

UNIVERSIDADE FEDERAL DO RIO GRANDE
CENTRO DE CIÊNCIAS COMPUTACIONAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO
CURSO DE MESTRADO EM ENGENHARIA DE COMPUTAÇÃO

Dissertação de Mestrado

**Redes Neurais Convolucionais e sua Aplicação Assistiva
como Auxílio para Identificação de Curvas Algébricas no
Plano Cartesiano**

Luiz Oscar Homann de Topin

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal do Rio Grande, como requisito parcial para a obtenção do grau de Mestre em Engenharia de Computação

Orientadora: Profa. Dra. Regina Barwaldt

Rio Grande, 2020

Ficha Catalográfica

T674r Topin, Luiz Oscar Homann de.
Redes neurais convolucionais e sua aplicação assistiva como auxílio para identificação de curvas algébricas no plano cartesiano / Luiz Oscar Homann de Topin. – 2020.
81 f.

Dissertação (mestrado) – Universidade Federal do Rio Grande – FURG, Programa de Pós-Graduação em Computação, Rio Grande/RS, 2020.

Orientadora: Dra. Regina Barwaldt.

1. Cegos 2. Álgebra 3. Assistivo 4. Aprendizagem Profunda
5. Redes Neurais Convolucionais I. Barwaldt, Regina II. Título.

CDU 512.772:004

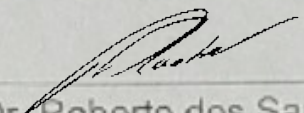
Catálogo na Fonte: Bibliotecário José Paulo dos Santos CRB 10/2344

DISSERTAÇÃO DE MESTRADO

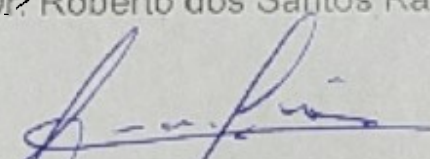
**Redes Neurais Convolucionais e sua Aplicação Assistiva como Auxílio
para Identificação de Curvas Algébricas no Plano Cartesiano**

Luiz Oscar Homann de Topin

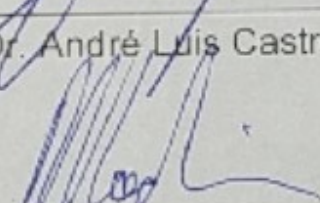
Banca examinadora:



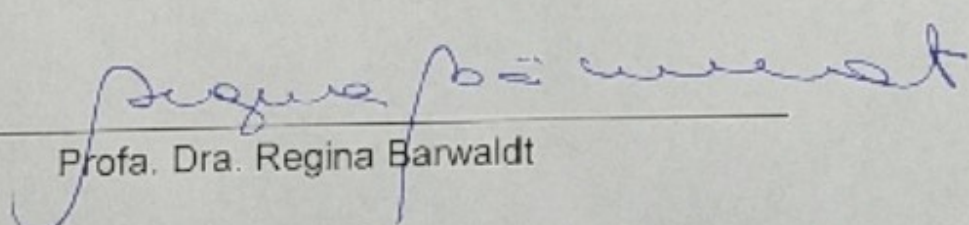
Prof. Dr. Roberto dos Santos Rabello



Prof. Dr. André Luis Castro de Freitas



Prof. Dr. Ricardo Nagel Rodrigues



Profa. Dra. Regina Barwaldt

AGRADECIMENTOS

Finalizar este trabalho só foi possível com a ajuda de todas as pessoas que de alguma maneira contribuíram direta e indiretamente. São tantas pessoas que pode ser que venha a esquecer de alguém, caso venha a esquecer de você caro amigo, sinta-se agradecido.

Gostaria de começar agradecendo aos meus colegas bolsistas do grupo de pesquisa InfoEduc - FURG, Sartori e Bruna, sem estes dois não teria sido possível avançar nas pesquisas.

Agradeço também aos outros colegas mestrandos também do grupo de pesquisa, ao Granada por sempre estar disponível para tirar umas dúvidas e ouvir as minhas diversas lamentações e reclamações ao longo do mestrado, aos colegas Tiago e Stefani pelas valiosas contribuições durante as reuniões sem fim realizadas ao longo desses dois anos. Aos meus colegas que hoje considero grandes amigos, Márcio e Paulo, com estes compartilhei diversos bons momentos e, sem dúvida alguma aprendi muito com as valiosas dicas de ambos, infelizmente o amigo Paulo acabou sofrendo um acidente ao longo da trajetória, abalando bastante os últimos dias de escrita deste trabalho, espero com todas as minhas forças que logo estejamos juntos rindo dos anos que se passaram e dos obstáculos que foram superados.

Um agradecimento mais que especial para a minha incansável orientadora, Prof. Regina que, ao longo destes anos sempre esteve disponível para sempre apontar a direção certa e sempre me manter focado nos objetivos que ainda pretendo alcançar, sou profundamente grato pelas reuniões que pareciam não ter fim e pelas discussões acaloradas, tenho um respeito e admiração enorme por ti Professora.

Sem dúvida o principal agradecimento vai para a minha família... os meus irmãos que, mesmo estando longe nunca deixaram de me fortalecer, a minha querida mãe que sempre esteve rezando e nunca mediu esforços para me ajudar e sempre me apoiou em absolutamente todas as decisões que tomei... mesmo as mais tolas, obrigado Nica. A minha esposa, minha melhor amiga, minha parceira ao meu lado sempre, Violeta meu pilar, confiou no meu potencial desde o início e toda vez que desanimei por algum motivo ela estava lá com uma palavra de apoio e um carinho. A ti eu devo muito. A minha filha que está prestes a nascer é a grande razão por trás de todo esforço para finalizar a escrita deste trabalho.

E por último queria agradecer ao meu Pai que, onde quer que ele esteja sei que está sempre de olho em mim e na nossa família, é até engraçado pensar que hoje sigo teus passos.

"My Precious !"

— SMEAGOL - THE LORD OF THE RINGS

RESUMO

No Brasil, particularmente, a interação com o Sistema de Coordenadas Cartesianas (SCC) inicia no ensino fundamental e, em muitas áreas, se estende ao ensino superior, o estudo de curvas Algébricas possui uma série de aplicações no campo da física, biologia, computação gráfica e outras. É possível explicar e entender fenômenos físicos, que podem variar da mecânica celeste à dinâmica complexa das partículas, permeando a análise das formas peculiares de organelas e os complexos padrões de fractais. No entanto entender a complexidade dos elementos apresentados e trabalhados nas grandes subáreas da geometria e do estudo do plano cartesiano, pode vir a ser uma tarefa complicada para a maioria dos seres humanos. Nesse sentido, existem técnicas computacionais que visam imitar a estrutura neural dos seres biológicos, por meio de séries de cálculos matemáticos, denominadas Redes Neurais Artificiais (RNA). Tais estruturas realizam trabalhos que podem, na maioria dos casos, ser muito dispendiosos para um ser humano realizar. A partir dos avanços no poder computacional e da geração de conjuntos de dados massivos, os sistemas baseados em RNAs são cada vez mais frequentes. Este trabalho teve como objetivo desenvolver uma RNA capaz de reconhecer curvas Algébricas utilizadas no ensino de geometria no Brasil. Para atingir o objetivo foi necessário realizar uma sequência de três etapas. O primeiro estágio refere-se à escolha da tecnologia, cada RNA possui arquiteturas diferentes, variando de acordo com a aplicação, a RNA convolucional é a arquitetura mais utilizada para realizar a detecção e classificação de imagens devido especialização na percepção de padrões. O segundo estágio diz respeito à criação do conjunto de dados. Redes neurais convolucionais precisam ser supervisionadas, necessitam aprender com séries de exemplos do objeto de estudo. A última etapa trata do treinamento e da análise dos dados, a rede neural aprende conforme alimentada pelos dados gerados e, como resposta, produz valores de saída referentes à precisão na identificação e classificação destes elementos. Em seguida, os valores resultantes são analisados a fim de realizar melhorias e atualizações na arquitetura. Para desenvolver essa solução, foi necessário arquitetar uma rede neural convolucional e desenvolver um conjunto de dados consistente. A rede neural proposta possui 16 camadas convolucionais no espaço 3×3 , enquanto o conjunto de dados possui classes representando cada curva cartesiana, compreendendo um total de mais de 30 mil imagens. Após realizar todas as etapas propostas o modelo de RNC proposto conseguiu atingir precisão acima de 90% e ritmo de perda inferior a 20%.

Palavras-chave: Cegos, Algebra, Assistivo, Aprendizagem Profunda, Redes Neurais Convolucionais.

ABSTRACT

In Brazil, in particular, interaction with the Cartesian Coordinate System (SCC) begins in elementary education and, in many areas, extends to higher education. The study of Cartesian Curves has a series of applications in the field of physics, biology, computing graphics and others. It is possible to explain and understand physical phenomena, which can vary from celestial mechanics to complex particle dynamics, including the analysis of peculiar forms of organelles and complex fractal patterns. However, understanding the complexity of the elements presented and worked on in the great subareas of geometry and the study of the Cartesian plane, can prove to be a complicated task for most human beings. In this sense, there are computational techniques that aim to imitate the neural structure of biological beings, through a series of mathematical calculations, called Artificial Neural Networks (ANN). Such structures do work that can, in most cases, be very expensive for a human being to do. Based on advances in computational power and the generation of massive datasets, RNA-based systems are most frequent. This work aims to develop an RNA capable of recognizing the main Cartesian curves used in the teaching of geometry in Brazil. It was necessary to carry out a sequence of three steps to achieve it. The first stage concerns the choice of technology. RNA has different architectures, varying according to the application. Convolutional RNA is the most used architecture to perform the detection and classification of images, that is, it has specialization in the perception of image patterns. The second stage concerns the creation of the dataset. Convolutional neural networks need to be supervised; that is, you must learn from a series of examples of the object of study. The last stage deals with training and data analysis, that is, the neural network learns as fed by the generated data and, in response, produces output values referring to the precision in the identification of these elements. Then, the resulting values are analyzed in order to make improvements and updates and updates to the neural network architecture. To develop this solution, it was necessary to architect a convolutional neural network and develop a consistent dataset. The proposed neural network has 16 convolutional layers in the 3x3 space, while the dataset has classes representing each Cartesian curve, comprising a total of more than 30 thousand images. After performing all the proposed steps, the proposed RNC model was able to achieve accuracy above 90% and loss rate below 20%.

Keywords: Blinds, Algebra, Assistive, Deep Learning, Convolution Neural Networks.

LISTA DE FIGURAS

Figura 1	Número de matrículas de alunos com deficiência, transtornos globais do desenvolvimento ou altas habilidades em classes comuns ou especiais exclusivas	15
Figura 2	Restultado da busca automática	25
Figura 3	Resultado após critérios de inclusão e exclusão	25
Figura 4	Relação de trabalhos por ano	26
Figura 5	Linguagens de programação utilizadas nos trabalhos	26
Figura 6	Tipo de aplicações desenvolvidas	27
Figura 7	Aplicações com viés para ensino e aprendizagem para cegos	27
Figura 8	Ferramentas utilizadas como meio de interação	28
Figura 9	Exemplo de Plano Cartesiano com Coordenadas Positivas e Negativas	29
Figura 10	Tipos de deficiência	33
Figura 11	Causas da Cegueira e Deficiência Visual no Mundo.	35
Figura 12	Modelo de Neurônio Artificial (MCCULLOCH; PITTS, 1943b)	40
Figura 13	Estrutura de uma rede neural Multicamada	41
Figura 14	Exemplo de <i>padding</i>	42
Figura 15	Exemplo de <i>stride</i>	43
Figura 16	Produto Convolutacional	44
Figura 17	<i>Pooling</i> médio	45
Figura 18	Exemplo de Camada Convolutacional	46
Figura 19	Exemplo de Camada Convolutacional	47
Figura 20	Camada <i>Fully Connected</i>	48
Figura 21	Camada <i>Fully Connected</i>	48
Figura 22	Rede Convolutacional 3d	49
Figura 23	Fluxo de Processos da Aplicação	50
Figura 24	Ciclo para Geração do <i>Dataset</i>	53
Figura 25	Gerador de Matriz Polinomial	54
Figura 26	Curva Cardioide	54
Figura 27	Exemplo <i>dataset</i> curva Cardioide	55
Figura 28	Diagrama de Experimentos	57
Figura 29	Ciclo de testes do primeiro experimento	58
Figura 30	Fluxo de Treinamento a cada Ciclo	59
Figura 31	VGG16 Precisão por Época	61
Figura 32	VGG19 Precisão por Época	61
Figura 33	MobileNet Precisão por Época	62
Figura 34	InceptionV3 Precisão por Época	62

Figura 35	VGG16 ritmo de perda por Época	63
Figura 36	VGG19 ritmo de perda por Época	63
Figura 37	InceptionV3 ritmo de perda por Época	64
Figura 38	MobileNETV2 ritmo de perda por Época	64
Figura 39	Arquitetura da RNC VGG16.	66
Figura 40	Fluxo de Treinamento por Ciclo Experimento 2	67
Figura 41	Demonstração do Critério de Parada	71
Figura 42	Dados do Ritmo de Perda no ciclo de Treino	72
Figura 43	Dados de Precisão no ciclo de Treino	72
Figura 44	Dados de Precisão no ciclo de Validação	73
Figura 45	Dados do Ritmo de Perda no ciclo de Validação	73
Figura 46	Matriz de Confusão	74
Figura 47	Teste de Avaliação	76

LISTA DE TABELAS

Tabela 1	Organização do Texto	18
Tabela 2	Bases de Dados selecionadas para Revisão Sistemática	20
Tabela 3	Resultados da busca nas Bases de Dados	21
Tabela 4	Resultados Final da busca nas Bases de Dados	22
Tabela 5	Dados extraídos a partir da estratégia definida	23
Tabela 6	Lista de Curvas Algébricas de Acordo com o Grau - Fonte autor . . .	31
Tabela 7	Quadro comparativo da classificação de TA.	37
Tabela 8	Parâmetros de avaliação das arquiteturas de RNC	65
9	Dados dos Ciclos de Treinamento Experimento 2	68
Tabela 10	Melhores Resultados Experimento 2	70
Tabela 11	Calculo Matriz de Confusão	75
Tabela 12	Dados Teste de Avaliação 1 - Fonte autor	77
Tabela 13	Dados Teste de Avaliação 2 - Fonte autor	77

LISTA DE ABREVIATURAS E SIGLAS

RNA	Rede Neural Artificial
RNC	Rede Neural Convolutacional
MEC	Ministério da Educação
INEP	Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira
RS	Revisão Sistemática
SciELO	Sociedade Brasileira de Computação
RENOTE	Revista Novas Tecnologias na Educação
SBIE	Simpósio Brasileiro de Informática na Educação
RBIE	Revista Brasileira de Informática na Educação
IBGE	Instituto Brasileiro de Geografia e Estatística
DMRI	Degeneração Macular Relacionada à Idade
API	Application Programming Interface
RNP	Rede Neural Profunda
DNN	Deep Neural Network

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Motivação	16
1.2	Objetivos	17
1.2.1	Objetivo Geral	17
1.2.2	Objetivos Específicos	17
1.2.3	Contribuições da Pesquisa	17
1.3	Organização do Texto	17
2	REFERENCIAL TEÓRICO	19
2.1	Revisão sistemática da literatura	19
2.1.1	Planejamento	19
2.1.2	Condução	28
2.1.3	Fase de Relato	28
2.1.4	Resultados da revisão	28
2.2	O Plano Cartesiano	29
2.3	Curvas Algébricas no Plano Cartesiano	30
2.4	Álgebra e a Educação	32
2.5	Deficiência Visual	32
2.5.1	Deficiência visual e suas causas	34
2.5.2	Deficiência Visual no Sistema Educacional	35
2.6	Tecnologia Assistiva para a Educação	36
2.7	Visão Computacional	39
2.8	Redes Neurais Artificiais	40
2.8.1	Redes Neurais Convolucionais	41
3	PROCEDIMENTOS METODOLÓGICOS	50
3.1	Proposta do Sistema	50
3.1.1	Desenvolvimento do Dataset	51
3.1.2	Preparação	55
3.1.3	Treinamento da Rede	57
3.2	Primeiro Experimento	57
3.2.1	Análise dos Dados - Primeiro Experimento	60
3.3	Segundo Experimento	67
3.3.1	Análise dos Dados - Segundo Experimento	68
3.4	Terceiro Experimento	70
3.4.1	Análise dos Dados - Terceiro Experimento	71
3.5	Confirmação	75

4 CONSIDERAÇÕES FINAIS	78
REFERÊNCIAS	79

1 INTRODUÇÃO

A visão é um dos mais complexos mecanismos do corpo humano e um dos sentidos mais importantes do sistema sensorial, com 140 milhões de neurônios (LEUBA; KRAFT-SIK, 1994), o córtex visual, responsável pelo processamento e interpretação de dados visuais, encarregado de dar percepção e formular memórias, ocupa uma parte significativa de todo o córtex cerebral.

Os seres humanos são uma espécie visualmente orientada, possuem visão binocular, capaz de distinguir cor, profundidade, identificar padrões e formatos. A partir destas características, orientamos a maioria das ações e eventos em nossa sociedade em torno da suposição de um certo grau de capacidade visual, e, aqueles sem esta competência enfrentam uma série de obstáculos na vida como resultado.

Os olhos e a capacidade visual são importantes não apenas porque os humanos usam e confiam neles. Nós evoluímos para, fomos criados para, e construímos nossas civilizações ao longo da história em torno deste sentido estar disponível.

De acordo com dados da OMS - Organização Mundial da Saúde (*WHO - World Health Organization*), em pesquisa divulgada em agosto de 2012, o número de pessoas com deficiência visual no Mundo é de aproximadamente 285 milhões, sendo 246 milhões com perda parcial (visão subnormal) e 39 milhões com perda total da visão (cegos)¹.

No Brasil, de acordo com o IBGE (Instituto Brasileiro de Geografia e Estatística) o Censo² de 2010 apontou que a deficiência visual totalizou aproximadamente 35,8 milhões de pessoas, onde 6,6 milhões de pessoas com grave comprometimento visual (cegueira ou visão subnormal), dos quais 506,3 mil eram cegos.

Dessa forma, é evidente que esta parcela significativa da população merece e necessita de mais atenção, respeito e cuidados.

A escola é um ambiente com uma série de estímulos visuais, e, enxergar faz parte não apenas da interação entre professor e aluno, mas também da construção de todo o processo de ensino e aprendizagem.

Existem inúmeros meios para melhorar a condição destes indivíduos, a criação e

¹<http://www.who.int/mediacentre/factsheets/fs282/en/>

²<https://ww2.ibge.gov.br/home/estatistica/populacao/censo2010/>

inserção de políticas públicas de acessibilidade e inclusão ou tecnologias assistivas (TA). As TAs são recursos e serviços que contribuem para proporcionar ou ampliar habilidades funcionais de pessoas com deficiência, e visam proporcionar maior independência, qualidade de vida e inclusão social, através da ampliação de sua comunicação, mobilidade, controle de seu ambiente, habilidades de seu aprendizado, trabalho e integração com a família, amigos e sociedade (BERSCH, 2017).

De acordo com o censo da educação básica de 2018³, o número de matrículas na educação especial chegou a 1,2 milhão em 2018, um aumento de 33,2% em relação a 2014. O maior número de matrículas está no ensino fundamental, que concentra 70,9% das matrículas na educação especial. Quando avaliado aumento no número de matrículas entre 2014 e 2018, percebe-se que as matrículas no ensino médio são as que mais cresceram, um aumento de 101,3%, conforme consta na Figura 1, ainda de acordo com (LAPLANE, 2015), as deficiências visuais (cegueira e baixa visão) representam 7% das matrículas ou um total de 84,000 alunos.

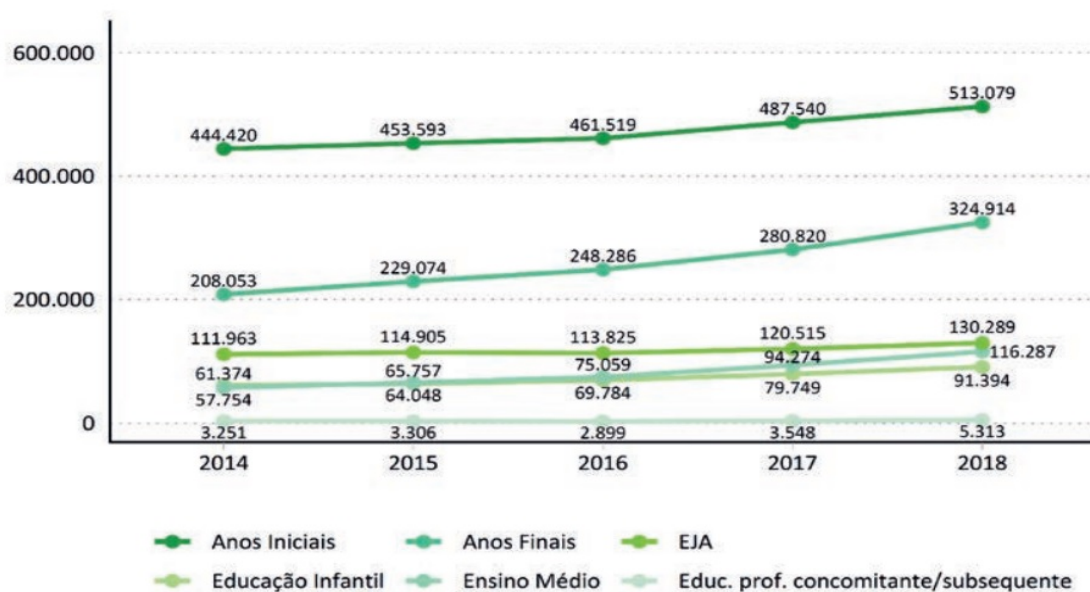


Figura 1: Número de matrículas de alunos com deficiência, transtornos globais do desenvolvimento ou altas habilidades em classes comuns ou especiais exclusivas

fonte: Elaborado pela Deed/Inep com base nos dados do Censo da Educação Básica.

Contribuir com o processo de aprendizado do indivíduo sem possuir o sentido da visão pode ser uma tarefa desafiadora. Embora existam ferramentas capazes de estimular e ensinar o cego algumas disciplinas escolares, tais ferramentas se limitam ao uso na sala de aula ou em ambientes controlados e com poucos recursos. Alunos com alguma restrição visual terão impedimentos no processo de aprendizado se não forem apresentados à recursos didáticos aos quais possam interpretar os conteúdos exibidos em sala de aula.

³<http://inep.gov.br/web/guest/resultados-e-resumos>

O ensino da matemática exige que o aluno seja capaz de identificar uma série de formas que necessitam ser abstraídas através da visão. Perceber uma forma geométrica ou um gráfico cartesiano sem efetivamente vê-lo pode ser uma tarefa bastante árdua, tanto para o professor, que precisa, em muitos casos utilizar a criatividade para fazer com que o cego consiga tocar e interagir com o objeto, quanto para o aluno com deficiência visual, que precisa abstrair tudo que foi falado e ensinado, ao toque em um simples objeto. (BAUNEL; RIBEIRO, 2007)

O desenvolvimento de novas tecnologias e o avanço do poder computacional fazem com que os cegos possam ter um novo tipo de interação com o ambiente, cada vez mais e melhores ferramentas capazes de identificar e classificar objetos sem a necessidade de tocá-los tem surgindo frequentemente. Parte dessa capacidade computacional se dá devido a utilização das Redes Neurais Artificiais (RNAs), estruturas computacionais que se assemelham, ou são inspiradas nas redes neurais dos seres biológicos.

