância, com variadas formas de dispersão, além de apresentarem resistência ao ataque de pragas e doenças(EMBRAPA, 2018).

Das 350.000 espécies conhecidas de plantas, apenas 3.000 são cultivadas; e aproximadamente 250 são universalmente consideradas plantas daninhas, das quais cerca de 40% pertencem a apenas duas famílias: *Poaceae* (gramíneas) e *Asteraceae* (compostas). Por causa do seu caráter competitivo, as plantas daninhas garantem sua perpetuação por meio de dormência e germinação desuniforme das sementes. Estas habilidades conferem um difícil controle das espécies invasoras pelo fato de não germinarem todas ao mesmo tempo, mesmo em condições ideais de temperatura, umidade e luz (EMBRAPA, 2018).

O desenvolvimento das plantas invasoras é rápido, sendo capaz de atingir sua maturidade em pouco tempo. A produção de sementes é elevada (produzem em grandes quantidades), porém, este não é o único meio de reprodução destas

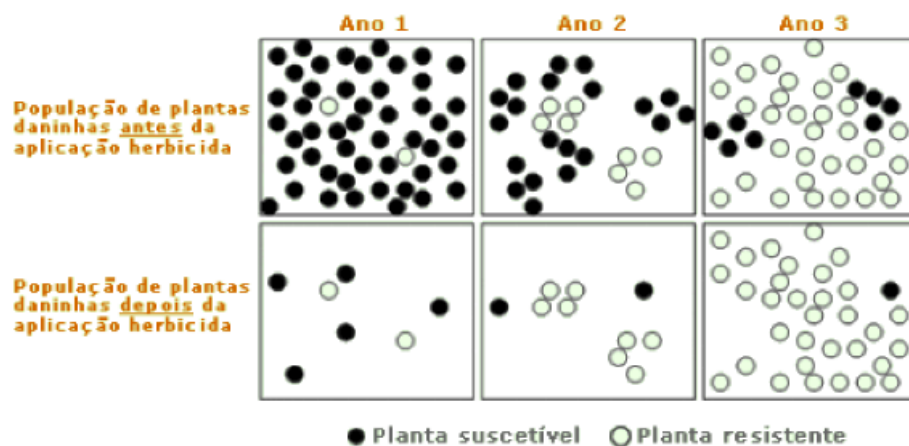
invasoras; algumas espécies apresentam capacidade reprodutiva também através de bulbos, tubérculos, rizomas(caules) e enraizamento.

2.2.1 Plantas Daninhas do Brasil Resistentes ao Herbicida Glifosato

As plantas daninhas, evolutivamente, já apresentam uma variabilidade genética natural. Ou seja, dentro de uma mesma espécie de planta daninha há diferenças genéticas entre cada indivíduo. Os herbicidas selecionam aqueles indivíduos que apresentam genes que resultam em resistência. A Resistência de uma planta daninha a um herbicida é a capacidade adquirida por uma planta em sobreviver e se reproduzir mesmo com a aplicação de um herbicida na dose registrada (dose indicada na bula) em condições normais e adequadas de aplicação.

Lavouras de soja com plantas daninhas resistentes ao glifosato, principal herbicida comercial utilizado no Brasil, possuem custos de 22 a 42% maiores, segundo dados da EMBRAPA. As aplicações de herbicidas não criam uma planta resistente, apenas selecionam. A seleção de biótipos resistentes ocorre através da aplicação de um mesmo herbicida repetidas vezes na mesma área. Como, por exemplo, nas culturas transgênicas resistentes ao glifosato, onde muitos produtores aplicam esse herbicida para toda a cultura plantada, por um período contínuo de diversas safras. A figura 3 ilustra o processo de seleção de plantas resistentes que ocorre ao longo dos anos de aplicação de um mesmo herbicida.

Figura 3 - Aumento de Resistência de Plantas Daninhas ao Mesmo Herbicida



Fonte: Christoffoleti (2008).

Conforme ilustrado na figura 4 a cada ano que o mesmo herbicida é aplicado, menos eficiente ele se torna na eliminação de plantas daninhas. No Brasil, foram

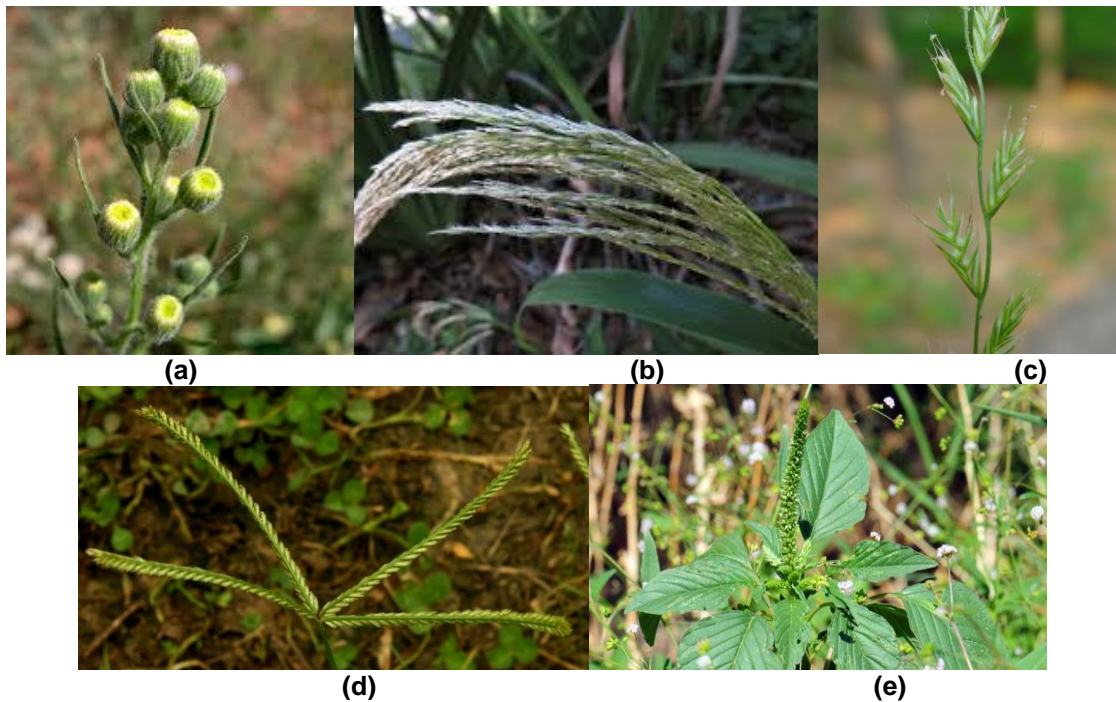
identificados biótipos resistentes ao glifosato das seguintes espécies apresentadas pelo quadro 2 e na figura 4:

Quadro 2 - Espécies de Plantas Daninhas Seleccionadas para este Trabalho

Nome Popular da Espécie	Nome Científico da Espécie
Buva	<i>Conyza Bonariensis</i>
Capim-Azevém	<i>Lolium Multiflorum</i>
Capim-Amargoso	<i>Digitaria Insularis</i>
Capim-Pé-de-Galinha	<i>Eleusine Indica</i>
Caruru Plameri	<i>Amaranthus Palmeri</i>

Fonte: Elaboração do próprio Autor

Figura 4 - Espécies de Plantas Daninhas Resistentes ao Glifosato, (a) Buva, (b) Capim Amargoso, (c) Capim-Azevém, (d) Capim-Pé-de-Galinha, (e) Caruru



Fonte: Elaboração do próprio Autor

As espécies contidas na figura 4 e descritas no quadro 2 foram seleccionadas para a criação do banco de imagens e treinamento das arquiteturas de redes neurais profundas. Cada espécie tem uma aparência singular se comparadas entre si, porém podem ser facilmente camufladas quando em meio a vegetação. Sendo que este será um dos pontos testados neste trabalho, que é a capacidade da rede convolucional

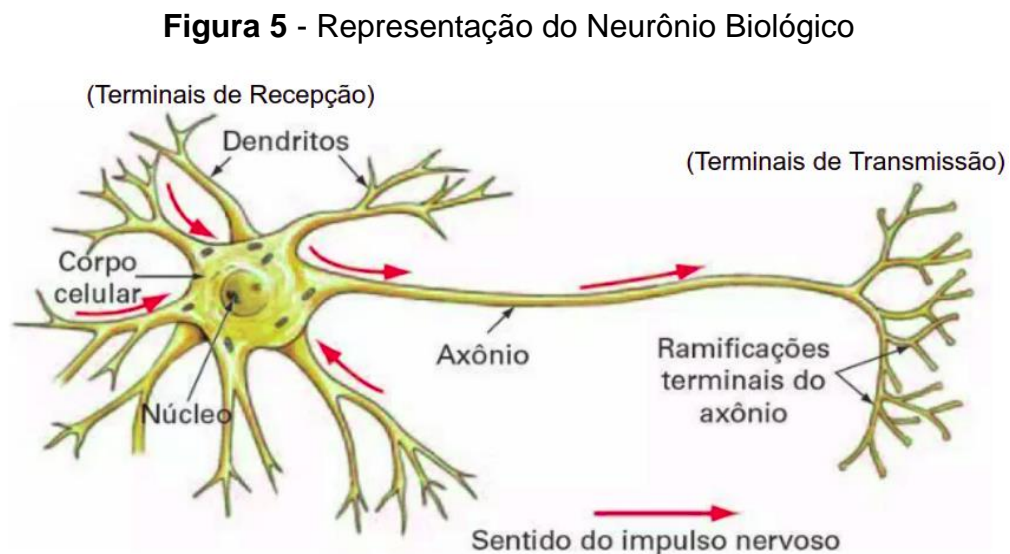
classificar corretamente a espécie, mesmo quando não segmentada do plano de fundo.

Segundo dados da Syngenta, empresa líder, na produção de semente e defensivos agrícolas, do total da área de soja plantada no Brasil, 60% tem espécies de plantas daninhas resistentes. O custo devido a resistência das plantas daninhas chega a R\$ 9 bilhões, podendo chegar a perda de 70% de produtividade caso não haja nenhum tipo de controle. Na safra 2016/2017 teve-se uma incidência de 50% de Buva e 40% de Capim-Amargoso, estima-se que até 2022 esse número aumente tendo-se 55% de incidência de Buva e 80% de Capim-Amargoso (Syngenta, 2017).

Uma vez definidas as espécies de plantas daninhas que serão utilizadas neste trabalho, os capítulos seguintes abordam o histórico e a evolução das redes neurais, até sua aplicação na agricultura de precisão.

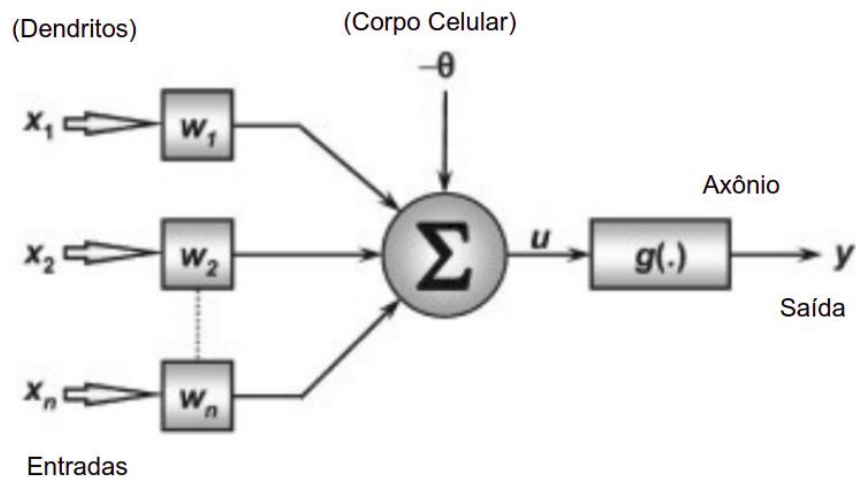
2.3 Redes Neurais Artificiais

Com base no cérebro do ser humano pesquisadores tentaram simular este funcionamento, principalmente o aprendizado por experiência para criar sistemas inteligentes capazes de realizar tarefas como classificação, reconhecimento de padrões, processamento de imagens, entre outras atividades. Como resultado dessas pesquisas surgiu o modelo do neurônio artificial e posteriormente um sistema com vários neurônios interconectados, a chamada Rede Neural. Na figura 5 tem-se uma representação do neurônio biológico.



O neurônio biológico pode ser dividido entre os dendritos que são os terminais de recepção, que recebem os impulsos nervosos (entradas), o corpo celular, que processa os sinais das entradas, e o axônio que são os terminais de transmissão (saída) do impulso nervoso. Os primeiros neurônios matemáticos artificiais datam de 1943, quando o neurofisiologista *Warren McCulloch* e o matemático *Walter Pitts* escreveram um artigo sobre como os neurônios poderiam funcionar e para isso, eles modelaram uma rede neural simples usando circuitos elétricos (W, S. McCulloch, 1943). Com base no neurônio *McCulloch* e *Pitts*, uma representação do neurônio matemático artificial, assim como seus componentes são apresentados na figura 6.

Figura 6 - Representação do Neurônio Matemático



Fonte: Deep Learning Book, adaptado pelo autor

O neurônio artificial pode ser dividido nos seguintes itens:

- Sinais de Entrada: Semelhante aos dendritos recebem os sinais externos, neste caso $\{X_1, X_2, \dots, X_n\}$, são os dados que alimentam o neurônio artificial.
- Pesos Sinápticos: Representados por $\{W_1, W_2, \dots, W_n\}$, são valores que ponderam cada entrada da rede, sendo estes valores aprendidos ao longo do treinamento (aprendizagem por experiência).
- Combinador Linear $\{\Sigma\}$: Une os sinais de entrada ponderados pelos pesos sinápticos com o intuito de gerar um potência de ativação.
- Limiar de Ativação $\{\Theta\}$: Determina o valor apropriado de resultado para o combinador linear, para dessa maneira gerar um disparo de ativação.

- Potencial de Ativação $\{u\}$: Valor determinado pela diferença entre o combinador linear a e o limiar de ativação. Caso o valor seja positivo $u \geq 0$, o neurônio produz um potencial excitatório, caso contrário, o potencial será inibitório.
- Função de ativação $\{g\}$: É responsável por limitar a saída de um neurônio em um intervalo valores.
- Sinal de saída $\{y\}$: Semelhante ao que ocorre no axônio, o valor de saída pode ser usado como entrada de outros neurônios que estão interligados.

Sintetizando, a arquitetura básica uma rede neural é dividida entre camada de entrada, uma ou mais camadas ocultas e camada de saída que apresenta o resultado pós treinamento da rede.

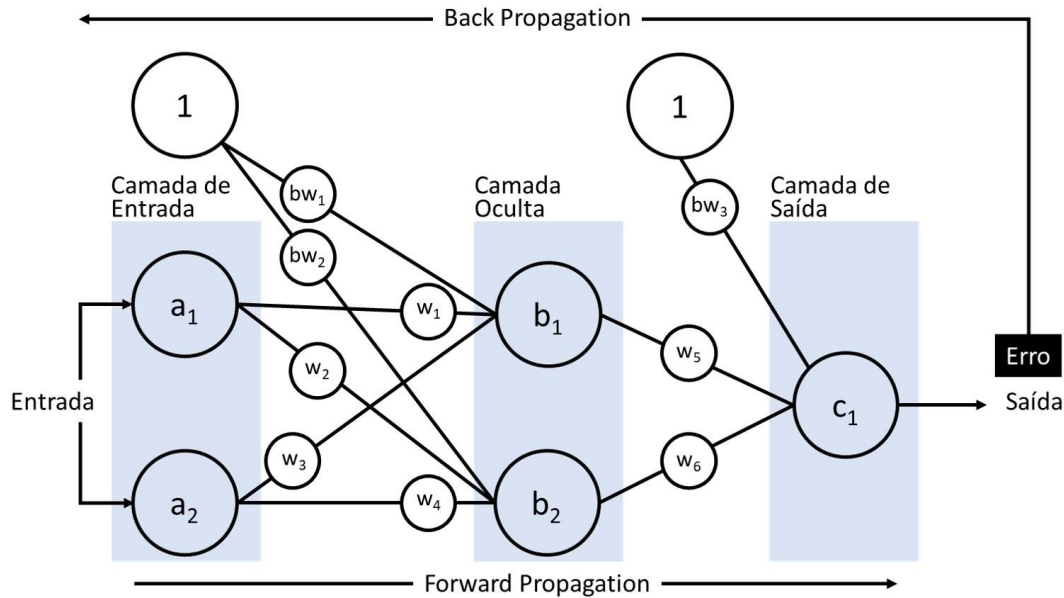
Em 1956 Frank Rosenblatt um neurologista começou a trabalhar no *Perceptron*. O *perceptron*, pode ser compreendido como um procedimento de aprendizado que examina os valores antes da mudança dos respectivos pesos sinápticos. Ao longo dos anos foram sendo inseridas mais camadas ao *perceptron*, dessa maneira criando o *perceptron* multicamadas (*Multilayer Perceptron*)(Rosenblatt, F., 1957).

Em 1959 foi desenvolvido o *ADALINE* (*Adaptive Linear Element*) ou Elemento Linear Adaptativo e o *MADALINE* (*Many ADALINE*) ou muitos Adalines pelos cientistas *Bernard Widrow* e *Marcian Hoff*, de *Stanford*. O modelo foi desenvolvido para reconhecer padrões binários de modo que, se ele estivesse lendo bits de transmissão de uma linha telefônica, poderia prever o próximo bit, já o *MADALINE* foi a primeira rede neural aplicada a um problema do mundo real, usando um filtro adaptativo que elimina ecos nas linhas telefônicas. Apesar do tempo, os sistemas ainda estão em uso comercial (W, Bernard et al., 1960).

Com maior difusão do *perceptron* multicamadas, Em 1982 *David Rumelhart*, propuseram o *Backpropagation*, neste algoritmo os pesos são retro propagados para as camadas anteriores da rede, dessa maneira calculando e atualizando os pesos sinápticos com base no erro que foi gerado. A criação do algoritmo de retropropagação (*Backpropagation*) foi um passo importante para o desenvolvimento das redes profundas e posteriormente das redes neurais convolucionais. Na figura 7 é apresentado tanto uma arquitetura de rede neural tradicional de propagação para frente (*Forward Propagation*) quanto uma arquitetura com a retropropagação

(*Backpropagation*) de seus pesos sinápticos com base no sinal de erro (Rumelhart, D. E., 1986)

Figura 7 - Arquitetura de Rede Neural com Propagação para Frente e Retro propagação



Fonte: Fagundes (2018)

Nas arquiteturas de rede de propagação para frente (*Forward Propagation*), não há a correção dos pesos sinápticos de acordo com o valor de erro, que caracteriza o aprendizado supervisionado. Já nas arquiteturas de rede por retro propagação (*Backpropagation*), existe a verificação do valor que se tem ao final da rede neural, e o valor de erro gerado, enquanto o valor de erro não diminuir a um patamar determinado aceitável haverá uma atualização contínua dos pesos sinápticos.

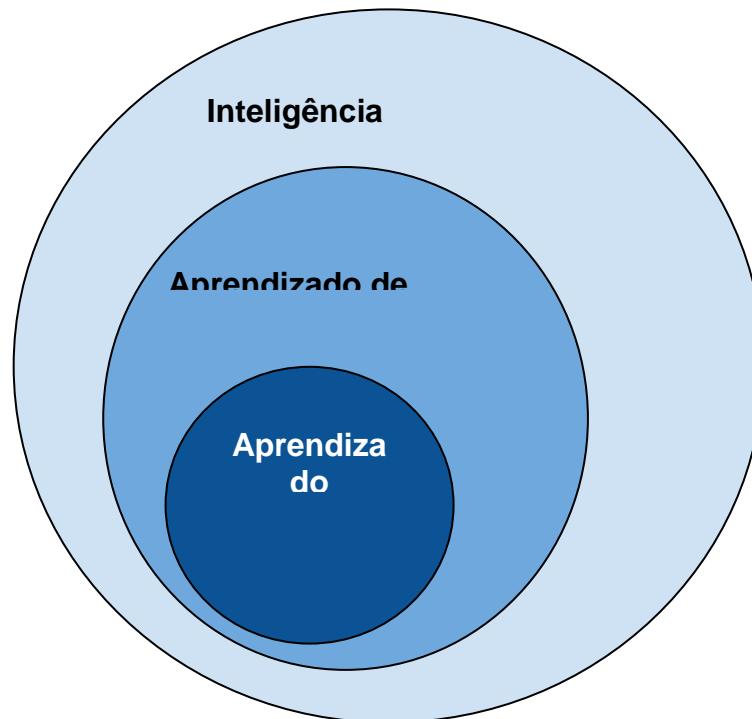
Em 1982, *John Hopfield* da *Caltech* apresentou um documento à Academia Nacional de Ciências (*National Science Foundation*), onde sua abordagem não buscava apenas modelar cérebros, mas criar dispositivos úteis. Já em 1985, o Instituto Americano de Física (*American Institute of Physics*) começou o que se tornou uma reunião anual de Redes Neurais para Computação.

Após esse período, a pesquisa em redes neurais passou por um hiato, uma vez que muitos dos resultados esperados não poderiam ser atingidos com a capacidade de processamento computacional da época, felizmente muitos cientistas continuaram suas pesquisas em redes neurais mesmo com a redução do investimento e interesse em pesquisa nesta área.

2.4 Aprendizado Profundo

O Aprendizado Profundo (*Deep Learning*), é uma subárea da Inteligência Artificial (*Artificial Intelligence*). Na inteligência artificial tem-se também o aprendizado de Máquina (*Machine Learning*), que faz uso de modelos estatísticos para a interpretação de dados. Já o aprendizado profundo emprega algoritmos para processar dados e imitar o processamento feito pelo cérebro humano (Deep Learning Book, Cap 3, 2018). Na figura 8 tem-se uma ilustração de como estas técnicas estão relacionadas no campo da inteligência artificial.

Figura 8 - Representação da Inteligência Artificial e suas subáreas



Fonte: Elaboração do próprio autor

No aprendizado profundo usa-se camadas de neurônios matemáticos para processar dados, compreender a fala humana e reconhecer objetos visualmente. Nesta técnica os dados passam por todas as camadas que estão interligadas. Semelhantemente as arquiteturas de redes neurais primeira camada representa as entradas, enquanto a última representa saída. Cada camada contém normalmente um algoritmo simples e uniforme contendo um tipo de função de ativação.

Uma vez definido o conceito de aprendizado profundo, as redes neurais convolucionais, que são uma das técnicas de aprendizado profundo e foco deste trabalho são apresentadas a seguir.

2.5 Redes Neurais Convolucionais

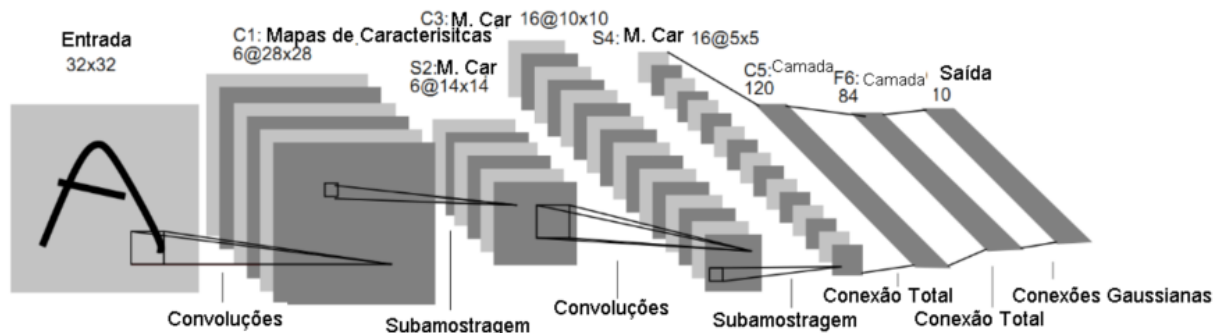
O primeiro modelo de rede neural convolutiva foi usado por *Kunihiko Fukushima* em 1979. *Fukushima* desenvolveu uma rede denominada *Neocognitron*, que usava um design hierárquico e multicamadas. Este design permite ao computador “aprender” a reconhecer padrões visuais. As redes se assemelhavam a versões modernas de redes convolucionais, porém, foram treinadas com uma estratégia de reforço de ativação recorrente em múltiplas camadas. A arquitetura de rede feita por *Fukushima* permitiu que os recursos importantes fossem ajustados manualmente aumentando o “peso” de certas conexões (Deep Learning Book, Cap 3, 2018).

Muitos dos conceitos de *Neocognitron* continuam a ser utilizados. O uso de conexões de cima para baixo e novos métodos de aprendizagem permitiram a realização de uma variedade de redes neurais. Um *Neocognitron* moderno não só pode identificar padrões com informações faltantes (por exemplo, um número 5 desenhado de maneira incompleta), mas também pode completar a imagem adicionando as informações que faltam. Isso pode ser descrito como “inferência”.

Uma das mais importantes pesquisas demonstrando a aplicabilidade das Redes Neurais Convolucionais foi realizado por *Yann Lecun*, após se unir aos laboratórios *AT&T Bell Labs* em 1988. Em seu experimento o autor fez uso das redes neurais convolucionais para reconhecimento de caracteres escritos à mão, assim, validando a hipótese proposta que as redes convolucionais poderiam ser utilizadas em aplicações práticas (Lecun, Y. et al., 1988)

A figura 9 apresenta estrutura da rede convolucional introduzida por *Yann Lecun*. Como entrada tem-se uma imagem em preto e branco de 32x32 pixels, nesta imagem aplica-se um filtro convolucional, onde uma convolução pode ser interpretada como uma operação matemática passando por um sistema ou filtro linear invariante ao tempo. Após o filtro convolucional, é aplicada uma técnica denominada subamostragem (*subsampling*), que reduz a tamanho da imagem, essa rotina se repete até a última camada que é uma rede neural totalmente conectada, com o respectivo número de saídas de acordo com o número de classes.

Figura 9 - Rede Neural Convolutiva proposta por *Lecun, Y*



Fonte: Lecun (1988)

Entretanto, uma limitação que existia era a capacidade de processamento dos computadores, por se tratar de processamento de imagens.

O próximo passo evolutivo significativo para o *Deep Learning* ocorreu em 1999, quando os computadores começaram a se tornar mais rápidos no processamento de dados e Unidades de Processamento de Gráfico ou GPUs (*Graphics Processing Units*) foram desenvolvidas. O uso de GPUs significou um salto no tempo de processamento, resultando em um aumento das velocidades computacionais em 1000 vezes ao longo de um período de 10 anos. Durante esse período, as redes neurais começaram a competir com máquinas de vetor de suporte. Enquanto que num estágio inicial uma rede neural poderia ser lenta em comparação com uma máquina de vetor de suporte, às redes neurais ofereciam melhores resultados usando os mesmos dados. As redes neurais também têm a vantagem de continuar a melhorar à medida que mais dados de treinamento são adicionados.

Uma vez apresentadas as redes neurais, seu histórico e princípio de funcionamento. A seguir trabalhos de destaque nacional e internacional utilizando redes neurais convolucionais para a detecção de plantas daninhas serão apresentados.

2.6 Redes Neurais Convolucionais na Agroindústria

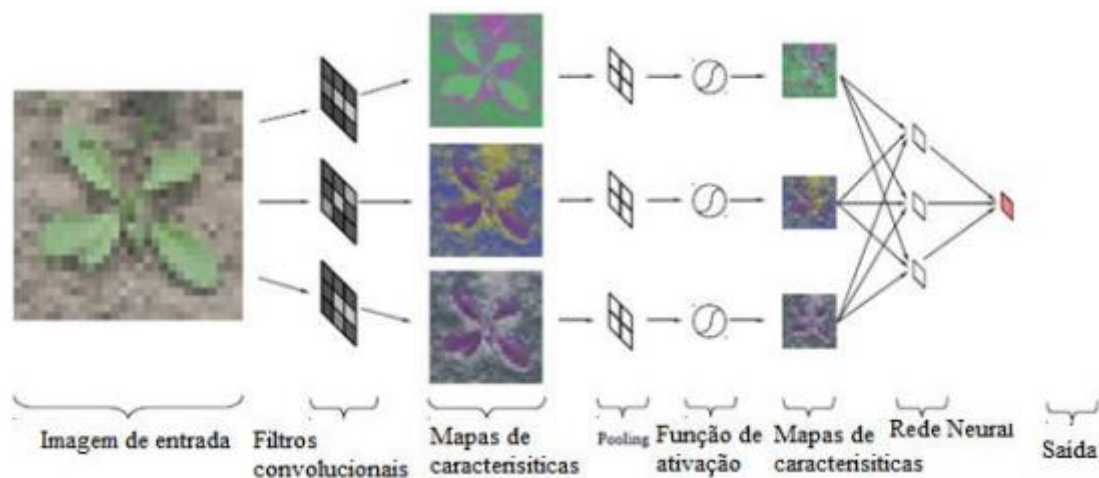
Após o aprimoramento genético das culturas agrícolas, o uso de *Internet* das Coisas e outras ferramentas de vanguarda da Tecnologia da informação, como aprendizado de máquina e grandes dados (*Big Data*) são consideradas a próxima grande revolução agroindustrial. Neste campo fértil de pesquisas e tecnologias, diversas têm sido a aplicação de redes neurais convolucionais na agricultura de precisão. Como este trabalho tem o foco na detecção de plantas daninhas usando

redes convolucionais, 3 trabalhos de destaque nacional e internacional são apresentados, por sua originalidade e contribuição para esta área de pesquisa.

2.6.1 Detecção Automática e Classificação de Plantas Daninhas sob Condições Naturais de Iluminação

Dyrmann (2017) em seu trabalho Detecção Automática e Classificação de Plantas Daninhas sob Condições Naturais de Iluminação (*Automatic Detection and Classification of Weed Seedlings under Natural Light Conditions*) apresenta-se como um dos precursores no uso de redes convolucionais para a detecção de plantas daninhas, em sua tese de Doutorado, em 2017 o autor fez a detecção de diversas espécies de plantas daninhas sob diferentes níveis de iluminação e sobreposição de folhas, ainda propondo uma arquitetura própria de rede neural convolucional. A estrutura do algoritmo utilizado pelo autor e apresentado na figura 10.

Figura 10 - Arquitetura de Aprendizagem Profunda Para Classificação de Plantas Daninhas



Fonte: Dyrmann (2017)

A arquitetura inicia-se com a aquisição das imagens para a construção do banco de dados, essas imagens em padrão RGB (*Red, Green, Blue*), também são as imagens de entrada da rede neural (*Input Image*), neste trabalho o autor coletou estas imagens manualmente utilizando um celular *Nokia Lumia 1020* e por meio de um Veículo Terrestre ATV (*All Terrain Vehicle*), com um computador integrado (*Nvidia TX1*), para a coleta automática das imagens. No total foram coletadas 13.976 imagens, 4.537 foram anotadas ao longo de 63 dias e três estações de crescimento. Essas 4.537 imagens foram utilizadas como entrada da rede neural.