Neste contexto o trabalho aqui apresentado, teve como objetivo propor um modelo de Rede Neural Artificial capaz de identificar e classificar curvas Algébricas. Este projeto visa conciliar áreas de aprendizagem profunda e visão computacional com a área da TA.

1.1 Motivação

Hoje existem uma série de leitores de tela disponíveis voltados para a comunidade cega, no entanto tais ferramentas apenas interpretam caracteres textuais, como letras e números e, ao se deparar com qualquer tipo de imagem, os leitores não realizam a classificação, tornando a informação passada para o indivíduo parcial ou incompleta.

A grande área da matemática hoje abrange uma série de conhecimentos, incluindo geometria, aritmética, álgebra, entre outras. Hoje não se trabalha apenas com números, mas também com relações, categorias, grupos e coleções. Para desempenhar esta série de conceitos, os sistemas de interpretação tornaram-se complexos, compreendendo níveis de subjetividade que desafiam os professores do ensino fundamental e médio. Para Reily, “sem recursos especiais alunos com cegueira terão bastante dificuldade de acompanhar a matéria nas primeiras séries do ensino fundamental, bem como a partir da 5ª série, quando as exigências começam a aumentar.” (REILY, 2004).

Desta maneira evidencia-se a necessidade de ferramentas de auxílio no processo de ensino e aprendizagem da matemática, portanto faz-se como principal fonte de motivação a necessidade de uma ferramenta de apoio aos leitores de tela atuais para realizar a identificação e classificação dos elementos não-textuais apresentados em textos discutidos na sala de aula.

1.2 Objetivos

1.2.1 Objetivo Geral

Este trabalho tem como objetivo arquitetar uma Rede Neural Convolutacional capaz de identificar e classificar curvas Algébricas.

1.2.2 Objetivos Específicos

- Identificar os principais modelos de redes neurais artificiais disponíveis na literatura;
- Desenvolver *datasets* acerca do tema proposto;
- Propor uma arquitetura de rede neural capaz de reconhecer as imagens presentes nos *datasets* desenvolvidos;
- Comparar o modelo de rede neural proposto com modelos consolidados a fim de avaliar e monitorar o desempenho;
- Validar o modelo.

1.2.3 Contribuições da Pesquisa

Os elementos apresentados nesta pesquisa evidenciam contribuições com caráter social, computacional e inovador conforme listadas abaixo:

- Criação de um *dataset* contendo uma série de figuras representando elementos matemáticos;
- Uma RNA pré treinada, capaz de identificar objetos apresentados na disciplina de matemática podendo contribuir em pesquisas futuras sobre esta temática;
- Cooperar na ampliação bibliográfica sobre o tema relacionado a RNCs e visão computacional;
- Propiciar conhecimento a possibilidades de aprendizado sobre gráficos cartesianos;
- Possibilitar, perspectivas de melhora no desempenho acadêmico.

1.3 Organização do Texto

Este trabalho é constituído por cinco capítulos, conforme exibido no Quadro 1

Tabela 1: Organização do Texto

INTRODUÇÃO	Capítulo 1 Introdução	<ul style="list-style-type: none"> 1.1 Motivação 1.2 Objetivos <ul style="list-style-type: none"> 1.2.1 Objetivo Geral 1.2.2 Objetivos Específicos 1.3 Organização do Texto
FUNDAMENTAÇÃO	Capítulo 2 Referencial Teórico	<ul style="list-style-type: none"> 2.1 Revisão Sistemática da Literatura <ul style="list-style-type: none"> 2.1.1 Planejamento 2.1.2 Condução 2.1.3 Fase de Relato 2.1.4 Resultado da Revisão 2.2 O Plano Cartesiano 2.3 Curvas Algébricas no Plano Cartesiano 2.4 Álgebra e a Educação 2.5 Deficiência Visual <ul style="list-style-type: none"> 2.5.1 Deficiência Visual e suas Causas 2.5.2 Deficiência Visual na Educação 2.6 Tecnologia Assistiva para a Educação 2.7 Visão Computacional 2.8 Redes Neurais Artificiais <ul style="list-style-type: none"> 2.8.1 Redes Neurais Convolucionais
PROCEDIMENTOS METODOLÓGICOS	Capítulo 3 Procedimentos Metodológicos	<ul style="list-style-type: none"> 3.1 Proposta do Modelo <ul style="list-style-type: none"> 3.1.1 Desenvolvimento do Dataset 3.1.2 Preparação 3.1.3 Treinamento da Rede 3.2 Primeiro Experimento <ul style="list-style-type: none"> 3.2.1 Análise dos Dados - Primeiro Experimento 3.3 Segundo Experimento <ul style="list-style-type: none"> 3.3.1 Análise dos Dados - Segundo Experimento 3.4 Terceiro Experimento <ul style="list-style-type: none"> 3.4.1 Análise dos Dados - Terceiro Experimento 3.5 Confirmação
RESULTADOS E CONSIDERAÇÕES	Capítulo 4 Considerações Finais	

2 REFERENCIAL TEÓRICO

2.1 Revisão sistemática da literatura

Os autores (KITCHENHAM; CHARTERS, 2007) propõem uma metodologia para o elaboração de estudos secundários, nos quais são dadas as diretrizes a seguir, a fim de realizar um estudo do estado da arte referindo-se a um determinado assunto na área da engenharia de software. Os autores desta metodologia propõem processos que seguem três etapas: i) planejamento, ii) condução e iii) relatórios. No entanto, eles apresentam dois tipos de resultados; o primeiro é uma Revisão Sistemática (RS) da literatura e a segunda, uma RS de mapeamento (ANTHONY; SEVETSON, 2018). No caso do estudo realizado e apresentado nesta seção, decidiu-se realizar um mapeamento sistemático da literatura, considerando e fontes atualizadas de informação.

2.1.1 Planejamento

A fase de planejamento é composta por cinco etapas:

1. Questão de pesquisa e subquestões;
2. Estratégia de busca;
3. Seleção dos estudos;
4. Extração de dados;
5. Métodos de síntese.

2.1.1.1 Questão de Pesquisa e Subquestões

A seguinte pergunta de pesquisa é proposta para a revisão sistemática:

”Quais tecnologias, estratégias e técnicas utilizando aprendizagem profunda estão sendo empregadas com a finalidade de aprimorar o ensino e aprendizagem da comunidade cega e como estas técnicas estão sendo aplicadas e estabelecidas.”

Conseqüentemente as subquestões são definidas por:

- **Q1:** Quais linguagens de programação estão atualmente disponíveis para o desenvolvimento de aprendizagem profunda ?
- **Q2:** Quais aplicações estão sendo desenvolvidas utilizando aprendizagem profunda como meio de auxílio à comunidade cega ?
- **Q3:** Quais aplicações estão sendo desenvolvidas utilizando aprendizagem profunda como meio de auxílio ao ensino a aprendizado voltado à comunidade cega ?
- **Q4:** Quais recursos foram utilizados como meio de integração entre o usuário e a aplicação ?

2.1.1.2 Estratégia de Busca

Esta seção descreve os critérios considerados e as Bases de Dados selecionadas a partir destes critérios. A busca abrange idiomas inglês e português, e o período de cobertura definido entre os anos de 2015 até 2019 totalizando 5 anos de pesquisa.

As Bases de Dados mais relevantes selecionadas e que possuem uma relação direta com o tema são os veículos de publicação científica, anais de eventos e periódicos, bem como revistas de renome científico possuindo publicações na área da informática e educação constam na Tabela 2. Neste escopo temos quatro Bases de Dados de abrangência nacional e três internacionais, totalizando sete Bases de Dados consultadas.

Tabela 2: Bases de Dados selecionadas para Revisão Sistemática

Bases de Dados Internacionais

IEEE Explore Digital Library

Springer Link

ScienceDirect

Bases de Dados Nacionais

Revista Brasileira de Informática na Educação (RBIE)

SciELO Sociedade Brasileira de Computação

Revista Novas Tecnologias na Educação (RENOTE)

Simpósio Brasileiro de Informática na Educação (SBIE)

Para realizar a busca automática nos repositórios online, as seguintes palavras-chave foram selecionadas: *assistive, blind people, deep learning, teaching, learning, visually impaired*.

A partir das palavras-chave descritas anteriormente, é necessário escolher os conectores lógicos (*and, or, not*) com a finalidade de otimizar a busca. Sete estratégias de busca foram traçadas, cada estratégia tem como objetivo cruzar as *strings* para maximi-

zar o número de trabalhos relacionados e garantir que apenas os artigos relevantes para a pesquisa sejam relacionados, as estratégias foram as seguintes:

- *(assistive OR blind AND people) AND deep learning*
- *(teaching AND learning) AND deep learning*
- *(blind AND teaching) AND deep learning*
- *(blind people and learning) AND deep learning*
- *Blind and deep learning*
- *(blind AND assistive) AND deep learning*
- *Visually Impaired AND deep learning*

A concatenação das sete estratégias de busca resultaram na seguinte *string* final: *((assistive OR blind AND people) AND deep learning) OR ((teaching AND learning) AND deep learning) OR ((blind and teaching) AND deep learning) OR ((blind people and learning) AND deep learning) OR (Blind and deep learning) OR ((blind AND assistive) AND deep learning) OR (Visually Impaired AND deep learning)* .

2.1.1.3 Seleção dos Estudos

A busca foi realizada aplicando a cadeia de pesquisa (*string*) aos mesmos metadados (título, resumo e palavras-chave) em cada uma das Bases de Dados mencionadas anteriormente, os resultados obtidos são apresentados na Tabela 3

Bases de Dados Internacionais	Artigos
<i>IEEE Explore Digital Library</i>	261
<i>Springer Link</i>	312
<i>ScienceDirect</i>	177
Bases de Dados Nacionais	Artigos
Revista Brasileira de Informática na Educação (RBIE)	0
SciELO Sociedade Brasileira de Computação	0
Revista Novas Tecnologias na Educação (RENOTE)	0
Simpósio Brasileiro de Informática na Educação (SBIE)	0
Total	750

Um total de 750 artigos foram encontrados nas bases internacionais e nenhum trabalho nas bases nacionais. Testes com *strings* simplificadas foram realizados a fim de

expandir as buscas nos repositórios nacionais, no entanto nenhum trabalho significativo e relacionado com o tema foi encontrado.

Para garantir a qualidade dos trabalhos apresentados, apenas os estudos que atendam à pelo menos um dos seguintes critérios serão incluídos:

- Estudos que apresentem o uso de *deep learning* como meio de auxílio ao ensino e aprendizagem de cegos;
- Estudos que utilizam *deep learning* como meio de identificação e classificação de objetos;
- Estudos que apresentem técnicas, estratégias e metodologias no uso de *deep learning* para detecção e classificação de objetos.

E os trabalhos que atendam à pelo menos um dos critérios abaixo serão excluídos:

- Artigos introdutórios para questões especiais;
- Livros e *workshops*;
- Relatórios duplicados do mesmo estudo em diferentes fontes;
- Artigos curtos com menos de cinco páginas e não escrito em inglês ou português;
- Artigos que não possuem relação com o tema;

Após realizar os processos de inclusão e exclusão, a seleção final dos trabalhos contou com um total de 79 estudos, divididos entre 71 anais de conferências e 8 artigos em revistas apresentado na Tabela 4

Tabela 4: Resultados Final da busca nas Bases de Dados

Bases de Dados Internacionais	Artigos
<i>IEEE Explore Digital Library</i>	58
<i>Springer Link</i>	17
<i>ScienceDirect</i>	4
Bases de Dados Nacionais	Artigos
Revista Brasileira de Informática na Educação (RBIE)	0
SciELO Sociedade Brasileira de Computação	0
Revista Novas Tecnologias na Educação (RENOTE)	0
Simpósio Brasileiro de Informática na Educação (SBIE)	0
Total	79

Deste total de 79 trabalhos, 58 foram selecionados da base *IEEE Explore Digital Library*, 17 do repositório *Springer Link* e 4 do *ScienceDirect*.

2.1.1.4 Extração de dados

A estratégia de extração dos dados baseia-se em fornecer o conjunto de respostas possíveis para cada sub-questão de pesquisa definida anteriormente. Esta estratégia garante a aplicação dos mesmos critérios de dados de extração para todos os documentos, facilitando sua classificação, conforme apresentado na Tabela 5

Tabela 5: Dados extraídos a partir da estratégia definida

Pergunta de pesquisa	Resultados
Q1:	
C	21
C++	3
CUDA	2
Python	53
Q2:	
Reconhecimento de linguagem de sinais	12
Navegação e Mobilidade	20
Deteção e classificação de objetos	28
Assistente para tarefas sociais	4
Outros	15
Q3:	
Plataforma de ensino	1
Reconhecimento de sinais	14
Navegação em ambiente acadêmico/escolar	2
Ferramentas para Braile	3
Q4:	
Dispositivo móvel	12
Vestimenta	23
Não especificado	15
Computador pessoal (webcam)	29

Todos os trabalhos selecionados apresentavam resposta para a pergunta **Q1: Quais linguagens de programação estão atualmente disponíveis para o desenvolvimento de aprendizagem profunda ?** As linguagens de programação encontradas foram C, C++, CUDA e Python. Trabalhos que não referenciavam diretamente a linguagem utilizada, faziam menção ao *framework* de desenvolvimento empregado, e, a partir desta informação foi possível relacionar a linguagem de programação aplicada. Desta forma

podemos entender melhor as preferências dos pesquisadores em se tratando de linguagens de programação. De acordo com as pesquisas a linguagem mais utilizada é o Python com 53 trabalhos, e a menos utilizada é a CUDA, com 2 trabalhos desenvolvidos.

Os trabalhos mais relevantes que contemplavam a pergunta **Q2: Quais aplicações estão sendo desenvolvidas utilizando aprendizagem profunda como meio de auxílio à comunidade cega ?** Apresentavam as seguintes aplicações: reconhecimento de linguagem de sinais, navegação e mobilidade, detecção e classificação de objetos e assistente para tarefas sociais. Estamos considerando o reconhecimento de linguagem de sinais dentro do escopo deste estudo por entender que o reconhecimento de linguagem de sinais pode ser utilizado como meio de comunicação entre cegos e surdos, também por ser um estudo bastante aprofundado na área de redes neurais artificiais e aprendizagem profunda, foco principal deste trabalho. Considera-se para esta análise assistente para tarefas sociais, auxílio para compras e identificação de prateleiras. Detecção e classificação de objetos foram os mais encontrados, com total de 28, já os trabalhos que menos apareceram nesta análise foram os assistentes para tarefas sociais, com apenas 4 artigos.

A pergunta **Q3: Quais aplicações estão sendo desenvolvidas utilizando aprendizagem profunda como meio de auxílio ao ensino a aprendizado voltado à comunidade cega ?** Apenas 20 trabalhos dos 79 apresentavam algum tipo de tecnologia de auxílio ao ensino e educação para cegos. O trabalho de (GRANADA; BARWALDT; ESPÍNDOLA, 2018) no desenvolvimento de um glossário de termos técnicos computacionais para surdos como auxílio no ensino e aprendizado, da mesma forma entendemos que linguagem de sinais, conforme citado anteriormente, pode e deve ser utilizado como meio de interação entre cegos e surdos auxiliando no processo de ensino e aprendizagem. Nesta análise temos a modalidade reconhecimento de sinais com 14 artigos com maior incidência e com menor, plataforma de ensino com apenas 1 trabalho.

Para responder a pergunta **Q4: Quais recursos foram utilizados como meio de integração entre o usuário e a aplicação ?** Encontramos 3 tipos de ferramentas, dispositivos móveis, vestimentas e computador pessoal. Consideramos vestimenta qualquer elemento que necessite ser vestido para realizar a interação, óculos, bengala, luvas podem ser exemplificados como itens vestíveis. Já computador pessoal representa o meio de interação a partir do recebimento de imagens de outras fontes. Alguns trabalhos não especificaram a ferramenta, portanto o campo 'não especificado' representa estes trabalhos. Nesta análise temos a ferramenta computador pessoal como mais representativo, com 29 artigos e, dispositivo móvel como menos representativo, relatado com 12 trabalhos.

2.1.1.5 Métodos de Síntese

Neste estágio é realizado o complemento das tabelas descritas anteriormente com gráficos para melhor visualização e entendimento dos dados, bem como novas informações serão adicionadas a fim de estender a compreensão da análise realizada.

O gráfico indicado pela Figura 2 apresenta os dados referentes a primeira rodada de buscas nos repositórios selecionados. Nesta etapa de busca, o maior número de trabalhos encontrados pertence a Base de Dados *SpringerLink*, esta representa 41.06% do total.

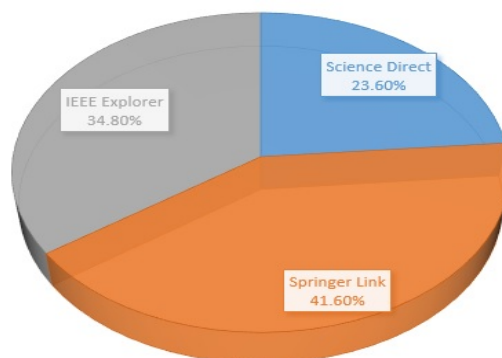


Figura 2: Restultado da busca automática

Fonte: autor

A Figura 3 apresenta os dados referentes aos trabalhos selecionados após a etapa de aplicação dos critérios de inclusão e exclusão. Nesta etapa o repositório *IEEE Explore* apresentou os trabalhos que melhor se enquadraram com a proposta desta revisão, constituindo 73.04% dos trabalhos.

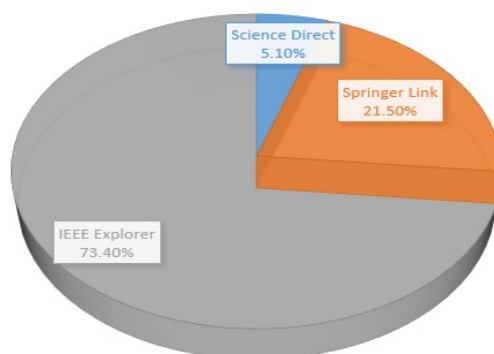


Figura 3: Resultado após critérios de inclusão e exclusão

Fonte: autor

A Figura 4 apresenta a disposição dos trabalhos publicados por ano. Dos 79 artigos selecionados após os critérios de inclusão e exclusão, seis artigos foram publicados no ano de 2015, sete em 2016, quinze em 2017, 39 em 2018 e até o final desta pesquisa em 2019, doze artigos foram publicados que atenderam aos critérios.

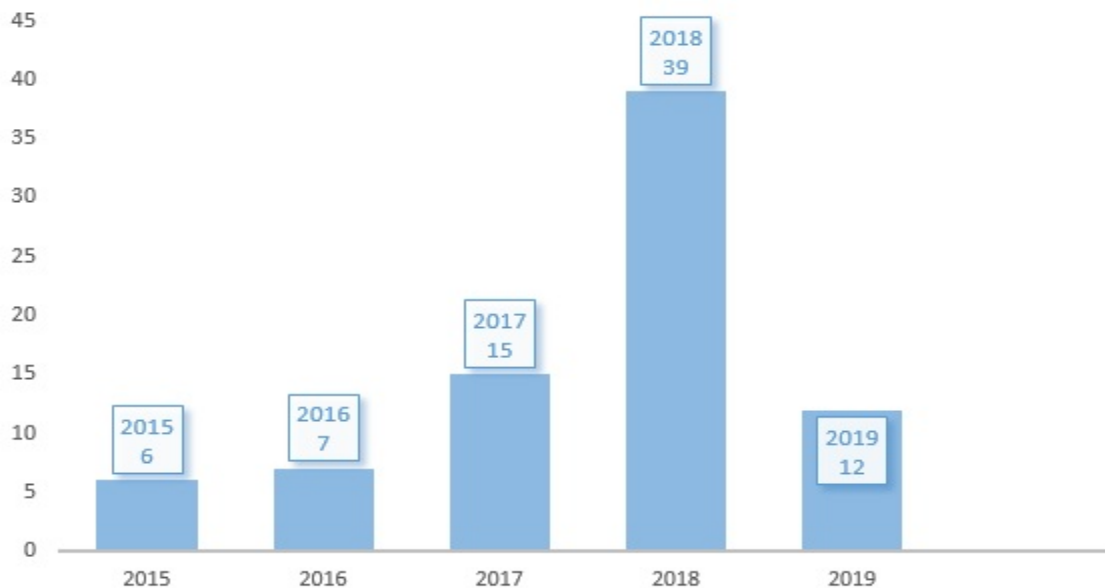


Figura 4: Relação de trabalhos por ano

Fonte: autor

Em relação as perguntas **Q1**, **Q2**, **Q3**, **Q4** foi possível encontrar nos trabalhos selecionados um total de 15 elementos foram identificados.

A Figura 5 representa as linguagens de programação encontradas nos 79 trabalhos selecionados que contemplam a pergunta **Q1**. Python se destacou como a linguagem mais utilizada pelos pesquisadores e desenvolvedores de aplicações utilizando aprendizagem profunda e redes neurais artificiais.

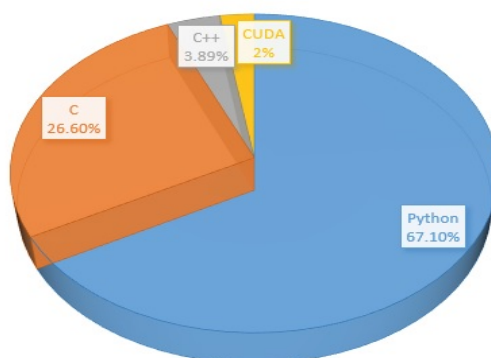


Figura 5: Linguagens de programação utilizadas nos trabalhos

Fonte: autor

A Figura 6 representa as aplicações encontradas nos 79 trabalhos selecionados que contemplam a pergunta **Q2**. Aqui temos a detecção e classificação de objetos se destacando como principal aplicação para aprendizagem profunda.

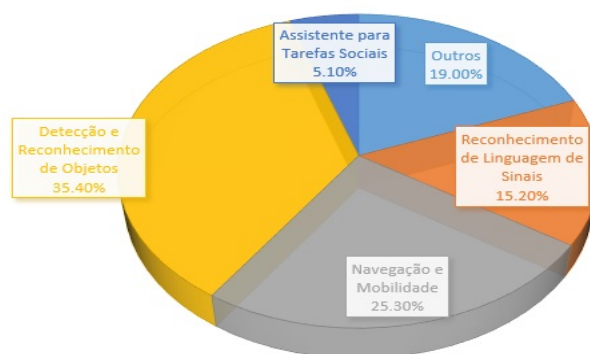


Figura 6: Tipo de aplicações desenvolvidas

Fonte: autor

Na Figura 7 podemos analisar as aplicações desenvolvidas que contemplam a pergunta **Q3**, este quesito leva em conta os trabalhos que utilizam aprendizagem profunda e possuem aplicações no contexto de ensino e aprendizagem para cegos. Conforme definido anteriormente, considera-se para este trabalho reconhecimento de linguagem de sinais com aplicação também para cegos, portanto, dos 20 artigos que apresentam algum tipo de auxílio no ensino e aprendizagem para cegos, as ações que destacaram-se foram as de reconhecimento de linguagem de sinais.

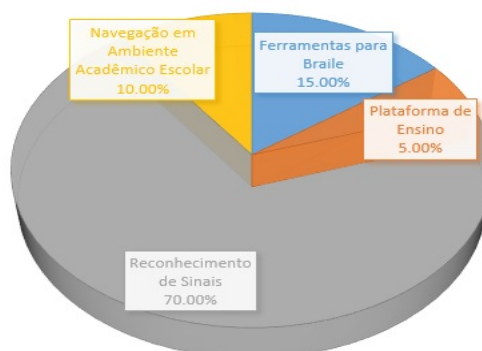


Figura 7: Aplicações com viés para ensino e aprendizagem para cegos

Fonte: autor

A Figura 8 representa as ferramentas encontradas que contemplam a pergunta **Q4**. Neste elemento temos o item computador pessoal como instrumento mais utilizado para realizar os estudos e aplicações dos artigos seleccionados.