Após a etapa de coleta e criação do banco de dados, filtros convolucionais foram aplicados nas imagens. Por exemplo, uma imagem de dimensões 224x224x3, tendo respectivamente 224 pixels de altura, 224 pixels de largura e 3 pixels de profundidade representando na camada de profundidade as cores vermelho, verde e azul, feita a aplicação de um filtro convolucional terá as dimensões 224x224x64, destacando as características principais em sua profundidade e assim gerando mapas de características.

Uma vez gerados os mapas de características é aplicada uma técnica de compressão (*Pooling*), assim, uma imagem com dimensões 224x224x64 após a compressão, terá as dimensões 112x112x128, dessa maneira, comprimindo a altura, largura e aumentando a profundidade, sendo etapa essencial para a extração e aquisição de características relevantes das imagens.

Após a etapa de compressão uma função de ativação é aplicada para inserir não linearidade a rede, tornando-a mais eficiente na extração de informações, as principais funções de ativação utilizadas são a ReLu (*Rectified linear unit*) ou Retificador de Unidade Linear e a função Sigmoid. As etapas anteriores são repetidas até a imagem ser utilizada como entrada de uma rede neural completamente conectada (*Fully Connected Neural Network*) para a classificação das plantas daninhas.

Ao final do trabalho o autor conclui que realizou a classificação de 17 espécies de plantas daninhas mais frequentes com uma exatidão total de 87% de maneira automática, o autor destaca também que o método representa uma economia importante para os agricultores em relação ao uso de herbicidas, uma vez que pode-se classificar plantas daninhas sob condições de iluminação natural.

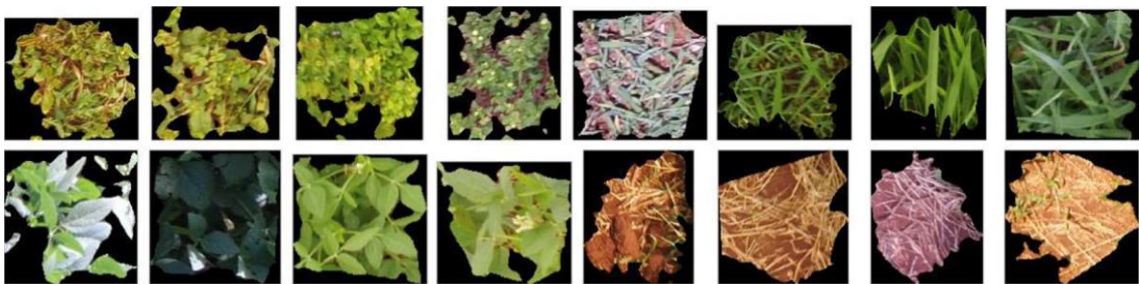
2.6.2 Detecção de Plantas Daninhas em Plantações de Soja usando Redes Convolucionais

Dos Santos Ferreira, A. (2017), em sua dissertação de mestrado que posteriormente resultou no artigo Detecção de Plantas Daninhas em Plantações de Soja usando Redes Convolucionais (*Weed detection in soybean crops using ConvNets*), fez uso de das redes neurais convolucionais para a classificação entre soja, solo, plantas daninhas de folha larga e folha estreita. Neste trabalho o autor criou um banco de imagens com mais de 15.000 imagens, além de ter realizado a

coleta com um VANT (Veículo Aéreo Não tripulado) *DJI Phantom3* a altura média de 2 metros do solo. O trabalho destaca-se principalmente por ser uma das primeiras pesquisas usando redes convolucionais para a detecção e classificação de plantas daninhas no Brasil.

O autor também realizou a segmentação das imagens do plano de fundo utilizando o algoritmo SLIC (*Simple Linear Iterative Clustering Superpixels* ou *Simple Clusterização Linear Iterativa de Superpixels*). A figura 11 apresenta algumas das imagens já segmentadas utilizadas no algoritmo.

Figura 11 - Imagens Segmentadas Utilizadas para Classificação de Plantas Daninhas



Fonte: Dos Santos Ferreira, A. (2017)

Para a etapa de validação o autor utilizou 1000 imagens divididas respectivamente em 250 imagens de solo, 250 de soja, 250 plantas daninhas de folha larga, 250 plantas daninhas de folha estreita. O melhor resultado obtido foi para o solo com 100% de acerto e o pior plantas daninhas de folha estreita com 98,4% de exatidão e 246 imagens classificadas corretas.

O trabalho destaca-se por ser um dos precursores no Brasil em relação ao uso de aprendizado profundo para a determinação e classificação de plantas daninhas.

2.6.3 Segmentação Semântica em Tempo Real de Plantações e Plantas Daninhas para Robôs de Agricultura de Precisão com o Conhecimento de Plano de Fundo em CNNs

Milioto em (2018) em seu trabalho *Segmentação Semântica em Tempo Real de Plantações e Plantas Daninhas para Robôs de agricultura de Precisão com o Conhecimento de Plano de Fundo em CNNs (Real-time Semantic Segmentation of Crop and Weed for Precision Agriculture Robots Leveraging Background Knowledge in CNNs)* aplicou as redes neurais convolucionais visando a automação na detecção e eliminação de plantas daninhas, neste trabalho o autor fez a classificação entre a

beterraba-sacarina em estágio de crescimento inicial, plantas daninhas e plano de fundo, para isso o autor fez uso de uma técnica denominada segmentação semântica (*Semantic Segmentation*) o que permitiu a detecção em tempo real das plantas daninhas a uma frequência de 20 frames por segundo.

O autor utilizou 15.197 de três locais distintos, as imagens foram coletadas em RGB e pelo Índice de Vegetação por Diferença Normalizada NDVI (*Normalized Difference Vegetation Index*), além de 14 índices usados como entradas da rede neural. Para efetuar o treinamento e detecção em tempo real foi utilizado um computador com processador Intel i7 e placa gráfica NVIDIA GTX1080Ti, o computador foi embarcado em uma plataforma denomina *Bonirob* (*Bosch*). Na figura 12 pode-se ver tanto a plataforma *Bonirob* quanto as imagens que são processadas.

Figura 12 - Plataforma de Coleta de Imagens *Bonirob* e Plantas Analisadas

Figura 12 - Plataforma de Coleta de Imagens *Bonirob* e Plantas Analisadas



Fonte: Milioto et al (2018)

O trabalho faz parte de uma iniciativa da União Européia denominada *Flourish*, provando-se eficiente na detecção e eliminação de plantas daninhas em tempo real, uma vez que o *Bonirob* conta com pistões pneumáticos que eliminam mecanicamente as plantas daninhas, atualmente o sistema trabalha com a frequência de 20hz.

Os 3 três trabalhos trouxeram importantes contribuições sobre a aplicação das redes neurais convolucionais na detecção e classificação de plantas daninhas, o trabalho de *Dyrmann* destaca-se pela classificação das plantas daninhas sob condições naturais de iluminação e sobreposição de folhas, demonstrando como as redes neurais profundas são confiáveis, mesmo em cenários difíceis.

O trabalho de Dos Santos Ferreira destaca-se por ser um dos precursores no uso de aprendizado profundo para a classificação de plantas daninhas no Brasil, além de ter obtido resultados consistentes na classificação no banco de imagens criado, o autor demonstra o potencial das redes neurais convolucionais para a classificação de plantas daninhas em plantações de soja, que atualmente é a cultura mais atingida, e também é a que consome mais herbicidas no Brasil.

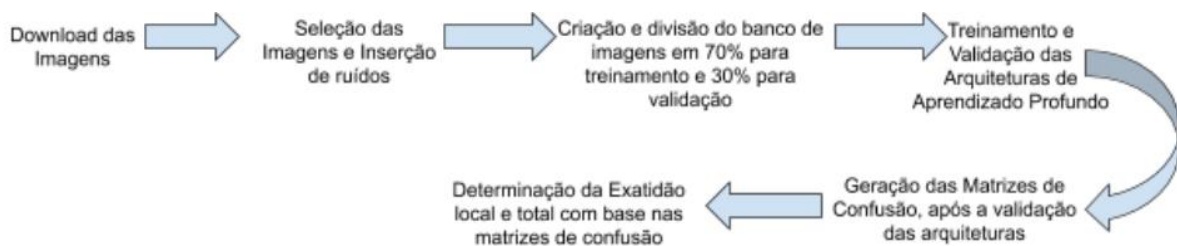
No trabalho de *Milioto* pode-se ver como será uma plataforma robótica que utiliza as redes neurais convolucionais para a classificação entre a cultura plantada, plantas daninhas e solo e que concomitantemente realiza a eliminação das plantas daninhas. Assim, este trabalho engloba tanto o desenvolvimento quanto a aplicação do algoritmo no campo para a correta eliminação de plantas daninhas.

No seguinte capítulo os materiais, como softwares, o banco de imagens e métodos como as arquiteturas de redes neurais convolucionais são apresentados.

3 MATERIAIS E MÉTODOS

Para a execução das arquiteturas de aprendizado profundo e geração de resultados, seguiu-se uma ordem de etapas, iniciando com o *download* das imagens, seleção e inserção de ruídos, criação e divisão do banco de imagens para treinamento e validação, e finalizando com a criação das matrizes de confusão e determinação da exatidão local e total de cada arquitetura. A figura 13 exemplifica o fluxo deste processo descrito.

Figura 13 - Fluxo metodológico de etapas para a aplicação das arquiteturas de aprendizado profundo e geração dos resultados



Fonte: Elaboração do próprio autor

Nos itens a seguir todos os elementos principais utilizados para a aplicação dos algoritmos de aprendizado profundo, parametrização das arquiteturas e criação do banco de imagens são descritos em maiores detalhes.

3.1 Biblioteca *Tensorflow*

Criado pela equipe do *Google Brain*, a biblioteca *TensorFlow* é uma biblioteca de código aberto para computação numérica e aprendizado de máquina em larga escala. O *TensorFlow* reúne vários modelos e algoritmos de aprendizado de máquina e aprendizagem profunda (também conhecido como redes neurais). É feito uso da linguagem de programação *Python* para fornecer uma API (*Application Programming Interface*) ou Interface de Programação de Aplicativos conveniente para construir aplicativos com a estrutura, enquanto executa-se esses aplicativos em C++ para alto desempenho (Abadi, M. et al., 2015)

Por meio do *TensorFlow* pode-se treinar e executar redes neurais profundas para classificação manuscrita de dígitos, reconhecimento de imagens, incorporação de palavras, redes neurais recorrentes, processamento de linguagem natural e simulações baseadas em equações diferenciais parciais. O *TensorFlow* também

suporta previsão de produção em escala, com os mesmos modelos usados para treinamento.

Possibilita a criação de gráficos de fluxo de dados, que descrevem como os dados se movem através de um gráfico ou uma série de nós de processamento. Cada nó no gráfico representa uma operação matemática e cada conexão ou borda entre os nós é uma matriz de dados ou um tensor multidimensional. Isso é fornecido por meio da linguagem *Python*. O *Python* fornece maneiras convenientes de expressar abstrações de alto nível podem ser acopladas.

As operações matemáticas reais, no entanto, não são executadas no *Python*. As bibliotecas de transformações que estão disponíveis através do *TensorFlow* são escritas como binários C++ de alto desempenho. O *Python* apenas direciona o tráfego entre as partes e fornece abstrações de programação de alto nível para conectá-las. Os modelos resultantes criados pelo *TensorFlow*, podem ser implantados na maioria dos dispositivos em que serão usados para exibir previsões. Para este trabalho, foram treinados modelos contidos no *Keras* e o *TensorFlow* foi utilizado como plano de fundo.

3.2 Biblioteca *Keras*

Keras é definida como uma API (*Application Programming Interface*) ou Interface de Programação de Aplicativos de alto nível para redes neurais, capaz de executar bibliotecas como o *TensorFlow*, que é uma biblioteca aberta para computação numérica de alta performance. Diferente do *TensorFlow* onde a modelagem da rede neural é feita graficamente, o *Keras* possibilita a construção modular, linear e sequencial das redes neurais convolucionais (Chollet., 2015).

O *Keras* propriamente não faz suas próprias operações de baixo nível, como produtos tensoriais e convoluções; Ele depende de um mecanismo de plano de fundo para isso, seu plano de fundo primário (e padrão) é o *TensorFlow*, a API *Keras* vem empacotada no *TensorFlow* como *tf.keras*.

A API foi criado para ser modular, fácil de entender e trabalhar com o *Python*. Camadas neurais, funções de custo, otimizadores, esquemas de inicialização, funções de ativação e esquemas de regularização são todos módulos independentes

que podem ser combinados para criar novos modelos. Os modelos são definidos em código *Python*, não em arquivos de configuração de modelo separados.

Além da simplicidade estrutural, através do *Keras* é possível importar arquiteturas de redes neurais convolucionais pré-treinadas, dessa maneira aprimorando a etapa de treinamento da rede, assim obtendo-se maior precisão com menor tempo de processamento. O quadro 3 apresenta as arquiteturas de redes neurais convolucionais pré-treinadas que foram utilizadas para este trabalho. Originalmente estas arquiteturas foram treinadas com o banco de imagens do *ILSVRC (Imagenet Large Scale Visual Recognition Challenge)* ou Imagenet Desafio de Reconhecimento Visual em Larga Escala (Deng, J. et al ., 2009).

Quadro 3 - Arquiteturas de Redes Neurais Profundas Armazenadas no *Keras* e usadas neste trabalho

Modelo	Tamanho	Exatidão Top-5	Parâmetros	Profundidade
VGG16	528 MB	0,901	138.357.544	23
ResNet50	99 MB	0,921	25.636.712	168
InceptionV3	92 MB	0,937	23.851.784	159
InceptionResnetV2	215 MB	0,953	55.873.736	572

Fonte: Elaboração do próprio autor

Quatro arquiteturas de redes neurais convolucionais foram treinadas no banco de imagens criado para este trabalho, seguindo a ordem cronológica do lançamento das arquiteturas treinou-se: *VGG16*, *ResNet50*, *InceptionV3* e *InceptionResnetV2*.

A rede *VGG16* possui mais parâmetros e maior tamanho, assim tendo um tempo de treinamento maior que as demais redes. Em contrapartida a rede *InceptionV3* é a rede com menos parâmetros e menor tamanho também, tendo seu tempo de treinamento reduzido. A rede mais profunda é a *InceptionResnetV2* com 572 camadas e menos da metade do tamanho e número de parâmetros da rede *VGG16*.

O item exatidão do Top-5 significa que qualquer uma das 5 respostas de probabilidade mais alta da arquitetura deve corresponder à resposta esperada. Estes resultados foram obtidos aplicando essas arquiteturas no banco de imagens do *ILSVRC*, sendo que a arquitetura *InceptionResnetV2* apresenta a melhor performance com 0,953 ou 95,3% seguida pela *InceptionV3* com 93,7%, *ResNet50* 92,1% e *VGG16*

com 90,1%. espera-se portanto obter resultados e uma ordem semelhantes aos obtidos pelas arquiteturas anteriormente.

Nos itens a seguir detalhes estruturais das arquiteturas pré-treinadas no ILSVRC, que foram utilizadas neste trabalho, assim como sua performance, construção e aplicação são descritos.