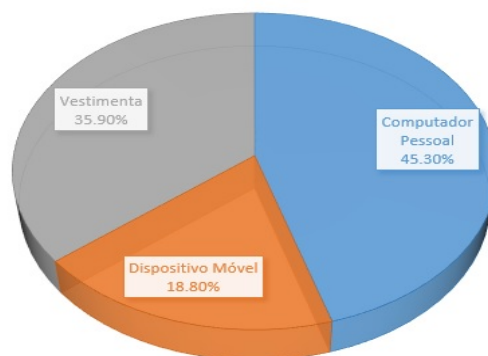


Figura 8: Ferramentas utilizadas como meio de interação

Fonte: autor

2.1.2 Condução

Com a aplicação do protocolo de revisão, são produzidos resultados preliminares. Os resultados serão selecionados de acordo com os critérios de inclusão. Nesta etapa, podem ser encontrados alguns estudos que possivelmente tiveram publicado em mais de uma revista / artigo e neste caso, é feita a seleção de apenas a versão mais completa do estudo. Quando encontrar alguns estudos em mais de uma fonte, deve-se levar em conta apenas uma vez de acordo com a ordem de pesquisa, que é o seguinte: *IEEE Xplore, Springer Link, ScienceDirect*.

2.1.3 Fase de Relato

A fase final de um mapeamento sistemático envolve escrever os resultados da revisão e apresentando os dados para os possíveis interessados. Estes desfechos foram mostrados sequencialmente em fases anteriores para o melhor entendimento.

2.1.4 Resultados da revisão

Esta seção apresenta os resultados das tabulações do mapeamento sistemático. Na tabela 5, serão apresentadas as sub-questões investigadas através da análise detalhada dos 79 artigos, selecionados a partir de 2015 até 2019.

Entende-se então que o principal meio de interação entre o usuário cego e a ferramenta desenvolvida é o computador pessoal, este realizando processos através da captura de imagens via câmeras. O reconhecimento de sinais em Libras é a aplicação mais pesquisada tratando-se em aprendizagem profunda voltada para o processo de ensino-aprendizagem de cegos. A detecção e classificação de objetos que utiliza aprendizagem profunda para realizar o processamento, e como linguagem de programação envolvendo todos os processos de desenvolvimento, a linguagem Python é a que mais apresentou resultados.

2.2 O Plano Cartesiano

Um plano cartesiano é um gráfico com um eixo x e um eixo y . Esses dois eixos são perpendiculares entre si. A origem (O) está no centro exato do gráfico. Os números à direita do zero no eixo x são positivos; números à esquerda de zero são negativos. Para o eixo y , números abaixo de zero são negativos e números acima são positivos.

René Descartes (1596 - 1650) um dos fundadores da geometria analítica na década de 1630, adaptando a álgebra de François Viète ao estudo geométrico. Ele seguiu em direção contrária à Viète, usando letras para representar distâncias variáveis em vez de fixas. Descartes demonstrou seu método em um problema clássico: encontrar todos os pontos P de modo que o produto das distâncias de P a certas linhas seja igual ao produto das distâncias para outras linhas (D'SOUZA; BIX, 2016), ele usou coordenadas negativas livremente, embora tenha sido Isaac Newton quem inequivocamente usou dois eixos (oblíquos) para dividir o plano em quatro quadrantes, como mostra a Figura 9.

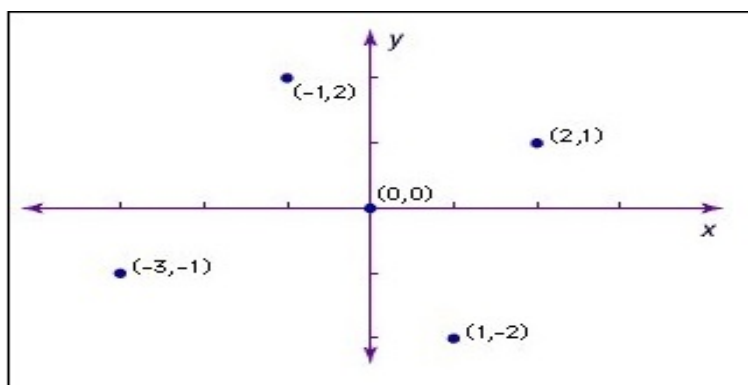


Figura 9: Exemplo de Plano Cartesiano com Coordenadas Positivas e Negativas

Fonte: adaptado de Encyclopædia Britannica

A localização de um ponto no plano é dada por suas coordenadas, um par de números entre parênteses: (x, y) . O primeiro número x indica a posição horizontal do ponto e o segundo número y indica a posição vertical. Todas as posições são medidas em relação a um ponto "central" chamado origem, cujas coordenadas são $(0,0)$. Por exemplo, o ponto $(5,2)$ é de 5 unidades à direita da origem e 2 unidades acima. Os números negativos das coordenadas nos dizem para ir para a esquerda ou para baixo.

As coordenadas no plano cartesiano fundamentam o campo da geometria analítica e contribui através das interpretações geométricas esclarecedoras para muitos outros ramos da matemática, como álgebra linear, análise complexa, geometria diferencial, cálculo multivariado, teoria de grupos e muito mais. As coordenadas cartesianas também são ferramentas essenciais para a maioria das disciplinas aplicadas que lidam com geometria, incluindo astronomia, física, engenharia e muito mais. Eles são o sistema de coordenadas mais comum usado em computação gráfica, desenho geométrico auxiliado por computa-

dor e outro processamento de dados relacionado à geometria (AXLER, 2015).

As coordenadas são uma abstração que possui inúmeras aplicações possíveis no mundo real. No entanto, três etapas construtivas estão envolvidas na sobreposição de coordenadas em um aplicativo com problemas. 1) As unidades de distância devem ser decididas definindo o tamanho espacial representado pelos números usados como coordenadas. 2) Uma origem deve ser atribuída a um local ou ponto espacial específico e 3) a orientação dos eixos deve ser definida usando pistas direcionais disponíveis para todos, exceto um eixo. E através das coordenadas no plano cartesiano são desenhadas as curvas cartesianas, discutido na próxima seção deste trabalho.

2.3 Curvas Algébricas no Plano Cartesiano

Uma curva é todo o lugar geométrico de pontos do espaço, cujas coordenadas cartesianas, em determinado referencial cartesiano, são funções contínuas de um parâmetro real, definidas num intervalo (SILVA, 1969). Pode ser representada também por uma figura geométrica gerada pelo movimento contínuo de um ponto no espaço. A curva algébrica é especificada em coordenadas cartesianas, Descartes usou equações para estudar curvas definidas geometricamente e enfatizou a necessidade de considerar curvas algébricas gerais - gráficos de equações polinomiais em x e y de todos os graus, tornando o estudo das curvas tão antigo quanto o desenvolvimento das coordenadas cartesianas.

A curva algébrica sobre um campo κ é uma equação $F(X, Y) = 0$, onde $F(X, Y)$ é um polinômio em X e Y com coeficientes em κ . Uma curva algébrica não singular é uma curva algébrica sobre κ que possui nenhum ponto singular sobre κ . Um ponto em uma curva algébrica é simplesmente uma solução da equação da curva. Um ponto racional κ é um ponto (X, Y) na curva, onde X e Y estão no campo κ (ARBARELLO et al., 2007). As curvas estão divididas em grupos de acordo com seu grau, conforme a tabela 6 demonstra.

Tabela 6: Lista de Curvas Algébricas de Acordo com o Grau - Fonte autor

Grau	Tipo	Curva
2	Curva Quadrática	Círculo, Elipse, Hipérbole, Parábola
3	Curva Cúbica	Cissoide de Diocles, Conchóide de Sluze, Fólio de Descartes, Trisectrix de Maclaurin, Curva da Cruz de Malta, Curva de Mordell, Curva de Ochoa, Estrofóide Direito, Parábola Semicúbica, Curva Serpentina, Tschirnhausen Cúbico, Bruxa de Agnesi
4	Curva Quártica	Curva e Comercial, Curva de Feijão, Bicórnio, Curva Bicúspide, Bifoliado, Bifólio, Curva Rica em Tangentes, Arco, Nariz de Bala, Curva de Borboleta, Capricornóide, Cardióide, Ovais Cartesianas, Ovais de Cassini, Conchóide de Nicomedes, Cruciforme, Deltóide, Curva do Diabo, Curva de Dürer Conchóide, Curva Oito, Curva de Peixe, Fólio, Hipopótamo, Kampyle de Eudoxus, Klein Quártica, Curva de nó, Lemniscate, Limaçon, Curva de Links, Curva em Forma de Pêra, Curva Piriforme, Curva de Suástica, Curva de Trifólio, Trifólio
5	Curva Quintica	Curva de Burnside, Curva de Catástrofe de Borboleta, Curva de Estribo
6	Curva Sextica	Astroide, Atrifalóide, Sexóide de Cayley, Cornoide, Ciclóide de Ceva, Curva de Halteres, Elipse Evoluída, Epiciclóide, Nefróide de Freeth, Curva do Coração, Limaçon Evolutora, Nefróide, Quadrifólio, Curva de Scarabaeus, Curva de Talbot
8	Curva Octica	Curva de Pêra
12	Curva Dodecica	Ranunculoide

Cada curva Algébrica é representada no plano cartesiano a partir de uma equação geral, esta pode possuir características singulares para um grupo de um determinado grau, mas cada curva individual possui a sua. Estas equações possuem relação direta com o cálculo, com a matemática e com a álgebra linear, assim trabalhada, em algum grau, ao longo de todo o ciclo acadêmico e, entender o processo de ensino acerca deste tema é o tópico discutido na próxima seção.

2.4 Álgebra e a Educação

O processo de ensino da álgebra sempre foi alvo de discussões, os diferentes pontos de vista acerca dos objetivos e abordagens são os principais tópicos abordados (SCHONFELD, 2004). Crucial nesses debates é a relação entre habilidades processuais e compreensão conceitual no ensino e aprendizagem de álgebra.

Alguns vêem o núcleo da educação em álgebra como o desenvolvimento de habilidades estratégicas de resolução de problemas e raciocínio, senso de símbolo e flexibilidade, em vez de fluência processual. Se o ensino e a aprendizagem da álgebra podem se beneficiar da integração da tecnologia, a pergunta subsequente deve ser qual tipo de tecnologia usar e quais critérios determinam essa escolha (BOKHOVE; DRIJVERS, 2010). Muitos tipos diferentes de ferramentas de software estão disponíveis, cada um oferecendo oportunidades e restrições para diferentes estruturas de atividades e até diferentes tipos de conhecimento a surgir. Não é uma questão simples prever esses efeitos e decidir adequadamente sobre quais ferramentas envolver no processo de aprendizagem e por quê. O que é adequado depende, é claro, dos objetivos e visões da educação em álgebra, das aquisições e aprendizado de conhecimento, bem como dos fatores situacionais.

Na Álgebra, no entanto, há um nível de abstração que provoca, tanto na educação básica quanto no ensino superior, um momento de ruptura com conceitos e procedimentos já internalizados pelos alunos. Temos diferentes estilos de aprendizagem, ou seja, características e preferências quanto à forma de apropriação e processamento das informações para construir novos conhecimentos (CURY et al., 2002)

Percebe-se uma grande dificuldade tanto no ensino quanto na aprendizagem da álgebra em todos os níveis do currículo acadêmico, existe uma questão de abstração envolvida no processo de aprendizagem onde, o aluno percebe através da visão os elementos algébricos por meio de gráficos, para então subjetivar em busca de "pistas" para dar a resposta. Neste contexto não está sendo considerado indivíduos cegos ou com baixa visão estes, não conseguem se valer da visão para realizar a abstração dos elementos trabalhados. Nas próximas seções serão apresentados alguns meios de interação entre o deficiente visual no processo educacional.

2.5 Deficiência Visual

A OMS classifica a deficiência visual em classes que compreendem desde a redução visual leve até a ausência completa da visão. A Classificação Estatística Internacional de Doenças e Problemas Relacionados à Saúde (CID-10) estipula o conceito de cegueira em função da acuidade visual corrigida no melhor olho. A Classificação Internacional de Funcionalidade, Incapacidade e Saúde (CIF) descreve a funcionalidade e a incapacidade relacionadas às condições de saúde, identificando o que uma pessoa pode ou não pode fazer na sua vida diária (FARIAS; BUCHALLA, 2005), considerando as funções

dos órgãos ou sistemas e estruturas do corpo, bem como as limitações de atividades e participação social no meio ambiente onde vive.

Uma pessoa é considerada deficiente quando apresenta em caráter permanente, perdas ou reduções significativas em suas funções físicas ou mentais, ocasionando dificuldades ou incapacidades para realizar determinadas tarefas consideradas comuns ou simples quando comparadas a uma pessoa dita 'normal'.

Conforme a Figura 10 demonstra os declarados com incapacidade visual no Brasil são 500 mil, já os que possuem grande dificuldade de enxergar são 6,5 milhões de habitantes, somando pelo menos 7 milhões de pessoas com grave deficiência visual. Dados atuais não foram divulgados pelo IBGE, portanto não estão presentes neste estudo, no entanto o número de deficientes visuais pode ter crescido consideravelmente de 2010 até a presente data.

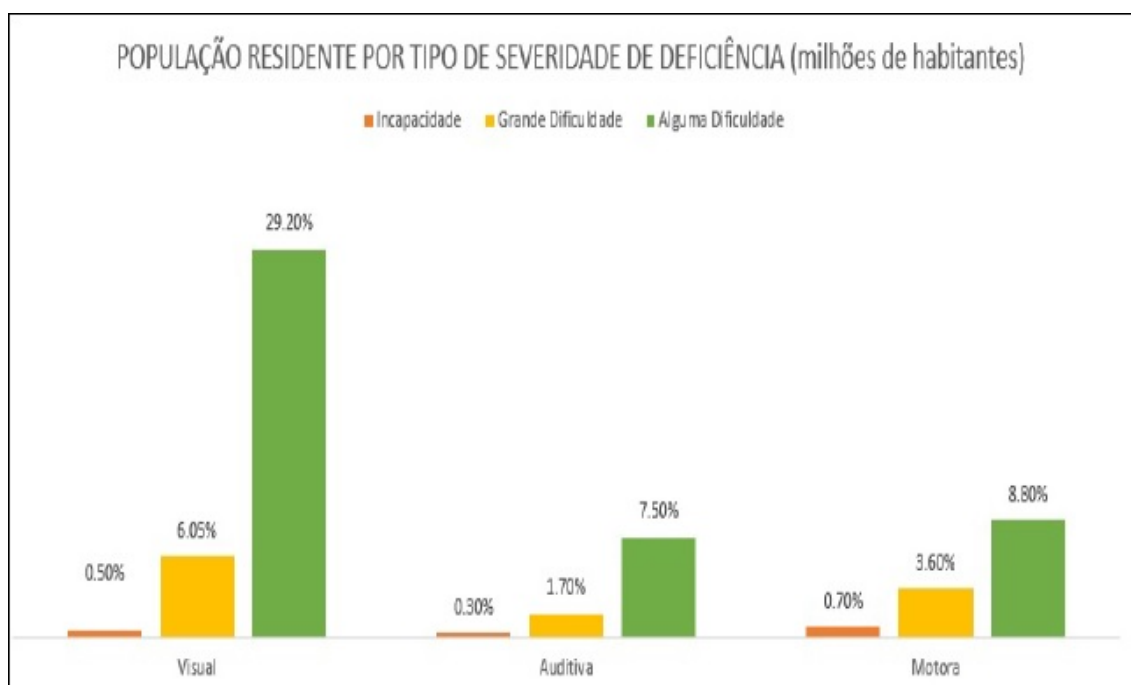


Figura 10: Tipos de deficiência
Fonte: IBGE, Censo Demográfico - 2010

Estes dados auxiliam na compreensão da necessidade da construção de uma sociedade inclusiva, e naturalmente o tamanho deste desafio, para que as diferenças sejam deixadas de lado e o direito a cidadania seja colocado acima de qualquer individualidade.

”As pessoas com deficiência têm os mesmos direitos que as pessoas sem deficiência. No entanto, elas nem sempre conseguem clamar por seus direitos nas mesmas condições em que o fazem as pessoas sem deficiência, devido a desvantagens impostas pela restrição de funcionalidades e pela sociedade, que lhes impõe barreiras físicas, legais e de atitude. Essas barreiras são responsáveis pelo distanciamento que existe na

realização dos direitos das pessoas com e sem deficiência.”(OLIVEIRA, 2012, p.14)

No Brasil, o documento oficial que define a deficiência visual é o Decreto¹ nº 3.298, de 20 de dezembro de 1999, no seu artigo 4o, inciso III que estabelece: “deficiência visual- acuidade visual igual ou menor que 20/200 no melhor olho, após a melhor correção, ou campo visual inferior a 20 na tabela de Snellen, ou ocorrência simultânea de ambas as situações”.

A visão é um aferente capaz de promover a integração com outras informações sensoriais. É um estímulo contínuo que possibilita condutas motoras auto-dirigidas e intencionais, críticas para localizar e identificar objetos distantes que não podem ser capturados por outros sentidos, entender as relações espaciais, a posição do corpo em relação ao espaço, capturar de forma eficaz e rápida o ambiente de forma segura e confiante, mantendo postura corporal adequada, entre outras funções (MANCINI et al., 2010).

Considerando que deficiência visual é um dos grandes desafios da atualidade, não apenas no Brasil mas no mundo todo, é de suma importância que esta parcela significativa da população receba mais respeito e atenção.

Então em relação a deficiência visual, podemos afirmar que os recursos tecnológicos disponíveis hoje em dia, representam um grande avanço para a inclusão social destas pessoas. Os autores (BARQUEIRO; BARQUEIRO, 2010) mencionam que a informática e outras tecnologias mudaram de forma significativa as oportunidades de acessibilidade e inclusão dos indivíduos com deficiência visual em todas as áreas, profissional, educacional, cultural entre outras.

2.5.1 Deficiência visual e suas causas

A cegueira é considerada a perda total da visão ou pouquíssima capacidade visual (cegueira parcial). A perda total da visão, também chamada de Amaurose, ocorre quando o indivíduo não possui nenhuma capacidade de percepção de luminosidade. No caso da cegueira parcial, existe percepção de luz, sendo possível apenas diferenciar a direção da fonte, o claro e o escuro. Já a visão subnormal é dada pela alteração da capacidade visual causada pelo comprometimento dos olhos, onde o indivíduo possui acuidade visual reduzida e não podendo ser corrigido com óculos, lentes ou intervenções cirúrgicas.

Os dados sobre as causas da deficiência visual e da cegueira constituem uma base importante para recomendações em políticas públicas de saúde, como o planejamento de orçamentos nacionais e serviços de saúde, e são importantes para a pesquisa científica.

Estudos populacionais realizados nos últimos 20 anos revelaram que catarata, glaucoma, degeneração macular, retinopatia diabética, tracoma e erro refrativo não corrigido são as causas mais comuns de cegueira e deficiência visual em todo o mundo (BOURNE

¹<http://portal.mec.gov.br/arquivos/pdf/decreto3298.pdf>

et al., 2013).

Diferenças importantes entre grupos étnicos, regiões de habitação, parâmetros demográficos, estilo de vida, exposição a fatores ambientais e outros fatores, no entanto, podem afetar as causas e a prevalência da perda de visão.

As estimativas das principais causas de deficiência visual em todo o mundo foram geradas por metanálises. A primeira delas realizada por (THYLEFORS et al., 1995) em 1995, estimou que 38 milhões de pessoas eram cegas, principalmente devido a catarata, tracoma e glaucoma. No entanto, devido ao crescimento populacional e ao aumento do número de adultos idosos, a população cega permaneceu estável e o número de indivíduos com deficiência visual pode ter aumentado (STEVENS; WHITE; FLAXMAN, 2013)

A pesquisa *Global Data on Visual Impairments*² apontou que globalmente as principais causas de deficiência visual são erros de refração não corrigidos e catarata, 43% e 33%, respectivamente. Outras causas são glaucoma, 2%, degeneração macular relacionada à idade (DMRI), a retinopatia diabética, tracoma e opacidades corneanas, todo cerca de 1%. Uma grande proporção de causas, 18%, são indeterminadas, conforme apontado a Figura 11.

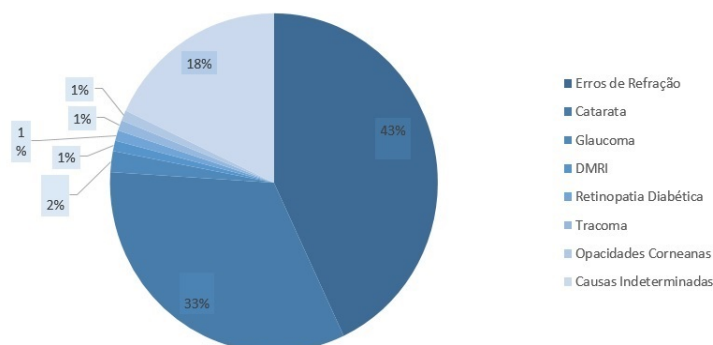


Figura 11: Causas da Cegueira e Deficiência Visual no Mundo.

Fonte: (ÁVILA; ALVES; NISHI, 2015)

2.5.2 Deficiência Visual no Sistema Educacional

Grande parte do esforço educacional voltado ao cego tem como premissa a sua inclusão social, a primeira escola para cegos conhecida foi fundada em Paris, em 1784. Um de seus alunos, Louis Braille acabou desenvolvendo um sistema de leitura tátil, que não apenas continha letras, mas também possuía números e notas musicais.

A partir deste sistema Braille, uma série de escolas para cegos vieram a ser fundadas por toda Europa, uma escola que se destacou bastante nesta época foi a inglesa *st. Dunstan*, conhecida por dar ênfase em aspectos psicológicos dos alunos.

A primeira escola residencial para cegos foi estabelecida na América em 1829, era o Asilo para Cegos da Nova Inglaterra. O termo asilo foi usado nos nomes da maioria das

²<https://www.who.int/blindness/publications/globaldata/en/>

primeiras escolas. Hoje, o *New England Asylum* é a Escola Perkins para Cegos, localizada em Watertown, Massachusetts.

Em 1832, uma instituição para a educação de crianças cegas, agora a *Overbrook School for the Blind*, foi estabelecida na Filadélfia, e o conceito de escolas residenciais para cegos varreu rapidamente para o Ocidente através da América.

Em 1918, nos Estados Unidos o Braille foi aceito como padrão nacional de leitura tátil para cegos. Antes de 1918, havia vários métodos táteis concorrentes e havia fortes diferenças de opinião entre os principais profissionais da cegueira da época.

No Brasil, a primeira instituição especializada foi o Imperial Instituto dos Meninos Cegos, criado em 1854, hoje Instituto Benjamin Constant, no Rio de Janeiro.

Vygotsky contribuiu para a educação através de várias questões apresentadas em sua pesquisa, no caso do Brasil, ficou conhecido a partir da divulgação de seus textos por (CARVALHO, 2002), (OLIVEIRA, 1992), (OLIVEIRA, 1995) e (ZANELLA, 1994). O autor tem várias análises sobre o papel da interação social na formação da compreensão do indivíduo, e suas colaborações no estudo da relação entre desenvolvimento e aprendizagem, e suas afirmações sobre o papel da educação no desenvolvimento psicológico são consideradas na pesquisa sobre educação.

Em seu trabalho, a preocupação frequente relacionada às pessoas com deficiência visual tem sido notada, e pode ser evidenciado a partir de textos em que o autor elabora não apenas princípios gerais da educação de pessoas com deficiência, mas também passagens em que lida exclusivamente com os problemas do desenvolvimento psicológico dos cegos, (VYGOTSKI, 1997) já aborda novas tecnologias como meio de acesso dos cegos às formas de geração de conhecimento na modernidade.

Ao questionar o fenômeno da cegueira, (VYGOTSKY, 1993) argumentou que a ausência de visão requer que se gere faculdades compensatórias tais que tornem a cegueira não meramente um defeito, 'um menos', uma fraqueza, mas em certo sentido é também fonte de manifestações e habilidades, um *plus*, uma força.

”as crianças em desenvolvimento tornavam-se não mais desenvolvidas, mas desenvolvidas de um modo diferente, fazendo uso de um outro conjunto de instrumentos. Vygotsky, conseqüentemente opunha-se a todos os procedimentos diagnósticos que fossem baseados em uma abordagem puramente quantitativa.”(VEER; VALSINER, 1996, p.84)

2.6 Tecnologia Assistiva para a Educação

O Ministério da Cultura sempre se preocupou com o uso de tecnologias como meio de ajudar o aluno cego, sugerindo em documentos oficiais, uso de ferramentas auxiliares para o ensino, seja usando técnicas digitais ou não.

”A predominância de recursos didáticos eminentemente visuais ocasiona uma visão fragmentada da realidade e desvia o foco de interesse e de motivação dos alunos cegos e com baixa visão. Os recursos destinados ao Atendimento Educacional Especializado desses alunos devem ser inseridos em situações e vivências cotidianas que estimulem a exploração e o desenvolvimento pleno dos outros sentidos. A variedade, a adequação e a qualidade dos recursos disponíveis possibilitam o acesso ao conhecimento, à comunicação e à aprendizagem significativa.”(Sá; CAMPOS; SILVA, 2007, p.26)

Segundo dados do Censo Escolar 2008/2009, realizado pelo Inep/MEC, mais de 55 mil alunos têm deficiência visual no país. Desse total, 51.311 são os chamados estudantes com baixa visão e 4.604 são cegos.

A seguir, na Tabela 7, são apresentadas as categorias de classificação de TA no contexto educacional, que assim como o texto explicativo de cada uma, essas foram adaptadas e extraídas de (BERSCH, 2017).

Categoria	Aplicação
1 - Comunicação Aumentativa e Alternativa	Destinada a atender pessoas sem fala ou escrita funcional ou em defasagem entre sua necessidade comunicativa e sua habilidade em falar e/ou escrever. Ex: Prancha de comunicação impressa e vocalizadores de mensagens gravadas.
2 - Recursos de Acessibilidade ao Computador	Conjunto de hardware e software especialmente idealizado para tornar o computador acessível a pessoas com privações sensoriais (visuais e auditivas), intelectuais e motoras. Ex: Mouses, teclado Braile e mapa tátil com impressão em relevo.
3 - Projetos Arquitetônicos para Acessibilidade	Projetos de edificação e urbanismo que garantem acesso, funcionalidade e mobilidade a todas as pessoas, independentemente de sua condição física e sensorial. Ex: Acessibilidade no banheiro.
4 - Auxílios de Mobilidade	A mobilidade pode ser auxiliada por bengalas, muletas, andadores, carrinhos, cadeiras de rodas manuais ou elétricas, scooters e qualquer outro veículo, equipamento ou estratégia utilizada na melhoria da mobilidade pessoal. Ex: Cadeira de rodas motorizada, andador com freio e carrinho de transporte infantil.
5 - Auxílios para Qualificação da Habilidade Visual e Recursos que Ampliam a Informação de Pessoas com Baixa Visão ou Cegas	São utilizados para auxiliar pessoas cegas ou com visão subnormal. Ex: Lupas manuais, lupa eletrônica, aplicativos para celulares com retorno de voz.

Tabela 7: Quadro comparativo da classificação de TA.

Fonte: adaptado de (BERSCH, 2017)

Existem uma série de meios informáticos que auxiliam educadores e educandos que possibilitam a comunicação, a pesquisa e o acesso ao conhecimento, programas como leitores de tela com síntese e comandos de voz desenvolvidos especificamente para usuários cegos, permitindo inclusive navegação na internet, leitura de elementos textuais e correio eletrônico, sem necessidade de uso do *mouse*, entre estes sistemas pode-se destacar:

DOSVOX³: Desenvolvido pelo Núcleo de Computação eletrônica da Universidade Federal do Rio de Janeiro, é um sistema para microcomputadores da linha PC que se comunica com o usuário por meio de síntese de voz, viabilizando, deste modo, o uso de computadores por deficientes visuais.

VIRTUAL VISION⁴: Este desenvolvido por uma empresa privada, a *Micropower*, opera utilitários do sistema operacional *Windows* juntamente com uma série de ferramentas. A navegação é realizada por meio de um teclado comum e o som é emitido através da placa de som presente no computador. Nenhuma adaptação especial é necessária para que o programa funcione e possibilite a utilização do computador pelas pessoas com deficiência visual, assim, o uso de sintetizadores externos é dispensado.

JAWS⁵: O *JAWS*, *Job Access With Speech*, leitor de tela desenvolvido para usuários de computador cuja perda de visão os impede de ver o conteúdo da tela ou navegar com um mouse. O *JAWS* fornece voz e saída em braille para os aplicativos de computador.

É possível destacar também a existência de outros *softwares* que possibilitam a reprodução de livros em formato digital, em áudio e em Braille, no caso *scanners*, aplicações para reconhecimento óptico de caracteres para digitalização de textos e programas que permitem converter textos em arquivos de áudio.

No currículo da matemática algumas atividades predominantemente visuais devem ser adaptadas com antecedência e outras durante sua execução por meio de descrição, informação tátil, auditiva, olfativa e qualquer outra referência que favoreça a compreensão do ambiente. Os esquemas, símbolos e diagramas presentes devem ser descritos oralmente. Os desenhos, gráficos e ilustrações devem ser adaptados e representados em relevo.

Existem instrumentos bastante conhecidos utilizados em sala de aula para representar elementos presentes no currículo, como o sorobã, o cubarítmico, blocos lógicos, material dourado e o cuisenaire.

Sorobã ou ábaco, este aparelho de cálculo, adaptado para o uso de deficientes visuais, vem recebendo crescente aceitação no ensino especializado, em virtude da rapidez e da eficiência na realização das operações matemáticas (adição, subtração, multiplicação, divisão, radiciação, potenciação), de seu baixo custo e de sua grande durabilidade.

³<http://intervox.nce.ufrj.br/dosvox/intro.htm>

⁴<https://www.virtualvision.com.br/Virtual-Vision/>

⁵<https://www.freedomscientific.com/products/software/jaws/>

O Cubartmo trata-se de uma placa plástica retangular com buracos para encaixar cubos, os quais, nas suas faces, têm representados todos os algarismos, a vírgula e o cifrão. Possui vinte orifícios em cada linha e quinze colunas, portanto dispondo de trezentos locais para introduzir os cubos.

Os blocos lógicos são um conjunto de 48 peças de plástico ou madeira criados por Zoltan Dienes, matemático russo. Este recurso é fundamental na construção das primeiras estrutura lógico matemáticas.

O material dourado foi criado para que as crianças pudessem perceber através de um material concreto como é o sistema de numeração decimal e para facilitar a compreensão sobre as operações(adição, subtração, multiplicação e divisão).

O objeto Cursenaire, também chamado de Escala Cuisenaire, é simples e ajuda o indivíduo a construir conceitos básicos de Matemática. Pode-se trabalhar sucessão numérica, comparação e inclusão, as quatro operações, o dobro e a metade de uma quantidade, frações.

O Multiplano é um instrumento que possibilita aos deficientes visuais aproximação maior com a matemática, é constituído por um tabuleiro retangular operacional no qual são encaixados pinos, fixados elásticos, hastes de corpo circular para sólidos geométricos, hastes para cálculo em funções ou trigonometria, base de operação, barras para gráficos de Estatística, disco circular que apresenta em sua periferia uma seqüência de orifícios circulares, onde podem ser combinadas duas ou mais peças pertinentes a uma determinada operação matemática que se pretenda aprender e compreender por meio da visão e ou do tato.

2.7 Visão Computacional

A visão computacional, pode ser considerada em determinados casos um subcampo da Inteligencia Artificial (IA) (SZELISKI, 2010), usada para descrever o mundo que vemos em uma ou mais imagens e reconstruir suas propriedades como forma, iluminação e distribuição de cores, visando compreender imagens estáticas ou sequencias de um vídeo. Exemplos destes fatores podem incluir o reconhecimento de pessoas ou objetos, auxílio na navegação em ambientes ou controlar algum dispositivo, situações que humanos e animais fazem sem esforço algum.

A visão computacional é um subcampo de aprendizado profundo que lida com imagens em todas as escalas. Ele permite que o computador processe e compreenda o conteúdo de um grande número de fotos através de um processo automático.

Um classificador de imagem é um procedimento que aceita um conjunto de recursos e produz um rótulo de classe, construídos através de uma série de exemplos pré rotulados, capaz de atribuir um rótulo a qualquer novo modelo inserido no sistema (PONCE; FORSYTH, 2002).

Dessa forma, a visão computacional se encarrega de extrair os principais atributos para posterior interpretação através de seus algoritmos, quando o objetivo é atingir um processamento semelhante aos humanos e com baixo custo computacional, o uso de redes neurais é adotado, pois em suas arquiteturas otimizam seu aprendizado por meio do mapeamento de padrões entre as camadas da rede.

2.8 Redes Neurais Artificiais

De acordo com (HAYKIN, 2017), uma rede neural é um processador paralelamente distribuído e constituído de séries de unidades de processamento simples, que tem a propensão natural para armazenar conhecimento experimental e tornar disponível para uso, se assemelha ao cérebro pois seu conhecimento é adquirido pela rede a partir de um ambiente, no caso o conjunto de dados exposto, caracterizando-se um processo de aprendizagem, e pelas forças de conexão entre neurônios, conhecidas como pesos sinápticos, utilizadas para armazenar o conhecimento adquirido.

A natureza de uma Rede Neural Artificial (RNA) é fundamentada em dois aspectos, seu algoritmo e sua arquitetura, a partir do momento em que a rede é criada, esta é treinada a partir de exemplos, e o algoritmo de aprendizagem então desenvolve estes dados e armazena os parâmetros ajustáveis da rede, os chamados pesos. Uma RNA é composta por uma serie de neurônios organizados em diversas camadas, seu processamento costuma ser linear, a partir da combinação das entradas com os pesos e, em seguida pela passagem por uma função de ativação.

O neurônio é a unidade de processamento fundamental para o funcionamento de uma rede neural, a Figura 12 demonstra sua estrutura, proposta em 1943 por (MCCULLOCH; PITTS, 1943a), é constituído por três elementos básicos, um conjunto de sinapses, ou conexões, um somador que soma os sinais de entrada e uma função de ativação que restringe a amplitude da saída.

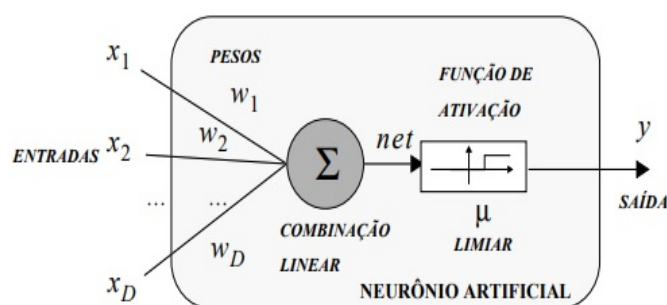


Figura 12: Modelo de Neurônio Artificial (MCCULLOCH; PITTS, 1943b)

As Redes Neurais Artificiais são baseadas em uma coleção de unidades chamadas de

Neurônios Artificiais ou apenas neurônios, esta coleção pode transmitir sinais entre si, então o receptor processa o sinal recebido e sinaliza os próximos conectados a ele.

Estes neurônios são comumente organizados em camadas, diferentes camadas podem realizar diferentes tipos de transformações nas suas entradas, organizadas na forma de um *Input Layer*, um ou mais *Hidden Layers* e um *Output Layer*, conforme apresentado na Figura 13, onde, cada nodo representa um, e as setas representam a transmissão dos sinais para os próximos

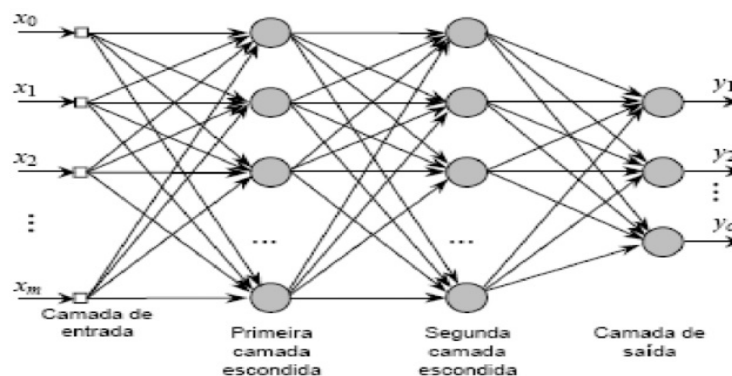


Figura 13: Estrutura de uma rede neural Multicamada
fonte: (OLIVEIRA et al., 2010)

Uma RNA não se limita a apenas uma ou duas camadas, pode conter inúmeras, dependendo da arquitetura proposta para a rede, em caso mais de duas camadas é denominada de Rede Neural Profunda (RNP) ou *Deep Neural Network* (DNN). Nestas redes a informação é transferida por meio de cada camada utilizando a saída da camada anterior como entrada para a próxima.

2.8.1 Redes Neurais Convolucionais

Segundo (LECUN et al., 1999) as redes convolutivas combinam três ideias arquitetônicas para garantir um certo grau de desvio, escala e invariância de distorção: campos receptivos locais, pesos compartilhados (ou replicação de peso) e subamostragem espacial. Nesse sentido, essas três ideias fundamentam um algoritmo de aprendizado profundo que aprende uma imagem de entrada e atribui seus respectivos pesos conforme a saída, para posteriormente, de acordo com sua arquitetura, resultar uma saída probabilística.

As peculiaridades das Redes Neurais Convolucionais derivam da complexidade em certos problemas de classificações, no qual para um amplo conjunto de dados, existem inúmeros níveis de oscilações sobre determinado alvo. Neste caso, a melhor estratégia é transformar as imagens (matrizes) em vetores, capazes de serem alimentadas por filtros relevantes (camadas) e funções de ativações, ambos designados especificadamente em reconhecer feições ou ocorrências mais relevantes nas cenas.

O primeiro processamento de imagens foi basicamente baseado em filtros que permi-

tiram, por exemplo, obter as bordas de um objeto em uma imagem usando a combinação de filtros de borda vertical e borda horizontal. Uma rede neural convolucional é uma série de camadas convolucionais e de agrupamento que permitem extrair os principais recursos das imagens que respondem melhor ao objetivo final. Na seguir, detalharemos cada filtro juntamente com suas equações matemáticas.

Mas antes de definir explicitamente o processo de convolução, é necessário começar estabelecer algumas operações básicas, como o *padding* e *stride*.

Normalmente os pixels no canto da imagem (de uma matriz 2D) são menos usados que os pixels no meio da imagem, isto significa que as informações das bordas costumam ser descartadas. Para resolver esse problema, geralmente adicionamos preenchimento ao redor da imagem para levar em consideração os pixels nas bordas, este processo é chamado de *padding*. Por convenção, zeros são adicionados e designado o parâmetro *padding* como "p", que representa o número de elementos adicionados em cada um dos quatro lados da imagem. A figura 14 ilustra o preenchimento de uma imagem em escala de cinza em que $p = 1$:

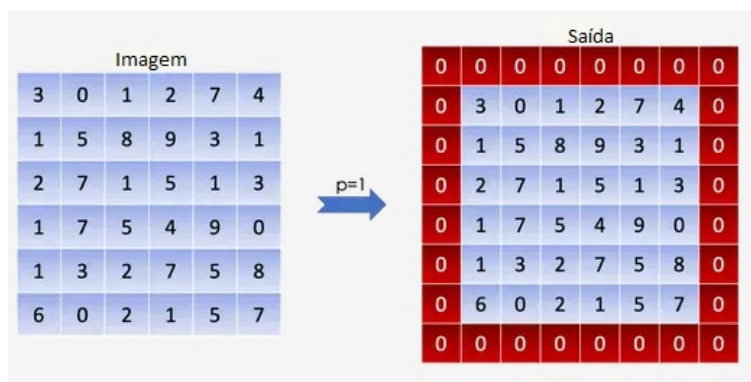


Figura 14: Exemplo de *padding*

fonte: adaptado de (GOODFELLOW; BENGIO; COURVILLE, 2016)

O *stride* obtido por meio do produto convolucional. Um *stride* grande permite reduzir o tamanho da saída e vice-versa. Denotamos com "S" o parâmetro *stride*. A Figura 15 ilustra um produto convolucional (somatório elemento por elemento por bloco) com $s = 1$

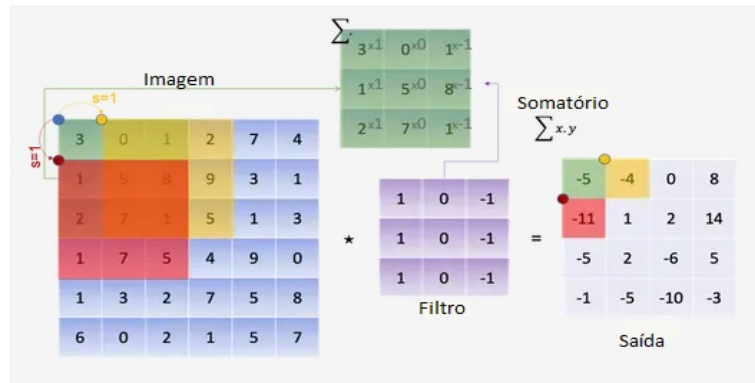


Figura 15: Exemplo de *stride*

fonte: adaptado de (GOODFELLOW; BENGIO; COURVILLE, 2016)

Depois de definir o *stride* e o *padding*, podemos estabelecer o produto da convolução entre um tensor e um filtro. Então Uma imagem, em geral pode ser matematicamente representada como um tensor com as seguintes dimensões:

$$\dim(imagem) = (\eta H, \eta W, \eta C) \quad (1)$$

Onde: ηH = Dimensão da altura ηW = Dimensão do comprimento ηC = Número de canais de cor

No caso de uma imagem RGB, por exemplo, temos $\eta C = 3$ vermelho, verde e azul. Por convenção, é considerado o filtro K ao quadrado e uma dimensão ímpar denotada por f , que permite que cada pixel seja centrado no filtro e, portanto, considere todos os elementos ao seu redor. Ao operar o produto convolucional, o filtro / kernel K deve ter o mesmo número de canais ηC da imagem, dessa forma aplicamos um filtro diferente para cada canal. Assim, a dimensão do filtro é a seguinte:

$$\dim(filtro) = (f, f, \eta C) \quad (2)$$

O produto convolucional entre a imagem e o filtro é uma matriz 2D em que cada elemento é a soma da multiplicação por elementos do cubo (filtro) e o subcubo da imagem fornecida, conforme ilustrado na Figura 16:

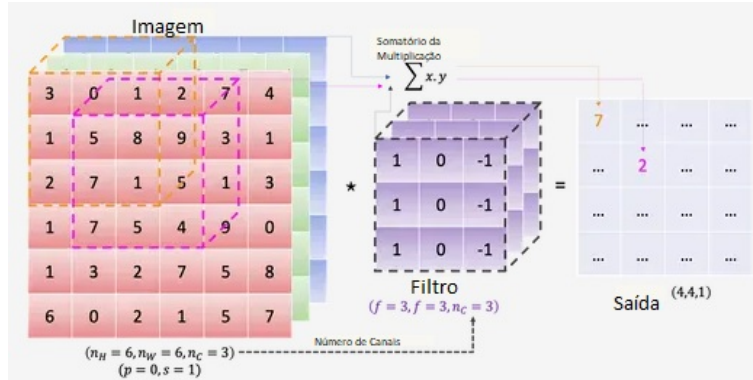


Figura 16: Produto Convolutacional

fonte: adaptado de (GOODFELLOW; BENGIO; COURVILLE, 2016)

Matematicamente falando, para uma determinada imagem I e filtro K , temos:

$$\text{conv}(I, K)_{x, y} = \sum_{i=1}^{\eta H} \eta H \sum_{j=1}^{\eta W} \eta W \sum_{k=1}^{\eta C} \eta C K_{i, j, k} I_{x + i, y + j - 1, k} \quad (3)$$

Mantendo as mesmas notações de antes, temos:

$$\begin{aligned} \dim(\text{conv}(I, K)) &= \left(\frac{\eta H + 2p - f}{s} + 1, \frac{\eta W + 2p - f}{s} + 1 \right); s > 0 \\ &= (\eta H + 2p - f, \eta W + 2p - f); s = 0 \end{aligned} \quad (4)$$

O *pooling* é o processo de reduzir a amostragem dos recursos da imagem, resumindo as informações. A operação é realizada através de cada canal e, portanto, afeta apenas as dimensões ηH e ηW e mantém ηC intacto. Dada uma imagem, um filtro é deslizado, sem parâmetros para aprender, seguindo um certo *stride*, e então uma função é aplicada nos elementos selecionados.

$$\begin{aligned} \dim(\text{pool}(img)) &= \left(\frac{\eta H + 2p - f}{s} + 1, \frac{\eta W + 2p - f}{s} + 1, \eta C \right); s > 0 \\ &= (\eta H + 2p - f, \eta W + 2p - f, \eta C); s = 0 \end{aligned} \quad (5)$$

E são frequentemente aplicados:

Pooling médio: calculado a média dos elementos presentes no filtro; *Pooling* máximo: considera todos os elementos no filtro, e retorna o máximo; Abaixo, uma ilustração de um pool médio representado pela Figura 17:

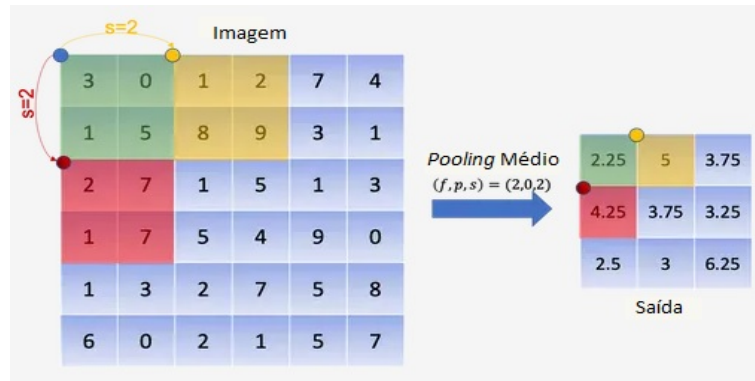


Figura 17: *Pooling* médio

fonte: Adaptado de (GOODFELLOW; BENGIO; COURVILLE, 2016)

Então combinando todas as operações descritas acima é possível construir uma RNC, arquitetando camada por camada.

Cada camada da rede neural convolucional pode ser:

- Camada convolucional -CONV- seguida de uma função de ativação;
- Camada de *pooling* -POOL- conforme detalhado acima;
- Camada -FC- totalmente conectada, basicamente uma camada semelhante à de uma rede neural feedforward.