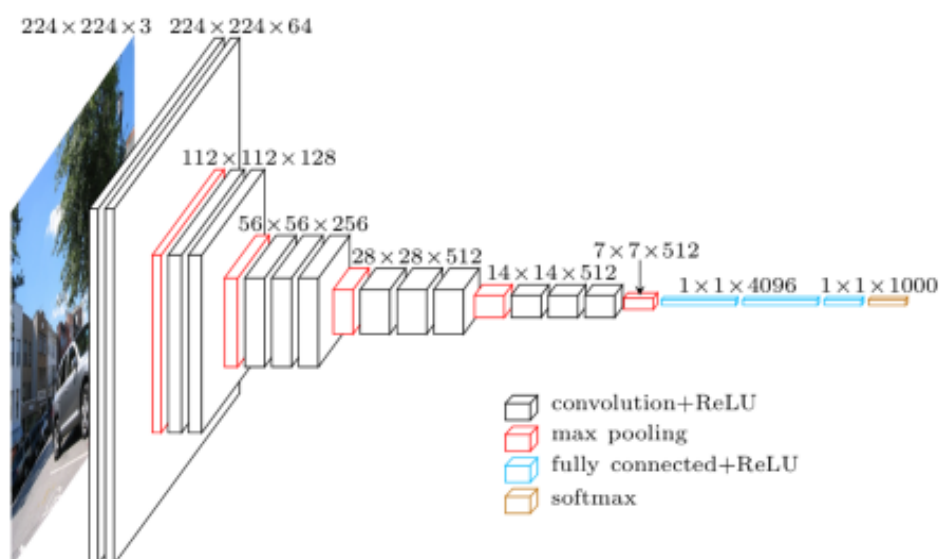
3.3 Arquiteturas de Redes Neurais Convolucionais

Desde os resultados obtidos por *Krizhevsky* em 2012, diversas foram as arquiteturas de redes neurais convolucionais desenvolvidas (Krizhevsky., 2012). Para este trabalho foram selecionadas 4 arquiteturas de rede, cujos pesos foram treinados com base no (*ILSVRC*) (Deng, J. et al ., 2009) e encontram-se disponíveis por meio da aplicação *Keras*, são estas redes:

3.3.1 VGG16: Redes Convolucionais Muito Profundas para Reconhecimento de Imagem em Grande Escala

Em 2015 (Simonyan, K; Zisserman, A. 2015), em seu trabalho (*VGG16: Very Deep Convolutional Networks for Large-Scale Image Recognition*) ou *VGG16*: Redes Convolucionais Muito Profundas para Reconhecimento de Imagem em Grande Escala, propuseram uma arquitetura de rede neural convolucional profunda, tendo 2 variantes *VGG16* e *VGG19*. A figura 14 representa a construção e estrutura desta arquitetura.

Figura 14 - Arquitetura VGG16



Fonte: K. Simonyan; A. Zisserman (2015)

A arquitetura tem como entradas imagens em RGB tendo dimensões de 224x224x3 pixels, que representa respectivamente altura, largura e profundidade da imagem. Logo em de início é aplicado um filtro Convolutacional e uma função de ativação de um Retificador de Unidade Linear (*Convolutional+ReLU*), tendo-se como resultado uma imagem de dimensões 224x224x64 pixels, aumentando a profundidade da imagem que está sendo analisada. Em seguida é aplicado um filtro para a redução de dimensionalidade da imagem, no exemplo acima após a aplicação do filtro de máximo local (*max pooling*) a entrada tem suas dimensões reduzidas de 224x224x64 para 112x112x128, pode-se perceber que há uma diminuição na altura e comprimento da imagem, porém não na profundidade, pelo contrário ao longo dos filtros convolucionais e os filtros de máximo local a imagem torna-se mais profunda. Isso deve-se pelo fato de que as características ou informações mais importantes sobre as imagem estão em sua profundidade e ainda visa evitar o sobreajuste, que é um termo usado em estatística para descrever quando um modelo estatístico se ajusta muito bem ao conjunto de dados anteriormente observado, mas se mostra ineficaz para prever novos resultados.

Após repetidos filtros convolucionais, funções de ativação de retificador de unidade linear chega-se a penúltima etapa que é uma rede neural completamente conectada e em sua saída um retificador de unidade linear nesta penúltima etapa a imagem tem dimensões 1x1x4096. Por fim é aplicada uma função de ativação *softmax*, na matemática, a função softmax usa um vetor não normalizado e a normaliza em uma distribuição de probabilidade, em aprendizado profundo a função *softmax* é usada para classificação. No caso das imagens do *ILSVRC* significa classificar as imagens entre as 1000 classes do desafio.

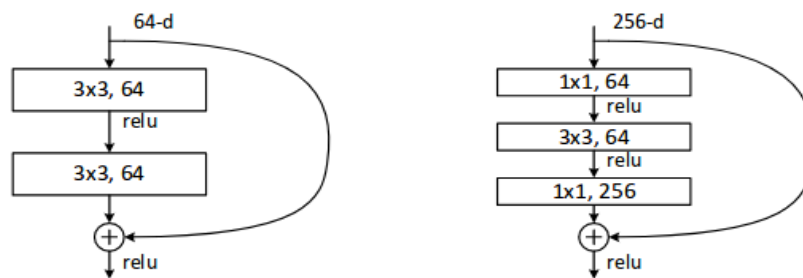
Para este trabalho foram usadas as dimensões recomendadas para a arquitetura VGG16, então todas as imagens de entrada desta rede tinham dimensões 224x224x3 pixels.

3.3.2 ResNet-50: Aprendizagem Residual Profunda para Reconhecimento de Imagem

Em 2016 (He. et al., 2016), propuseram uma arquitetura de rede neural profunda residual (*Resnet*) em seu artigo (*Deep Residual Learning for Image Recognition*) ou Aprendizagem Residual Profunda para Reconhecimento de Imagem. Nesta

arquitetura, os blocos residuais foram propostos com o propósito de resolver o seguinte problema, com o aumento da profundidade da rede, a exatidão fica saturada e depois se degrada rapidamente. Assim como a rede *VGG16* a rede *Resnet* tem como entradas imagens de dimensão $224 \times 224 \times 3$ pixels. A ideia central da *ResNet* é introduzir a chamada “conexão de atalho de identidade” que pula uma ou mais camadas, como mostra a figura 15.

Figura 15 - Bloco Residual Rede *Resnet*



Fonte: He, K. et al. (2016)

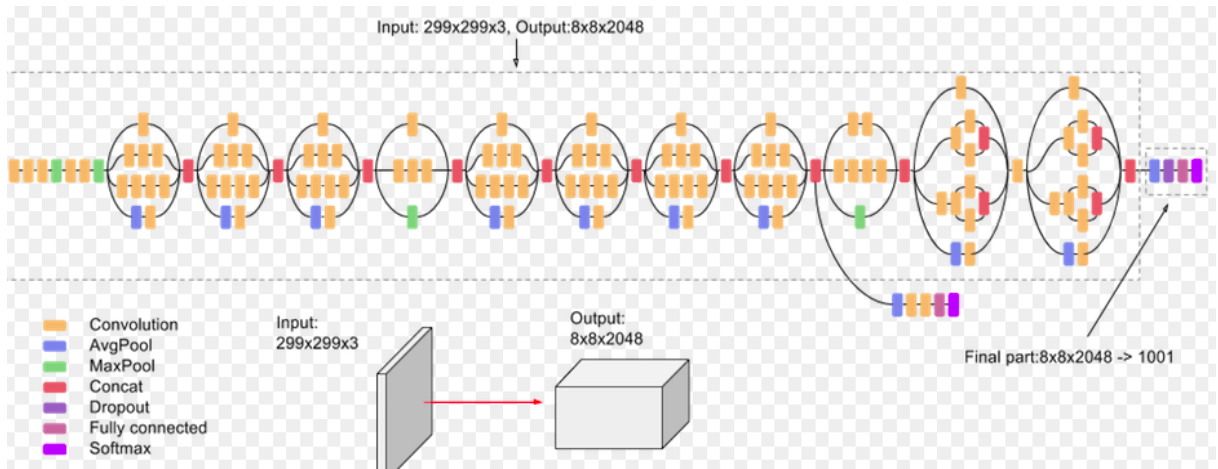
Os atalhos de identidade podem ser usados diretamente quando a entrada e saída são das mesmas dimensões. Quando as dimensões mudam: A) O atalho ainda executa o mapeamento de identidade, com entradas extras de zero preenchidas com a dimensão aumentada; B) O atalho de projeção é usado para coincidir com a dimensão.

Cada bloco do *ResNet* tem 2 camadas de profundidade (usado em redes pequenas como *ResNet* 18, 34) ou 3 camadas de profundidade (*ResNet* 50, 101, 152). Os demais componentes da arquitetura mantêm-se como os descritos anteriormente na arquitetura *VGG16*, tendo ao final uma função de ativação *softmax* para realizar a classificação.

3.3.3 InceptionV3: Repensando a Arquitetura Inception para Visão Computacional

A *InceptionV3* (Szegedy, C. et al., 2016) é um modelo de reconhecimento de imagem amplamente utilizado, o modelo é o culminar de muitas ideias desenvolvidas por vários pesquisadores ao longo dos anos. É baseado no artigo original: “Repensando a Arquitetura Inception para Visão Computacional (*Rethinking the Inception Architecture for Computer Vision*)” de Szegedy, et. al. A figura 16 ilustra a arquitetura da rede.

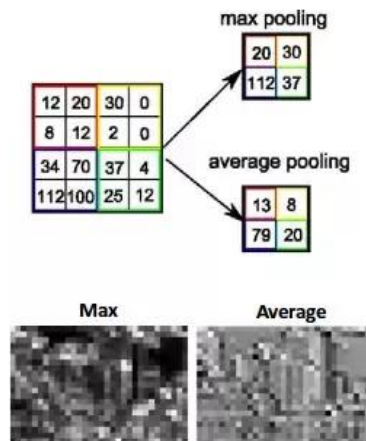
Figura 16 - Arquitetura InceptionV3



Fonte Szegedy, C. et al. (2015)

Nesta arquitetura tem-se como entrada imagens de dimensões 299x299x3 pixels, o principal diferencial desta arquitetura foi a introdução de módulos Inception, que introduziu a concatenação de diferentes filtros convolucionais (*Concat*) de distintos tamanhos, por exemplo, filtros 5x5, 3x3 ou até 1x1, fazendo parte de um módulo Inception. Foi introduzido também o módulo abandono (*Dropout*), o abandono é uma técnica de regularização patenteada pelo Google para reduzir o sobreajuste (*overfitting*) em redes neurais, evitando co-adaptações complexas nos dados de treinamento. É uma maneira muito eficiente de executar a média do modelo com redes neurais. Também usa-se agrupamento médio (*AvgPool*). O agrupamento máximo (*MaxPool*) extrai os características mais importantes da imagem, como as bordas, enquanto o agrupamento médio (*AvgPool*) extrai os recursos de maneira mais suave. A figura 17 mostra a diferença entre o agrupamento máximo e médio.

Figura 17 - Diferença entre Agrupamento Médio e Máximo

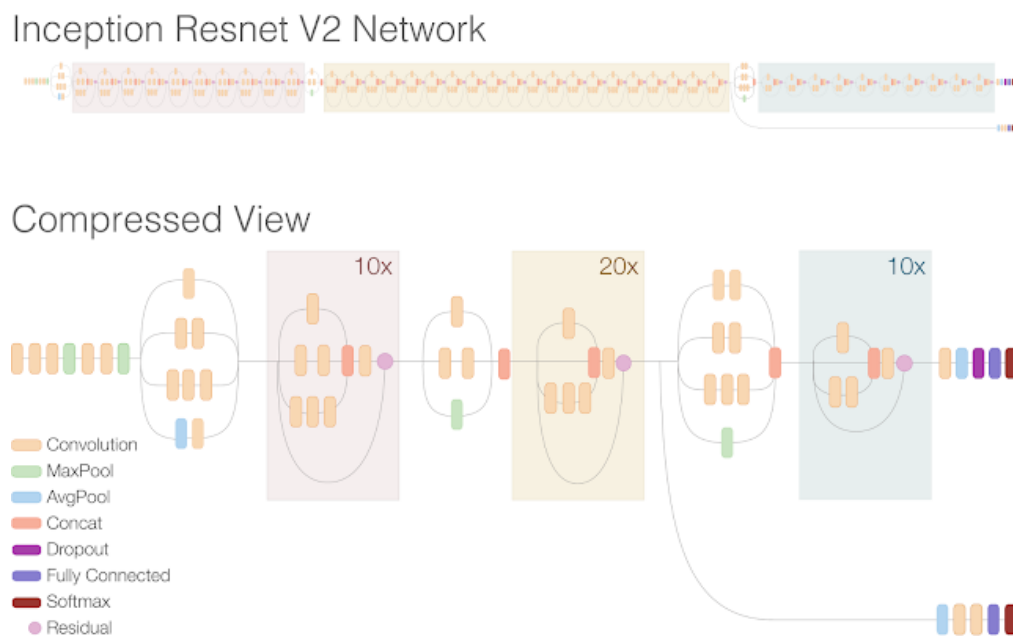


Fonte: Rahman, N (2017)

3.2.4 Inception-v4, Inception-ResNet e o Impacto das Conexões Residuais no Aprendizado

Inception-ResNet-v2 é baseada no artigo *Inception-v4, Inception-ResNet e o Impacto das Conexões Residuais no Aprendizado* ou (*Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning*) (Szegedy, C; Ioffe, S; Vanhoucke, V., 2016) é uma variação do modelo *InceptionV3*, com alguns dos itens da rede *ResNet*. Como a rede *InceptionV3* as imagens de entrada possuem dimensão 299x299x3, na figura 18 é exemplificada a construção da rede.

Figura 18 - Arquitetura *InceptionResnetV2*



Fonte: Szegedy, C; Ioffe, S; Vanhoucke, V. (2016)

Como principal inovação desta arquitetura em comparação com a *InceptionV3* destaca-se a inclusão de um módulo residual, realizando a operação descrita anteriormente na arquitetura *ResNet50*. Todos os modelos citados acima podem ser encontrados no *Keras*, assim possibilitando o uso de redes pré-treinadas.

3.4 Obtenção do Banco de Imagens

Para a aplicação das arquiteturas de redes neurais descritas, criou-se um banco de imagens. Como é intenção deste trabalho analisar a capacidade das redes de extrair informações relevantes e efetuar a classificação entre as 5 espécies de plantas daninhas de maneira eficaz, foi realizada a obtenção de diferentes imagens das 5 espécies. Para realizar esta descarga (*download*) aplicou-se um algoritmo para

que automaticamente descarrega-se as imagens baixadas com base em seu nome científico que foi pesquisado no buscador *Google* Imagens.

O banco de dados inicial continha centenas de imagens de cada espécie, foi realizada uma filtragem manual destas imagens uma vez que muitas das imagens baixadas não tinham conexão com as espécies de planta daninhas que serão classificadas. Após as etapas de descarga e filtragem das imagens foi inserido ruídos artificialmente nas imagens.

3.5 Inserção de Ruídos

Foi inserido artificialmente ruídos nas imagens baixadas por dois motivos, o primeiro é de fornecer mais dados (imagens) para o treinamento. A técnica do aumento dos dados através da inserção de ruído é conhecida como aumento de dados ou (*data augmentation*). Essa técnica possibilita o aumento do banco de imagens, assim, permitindo que a arquitetura obtenha resultados mais consistentes tanto em seu treinamento quanto na validação.

O segundo motivo trata-se de treinar as arquiteturas aqui estudadas sob imagens em diferentes condições, como rotação, embasamento, diferentes planos de fundo, iluminação e diferentes dimensões de objetos. Assim tornando as redes invariantes quanto às mudanças de iluminação, rotação e dimensionamento, tornando-as menos propensas a falhas na classificação ou até mesmo sobreajuste. Exemplos de imagens com e sem ruídos são apresentados nas figuras 19 e 20.