Conforme demonstrado anteriormente, na camada convolucional são aplicamos produtos convolucionais, utilizando uma série de filtros, dessa vez, na entrada seguida por uma função de ativação Ψ , portanto para a primeira camada temos:

- Entrada: a^{l-1} de tamanho $(\eta H^{l-1}, \eta W^{l-1}, \eta C^{l-1}), a^{[0]}$;
- *Padding*: p^l ;
- *stride*: s^l ;
- Número de Filtros: ηC^l onde, a cada K^n possui dimensão: $(f^l, f^l, \eta C^l)$;
- *bias*: da n^e sima convolução $b\eta^l$
- Função de Ativação: Ψ^l
- Saída: a^l de tamanho $(\eta H^l, \eta W^l, \eta C^l)$

Então temos:

$$\text{conv}(a^{l-1}, K^n)_{x,y} = \Psi^{[l]} \left(\sum_{i=1}^{\eta H^l} \sum_{j=1}^{\eta W^l} \sum_{k=1}^{\eta C^l} K^{n_i j k} a^{l-1}_{x+i-1, y+j-1, k+\eta B^l} \right)$$

$$\dim(\text{conv}(a^{l-1}, K^n)) = (\eta H^l, \eta W^l) \quad (6)$$

Neste caso os parâmetros aprendidos pela rede foram os **filtros** na forma de $(f^{[l]} x f^{[l]} x \eta N^{[l-1]}) x \eta C^{[l]}$ e o **bias** na forma de $(1 \times 1 \times 1) \times \eta C^{[l]}$.

Todos os processos matemáticos descritos acima podem ser melhor visualizados por meio da Figura 18



Figura 18: Exemplo de Camada Convolucional

fonte: adaptado de (GOODFELLOW; BENGIO; COURVILLE, 2016)

A camada de *pooling*, conforme mencionado anteriormente visa reduzir a amostragem dos recursos da entrada sem afetar o número de canais.

Considerando a seguinte notação:

- Entrada: a^{l-1} de tamanho $(\eta H^{l-1}, \eta W^{l-1}, \eta C^{l-1}), a^{[0]}$ representando uma imagem;
- *Padding*: p^l ;
- *stride*: s^l ;
- Tamanho: do filtro de *pooling* $f^{[l]}$
- Função de *pooling*: $\phi^{[l]}$
- Saída: a^l de tamanho $(\eta H^l, \eta W^l, \eta C^l = \eta C^{l-1})$

A partir disso é possível afirmar:

$$\begin{aligned} \text{pool}(a^{[l-1]})_{x,y,z} &= \phi^{[l]}((a^{[l-1]}_x + i - 1, y + j - 1, z)_{(i,j) \in [1, 2 \dots f^{[l]}]^2}) \dim(a^{[l]}) \\ &= (\eta^H, \eta^W, \eta^C) \end{aligned} \quad (7)$$

A camada de *pooling* não possui parâmetros para aprender, a junção entre as operações demonstradas anteriormente pode ser visualizada por meio da Figura 19

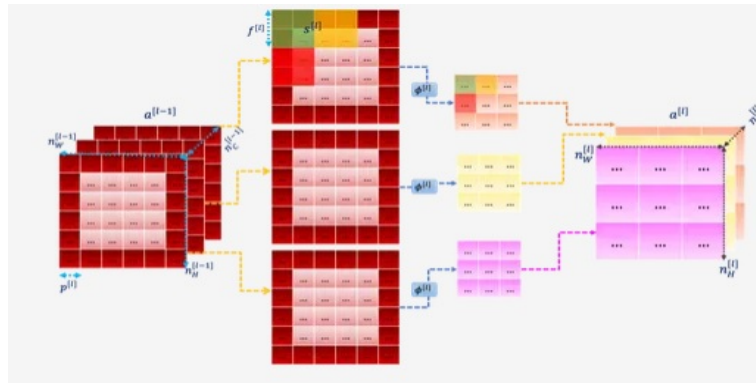


Figura 19: Exemplo de Camada Convolucional

fonte: adaptado de (GOODFELLOW; BENGIO; COURVILLE, 2016)

As redes convolutivas costumemente possuem em suas últimas camadas a chamada *fully connected layer*, sendo um número finito de neurônios que recebe um vetor $a^{[i-1]}$ e retorna um vetor de $a^{[i]}$. Em geral considerando o J^{esimo} nodo da I^{esima} camada, gerando a seguinte equação:

$$z_j^{[i]} = \sum_{l=1}^{n_{i-1}} w_{j,l}^i a_l^{i-1} + b_j^i \xrightarrow{a_j^i} \psi^{[i]}(z_j^{[i]}) \quad (8)$$

Neste caso a entrada $a^{[i-1]}$ pode ser o resultado de uma convolução ou de uma camada de *pooling* com as dimensões $(n_H^{[i-1]}, n_W^{[i-1]}, n_C^{[i-1]})$. Para poder conectá-lo à camada *fully connected*, é necessário passar ele por uma camada de *flatten*, "achatando" o tensor para um vetor 1D com a dimensão: $(n_H^{[i-1]} \times n_W^{[i-1]} \times n_C^{[i-1]}, 1)$, gerando a seguinte equação:

$$n_i - 1 = n_H^{[i-1]} * n_W^{[i-1]} * n_C^{[i-1]} \quad (9)$$

Os parâmetros aprendidos pela l^{esima} camada foram os pesos na forma de $w_{j,l}$, com $n_{l-1} * n_l$ parâmetros e o *bias* com n_l propriedades.

A camada *fully connected* é ilustrada a partir da Figura 20.

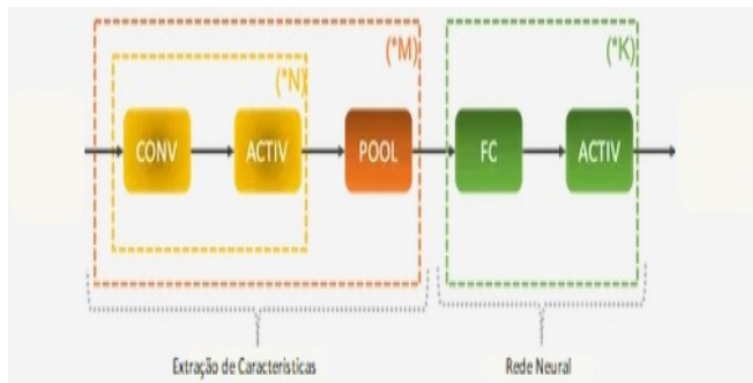


Figura 20: Camada *Fully Connected*

fonte: adaptado de (GOODFELLOW; BENGIO; COURVILLE, 2016)

Desta forma, em geral uma RNC é uma sequência de todas as operações descritas anteriormente, conforme a Figura 21 demonstra.

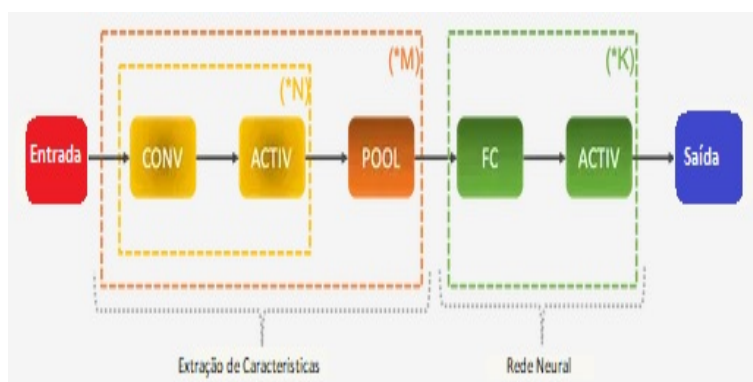


Figura 21: Camada *Fully Connected*

fonte: adaptado de (GOODFELLOW; BENGIO; COURVILLE, 2016)

Depois de repetir uma série de convoluções, seguidas pelas funções de ativação, uma camada de *pooling* é aplicada e então, este processo é repetido por um certo número de vezes. Essas operações permitem extrair características da imagem que serão alimentadas em uma rede neural descrita pelas camadas totalmente conectadas, que são regularmente seguidas pelas funções de ativação. A idéia principal é diminuir n_H e n_W ao passo que aumentamos o n_C conforme a rede se torna cada vez mais profunda. Uma rede convolucional 3d possui o formato representado na Figura 22

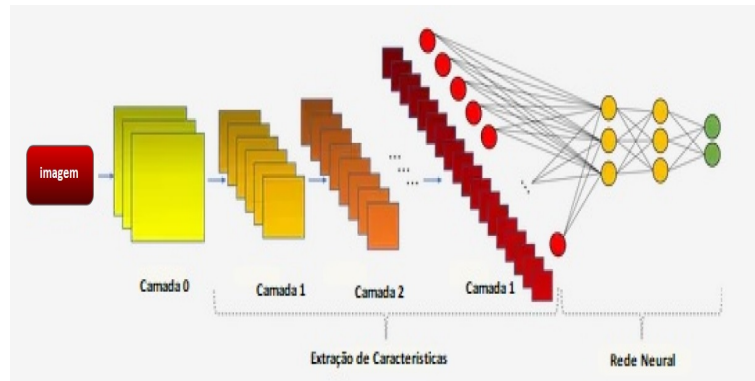


Figura 22: Rede Convolucional 3d

fonte: adaptado de (GOODFELLOW; BENGIO; COURVILLE, 2016)

As redes convolucionais foram algumas das primeiras redes profundas treinadas com *backpropagation*. Não está totalmente claro por que as redes convolucionais foram bem-sucedidas quando se considerou que as redes gerais de retropropagação falharam (GOODFELLOW; BENGIO; COURVILLE, 2016). Pode ser que as redes convolucionais tenham sido mais eficientes em termos computacionais do que redes conectadas com facilidade, por isso foi mais fácil executar várias experiências com o ajuste da implementação e hiperparâmetros.

3 PROCEDIMENTOS METODOLÓGICOS

3.1 Proposta do Sistema

A RNC proposta neste trabalho opera como uma API (*Application Programming Interface*), com propósito de permitir acesso aos dados remotamente, através de uma aplicação móvel, a API responsável pelo *backend* realizando a lógica de programação, pré-processamento, aplicação da RNA e classificação das imagens recebidas. A Figura 23 representa o fluxo de processos do modelo de uma possível aplicação móvel, encarregado pela interface gráfica, interação com o usuário, captação e geração das imagens.

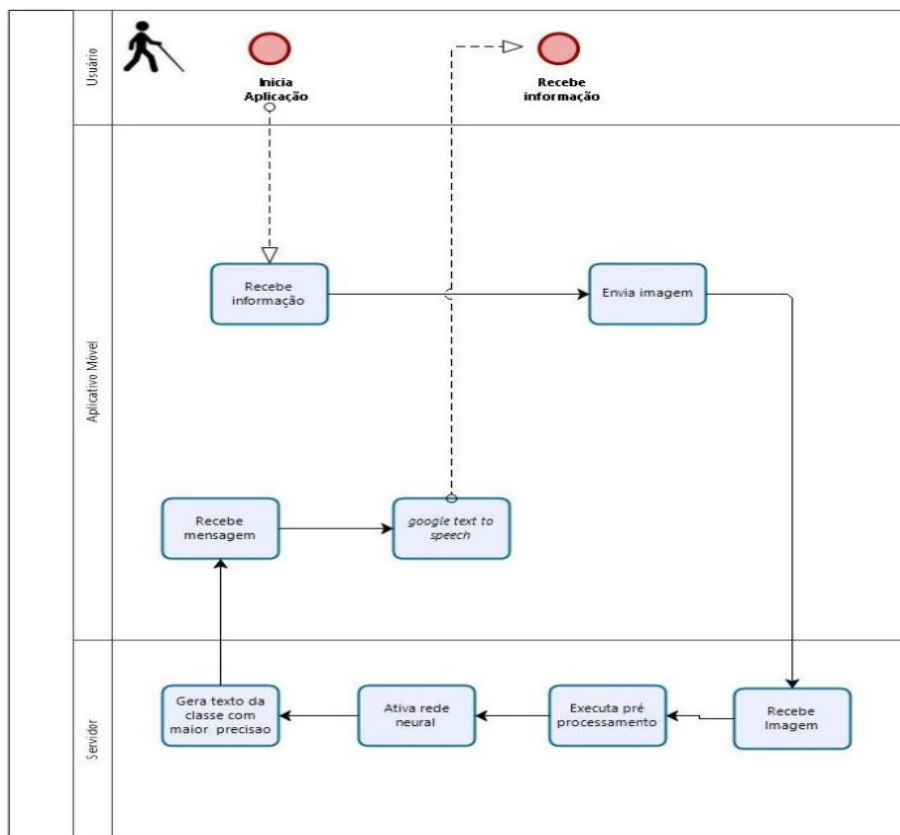


Figura 23: Fluxo de Processos da Aplicação

Fonte:autor

3.1.1 Desenvolvimento do Dataset

Para realizar a elaboração do *dataset* foi necessário desenvolver uma técnica computacional capaz de desenhar uma quantidade significativa de imagens de curvas cartesianas em um curto espaço de tempo.

O *dataset* proposto neste trabalho possui 10 classes de curvas racionais cujos graus variam de 1 até 4 e compõem o plano cartesiano e suas definições, bem como as equações (LAWRENCE, 2014) são descritas abaixo:

- **Parábola:** Curva de segundo grau em formato cônico, é uma curva plana, cujos pontos são equidistantes de um ponto fixo (foco) e de uma reta fixa (diretriz) ou curva resultante de uma secção feita num cone por um plano paralelo à geratriz. curva que um projétil descreve (SILVA, 1969), representada pela equação 10.

$$y^2 = 4px \quad (10)$$

- **Folium de Descartes:** Curva de terceiro grau, é uma curva plana proposta por Descartes em 1638 para desafiar as técnicas de localização extrema de Fermat. de forma paramétrica, representada pela equação 11.

$$x^3 + y^3 = 3axy \quad (11)$$

- **Trissectriz de Maclaurin:** Curva de terceiro grau, esta curva estabelecida por Colin Maclaurin em 1742. Tal como tantas outras, o seu estudo foi abordado no sentido de encontrar uma solução para um dos antigos problemas, neste caso o da trissecação de um ângulo, representada pela equação 12

$$y^2(a + x) = x^2(3a - x) \quad (12)$$

- **Lemniscata de Bernoulli:** Curva de quarto grau, descrita por Jacob Bernoulli em 1694, similar ao número 8 e ao símbolo do infinito, representada pela equação 13.

$$(x^2 + y^2)^2 = 2a^2(x^2 - y^2) \quad (13)$$

- **Cardioide:** Curva de quarto grau, descrita em 1741 pelo matemático Castillon. batizada a partir de seu formato de coração, representada pela equação 14.

$$(x^2 + y^2 - 2ax)^2 = 4a^2(x^2 + y^2) \quad (14)$$

- Curva Quartica: uma curva quártica é uma curva algébrica do quarto grau. O quártico é o polinômio de mais alta ordem que pode ser resolvido de maneira geral, por meio de raízes, representada pela equação 15.

$$b^2y^2 = x^3(a - x) \quad (15)$$

- Folium dupla: Parte da família das curvas Folium esta, com $b=0$, representada pela equação 16.

$$(x^2 + y^2)(y^2 + x(x + a)) = 4axy^2 \quad (16)$$

- Figura oito ou Lemniscata de Gerono: Curva de quarto grau, estudada por Grégoire de São Vicente em 1647 e Cramer em 1750. No entanto foi batizada apenas em 1985, representada pela equação 17.

$$x^4 = a^2(x^2 - y^2) \quad (17)$$

- Curva de Kappa: Curva de quarto grau, é também conhecida como curva de Gutschoven, que foi estudada por G. van Gutschoven por volta de 1662. Também foi estudado por Newton e, alguns anos depois, por Johann Bernoulli, representada pela equação 18.

$$(x^2 + y^2)y^2 = a^2x^2 \quad (18)$$

- Curva Trifólio: Curva de quarto grau, "O" trifólio é o fólio de três lóbulos com $b = a$, isto é, a rosa de três pétalas. Os trifolia são os pedais do deltóide em relação a um ponto dentro do deltóide; portanto, são casos especiais de folia. São as curvas compostas por três folhas que se juntam em um ponto triplo, representada pela equação 19.

$$(x^2 + y^2)(y^2 + x(x + a)) = 4axy^2 \quad (19)$$

Totalizando cerca de 30 mil imagens divididas entre treino teste e validação. O *dataset* está estruturado contendo 596 imagens para teste, 2381 para treino e 744 para validação totalizando 3.721 imagens para cada classe.

A proposta para o *dataset* apresentado neste trabalho foi inspirada a partir de um dos conjuntos de dados mais apresentados na literatura acerca do tema, o *Coco dataset* (LIN et al., 2015), este conjunto possui 80 classes e cerca de 300 mil imagens, totalizando em média 4.125 imagens para cada classe. Procurou-se então atingir uma quantidade de

imagens por classe similar ao *dataset* citado sem perder desempenho tanto da rede neural quanto computacional.

O procedimento descrito abaixo foi realizado para cada classe individual conforme especificado na Figura 24.



Figura 24: Ciclo para Geração do *Dataset*

Fonte:autor

A estruturação do *dataset* consiste em 5 fases distintas, aqui exemplificado utilizando a curva **Cardioide** conforme abaixo:

A primeira fase visa obter a equação descomposta 21 a partir da equação geral 20 de cada uma das curvas cartesianas.

$$(x^2 + y^2 - 2ax)^2 = 4a^2(x^2 + y^2) \quad (20)$$

$$x^4 + y^4 + 2x^2y^2 - 4ax^3 - 4axy^2 - 4a^2y^2 \quad (21)$$

A segunda fase aplica a a equação descomposta a uma função polinomial representada pela Figura 25, esta função gera uma matriz / lista $(m + 1) \times (n + 1)$ de listas de coeficientes (reais).

$$\begin{bmatrix}
 1 * 1, & 1 * x, & 1 * x^2, & 1 * x^3, & \dots, & 1 * x^n \\
 y * 1, & y * x, & y * x^2, & y * x^4, & \dots, & y * x^n \\
 \dots, & \dots, & \dots, & \dots, & \dots, & \dots \\
 y^m * 1, & y^m * x, & y^m * x^2, & y^m * x^3, & \dots, & y^m * x^n
 \end{bmatrix}$$

Figura 25: Gerador de Matriz Polinomial

Fonte:autor

A matriz gerada com base na equação da curva Cardioide pode ser visualizada a seguir.

$$\begin{bmatrix}
 0 & 0 & -1 & 0 & 1 \\
 0 & 0 & 0 & 0 & 0 \\
 1 & 0 & 2 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 \\
 1 & 0 & 0 & 0 & 0
 \end{bmatrix}$$

A fase três desenha a curva cartesiana conforme demonstrado abaixo na Figura 26.

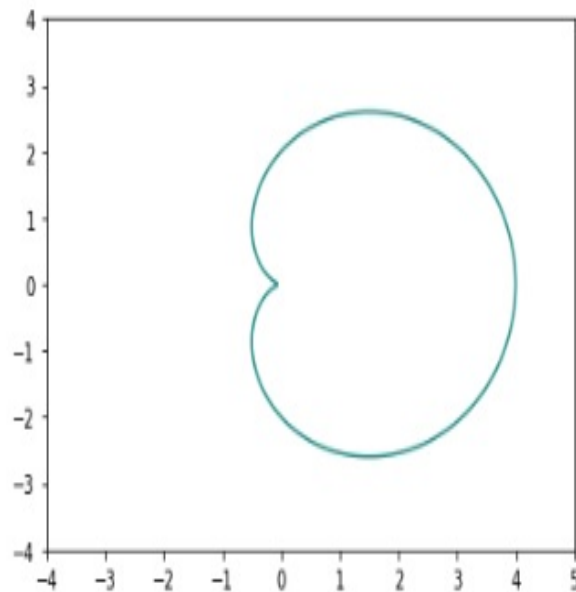


Figura 26: Curva Cardioide

Fonte:autor

No entanto para garantir a consistência no treinamento da RNC e garantir que diferentes imagens do elemento fossem corretamente classificados. Cada parâmetro foi escolhido de forma que não alterasse as características principais de cada curva, mantendo principalmente o seu formato. Os parâmetros escolhidos foram cor, formato, espessura e estilo da linha, gerando as seguintes imagens conforme mostra a Figura 27.

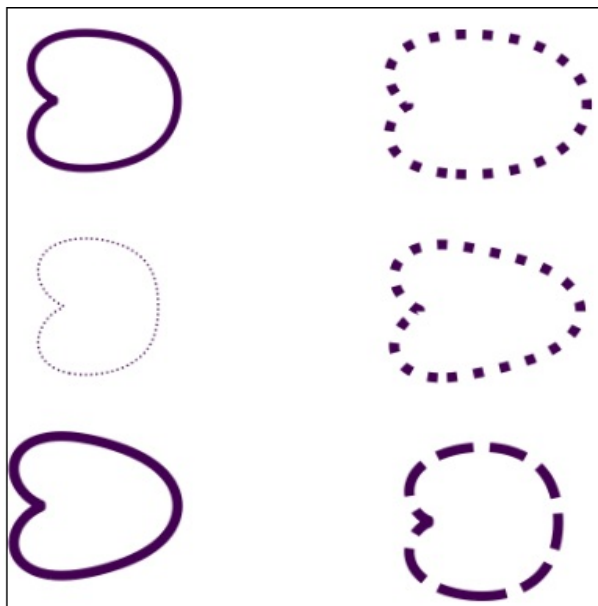


Figura 27: Exemplo *dataset* curva Cardioide

Fonte:autor

A fase quatro avalia as imagens geradas, verificando-as individualmente na busca por inconsistências que descaracterizem a curva, ou seja, as imagens necessitam ser aleatórias suficiente mas sem perder suas características principais, as imagens que não atingiram a consistência desejada são descartadas e refeitas.

A quinta e última fase divide o *dataset* em três, treino, teste e validação.

3.1.2 Preparação

A proposta apresentada neste trabalho utiliza redes neurais convolucionais para realizar a detecção e classificação de curvas cartesianas. Para realizar esta tarefa, é necessário definir a arquitetura de rede neural a ser utilizada, tanto para realizar o treino, quanto para executar a classificação.

Então foi preciso estudar os principais modelos de RNCs e, definir quais seriam utilizados neste projeto. A seleção das redes foi realizada considerando o trabalho de (CANZIANI; PASZKE; CULURCIELLO, 2016), os autores desempenharam uma análise abrangente acerca de métricas importantes em aplicações práticas: precisão, uso de memória, parâmetros, contagem de operações, tempo de inferência e consumo de energia, de um total de quinze RNCs.

É importante ressaltar que todos os testes realizados por Canziani estavam usando o conjunto de dados ImageNet¹, este tem cerca de 14 milhões de imagens e 20 mil classes. Portanto, a escolha dos modelos de rede foi realizada com base na precisão e no custo computacional apresentado.

¹<http://www.image-net.org/>

Devido à diferença de tamanho dos conjuntos de dados, foram escolhidos os modelos intermediários na classificação, uma vez que o conjunto de dados proposto é consideravelmente pequeno em relação ao utilizado no trabalho de base, as redes selecionadas foram: VGG-16, VGG-19, InceptionV3 e MobileNETV2.

As redes VGG-16 e VGG19 desenvolvidas pelo Grupo de Geometria Visual (Grupo VGG), fazem parte do Departamento de Ciências da Engenharia da Universidade de Oxford (SIMONYAN; ZISSERMAN, 2015). Esses modelos VGG16 e VGG19 foram nomeados a partir do número de camadas convolucionais, dezesseis e dezenove respectivamente. Estes modelos foram testados e validados a partir do Concurso de Reconhecimento Visual de Grande Escala do *ImageNet* de 2014 ² (ILSVRC-2014).

O modelo *InceptionV3*, proposto por (SZEGEDY et al., 2015) para a mesma competição, *ImageNet* no ano de 2015. Este modelo possui 42 camadas convolucionais, e sua arquitetura foi introduzida pela primeira vez como *GoogLeNet* e posteriormente denominada *InceptionV1*, uma série de refinamentos foram realizados em duas outras versões até a atual, que foi utilizada neste trabalho.

A arquitetura *MobileNetV2* procura maior eficiência em dispositivos móveis, maximizando a precisão enquanto trata os recursos restritos e limitados de dispositivos embarcados ou móveis. Esses modelos neurais *Mobilenet* são propostos e desenvolvidos pelo *Google AI Research* ³, o primeiro lançamento de junho de 2017 (HOWARD et al., 2017), este trabalho usa a versão atual desta arquitetura, sua segunda geração, classificada como estado da arte por seus desenvolvedores (SANDLER et al., 2018).

Os experimentos foram realizados em 3 etapas, buscando atingir objetivos individuais conforme demonstra a Figura 28, na primeira rodada de experimentos explorou-se 4 arquiteturas de RNCs e diferentes configurações de hiperparâmetros presentes nestes modelos realizando o cruzamento entre si com a intenção de selecionar o modelo mais eficiente, na segunda rodada tratou-se de incrementar o número de classes disponíveis na RNC e, ao longo da experiência realizando ajustes de configuração, e finalmente a terceira rodada culmina na junção dos 2 experimentos iniciais, assim efetivando o treinamento da rede com a totalidade das classes desenvolvidas e avaliando os resultados

²<http://image-net.org/challenges/LSVRC/2014/>

³<https://ai.googleblog.com/2017/06/mobilenets-open-source-models-for.html>



Figura 28: Diagrama de Experimentos

Fonte: autor

3.1.3 Treinamento da Rede

Para executar o treinamento da rede, um computador com processador core i7-3770 3.4 ghz, 16gb de memória RAM, placa de vídeo Nvidia Titan XP 12gb de memória e um disco rígido de 1 *terabyte* foi utilizado, bem como, as linguagens de programação Python e C embarcadas no *framework Keras*⁷.

Dois principais parâmetros são levados em conta em todos os experimentos, a precisão (*accuracy*) e o ritmo de perda (*loss rate*), este segundo parâmetro não foi estipulado valor mínimo, ou seja ele é utilizado como critério de parada mas não possui um valor a ser atingido, já o valor de precisão se encaixa na regra dos 85% proposta por (WILSON et al., 2019), neste caso para todos os experimentos o nível de precisão precisa necessariamente atingir um valor mínimo de 85%, neste caso é importante ressaltar que um valor mínimo é estipulado mas não um valor máximo.

3.2 Primeiro Experimento

Este primeiro experimento possui dois objetivos fundamentais, 1) avaliar arquiteturas de RNCs com a finalidade de definir uma para realizar futuras experiências e 2) definir valores dos principais hiperparâmetros presentes nas configurações das RNCs.

Para realizar os testes do primeiro experimento, foi necessário ajustar parâmetros em todos os modelos testados com a finalidade de obter a melhor precisão, respeitando o mínimo de 85% e o menor ritmo de perda.

⁷<https://github.com/keras-team/keras>

Cada ciclo de treinamento deste experimento busca avaliar a configuração da RNC utilizando os hiperparâmetros descritos na Figura 29.

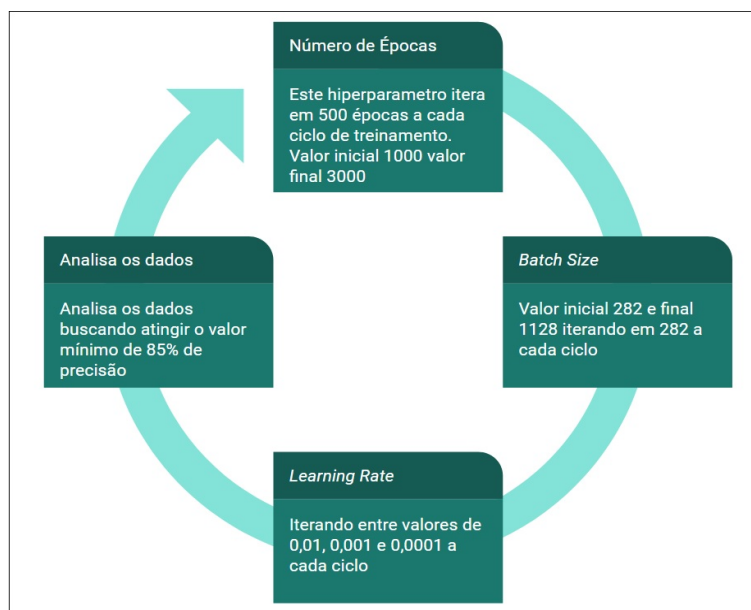


Figura 29: Ciclo de testes do primeiro experimento

Fonte: autor

Este experimento busca cruzar os hiperparâmetros entre si, cada ciclo de treinamento precisa possuir configurações diferentes do anterior até não existir uma combinação não testada, então é necessário realizar a análise dos dados e determinar a melhor combinação. O fluxo de treinamento para cada ciclo pode ser visualizado na Figura 30.

Ao final de todos os ciclos de treinamento, era esperado que, apesar de possuírem características distintas, todas as arquiteturas de RNCs testadas apresentassem resultados superiores ao mínimo estipulado de 85% para a precisão.

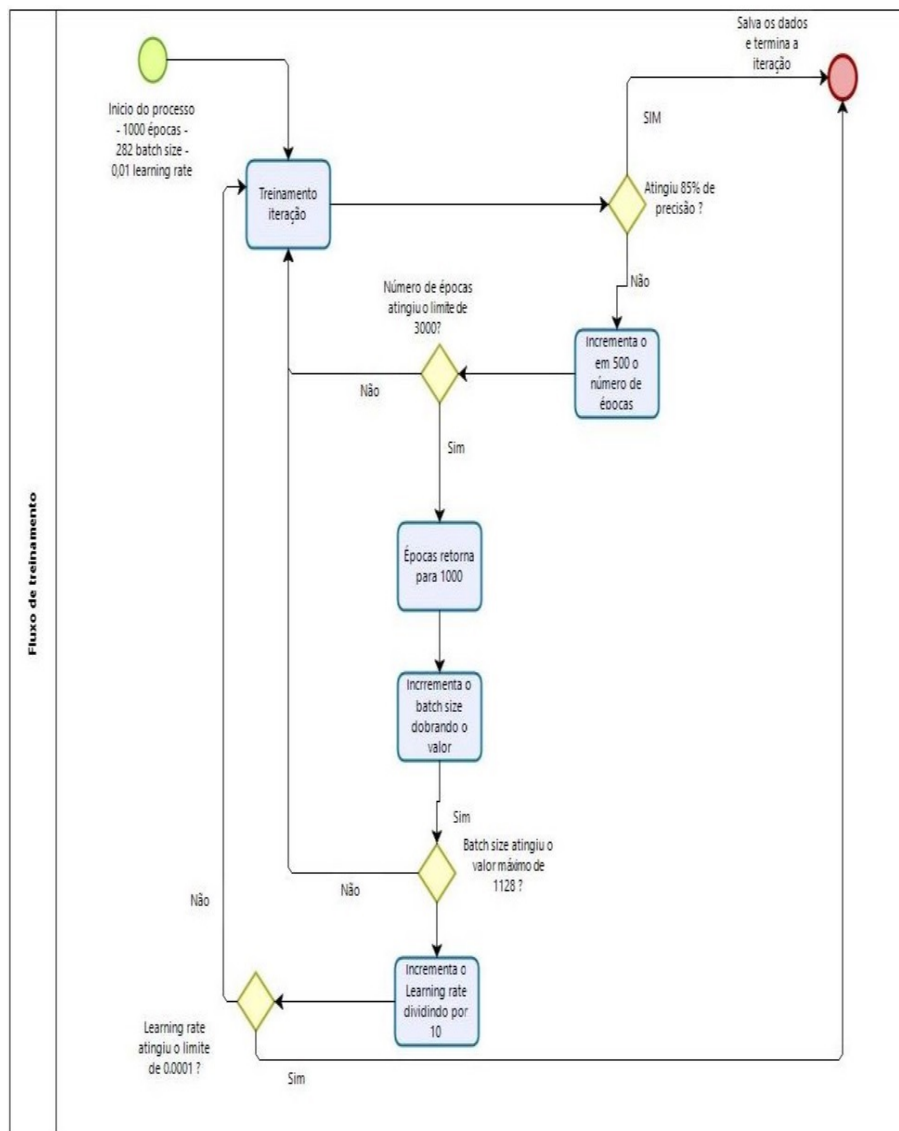


Figura 30: Fluxo de Treinamento a cada Ciclo

Fonte: autor

- **Método de Gradiente Descendente Estocástico:** Neste tópico, foi necessário selecionar entre os métodos de gradiente descendente estocástico e descontinuidade de gradiente de lote, o método foi determinado a partir de estudos na literatura sobre o tema (JING; GUANCI, 2018).
- **Função de ativação sigmóide:** Esta função é responsável por realizar uma aproximação entre zero e um, ou seja, valores abaixo de 0,5 serão considerados zero e acima de 0,5 considerados 1, função de ativação determinada a partir de estudos na literatura sobre o tema (CHUNG; LEE; PARK, 2016).
- **Taxa de Aprendizado:** Este hiper-parâmetro controla os pesos da rede neural com relação a uma função de perda, os valores são regularmente baixos para não perder

nenhum mínimo local, mas pode fazer com que a rede demore muito tempo para convergir, os valores selecionados para os testes foram 0,01, 0,001 e 0,0001.

- **Tamanho do lote:** O número de exemplos de treinamento realizando uma única passagem pela RNC para frente e retornando. Quanto maior o tamanho do lote, mais espaço de memória você precisará. Quanto menor o tamanho do lote, melhor, neste trabalho, devido ao tamanho do *dataset* e de limitações técnicas, optou-se por iniciar em 282 e incrementar sempre em 282 o valor a cada ciclo até um total de 1128
- **Épocas:** Cada a época representa o número de vezes que um elemento passa por meio de todas as camadas da rede e depois retornando, o número de épocas iterou entre 1000 até 3000 iterando em 500 a cada ciclo.

Cada sessão de treinamento tem um total de 100% de imagens, cada rede neural usa 25% como o conjunto de dados de teste e os 75% restantes são utilizados para realizar o treinamento.

Embora apenas o modelo com os melhores valores de precisão e taxa de perda selecionados para o desenvolvimento da aplicação, os arquivos de rede treinados com os pesos possam ser visualizados e baixados deste link ⁴.

3.2.1 Análise dos Dados - Primeiro Experimento

Os gráficos apresentam os valores que representam os dados referentes à precisão e o ritmo de perda de cada rede, todos os testes realizados em um total de três mil vezes. As linhas azul e laranja representam o conjunto de dados de treinamento e teste respectivamente, o eixo X em todos os casos representa o número de épocas e o eixo Y indica o valor de precisão ou perda dependendo do gráfico apresentado.

A precisão de uma rede é determinada após todos os parâmetros de rede terem aprendido e adequadamente corrigido, ou seja, o modelo já realizou todo o aprendizado com as especificações do arquiteto, então as amostras de treinamento são carregadas e novamente alimentam o modelo, a rede então anota o número de erros encontrados, em comparação com as imagens reais. Posteriormente, é gerado uma porcentagem com os erros de classificação, por exemplo se o número de imagens de teste for 100 e o modelo classificou corretamente 85 dessas imagens, então a precisão é considerada de 85%.

Em relação ao ritmo de perda gerada pelo modelo, quanto menor o valor, melhor a rede neural é considerada, esse valor é calculado a partir do conjunto de treinamento (*dataset* de treino) e validação (*dataset* de teste) e de quão bem o modelo da rede está relacionando esses dois conjuntos. O valor gerado pela função de perda não é uma porcentagem, assim

⁴<https://drive.google.com/open?id=1aczyOPZvi6EaJLmKeedK-QTpLzmikikJ>

como a precisão, a perda é a soma de todos os erros causados para cada amostra dos conjuntos de treinamento e teste.

As Figuras 31 e 32 representam os gráficos para os modelos VGG16 e VGG19 respectivamente. Estes gráficos apontam os dados gerados a partir do *dataset* de treinamento e teste, indicando a precisão destas arquiteturas.

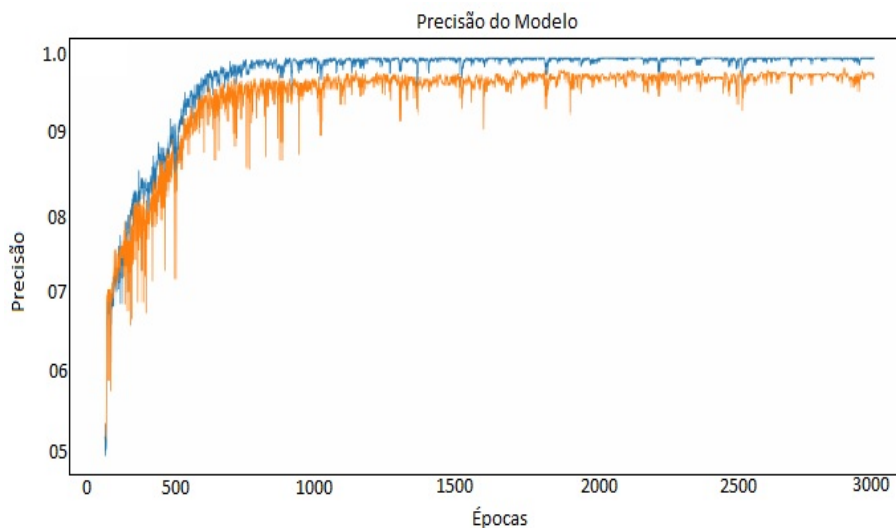


Figura 31: VGG16 Precisão por Época

Fonte: autor

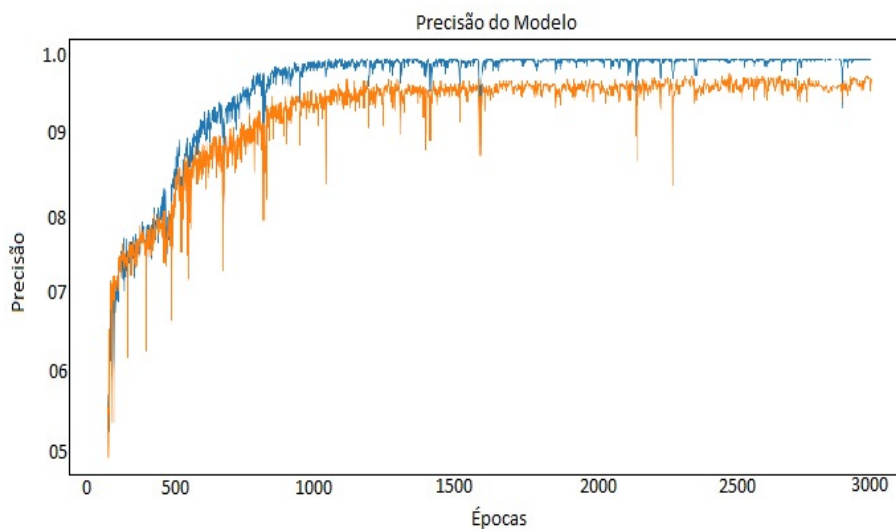


Figura 32: VGG19 Precisão por Época

Fonte: autor

Nos gráficos gerados é possível identificar semelhança natural entre as redes, uma vez que são versões diferentes do mesmo modelo, uma com 16 camadas e outra com 19. A rede VGG16 convergiu para o máximo de precisão com oitocentas épocas, a o valor médio de 97%, a VGG19 convergiu com mil e cem épocas, e atingiu valor máximo de 96%, estes

valores gerados a partir do conjunto de dados de treinamento. Com o conjunto de dados de teste, há uma oscilação distinta do linha laranja, indicando que a rede ainda encontrou erros nos dados de treinamento e precisa ajustar os pesos. Especialmente na rede VGG19, esta novamente convergiu lentamente e estabilizado posteriormente em relação à rede VGG16, a média os valores encontrados no conjunto de dados de teste foram 94% e 92% respectivamente para os modelos VGG16 e VGG19. As Figuras 33 e 34 representam os gráficos para os modelos *MobileNet* e *InceptionV3* respectivamente. Estes gráficos apontam os dados gerados a partir do *dataset* de treinamento e teste, indicando a precisão destas arquiteturas. Em ambos gráficos o eixo X representa o número de épocas e o eixo Y os valores médios da precisão.

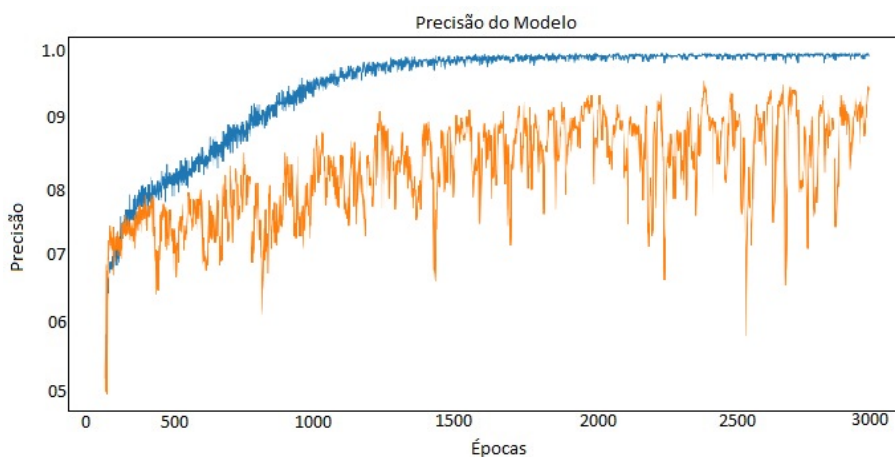


Figura 33: MobileNet Precisão por Época

Fonte: autor

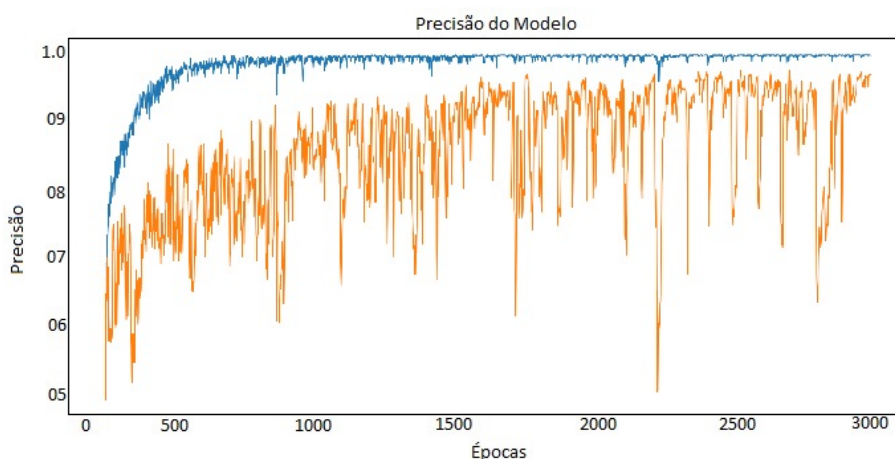


Figura 34: InceptionV3 Precisão por Época

Fonte: autor

As redes *MobileNETV2* e *InceptionV3* representadas pelas Figuras 34 e 33 não possuem *link* entre si e são modelos completamente diferentes e independentes. Neste caso, a rede *MobileNETV2* ao utilizar o conjunto de dados de treinamento convergiu em 1500 épocas, com uma precisão média de 95%. Enquanto a rede *InceptionV3* rapidamente convergiu em relação aos outros modelos, com apenas 500 épocas atingiu o valor máximo, e média em 98%. Com o conjunto de dados de teste, os modelos 34 e 33 estabeleceram uma média em 86% e 82%, respectivamente.

Os gráficos apresentados nas Figuras 35 e 36 representam os valores do ritmo de perda dos modelos VGG16 e VGG19 respectivamente. As linhas laranja e azul, indicam o conjunto de dados de teste e treinamento respectivamente, o eixo X representa o número de épocas e o eixo Y o valor do ritmo de perda.

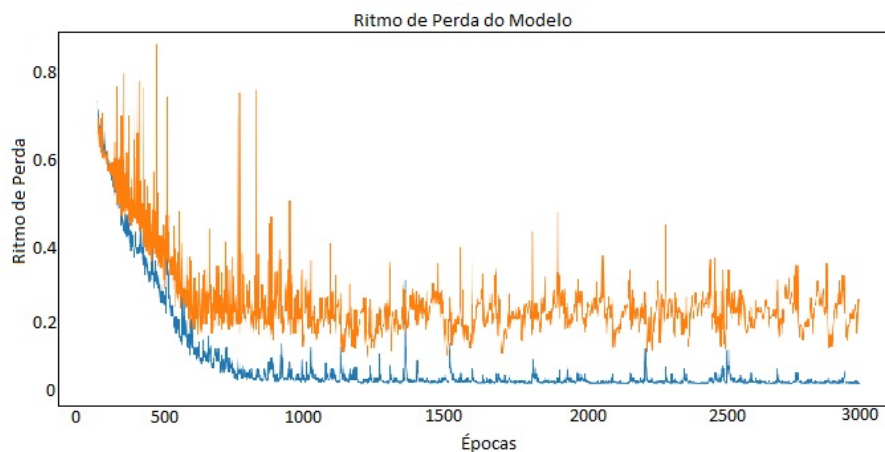


Figura 35: VGG16 ritmo de perda por Época

Fonte: autor

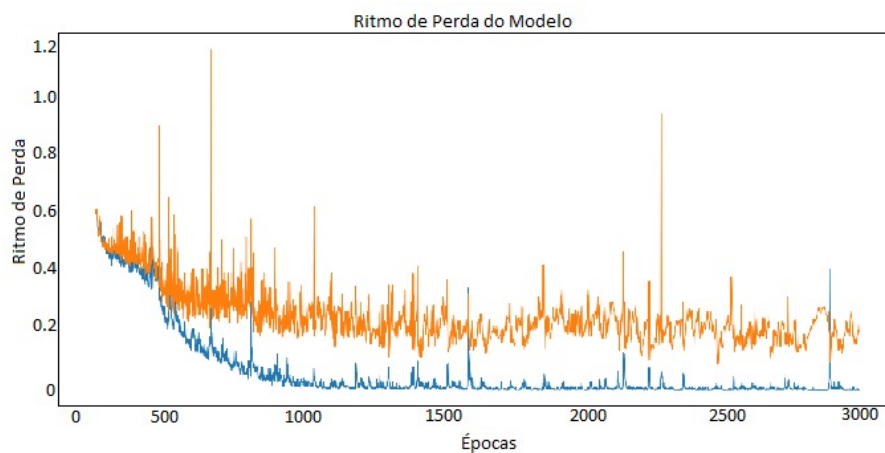


Figura 36: VGG19 ritmo de perda por Época

Fonte: autor

Os gráficos da rede VGG16 apresentam picos bastante regulares e uma considerável

oscilação da linha laranja, que indica o conjunto de dados de teste. É possível analisar que a rede não se estabilizou no final das três mil épocas. Com valor médio em 0.05 e máximo em 0.69. Com o conjunto de dados de teste, representado pela linha laranja, a variância é maior conforme a oscilação da linha, mesmo após três mil vezes o valor médio deste conjunto de dados foi de 0,20 e o máximo foi 0,83. A rede VGG19, embora demonstre aparentemente dados mais estáveis do que o modelo anterior, os valores foram ligeiramente superiores, com média em 0.07 e máximo de 0.6 no conjunto de dados de treino. Com o *dataset* de teste também atingiu valores levemente superiores, com média em 0.2 e máximo em 1,3. Os gráficos apresentados nas Figuras 37 e 38 representam os valores do ritmo de perda dos modelos *InceptionV3* e *MobileNETV2* respectivamente. As linhas laranja e azul, indicam o conjunto de dados de teste e treinamento respectivamente, o eixo X representa o número de épocas e o eixo Y o valor do ritmo de perda.