Figura 19 - Comparação entre uma imagem da espécie Capim Azevém sem ruído (a), e com ruído inserido (b)



(a)



(b)

Fonte: Elaboração do próprio autor

Figura 20 - Comparação entre uma imagem da espécie Capim Pé de Galinha sem ruído (a), e com ruído inserido (b)



(a)



(b)

Fonte: Elaboração do próprio autor

Na figura 19 tem-se uma imagem da espécie Capim Azevém sem ruídos (a) e ao seu lado outra imagem com inserção de ruídos (b), percebe-se também na figura 20 uma planta daninha da espécie Capim pé de Galinha sem inserção de ruídos (a) e com inserção de ruídos (b). Cabe salientar que não foi aplicada técnica alguma de segmentação (separação do objeto de interesse do plano de fundo), ou seja as arquiteturas de rede são alimentados com os dados brutos, o que torna a tarefa mais dificultosa para a classificação, uma vez que cabe a rede distinguir entre a espécie de planta daninha e todo o plano de fundo. Ao total foram utilizadas 3.500 imagens divididas entre as 5 classes (Buva, Capim Amargoso, Capim Azevém, Capim pé de Galinha e Caruru) para o treinamento e 1.500 imagens divididas entre as 5 classes para a validação, totalizando 5.000 do banco de imagens criado.

3.6 Parametrização do Algoritmo

Para a aplicação das 4 arquiteturas de redes no banco de imagens descrito anteriormente, foi implementado um algoritmo de aprendizado profundo com o intuito de atingir-se os objetivos propostos. O algoritmo foi desenvolvido e implementado na linguagem de programação *Python*, sendo que é constituído de diversos itens, tendo destaque:

- 1) Número de Épocas: Inicialmente todas as arquiteturas foram treinadas com 20 épocas. Cada época representa um ciclo de treinamento, desta maneira as 3.500 imagens selecionadas para o treinamento foram treinadas uma vez, assim, esse processo repetiu-se 20 vezes.

- 2) Taxa de Aprendizado: Foi utilizada uma taxa de aprendizado constante de 0,0001. A taxa de aprendizado determina a velocidade do treinamento da rede neural, uma taxa muito alta, pode causar a perda de dados substanciais. enquanto uma taxa muito baixa poderá deixar o aprendizado da rede estagnado.
- 3) Comprimento e Largura das imagens: Nesta seção determina-se o comprimento e altura das imagens, como foi visto anteriormente as imagens variam de 224x224 a 299x299 de comprimento e altura de acordo com a arquitetura que foi utilizada.
- 4) Taxa de Ajuste Fino: O ajuste fino é um conceito de aprendizagem de transferência. Aprendizagem de transferência é uma técnica de aprendizado de máquina, em que o ganho de conhecimento durante o treinamento em um tipo de problema é usado para treinar em outra tarefa ou domínio relacionado. Nesse caso possibilitando que as arquiteturas utilizem pesos pré-treinados, com base nos resultados obtidos anteriormente na competição *ILSVRC*.
- 5) Perda de Entropia Cruzada Categórica: Para determinar a perda foi utilizado a Perda de Entropia Cruzada Categórica, que é própria para a determinação de perda em classificação de várias classes (Espécies de Plantas Daninhas). Também conhecida como perda de log usa-se essa métrica para medir o desempenho de um modelo de classificação. A perda de entropia cruzada aumenta à medida que a classe prevista diverge do rótulo real.
- 6) Matriz de confusão: Ao final do treinamento e validação uma matriz de confusão é gerada para cada arquitetura, possibilitando principalmente verificar a exatidão de cada arquitetura.

A seguir são apresentadas tanto as matrizes de confusão geradas para cada arquitetura, quanto a exatidão, perda e os resultados gerais das arquiteturas analisadas.

4 RESULTADOS E DISCUSSÕES

Para a obtenção dos resultados, 4 arquiteturas de redes neurais convolucionais foram treinadas em 3.500 imagens. A seguir os gráficos criados a partir dos dados das 20 épocas de treinamento são apresentados. Estes gráficos foram criados com base nas informações que cada arquitetura exibe a cada época, os dados são perda e exatidão referente a etapa de treinamento com 3.500 imagens, perda de validação e exatidão de validação referente a validação das 1.500 imagens distribuídas entre as 5 espécies de plantas daninhas.

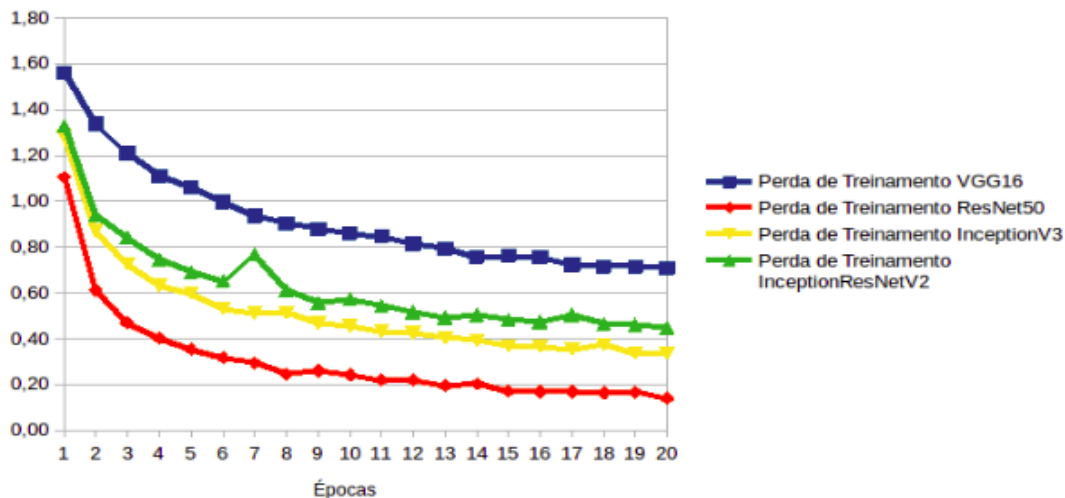
A primeira métrica computada e apresentada para as 4 arquiteturas é a perda na etapa de treinamento. Para o cálculo da perda utilizou-se a função de Perda de Entropia Cruzada Categórica, que é própria para a determinação de perda em classificação de várias classes. Esta função pode ser definida de acordo com a seguinte equação.

$$P = -\frac{1}{n} \sum_x [y \ln a + (1 - y) \ln(1 - a)]$$

Nessa equação a saída do neurônio é $\mathbf{a} = \sigma(\mathbf{z})$, onde $\mathbf{z} = \sum \mathbf{w}_j \mathbf{x}_j + \mathbf{b}$ é a soma ponderada das entradas. O número total de itens de dados de treinamento é representado por n , sendo somadas todas as entradas de treinamento da rede neural em \mathbf{x} , tendo \mathbf{y} como saída desejada. A função de perda por entropia cruzada categórica comporta-se da seguinte maneira, à medida que a arquitetura de aprendizado profundo se torna mais exata na determinação das distintas classes de plantas daninhas a perda de entropia cruzada categórica diminui tendendo ao zero. Por outro lado caso a arquitetura passe a falhar na classificação a perda de entropia cruzada categórica aumentará, isso ocorre devido a uma penalização desta função de perda. Caso fosse necessário fazer a classificação entre 2 classes poderia-se utilizar uma função de Perda de Entropia Cruzada Binária.

Para todas as arquiteturas a perda de entropia cruzada categórica inicia-se com valores acima de 1. Esta métrica é importante principalmente para compreender como a rede se comportou na classificação, uma vez que uma perda alta, ou seja igual ou acima de 1 resultará em uma classificação falha. Na figura 21 são apresentados os valores de perda ao longo das 20 épocas de treinamento para as arquiteturas.

Figura 21 - Gráfico de Perda de Treinamento das Arquiteturas

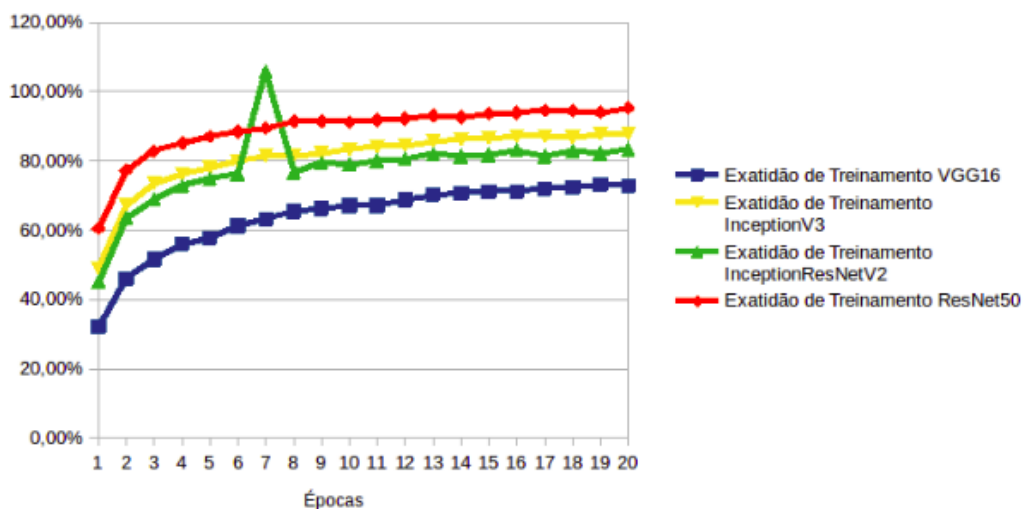


Fonte: Elaboração do próprio autor

Os resultados de perda distinguem-se entre as arquiteturas, enquanto que para a arquitetura *VGG16* o mínimo de perda foi de 0,7088, na arquitetura *ResNet50* teve-se um mínimo de perda de 0,1407. Foi percebido também uma queda drástica no valor de perda na rede *ResNet50*, na primeira época apresentava perda de 1,1040 e na segunda época 0,6134. As arquiteturas *InceptionV3* e *InceptionResNetV2* apresentaram uma diminuição no valor de perda gradual, tendo como mínimo de perda respectivamente 0,3350 e 0,4460.

Na figura 22 a exatidão de treinamento das arquiteturas de redes neurais convolucionais é apresentada.

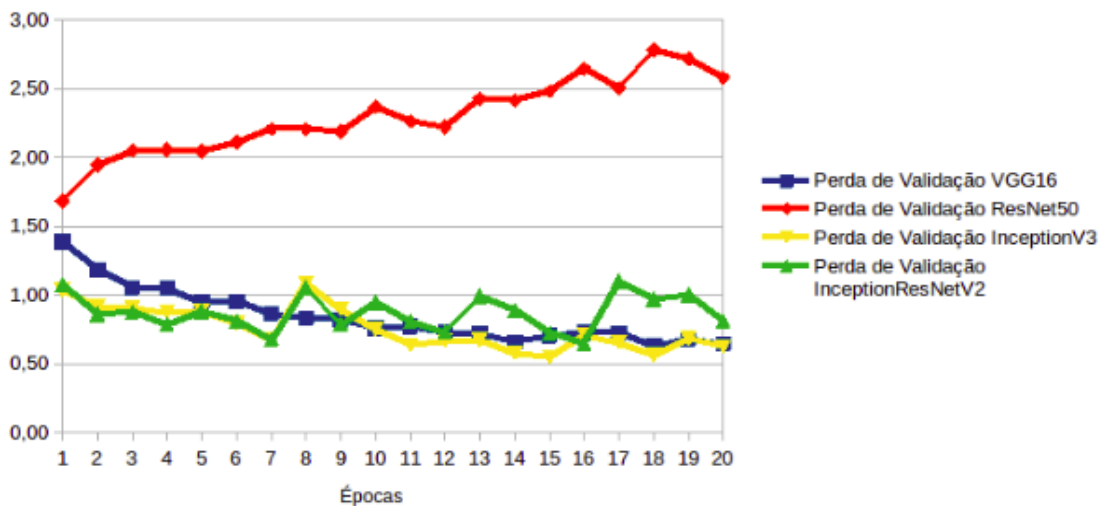
Figura 22 - Gráfico de Exatidão de Treinamento das Arquiteturas



Fonte: Elaboração do próprio autor

Das arquiteturas analisadas a *ResNet50* teve maior exatidão com 95,29% ao final, seguida pela *InceptionV3* com 87,81%, *InceptionResNetV2* com 83,53% e por último *VGG16* com 73,14%. É importante mencionar que não necessariamente uma performance muito boa na etapa de treinamento resultará em uma alta exatidão na etapa de validação. Isso se deve pelo fenômeno denominado sobreajuste (*overfitting*). Quando o sobreajuste ocorre, representa que a arquitetura de rede neural testada decorou o conjunto de dados propostos para o treinamento, mas não aprendeu a detectar características importantes do conjunto de dados que levaria a arquitetura obter bons resultados na etapa de validação. Outra informação que pode ser melhor verificado nas matrizes de confusão é a performance das arquiteturas para classificar cada espécie de planta daninha, uma vez que a arquitetura pode ter uma performance alta para classificação de uma ou mais espécies, entretanto ter uma performance inferior para a classificação das demais espécies. Na figura 23 é apresentada a perda de validação, possibilitando verificar o desempenho das rede na etapa de validação.

Figura 23 - Gráfico de Perda de Validação das Arquiteturas

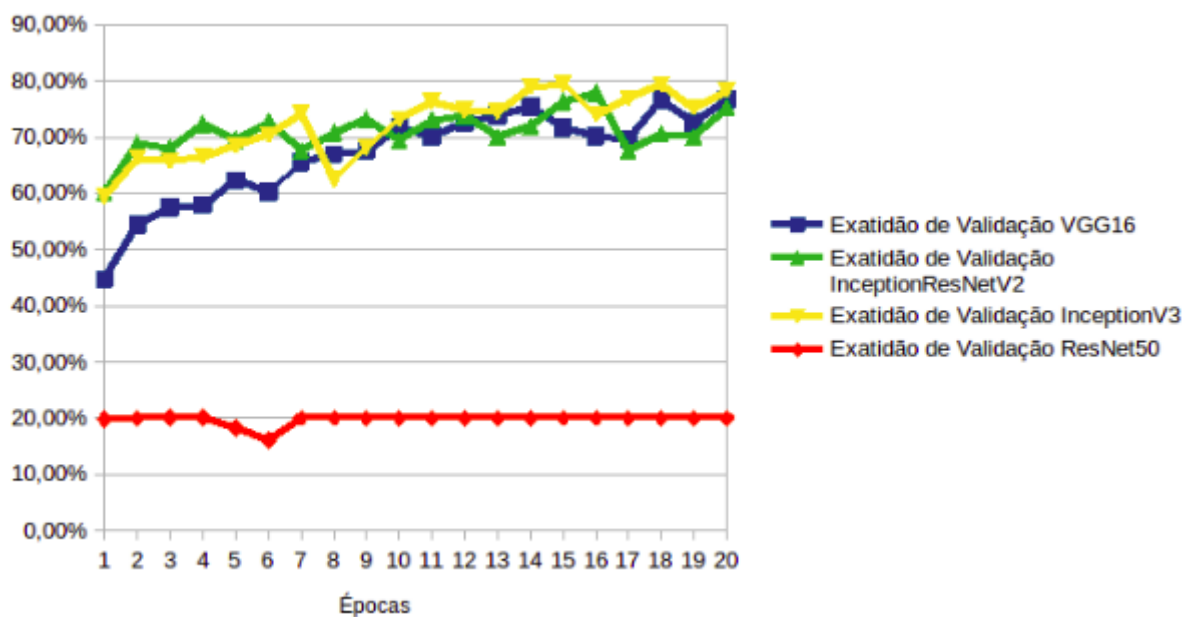


Fonte: Elaboração do próprio autor

Conforme mencionado anteriormente, somente a exatidão de treinamento, não é o suficiente para determinar a performance de uma arquitetura. Nesse caso de acordo com as informações contidas na figura 23, a perda de validação na rede *ResNet50* aumentou, o que significa que a rede não aprendeu, pelo contrário “decorou as informações”. Em relação às outras arquiteturas rede *InceptionV3* teve melhor performance com 0,6253 de perda de validação, seguida pela rede *VGG16* com

0,6425 e *InceptionResNetV2* com 0,8120. Nos gráficos a seguir pode-se verificar a exatidão de validação de cada arquitetura, assim, podendo-se inferir, quais destas obtiveram melhor performance. Na figura 24 a exatidão de verificação das arquiteturas é apresentado, entretanto com base nos dados da perda de validação pode-se concluir que a arquitetura *ResNet50* terá uma performance ruim e a arquitetura *InceptionV3* terá uma performance boa.

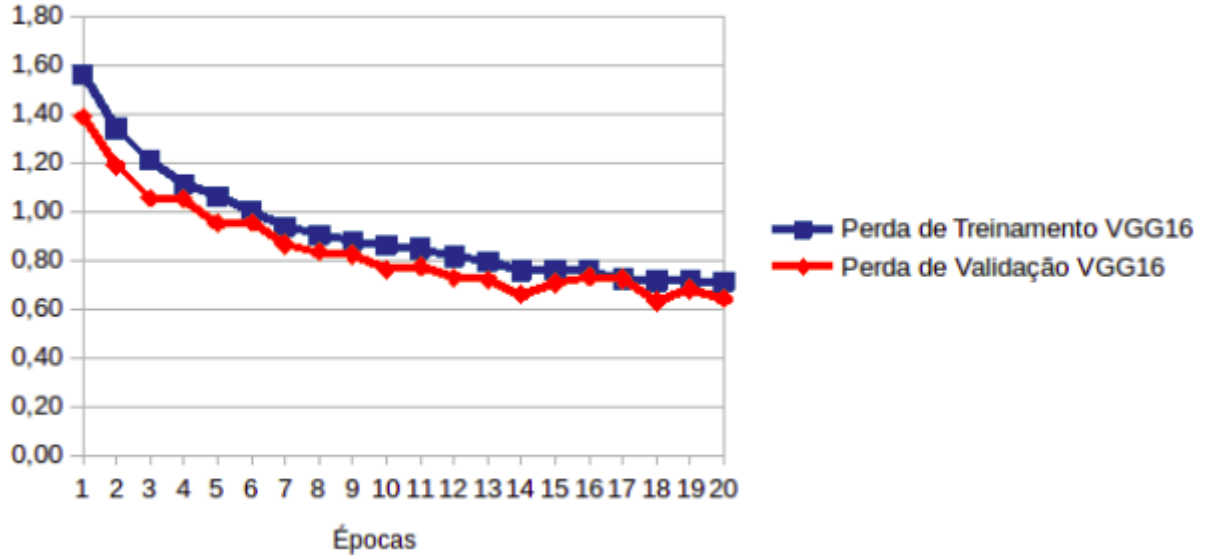
Figura 24 - Gráfico de Exatidão de Validação das Arquiteturas



Fonte: Elaboração do próprio autor

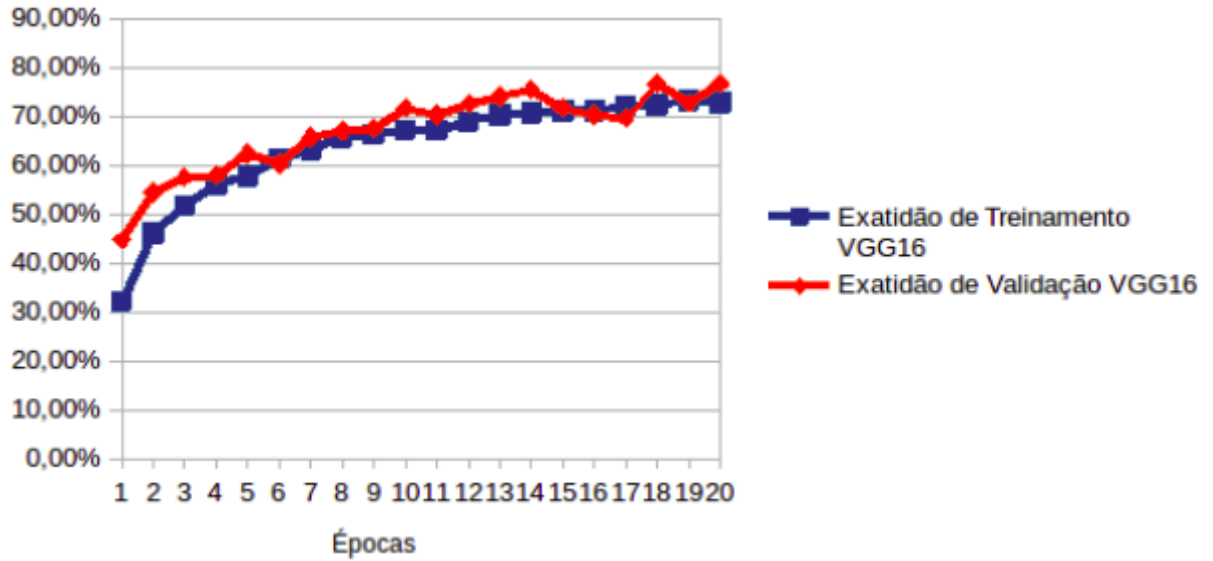
As arquiteturas *VGG16*, *InceptionV3* e *InceptionResnetV2*, apresentaram um crescimento na exatidão de validação, cada uma com respectivamente 76,68% na décima oitava época, 79,37 na décima oitava época e 77,89% na decima sexta época. Esse dado é importante, pois através dele já se pode detectar que não houve sobreajuste nessas arquiteturas. Uma vez que se atinja seja o valor mínimo de perda ou máximo de exatidão esse valor mantém-se, mesmo nas próximas épocas de treinamento o valor seja superior no caso da perda ou inferior no caso da exatidão. A arquitetura *ResNet50* apresentou uma estagnação mantendo-se nos 20,03% de exatidão de validação. Para sumarizar a performance das redes, nas figuras 25 e 26 são apresentados os valores de perda e exatidão para a arquitetura *VGG16*, tanto na etapa de treinamento quanto de validação, nas figuras 27 e 28 para a arquitetura *ResNet50*, nas figuras 29 e 30 para a arquitetura *InceptionV3* e nas figuras 31 e 32 para a arquitetura *InceptionResNetV2*.