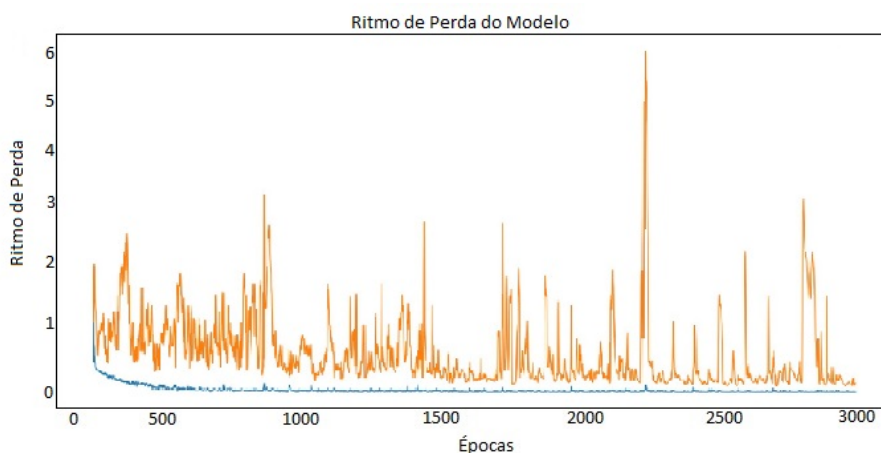


Figura 37: InceptionV3 ritmo de perda por Época

Fonte: autor

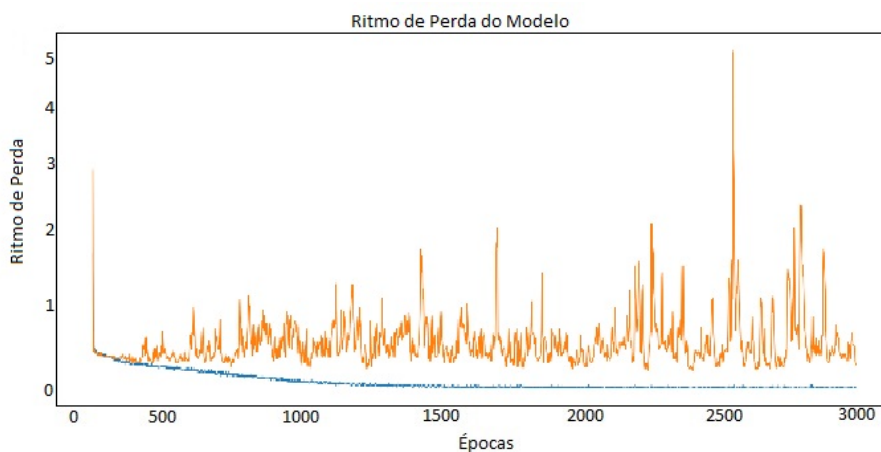


Figura 38: MobileNETV2 ritmo de perda por Época

Fonte: autor

O modelo *InceptionV3* é otimizado o suficiente para diminuir a taxa de perda, é possível ver apenas olhando para o gráfico, a linha azul que representa o conjunto de dados de treinamento manteve-se estável para todas as três mil épocas, o valor médio foi de 0,03, enquanto o valor o máximo foi 0.19 com o conjunto de dados de teste a oscilação é bastante considerável, conforme demonstrado no gráfico referente a Figura 37, apresentando valores máximos de 65 e média de 23.

O gráfico na Figura 38 indica os valores em rede *MobileNetV2*, embora a linha azul indicando o conjunto de dados de treinamento seja bastante estável, os dados de teste não apresentam valores relevantes, com média em 0,03 e máximo de 2,2 nos dados de treinamento, já no conjunto de teste, a média é 0,7 e máxima em 5,6.

O melhor desempenho a partir dos parâmetros definidos, foi alcançado pela arquitetura VGG-16, com média de 94% de precisão e ritmo de perda de 20% , conforme mostra a tabela 8. Dessa forma, tendo como critério o resultado obtido, justifica-se a adesão desta arquitetura de RNC para futuros experimentos.

Tabela 8: Parâmetros de avaliação das arquiteturas de RNC

Parâmetro	VGG16	VGG19	InceptionV3	MobileNet
Precisão Média (Treino)	97%	96%	98%	95%
Precisão Média (Validação)	94%	92%	74%	82%
Ritmo de Perda Médio (Treino)	5%	7,5%	3%	10%
Ritmo de Perda Médio (Validação)	20%	28%	74%	76%

A arquitetura da RNC VGG16 conta com:

- contém um total de 16 camadas nas quais os parâmetros de pesos e *bias* são aprendidos;
- Um total de 13 camadas convolucionais são ordenadas uma após a outra e 3 camadas densas realizam a classificação;
- O número de filtros nas camadas de convolução segue um padrão crescente;
- Os recursos informativos são obtidos por camadas *max pooling* aplicadas em diferentes etapas da arquitetura;
- As camadas densas compreendem 4096, 4096 e 1000 nodos cada.
- Um ponto contra esta arquitetura é o fato de ser lenta para treinar e produzir um modelo com tamanho muito grande.

Sua arquitetura pode ser visualizada a partir da Figura 39.

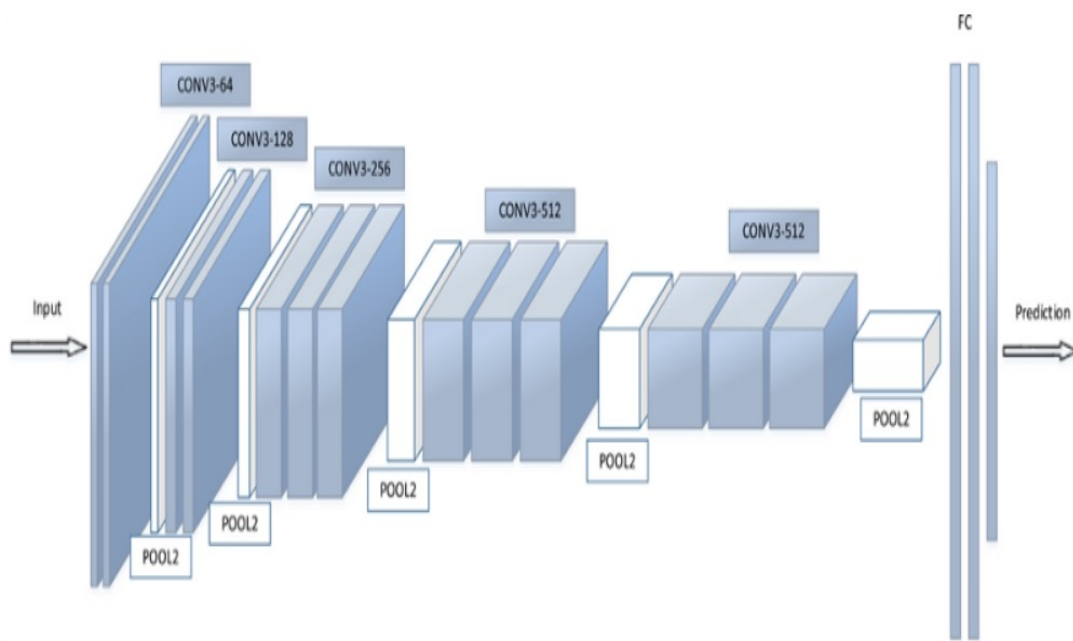


Figura 39: Arquitetura da RNC VGG16.

Fonte: Kaggle

3.3 Segundo Experimento

No segundo experimento buscou-se aumentar o número de classes nos ciclos de treinamento da RNC e avaliar o resultado a cada nova iteração. A partir dos dados gerados no primeiro experimento, o número de épocas pôde ser subtraído. Neste segundo caso as iterações começam em 100 épocas e terminam em 500 épocas iterando em 100 e, ao fim de cada ciclo de cruzamento entre os hiperparâmetros, uma nova classe é adicionada ao *dataset* conforme exemplificado na Figura 40. Os ciclos de treinamento e adição de classes encerram em dois casos: A) atingir o número máximo de classes para este experimento e B) não ocorrer mais alterações nos valores de precisão entre dois ciclos. Para este experimento o número de classes máximo estipulado é de cinco.

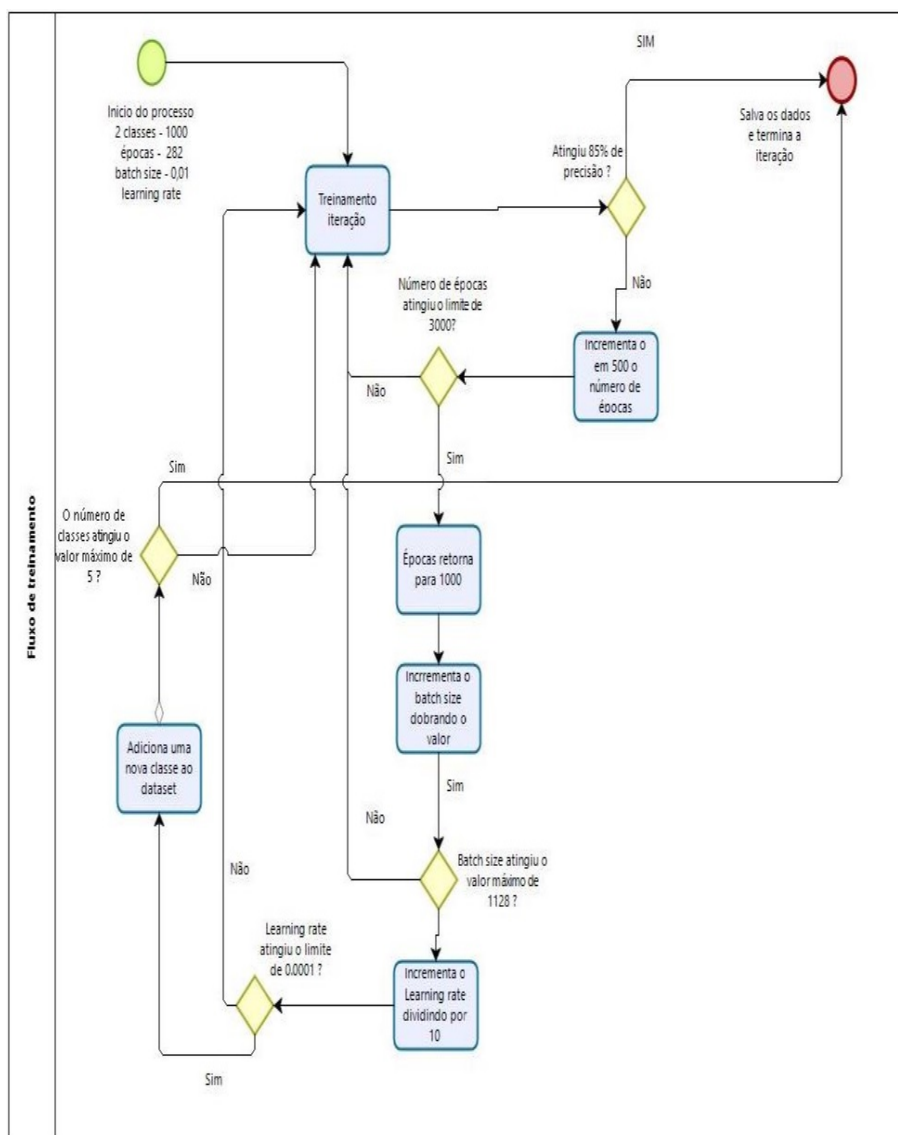


Figura 40: Fluxo de Treinamento por Ciclo Experimento 2

Fonte: autor

O *dataset* utilizado nesta segunda experiência conta com um total de 18,605 imagens dividido em 5 classes. Novamente os hiperparâmetros avaliados são, além do número de épocas, o *learning rate*, *batch size*. Os dados gerados serão analisados e discutidos na próxima seção.

3.3.1 Análise dos Dados - Segundo Experimento

A Tabela 9 refere-se aos valores gerados a partir de cada ciclo de treinamento executado no experimento dois. O critério de parada definido para esta experiência foi atingir o número máximo de classes disponíveis, ou seja cinco classes.

Tabela 9: Dados dos Ciclos de Treinamento Experimento 2

Epocas	<i>Batch Size</i>	<i>Learning Rate</i>	Precisão
500	282	0.01	0.49
400	282	0.01	0.51
300	282	0.01	0.51
200	282	0.01	0.51
100	282	0.01	0.51
500	564	0.01	0.51
400	564	0.01	0.51
300	564	0.01	0.51
200	564	0.01	0.51
100	564	0.01	0.55
500	846	0.01	0.51
400	846	0.01	0.51
300	846	0.01	0.51
200	846	0.01	0.49
100	846	0.01	0.51
500	1128	0.01	0.89
400	1128	0.01	0.51
300	1128	0.01	0.51
200	1128	0.01	0.51
100	1128	0.01	0.51
500	282	0.001	0.96
400	282	0.001	0.96
300	282	0.001	0.96
200	282	0.001	0.96
100	282	0.001	0.84

500	564	0.001	0.97
400	564	0.001	0.96
300	564	0.001	0.96
200	564	0.001	0.93
100	564	0.001	0.83
500	846	0.001	0.95
400	846	0.001	0.95
300	846	0.001	0.91
200	846	0.001	0.93
100	846	0.001	0.81
500	1128	0.001	0.93
400	1128	0.001	0.90
300	1128	0.001	0.95
200	1128	0.001	0.89
100	1128	0.001	0.92
500	282	0.0001	0.96
400	282	0.0001	0.97
300	282	0.0001	0.97
200	282	0.0001	0.95
100	282	0.0001	0.93
500	564	0.0001	0.97
400	564	0.0001	0.96
300	564	0.0001	0.96
200	564	0.0001	0.96
100	564	0.0001	0.90
500	846	0.0001	0.97
400	846	0.0001	0.95
300	846	0.0001	0.96
200	846	0.0001	0.93
100	846	0.0001	0.82
500	1128	0.0001	0.97
400	1128	0.0001	0.97
300	1128	0.0001	0.92
200	1128	0.0001	0.92
100	1128	0.0001	0.82

A Tabela 10 apresenta os hiperparâmetros que obtiveram os melhores resultados de precisão. Em um total de 35 ciclos de treinamento, atingiram valores superiores à 85% e, destes resultados, 7 possuem os melhores resultados, todos atingiram valores em 0.97 ou 97%.

Tabela 10: Melhores Resultados Experimento 2

Épocas	Batch Size	Learning Rate	Precisão
500	564	0.001	0.97
500	1128	0.0001	0.97
400	1128	0.0001	0.97
500	846	0.0001	0.97
500	564	0.0001	0.97
400	282	0.0001	0.97
300	282	0.0001	0.97

É possível perceber que 57% dos melhores resultados foram atingidos a utilizando 500 épocas e, o hiperparâmetro *learning rate* 0.0001 foi o que mais apareceu, um total de 87% entre os melhores. Já o *batch size* ficou fragmentado afetando uma conclusão precisa. Neste caso para o terceiro e último experimento os dados utilizados foram 300 épocas, um *batch size* de 282 e *learning rate* 0.0001, a escolha destes hiperparâmetros visa reduzir o custo computacional (limitação do *hardware*).

3.4 Terceiro Experimento

O terceiro experimento conta com os hiperparâmetros definidos na experiência anterior, 300 épocas, um *batch size* de 282 e *learning rate* 0.0001, e conforme já mencionado a escolha destes hiperparâmetros visa reduzir o custo computacional.

Este ciclo de treinamento conta com o *dataset* completo, totalizando 37,210 imagens divididas em 10 classes onde, cada classe segmentada em 75% e 25% para treino e teste respectivamente.

Um critério de parada também foi estabelecido para este ciclo de treinamento, caso o valor da precisão não sofra alterações em 10 épocas o ciclo de treinamento é interrompido automaticamente conforme mostra a Figura 41, o dado correspondente a precisão não foi aprimorado, portanto o algoritmo entendeu que não é necessário terminar o treinamento, então interrompeu o ciclo precocemente.

```

Epoch 00053: val_acc did not improve from 0.99049
Epoch 54/100
93/93 [=====] - 97s 1s/step - loss: 0.0319 - acc: 0.9880 - val_loss: 0.0240 - val_acc: 0.9929

Epoch 00054: val_acc improved from 0.99049 to 0.99294, saving model to best_model.h5
Epoch 55/100
93/93 [=====] - 99s 1s/step - loss: 0.0260 - acc: 0.9908 - val_loss: 0.0415 - val_acc: 0.9840

Epoch 00055: val_acc did not improve from 0.99294
Epoch 56/100
93/93 [=====] - 98s 1s/step - loss: 0.0353 - acc: 0.9866 - val_loss: 0.0629 - val_acc: 0.9788

Epoch 00056: val_acc did not improve from 0.99294
Epoch 57/100
93/93 [=====] - 99s 1s/step - loss: 0.0276 - acc: 0.9898 - val_loss: 0.0269 - val_acc: 0.9908

Epoch 00057: val_acc did not improve from 0.99294
Epoch 58/100
93/93 [=====] - 98s 1s/step - loss: 0.0296 - acc: 0.9896 - val_loss: 0.0565 - val_acc: 0.9827

Epoch 00058: val_acc did not improve from 0.99294
Epoch 59/100
93/93 [=====] - 99s 1s/step - loss: 0.0278 - acc: 0.9899 - val_loss: 0.0390 - val_acc: 0.9848

Epoch 00059: val_acc did not improve from 0.99294
Epoch 60/100
93/93 [=====] - 98s 1s/step - loss: 0.0298 - acc: 0.9882 - val_loss: 0.0491 - val_acc: 0.9793

Epoch 00060: val_acc did not improve from 0.99294
Epoch 61/100
93/93 [=====] - 99s 1s/step - loss: 0.0293 - acc: 0.9887 - val_loss: 0.0365 - val_acc: 0.9859

Epoch 00061: val_acc did not improve from 0.99294
Epoch 62/100
93/93 [=====] - 98s 1s/step - loss: 0.0310 - acc: 0.9892 - val_loss: 0.0297 - val_acc: 0.9901

Epoch 00062: val_acc did not improve from 0.99294
Epoch 63/100
93/93 [=====] - 99s 1s/step - loss: 0.0273 - acc: 0.9906 - val_loss: 0.0418 - val_acc: 0.9834

Epoch 00063: val_acc did not improve from 0.99294
Epoch 64/100
93/93 [=====] - 98s 1s/step - loss: 0.0263 - acc: 0.9902 - val_loss: 0.0291 - val_acc: 0.9887

Epoch 00064: val_acc did not improve from 0.99294
Epoch 00064: early stopping

```

Figura 41: Demonstração do Critério de Parada

Fonte: autor

Após completar todas as etapas dos ciclos de treinamento por meio dos experimentos, é necessário realizar a análise final dos dados. Na próxima seção são apresentados e discutidos os dados gerados.

3.4.1 Análise dos Dados - Terceiro Experimento

A primeira análise realizada refere-se aos dados do ritmo de perda e a precisão coletados a partir do treinamento. Conforme é possível visualizar nos gráficos 42 e 43 representando ritmo de perda e precisão respectivamente, ambos atingiram valor máximo em 64 épocas e, conforme já esperado, por ser um *dataset* de treinamento e a RNC já adquiriu valores dos padrões para estas imagens, os dados tendem a oscilar menos, ficando claro ao visualizar os gráficos, portanto ao final do ciclo de treinamento, ritmo de perda atingiu valor de 0.0263 e a precisão obteve 0.9902. Conforme discutido nas seções anteriores, um mínimo de 85% de precisão foi estipulado como aceitável, portanto, os dados de treinamento estão bastante acima da fronteira admissível e, assim encerrando sem a necessidade de novos ciclos de treinamento.

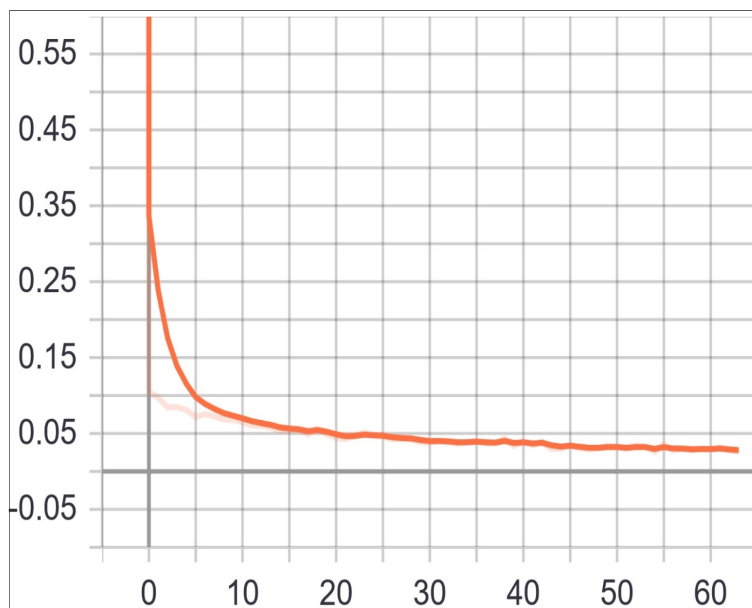


Figura 42: Dados do Ritmo de Perda no ciclo de Treino

Fonte: autor

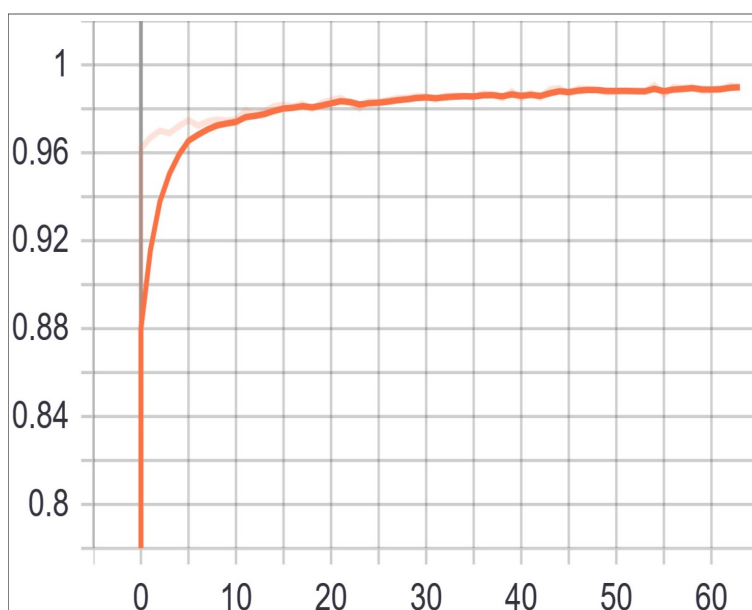


Figura 43: Dados de Precisão no ciclo de Treino

Fonte: autor

Na segunda investigação e considera-se mais importante, tratamos sobre a validação, os gráficos 44 e 45, é possível observar, devido a oscilação que os valores não se mantiveram estáveis, neste caso devido a RNC utilizar o *dataset* de validação, ou seja, imagens novas que não foram utilizadas para o ciclo de treinamento portanto a rede nunca havia trabalhado este conjunto. Os dados atingidos de precisão atingiram máximo de 0.9887 também acima do mínimo estipulado de 85% e, os valores referentes ao ritmo de perda

finalizaram em 0.0291, neste caso, assim como no ciclo de treinamento após 64 épocas foi possível encerrar sem a necessidade de repetições.