Figura 25 - Gráfico de Perda de Treinamento e Validação VGG16



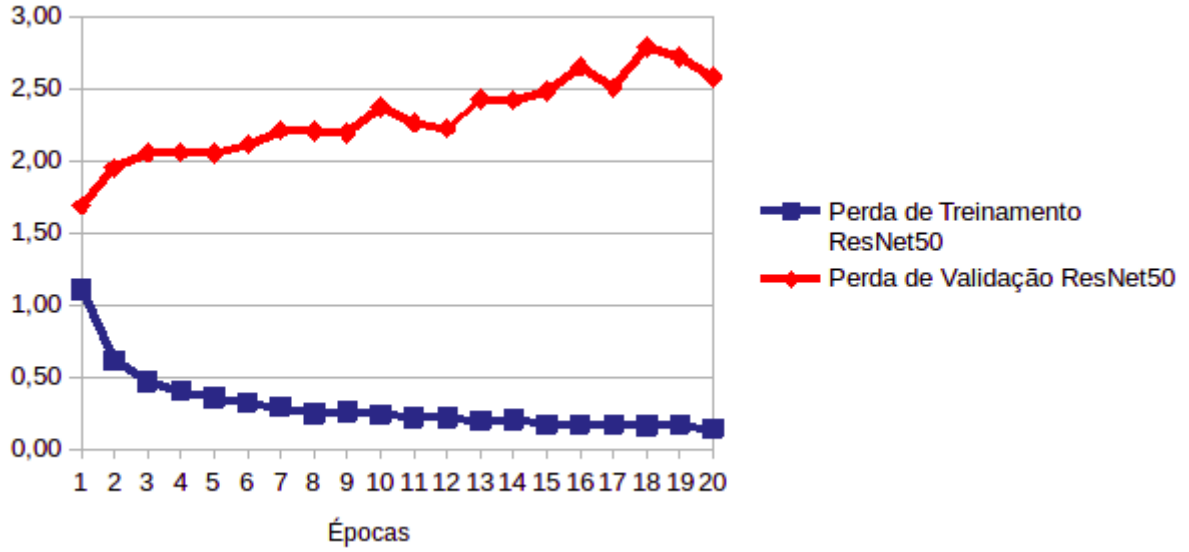
Fonte: Elaboração do próprio autor

Figura 26 - Gráfico de Exatidão de Treinamento e Validação VGG16



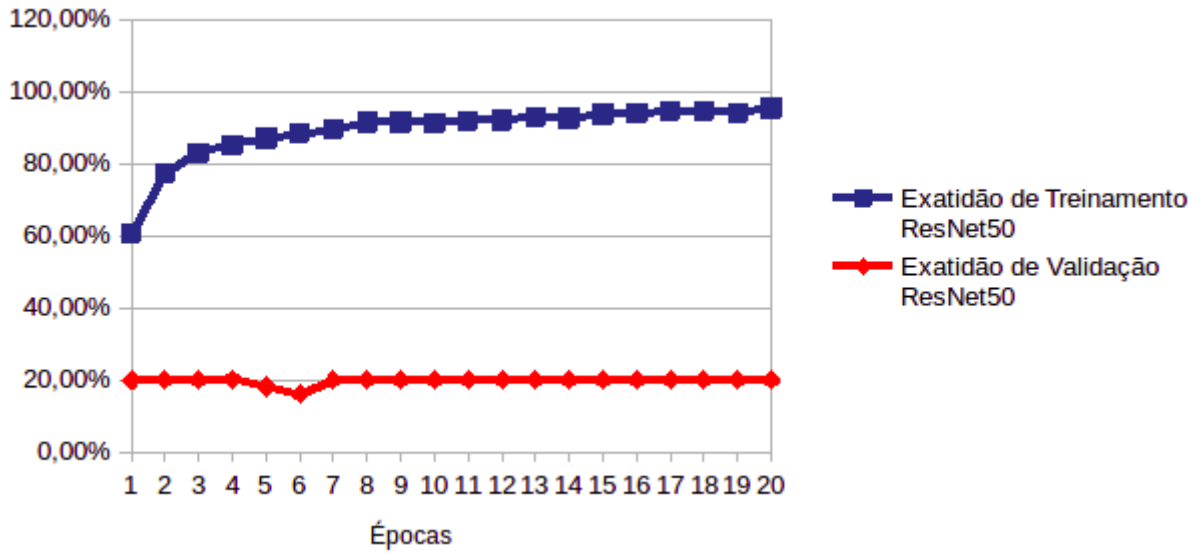
Fonte: Elaboração do próprio autor

Figura 27- Gráfico de Perda de Treinamento e Validação *ResNet50*



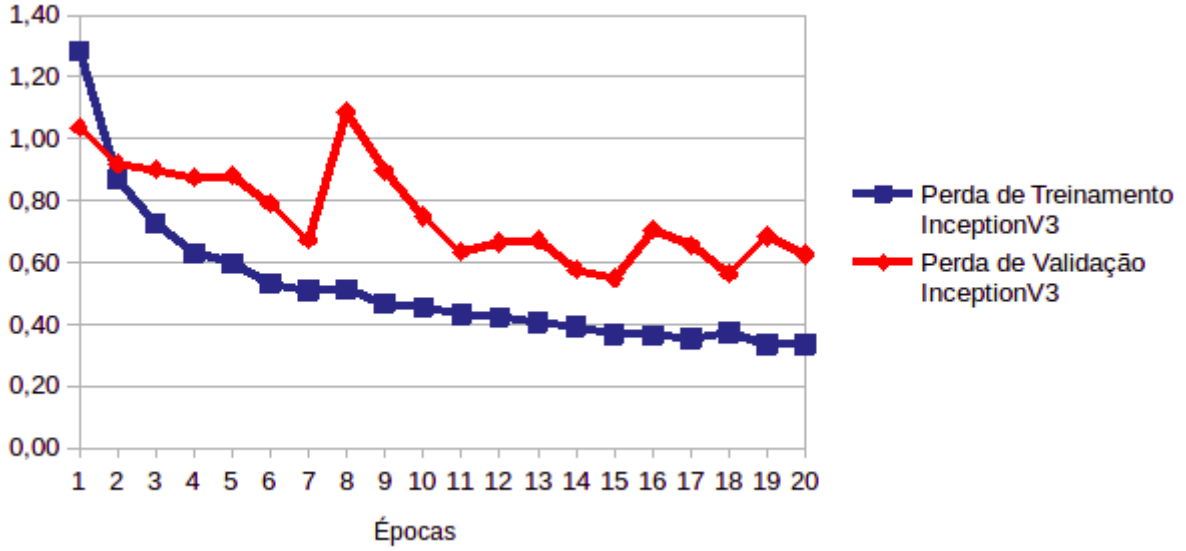
Fonte: Elaboração do próprio autor

Figura 28 - Gráfico de Exatidão de Treinamento e Validação *ResNet50*



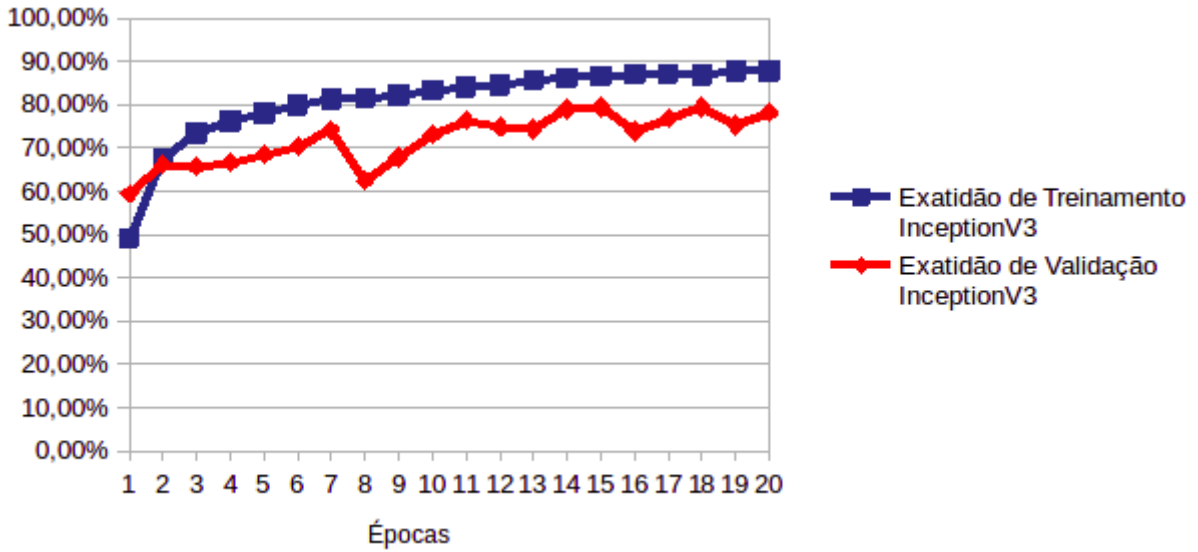
Fonte: Elaboração do próprio autor

Figura 29 - Gráfico de Perda de Treinamento e Validação *InceptionV3*



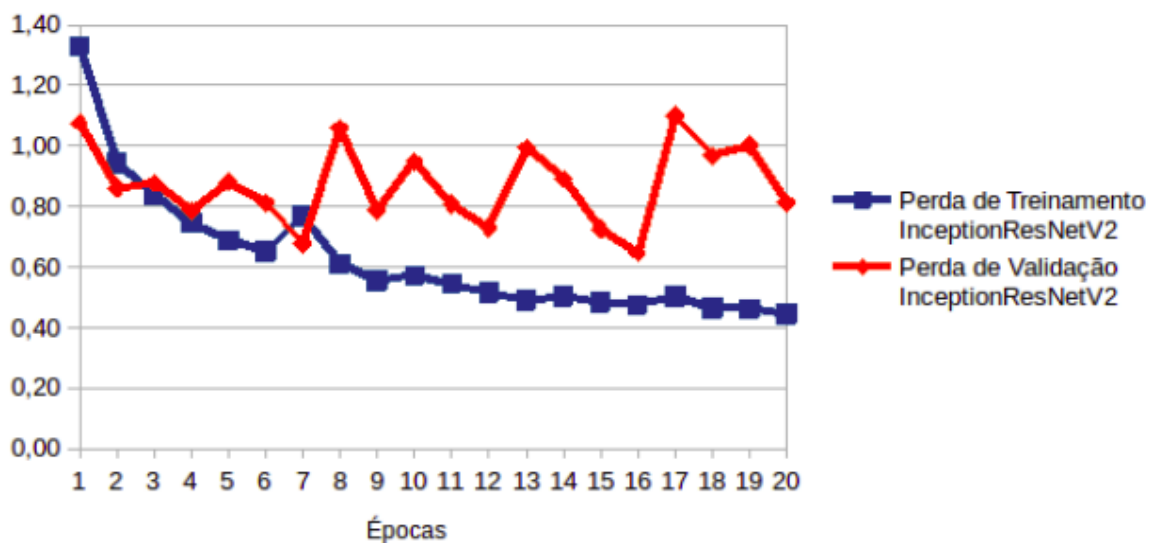
Fonte: Elaboração do próprio autor

Figura 30 - Gráfico de Exatidão de Treinamento e Validação *InceptionV3*



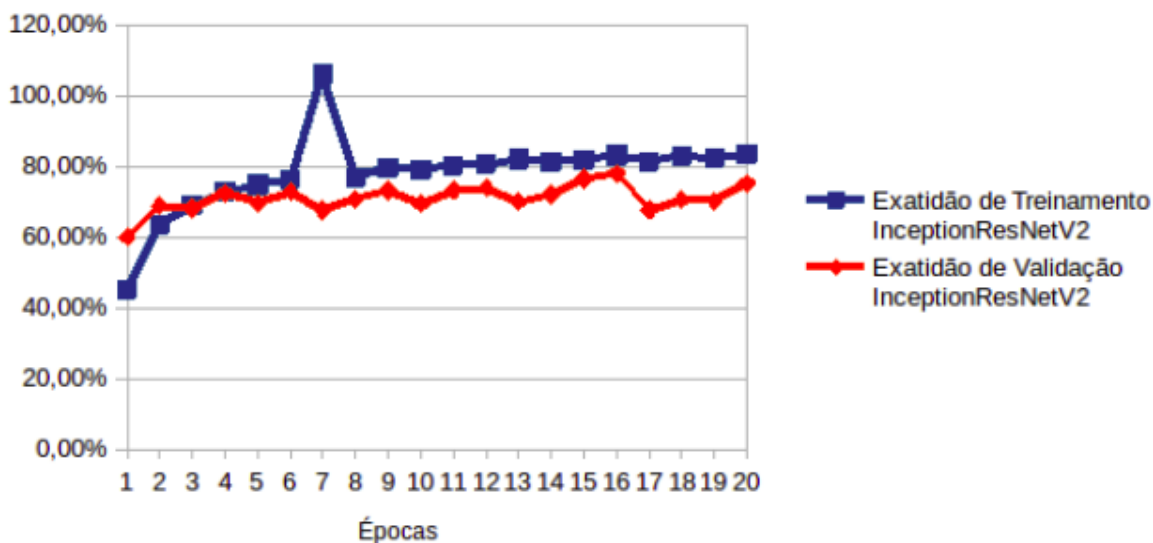
Fonte: Elaboração do próprio autor

Figura 31 - Gráfico de Perda de Treinamento e Validação *InceptionResNetV2*



Fonte: Elaboração do próprio autor

Figura 32- Gráfico de Exatidão de Treinamento e Validação *InceptionResNetV2*



Fonte: Elaboração do próprio autor

Das quatro arquiteturas a rede *VGG16* tem o desempenho mais linear conforme as figuras 25 e 26, esse desempenho deve-se principalmente pelo número de parâmetros que a rede tem 138.357.544, conforme apresentado no quadro 3, o número elevado de parâmetros torna o tempo de treinamento mais longo para esta rede, em contrapartida tanto os resultados de treinamento e validação são lineares, estando próximos ao longo das 20 épocas de treinamento e validação.

No caso da arquitetura *ResNet50* o que pode-se verificar nas figuras 27 e 28 é o contraste dos resultados nas etapas de treinamento e validação. Se por um lado a

perda e exatidão de treinamento da *ResNet50* são os melhores entre as quatro arquiteturas, a perda e exatidão de validação são os piores, demonstrando o sobreajuste desta rede.

De acordo com os gráficos 29 e 30, a rede *InceptionV3* apresenta um desempenho linear nas etapas de perda e exatidão de treinamento, porém estes resultados oscilam mais na etapa de perda e exatidão de validação. Essa performance deve-se pelo número de parâmetros da rede, 23.851.784, tornando à rede mais rápida em tempo de treinamento, porém que necessita de mais dados. Por exemplo na etapa de treinamento usou-se 3.500 imagens e na etapa de validação 1.500, ou seja, um conjunto menor de dados para uma rede com menos parâmetros.

Os resultados da arquitetura *InceptionResNetV2* são apresentados nas figuras 31 e 32, por ter sua construção semelhante a rede *InceptionV3* os resultados obtidos foram semelhantes ao da arquitetura. Entretanto a rede *InceptionV3* foi superior a *InceptionResNetV2* se comparados principalmente os gráficos de perda de validação das figuras 29 para *InceptionV3* e 31 para *InceptionResNetV2*.

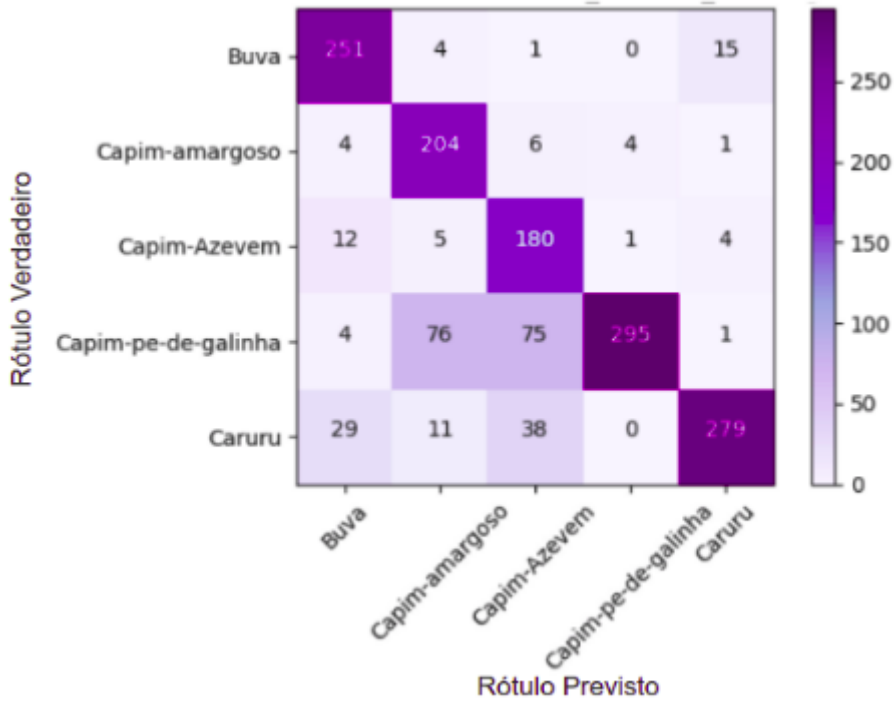
Das arquiteturas analisadas a *InceptionV3* apresenta melhor performance, seguida pela *InceptionResNetV2*. Ambas possuem desempenho similar, sendo que esses resultados próximos devem-se principalmente por compartilharem muitos pontos estruturais em comum. Além dos gráficos que foram criados com base nas informações que cada arquitetura gera ao longo das 20 épocas de treinamento, matrizes de confusão foram plotadas para uma melhor verificação da performance das arquiteturas, principalmente na classificação individual de cada espécie.

Após o treinamento na etapa de validação das 1.500 imagens o algoritmo desenvolvido cria matrizes de confusão para cada arquitetura, podendo-se averiguar a performance de cada arquitetura na classificação das diferentes classes de plantas daninhas. Com base nas informações das matrizes de confusão foram criados quadros que sintetizam a exatidão local e total de cada arquitetura estuda.

A seguir as matrizes de confusão para cada arquitetura são descritas. As matrizes de confusão foram criadas com base no banco de imagens de validação que contém 1.500 imagens divididas em 5 espécies de plantas daninhas, tendo assim 300 imagens para cada espécie. A matrizes são divididas em 2 rótulos o rótulo verdadeiro, ou seja, quantas espécies realmente pertencem a cada classe e o rótulo previsto, ou

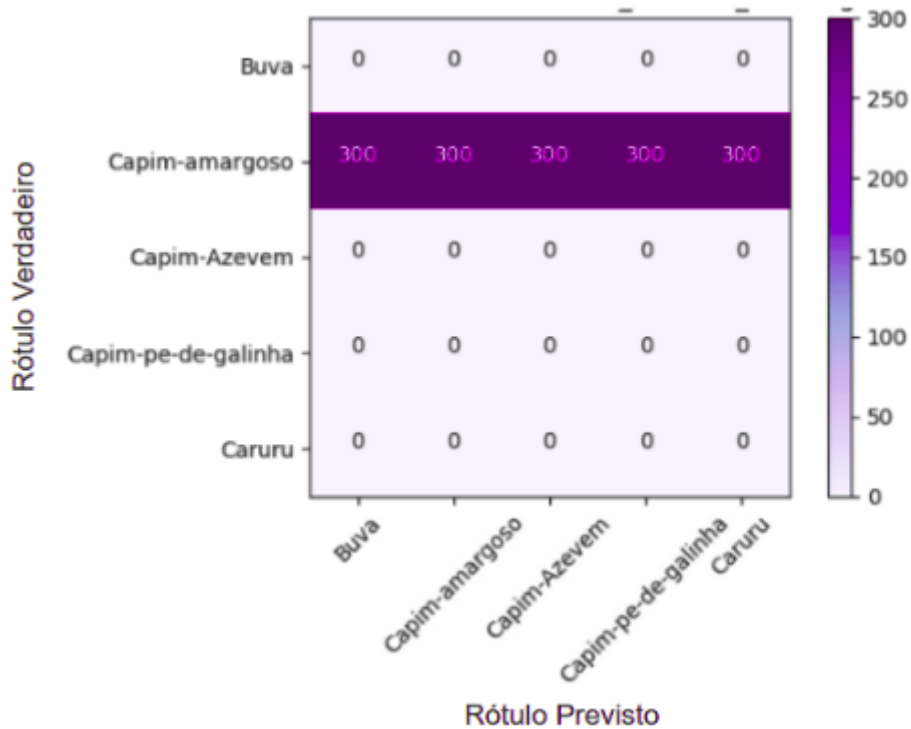
seja, quantas das espécies classificadas estão de acordo com o valor do rótulo verdadeiro. Os resultados podem ser verificados nas figuras 33, 34, 35 e 36.

Figura 33 - Matriz de Confusão Arquitetura VGG16



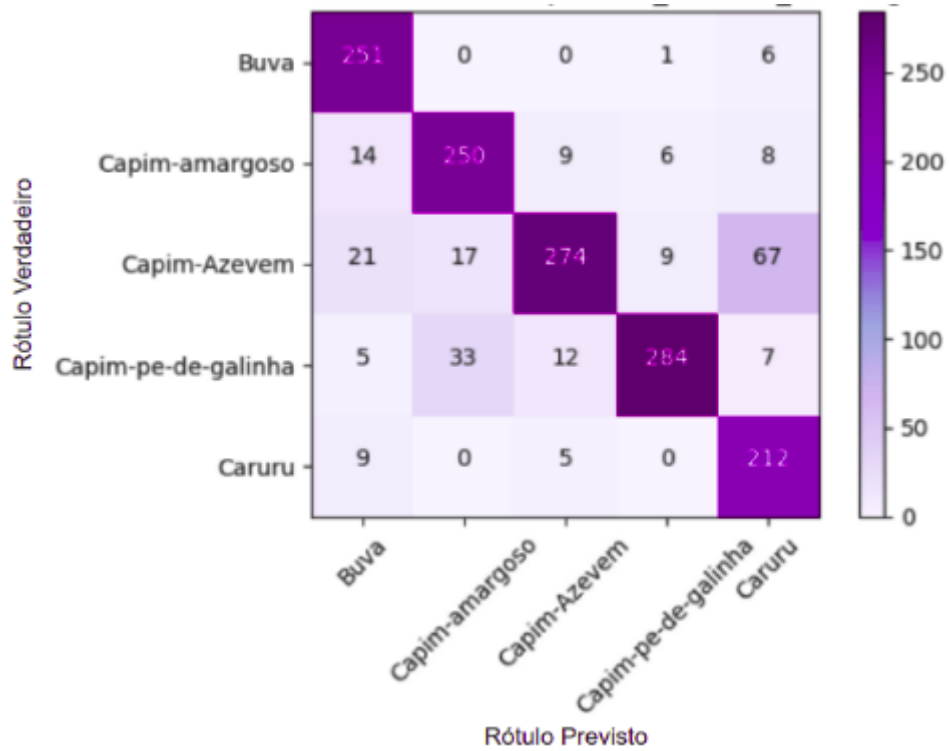
Fonte: Elaboração do próprio autor

Figura 34 - Matriz de Confusão Arquitetura ResNet50



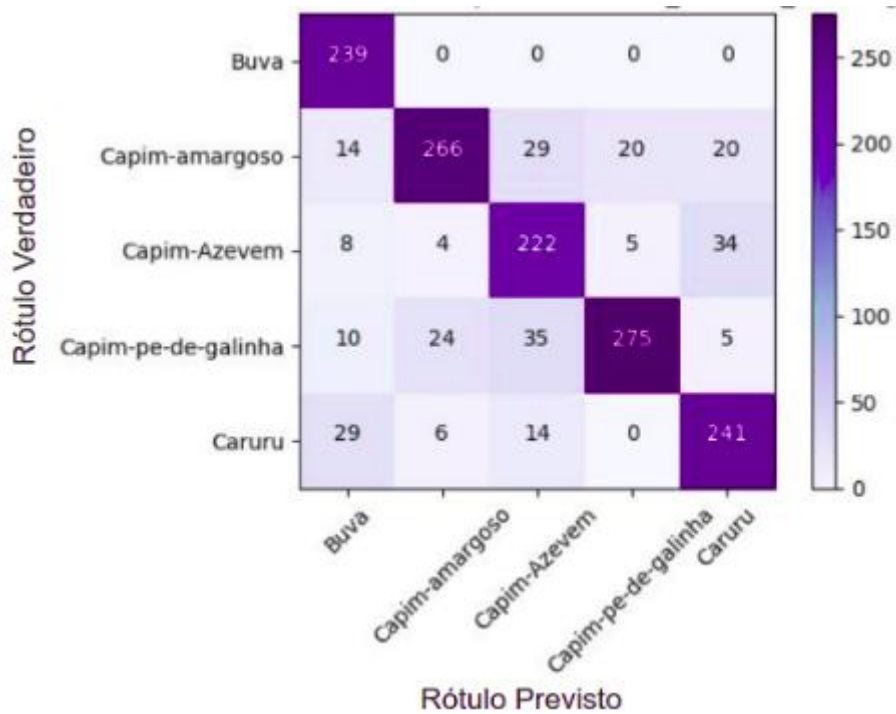
Fonte: Elaboração do próprio autor

Figura 35 - Matriz de Confusão Arquitetura *InceptionV3*



Fonte: Elaboração do próprio autor

Figura 36 - Matriz de Confusão Arquitetura *InceptionResNetV2*



Fonte: Elaboração do próprio autor

De todas as arquiteturas a rede *InceptionV3* teve a melhor performance, tendo seu melhor resultado para classificar a espécie capim pé de galinha acertando 284 imagens de 300, e pior resultado na classificação de caruru com 212 imagens de 300.