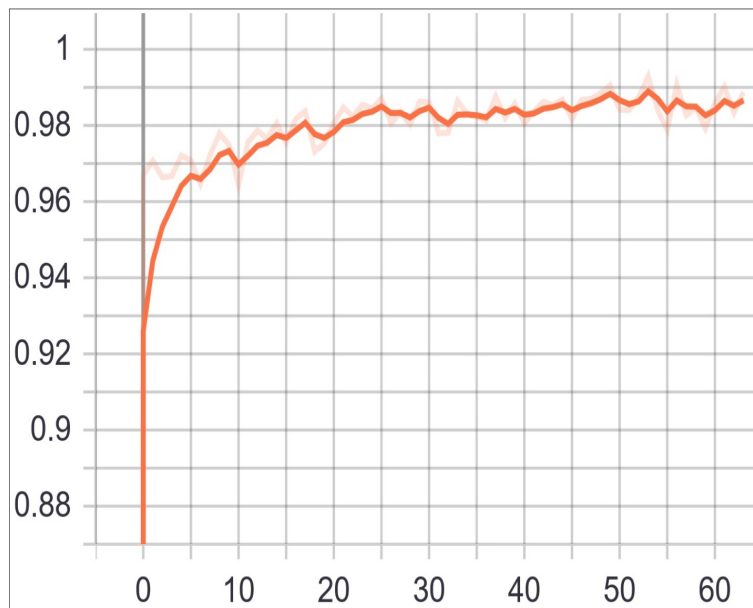


Figura 44: Dados de Precisão no ciclo de Validação

Fonte: autor

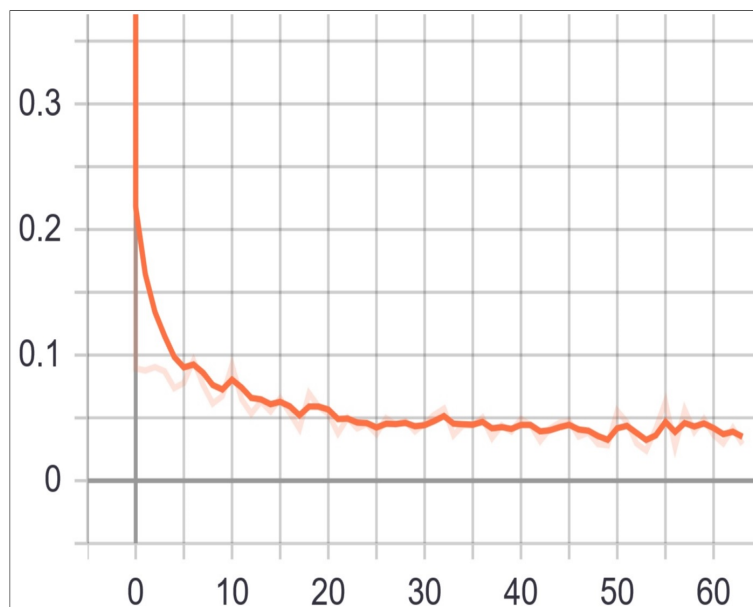


Figura 45: Dados do Ritmo de Perda no ciclo de Validação

Fonte: autor

Abaixo na Figura 46, pode-se visualizar a matriz de confusão do último e final experimento, este contanto com as 10 classes, ordenadas de 0 até 9, compreendendo todas as curvas apresentadas neste trabalho. As curvas estão ordenadas sequencialmente e nesta respectiva ordem: Cardioide, Folium Dupla, Figura Oito, Folium de Descartes, Curva

Kappa, Lemniscata de Bernoulli, Parábola, Curva Quartica, Curva Trifólio e Trissectriz de Maclaurin.

Nesta matriz indica os valores reais e os preditos pelo classificador, a diagonal principal indica os valores principais ou seja, os valores que a rede classificou corretamente, já as linhas representam os valores classificados incorretamente. Exemplificando utilizando a **linha 2** da matriz, esta representando a curva **Figura Oito**, foi classificado corretamente 294 imagens e, 23 imagens foram consideradas pela RNC como sendo da classe 5, Lemniscata de Bernoulli.

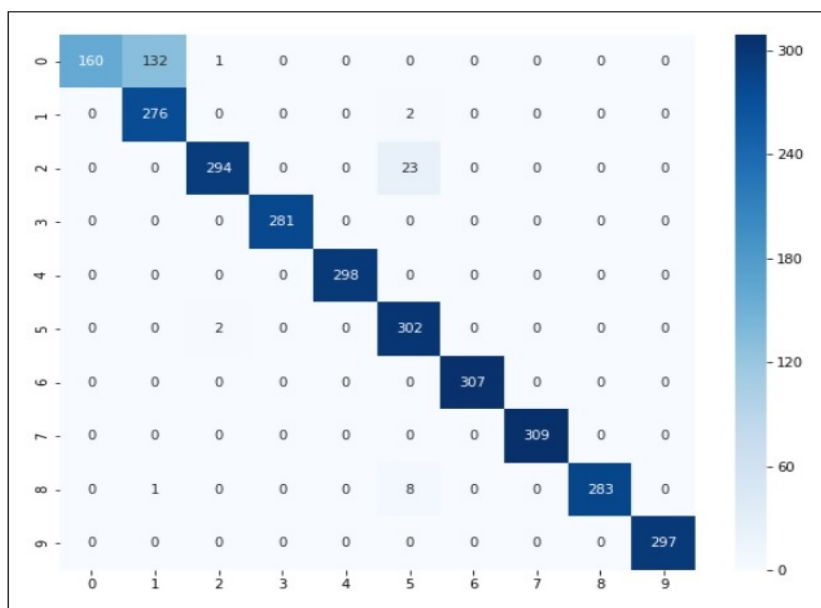


Figura 46: Matriz de Confusão

Fonte: autor

A partir destes dados podem ser quantificado as seguintes frequências: (SOKOLOVA; LAPALME, 2009):

- Verdadeiro positivo (*true positive* — TP): Sugerido quando no conjunto real, o elemento requisitado foi classificado corretamente.
- Falso positivo (*false positive* — FP): Sugerido quando no conjunto real, o elemento requisitado foi classificado incorretamente.
- Falso verdadeiro (*true negative* — TN): Sugerido quando no conjunto real, o elemento requisitado foi classificado corretamente.
- Falso negativo (*false negative* — FN): Sugerido quando no conjunto real, o elemento que não estamos requisitando foi classificado incorretamente.

A partir dos elementos descritos acima é possível calcular a *precision*, o *recall*, a *accuracy* e o *f-score* (SOKOLOVA; LAPALME, 2009) da matriz de confusão. Suas fórmulas podem ser visualizadas abaixo:

$$Accuracy = \frac{VP + TN}{TP + FP + TN + FN}$$

Neste caso seriam todas as predições corretas divididas pelo total de predições. Este indica quanto o modelo acertou das possíveis previsões.

$$Recall = \frac{TP}{TP + FN}$$

O *Recall* indica o quão bom o modelo é em prever os positivos, no caso o positivo refere-se a classe que se está tentando prever.

$$Precision = \frac{TP}{TP + FP}$$

A precisão, diferentemente da *accuracy* indica a proporção das classificações positivas estavam realmente corretas, ou seja, o quão bem o modelo proposto trabalha.

$$F - score = 2 * \frac{precision * recall}{precision + recall}$$

A formula *f-score* demonstra um balanço entre o *precision* e o *recall* do modelo.

A matriz 11 demonstra os valores gerados a partir dos dados apresentados na matriz de confusão aplicados as formulas exibidas anteriormente.

Tabela 11: Calculo Matriz de Confusão

	Recall	Accuracy	F-Score	Precision
Cardioide	1.000	0.5479	0.7080	0.5479
Folium Dupla	0.9964	0.9892	0.9946	0.9928
Figura Oito	0.9899	0.9188	0.9577	0.9274
Folium de Descartes	1.000	1.000	1.000	1.000
Curva Kappa	1.000	1.000	1.000	1.000
Lemniscata de Bernoulli	0.9015	0.8961	0.9452	0.9934
Parábola	1.000	1.000	1.000	1.000
Curva Quartica	1.000	1.000	1.000	1.000
Curva Trifólio	1.0000	0.9692	0.9843	0.9692
Trissectriz de Maclaurin	1.000	1.000	1.000	1.000

3.5 Confirmação

Nas seções referentes aos experimentos, foram apresentados os dados referentes ao ritmo de perda e precisão da RNC, a matriz de confusão e os resultados de *accuracy*, *precision*, *recall* e *F-score*, estes dados foram gerados a partir de imagens do conjunto de dados de teste que, apesar de serem utilizados apenas como validação dos dados, são

imagens geradas a partir da criação do *dataset* das curvas, ou seja ele se assemelha em determinadas características aos conjunto de treinamento. Então como meio de realizar a confirmação do modelo foi criada a estratégia de utilizar imagens geradas manualmente. Esta avaliação tem como objetivo ponderar o funcionamento da RNC proposta em condições reais.

Esta avaliação foi realizada com 10 indivíduos, com a tarefa de desenhar uma curva de cada presente no *dataset*. O gráfico demonstrando os resultados podem ser visualizados na Figura 47

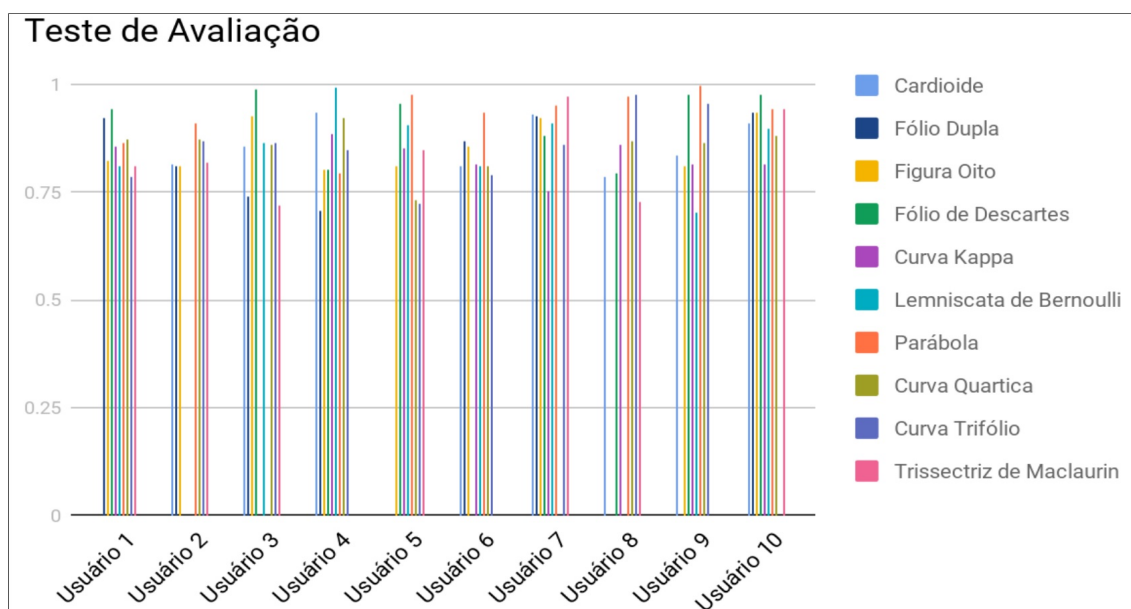


Figura 47: Teste de Avaliação

Fonte: autor

As tabelas 12 e 13 respectivamente apresentam os valores gerados pela rede para cada um dos desenhos realizados pelos usuários, neste caso os valores zero indicam um erro de classificação, ou seja a rede não classificou corretamente a curva algébrica presente na imagem.

Tabela 12: Dados Teste de Avaliação 1 - Fonte autor

	Cardioide	Fólio Dupla	Figura Oito	Fólio de Descartes	Curva Kappa
Usuário 1	0	0.9234	0.8236	0.9441	0.8562
Usuário 2	0.8153	0.8116	0.8121	0	0
Usuário 3	0.8583	0.7428	0.9272	0.9907	0
Usuário 4	0.9348	0.7081	0.8042	0.8049	0.8842
Usuário 5	0	0	0.8097	0.9557	0.8542
Usuário 6	0.8099	0.8708	0.8578	0	0.8148
Usuário 7	0.9324	0.9252	0.9245	0.8836	0.7536
Usuário 8	0.7851	0	0	0.7957	0.8597
Usuário 9	0.8367	0	0.8096	0.9782	0.8166
Usuário 10	0.9108	0.9356	0.9359	0.9755	0.8166

Tabela 13: Dados Teste de Avaliação 2 - Fonte autor

	Lemniscata	Parábola	Curva Quartica	Curva Trifólio	Trissectriz
Usuário 1	0.8123	0.8645	0.8716	0.7862	0.8121
Usuário 2	0	0.909	0.874	0.8709	0.8178
Usuário 3	0.8639	0	0.8621	0.8645	0.721
Usuário 4	0.9923	0.7952	0.9249	0.8499	0
Usuário 5	0.9075	0.9755	0.7313	0.7252	0.8468
Usuário 6	0.8121	0.9335	0.8119	0.7903	0
Usuário 7	0.9124	0.9519	0	0.8604	0.9741
Usuário 8	0	0.9717	0.8695	0.9755	0.7272
Usuário 9	0.702	0.9986	0.8645	0.9553	0
Usuário 10	0.8987	0.9441	0.8798	0	0.9441

Apesar do valor de precisão na validação da rede no último experimento atingiu 98%, os dados de avaliação apresentam uma outra realidade. O valor mínimo de precisão encontrado foi de 0.702 ou 70.2% por meio do desenho Lemniscata de Bernoulli do usuário 9, já o valor máximo foi de 0.9907 ou 99% no desenho Fólio de Descartes do usuário 3. As curvas que apresentaram o maior índice de erros foram a Fólio Dupla e a Trissectriz de Maclaurin ambas com 3 erros cada, e 4 curvas apresentaram 1 erro apenas cada, Figura Oito, Parábola, Curva Quártica e Curva Trifólio.

4 CONSIDERAÇÕES FINAIS

Este trabalho buscou propor uma abordagem inovadora para realizar a detecção e classificação de curvas algébricas no plano cartesiano utilizando Redes Neurais Convolucionais. Para atingir o objetivo foi necessário criar e estruturar um conjunto de dados consistente e arquitetar uma estrutura de CNN a partir de três experimentos fundamentais.

Ao final dos experimentos apresentados neste trabalho foi possível propor uma arquitetura neural que, inspirada na arquitetura convolucional VGG-16, conseguiu atingir valores superiores a 90% de precisão a partir do dataset apresentado.

Pode-se então concluir que é possível implementar e arquitetar uma Rede Neural Convolucional utilizando linguagens, bibliotecas e ambientes disponíveis gratuitamente na web.

Estas estruturas neurais são flexíveis e podem ser utilizadas em diferentes tipos de classificação e detecção de imagens em inúmeros contextos.

Imagens sintéticas, desenvolvidas a partir de estratégias computacionais podem ser utilizadas e aplicadas a fim de representar elementos reais e cotidianos.

A arquitetura proposta poderia ser avaliada e validada pela comunidade acadêmica a fim de mensurar os efeitos no aprendizado e mais testes precisam ser realizados em situações complexas e em situações de maior intensidade.

Como trabalhos futuros, espera-se ampliar o número de classes no conjunto de dados, aplicar a CNN a um dispositivo móvel e validar o sistema por meio de testes com a comunidade acadêmica.

REFERÊNCIAS

- ANTHONY, E.; SEVETSON, E. *LibGuides: Systematic and Literature Reviews*. Providence, Rhode Island 02912, 2018.
- ARBARELLO, E. et al. *Geometry of Algebraic Curves: Volume I*. [S.l.]: Springer, 2007.
- AXLER, S. *Linear Algebra Done Right*. [S.l.]: Springer, Cham, 2015.
- BARQUEIRO, R. R. M.; BARQUEIRO, A. C. Inclusão da pessoa com deficiência visual no mercado de trabalho. *Cultura médica* ., p. 481–494, 2010.
- BAUNEL, R. C. R. d. C.; RIBEIRO, M. L. S. *Educação Especial: do Querer ao Saber*. [S.l.: s.n.], 2007.
- BERSCH, R. *Introdução a Tecnologia Assistiva*. 2017. Porto Alegre - RS. ASSISTIVA • TECNOLOGIA E EDUCAÇÃO.
- BOKHOVE, C.; DRIJVERS, P. Digital tools for algebra education: Criteria and evaluation. *International Journal of Computers for Mathematical Learning*, 2010.
- BOURNE, R. R. A. et al. Causes of vision loss worldwide, 1990–2010: a systematic analysis. *Lancet Glob Health*, v. 1, n. 6, p. 339–349, dez. 2013.
- CANZIANI, A.; PASZKE, A.; CULURCIELLO, E. An analysis of deep neural network models for practical applications. *CoRR*, 2016.
- CARVALHO, D. C. de. A psicologia frente a educação e o trabalho docente. *Psicologia em Estudo*, 2002.
- CHUNG, H.; LEE, S. J.; PARK, J. G. Deep neural network using trainable activation functions. *International Joint Conference on Neural Networks (IJCNN)*, 2016.
- CURY, H. N. et al. Álgebra e educação algébrica: Concepções de alunos e professores de matemática. *Educação Matemática em Revista*, 2002.
- D'SOUZA, H. J.; BIX, R. A. Analytic geometry. *Encyclopædia Britannica*, 2016.
- FARIAS, N.; BUCHALLA, C. M. A classificação internacional de funcionalidade, incapacidade e saúde da organização mundial da saúde: Conceitos, usos e perspectivas. *Revista Brasileira de Epidemiologia*, 2005.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning (Adaptive Computation and Machine Learning series)*. [S.l.]: The MIT Press, 2016.

- GRANADA, R. P.; BARWALDT, R.; ESPÍNDOLA, D. B. Glossary of computational terms as a stimulus to programming logic: a case study with deaf students. *IEEE Frontiers in Education Conference (FIE)*, 2018.
- HAYKIN, S. *Redes Neurais: Princípios e Prática*. [S.l.]: Bookman, 2017.
- HOWARD, A. G. et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *CoRR*, abs/1704.04861, 2017. Disponível em: <http://arxiv.org/abs/1704.04861>.
- JING, Y.; GUANCI, Y. Modified convolutional neural network based on dropout and the stochastic gradient descent optimizer. *Algorithms*, 2018.
- KITCHENHAM, B.; CHARTERS, S. Guidelines for performing systematic literature reviews in software engineering. *EBSE 2007-001*, 2007.
- LAPLANE, A. L. F. de. O que os dados do censo escolar revelam sobre as barreiras à inclusão ? *Educação e Fronteiras On-Line*, 2015.
- LAWRENCE, J. D. *A Catalog of Special Plane Curves (Dover Books on Mathematics)*. [S.l.]: Dover, 2014.
- LECUN, Y. et al. Object recognition with gradient-based learning. In: *Shape, Contour and Grouping in Computer Vision*. London, UK, UK: Springer-Verlag, 1999. p. 319–. ISBN 3-540-66722-9. Disponível em: <http://dl.acm.org/citation.cfm?id=646469.691875>.
- LEUBA, G.; KRAFTSIK, R. Changes in volume, surface estimate, three-dimensional shape and total number of neurons of the human primary visual cortex from midgestation until old age. *Anatomy and Embryology*, v. 190, n. 4, p. 351–366, Oct 1994. ISSN 1432-0568. Disponível em: <https://doi.org/10.1007/BF00187293>.
- LIN, T.-Y. et al. Microsoft coco: Common objects in context. *arXiv Computer Vision and Pattern Recognition*, 2015.
- MANCINI, M. et al. Comparação do desempenho funcional de crianças com visão subnormal e crianças com desenvolvimento normal aos 2 e 6 anos de idade. *Revista Terapia Ocupacional Universidade de São Paulo*, 2010.
- MCCULLOCH, W.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, v. 5, p. 115–133, 1943.
- MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 1943.
- OLIVEIRA, A. Castro Soares de et al. Aplicação de redes neurais artificiais na previsão da produção de álcool. *Ciencia E Agrotecnologia - CIENC AGROTEC*, 2010.
- OLIVEIRA, L. M. B. *Cartilha do Censo 2010 – Pessoas com Deficiência*. Secretaria de Direitos Humanos da Presidência da República. [S.l.]: Secretaria Nacional de Promoção dos Direitos da Pessoa com Deficiência, 2012.
- OLIVEIRA, M. K. de. Vygotsky: alguns equívocos na interpretação de seu pensamento. *Cadernos de Pesquisa*, 1992.

- OLIVEIRA, M. K. de. O pensamento de vygotsky como fonte de reflexão. *Cadernos CEDES*, 1995.
- PONCE, J.; FORSYTH, D. *Computer vision: a modern approach*. [S.l.]: Prentice Hall, 2002.
- REILY, L. *Escola Inclusiva - Linguagem e Mediação*. [S.l.]: PAPIRUS, 2004.
- Sá, E. D. de; CAMPOS, I. M. de; SILVA, M. B. C. *Atendimento Educacional Especializado - Deficiência Visual*. [S.l.]: Editora Cromos, 2007.
- SANDLER, M. et al. Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. *CoRR*, abs/1801.04381, 2018. Disponível em: <http://arxiv.org/abs/1801.04381>.
- SCHOENFELD, A. H. The math wars. *Educational Policy*, 2004.
- SILVA, A. P. e. *Novo Dicionário Brasileiro Melhoramentos*. [S.l.]: Editora Globo, 1969.
- SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations*, 2015.
- SOKOLOVA, M.; LAPALME, G. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, n. 4, p. 427–437, jul. 2009.
- STEVENS, G.; WHITE, R.; FLAXMAN, S. Global prevalence of visual impairment and blindness: magnitude and temporal trends, 1990–2010. *Ophthalmology*, 2013.
- SZEGEDY, C. et al. Rethinking the inception architecture for computer vision. *CoRR*, abs/1512.00567, 2015. Disponível em: <http://arxiv.org/abs/1512.00567>.
- SZELISKI, R. *Computer vision: algorithms and applications*. [S.l.]: Business Media, 2010.
- THYLEFORS, B. et al. Global data on blindness. *Bull World Health Organ*, v. 73, p. 115–21, 1995.
- VEER, R. V. D.; VALSINER, J. *Vygotsky - Uma Síntese*. [S.l.: s.n.], 1996.
- ÁVILA, M.; ALVES, M. R.; NISHI, M. *As condições de Saúde Ocular no Brasil*. 1ª. ed. [S.l.]: Conselho Brasileiro de Oftalmologia, 2015.
- VYGOTSKI, L. S. *Obras Escogidas - V: Fundamentos de defectología*. [S.l.]: Antonio Machado Libros, 1997.
- VYGOTSKY, L. S. *The collected works of L. S. Vygotsky. Volume 2: The fundamentals of defectology (abnormal psychology and learning disabilities)*. [S.l.]: Springer US, 1993.
- WILSON, R. C. et al. The eighty five percent rule for optimal learning. *Nature Communications*, 2019.
- ZANELLA, A. V. Zona de desenvolvimento proximal: análise teórica de um conceito em algumas situações variadas. *Temas psicologia*, 1994.