Em segundo lugar está a rede *InceptionResNetV2* com 82,86% de exatidão total, levando em conta as cinco espécies e em terceiro lugar a rede *VGG16* com 80,6% de exatidão total.

O pior resultado foi da arquitetura *ResNet50* acertando 100% na categoria capim amargoso, porém errando todas as outras espécies, mais uma vez comprovando que a rede não “aprendeu” e sim decorou.

No quadro 4 são apresentados os valores de exatidão local, referente a exatidão na classificação de cada espécie de planta daninha com base nos dados das matrizes de confusão de cada arquitetura, e a exatidão total, que representa qual foi a porcentagem total de imagens classificadas corretamente em cada arquitetura de acordo com o total das 1.500 imagens usadas para a validação.

Quadro 4 - Performance das Arquiteturas com base nas matrizes de confusão

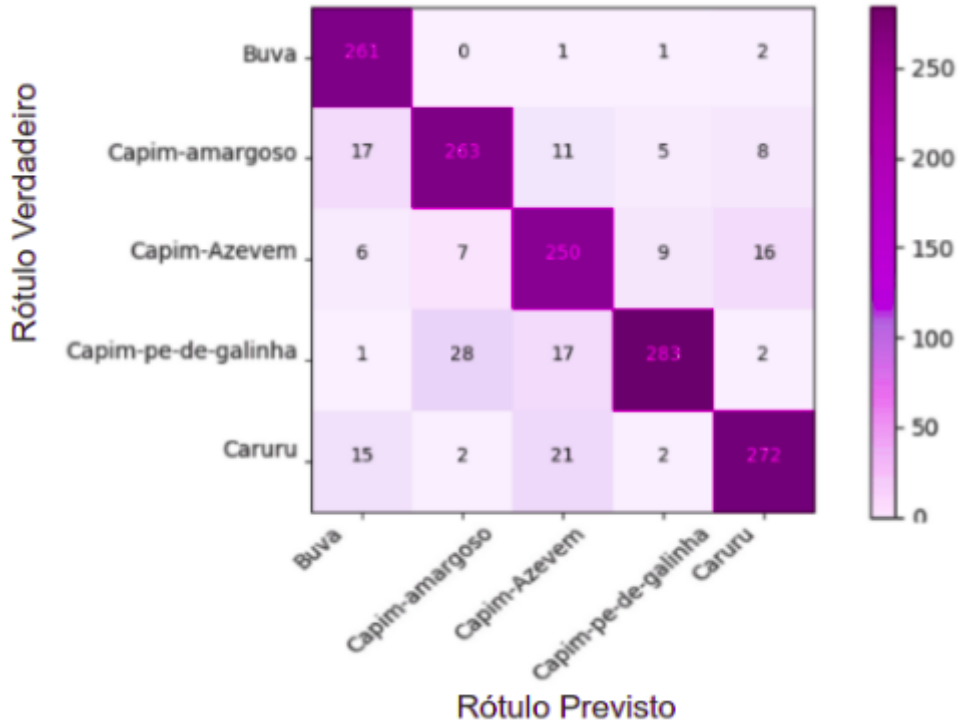
	VGG16	ResNet50	InceptionV3	InceptionResNet
Espécie de Planta Daninha	Exatidão Local(%)	Exatidão Local(%)	Exatidão Local(%)	Exatidão Local(%)
Buva	83,67	0,00	83,67	79,67
Capim-Amargoso	68,00	100,00	83,33	88,67
Capim-Azevém	60,00	0,00	91,33	74,00
Capim-pé-de-Galinha	98,33	0,00	94,67	91,67
Caruru	93,00	0,00	70,66	80,33
Exatidão Total(%)	80,60	20,00	84,73	82,87

Fonte: Elaboração do próprio autor

Com 84,73% de exatidão total e 1.271 imagens classificadas corretamente das 1.500 usadas na validação, a arquitetura *InceptionV3* obteve os melhores resultados. Após a identificação da melhor arquitetura, a rede *InceptionV3* foi treinada pelo período de 40 épocas. Na figura 37 matriz de confusão resultante deste treinamento é apresentada.

Figura 37 - Matriz de Confusão Arquitetura InceptionV3 40 Épocas

Figura 37 - Matriz de Confusão Arquitetura InceptionV3 40 Épocas



Fonte: Elaboração do próprio autor

Após o treinamento com 40 épocas a exatidão total para as 5 espécies analisadas aumentou de 84,73% para 88,6%. Foi identificado também uma melhora na classificação da espécie caruru, que aumentou a assertividade de 212 imagem para 274 de um total de 300. No quadro 5 é apresentada a performance da rede *InceptionV3* com 40 épocas de treinamento para a classificação das 5 espécies.

Quadro 5 - Performance da Arquitetura InceptionV3 com 40 épocas de treinamento, com base nas matrizes de confusão

Espécie	Exatidão Local(%)
Buva	87,00
Capim-Amargoso	87,67
Capim-Azevem	83,33
Capim-pé-de galinha	94,33
Caruru	90,67
Exatidão Total(%)	88,60

Fonte: Elaboração do próprio autor

5 CONCLUSÕES

De acordo com os objetivos propostos e resultados obtidos conclui-se que dentre as quatro arquiteturas de redes neurais profundas estudadas, ou seja, as arquiteturas *VGG16*, *InceptionV3*, *InceptionResNetV2* e *ResNet50*, as três primeiras obtiveram resultados promissores na classificação de imagens de cinco espécies de plantas daninhas com ruídos artificialmente adicionados, excetuando-se a arquitetura *ResNet50* que teve desempenho bastante pobre.

Conforme observado no capítulo anterior, observa-se que a exatidão total das arquiteturas na classificação das 1.500 imagens utilizadas na validação foram respectivamente *VGG16* 80,6%, *ResNet50* 20,0%, *InceptionV3* 84,73% e *InceptionResNetV2* 82,86%. A arquitetura *InceptionV3*, que apresentou o melhor resultado, foi treinada novamente com 40 épocas, obtendo uma exatidão total de 88,6%.

Nesse contexto, as arquiteturas que apresentaram os melhores resultados podem ser empregadas na identificação e classificação de plantas daninhas auxiliando na erradicação das mesmas, seja com a aplicação localizada de defensivos ou com a eliminação mecânica reduzindo o risco de contaminação ao meio ambiente e a cultura.

Entretanto, existem fatores limitantes para a utilização dessas abordagens, visto que o tempo necessário para a identificação e classificação ainda é demasiadamente longo quando empregado em computadores comerciais, limitando aplicação em campo.

Como trabalho futuro, propõe-se ampliar os ensaios para um maior número de épocas, arquiteturas de redes e pretende-se aprimorar a resposta destas arquiteturas aplicando segmentação no banco de imagens, visando otimizar o emprego destas redes em sistemas de tempo real, bem como estender o banco de imagens para outras espécies miscigenado com culturas de soja, milho dentre outras.

REFERÊNCIAS

ABADI, M. et al. TensorFlow: **Large-scale machine learning on heterogeneous systems**. 2015. Disponível em: <<http://tensorflow.org/>> . Acesso 12/12/2018.

BADRINARAYANAN, V et al. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. **IEEE Transactions on Pattern Analysis & Machine Intelligence**, v. 39, p. 2481 - 2495, 2017.

BRASIL. Ministério da Saúde. Secretaria de Vigilância em Saúde. Departamento de Vigilância em Saúde Ambiental e Saúde do Trabalhador. Relatório Nacional de Vigilância em Saúde de Populações Expostas a Agrotóxicos. – v. 1. t. – Brasília (DF), 2018. Disponível em: <http://bvsms.saude.gov.br/bvs/publicacoes/relatorio_nacional_vigilancia_populacoes_expostas_agrotoxicos.pdf> Acesso em 15/12/2018.

CHOLLET, F. **Keras**. 2015. Disponível em: <<https://github.com/fchollet/keras>>. Acesso em 10 jan.2019.

CHRISTOFFOLETI, P.J. (Coord.). **Aspectos de resistência de plantas daninhas a herbicidas**. 3.ed. Campinas, 2008.

DENG, J. et al. ImageNet: A Large-Scale Hierarchical Image Database. **IEEE Conference on Computer Vision and Pattern Recognition**, Miami, v. 1, p. 248 - 255, 2009.

DEEP LEARNING BOOK. **Capítulo 3 - O Que São As Redes Neurais Artificiais Profundas Ou Deep Learning?** Disponível em: <<http://deeplearningbook.com.br/o-que-sao-redes-neurais-artificiais-profundas/>> . Acesso 05 jan.2019.

DEEP LEARNING BOOK. **Capítulo 4 – O Neurônio, Biológico e Matemático**. 2018. Disponível em: <<http://deeplearningbook.com.br/o-neuronio-biologico-e-matematico/>> Acesso em : 10/01/2019.

DOS SANTOS FERREIRA, A. et al. Weed detection in soybean crops using ConvNets. **Computers and Electronics in Agriculture**. v. 143, p. 314–324, Mato Grosso, MT, dez. 2017.

DYRMANN, M. Automatic Detection and Classification of Weed Seedlings under Natural Light. Phd Thesis, **University of Southern Denmark**. jun. 2017.

EMBRAPA - EMPRESA BRASILEIRA DE PESQUISA AGROPECUÁRIA. **Plantas Daninhas**. Brasília (DF). Disponível em:

<<https://www.embrapa.br/tema-plantas-daninhas/sobre-o-tema>>. Acesso 10/12/2018.

ESTADOS UNIDOS. Assembleia Geral das Nações Unidas. Conselho de Direitos Humanos. Report of the Special Rapporteur on the right to food. Nova York (NY), 2017. Disponível em:<<https://documents-dds-ny.un.org/doc/UNDOC/GEN/G17/017/85/PDF/G1701785.pdf?OpenElement>> 15/12/2018.

FAGUNDES, E. **Uso de Redes Neurais Artificiais para aumentar a confiabilidade dos sistemas elétricos**. 2018. Disponível em:<<https://efagundes.com/blog/uso-de-redes-neurais-artificiais-para-aumentar-a-confiabilidade-dos-sistemas-eletricos/>> Acesso 12/12/2018.

GIRSHICK, R et al. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. **IEEE Conference on Computer Vision and Pattern Recognition**, Columbus, v. 1, p. 580 - 587, 2014.

GIRSHICK, R. Fast R-CNN. **IEEE International Conference on Computer Vision**, Santiago, v.1, p. 580 - 1440 - 1448 , 2015.

HE, K. et al. Deep Residual Learning for Image Recognition. **IEEE Conference on Computer Vision and Pattern Recognition**, Las Vegas,v. 1, p. 770-778, 2016.

HE, K et al. Mask R-CNN. **IEEE International Conference on Computer Vision**, Veneza, 2017.

IBAMA - INSTITUTO BRASILEIRO DE MEIO AMBIENTE E DOS RECURSOS NATURAIS RENOVÁVEIS. **Relatórios de Comercialização de Agrotóxicos**. Brasília (DF), 2013.

IBAMA - INSTITUTO BRASILEIRO DE MEIO AMBIENTE E DOS RECURSOS NATURAIS RENOVÁVEIS. **Relatórios de Comercialização de Agrotóxicos**. Brasília (DF), 2017.

KRIZHEVSKY, A; Sutskever, I; Hinton, G. E. ImageNet Classification with Deep Convolutional Neural Networks. **Advances in Neural Information Processing Systems**, San Diego, v. 25, p. 1097-1105, 2012.

LECUN, Y. et al. Gradient-based learning applied to document recognition. **Proceedings of the IEEE**, v. 88, p. 2278–2324, nov. 1998.

MILIOTO, A. et al. Real-time Semantic Segmentation of Crop and Weed for Precision Agriculture Robots Leveraging Background Knowledge in CNNs. **IEEE**

International Conference on Robotics & Automation, Austria, v. 1, p. 2229 - 2235, 2018.

RAHMAN, N. **What is the benefit of using average pooling rather than max pooling?**. 2017. Disponível em: <<https://www.quora.com/What-is-the-benefit-of-using-average-pooling-rather-than-max-pooling>> Acesso 12/12/2018.

REDMON, J et al. You Only Look Once: Unified, Real-Time Object Detection. **IEEE Conference on Computer Vision and Pattern Recognition**, Las Vegas, v. 1, p. 779 - 788, 2016.

ROSENBLATT, F. The perceptron, perceiving and recognizing automaton Project Para. **Cornell Aeronautical Laboratory**, v. 85, Buffalo, NY, 1957.

RUMELHART, D. E.; Hinton, G. E.; Williams, R. J. Learning representations by back-propagating errors. **Nature**, v. 323, p. 533–536, out.1986.

SERMANET, P et al. Overfeat: Integrated recognition, localization and detection using convolutional networks. **International Conference on Learning Representations**, Banff, 2014.

SILVA, S. L. A .; Junior, S. B. V. Aplicações E Benefícios Obtidos Através Das Redes Neurais Artificiais (RNA). **Revista Facima Digital Gestão**, Maceió, v. 2, p. 29 - 43, 2017.

SIMONYAN, K; Zisserman, A. Very deep convolutional networks for Large-Scale image Recognition. **International Conference on Learning Representations**, San Diego, 2015.

SINDIVEG - SINDICATO NACIONAL DA INDÚSTRIA DE PRODUTOS PARA A DEFESA VEGETAL. **O Que Você Precisa Saber Sobre Defensivos Agrícolas**. São Paulo, 2017. Disponível em <<http://sindiveg.org.br/wp-content/uploads/2018/08/oquevoceprecisasabersobredefensivosagricolas.pdf>> Acesso em : 12/12/2018.

SYNGENTA. **Tecnologia é aliada no controle de daninhas, como buva e amargoso**. 2017. Disponível em: <<https://www.portalsyngenta.com.br/noticiasdocampo/tecnologia-e-aliada-no-controle-de-daninhas-como-buva-e-amargoso>> . Acesso em : 12/12/2018.

SZEGEDY, C. et al. Going Deeper with Convolutions. **IEEE Conference on Computer Vision and Pattern Recognition**, Boston, v. 1, p. 1 - 9, 2015.

SZEGEDY, C. et al. Rethinking the Inception Architecture for Computer Vision. **IEEE Conference on Computer Vision and Pattern Recognition**, Las Vegas, v. 1, p.2818 - 2826, 2016.

SZEGEDY, C; Ioffe, S; Vanhoucke, V. Inceptionv4, inception-resnet and the impact of residual connections on learning. **International Conference on Learning Representations**, San Juan, 2016.
CoRR, abs/1602.07261.

UIJLINGS, J. R. R. et al. Selective Search for Object Recognition. **International Journal of Computer Vision**, v. 104, p. 154 - 171, 2013.

VASCONCELOS, Yuri. Agrotóxicos na Berlinda. **Pesquisa FAPESP**. São Paulo, Edição 271, set. 2018. Disponível em:
<<http://revistapesquisa.fapesp.br/2018/09/18/agrotoxicos-na-berlinda/>>. Acesso em: 15/12/2018.

W, S. MCCULLOCH.; Pitts, W. A logical calculus of the ideas immanent in nervous activity. **The bulletin of mathematical biophysics**, v. 5, p.115–133, dez. 1943.

W, BERNARD. et al. Adaptive "Adaline" neuron using chemical "memistors". **Stanford Electron. Labs, Technical Report 1553-2**, Stanford, CA, out. 1960.