

CENTRO UNIVERSITÁRIO FEI
GLAUCO RIGOLIN

**PROTÓTIPO INTELIGENTE DE PRÓTESE MIOELÉTRICA DE MÃO DE BAIXO
CUSTO AUXILIADA POR SISTEMA DE VISÃO**

São Bernardo do Campo

2020

GLAUCO RIGOLIN

**PROTÓTIPO INTELIGENTE DE PRÓTESE MIOELÉTRICA DE MÃO DE BAIXO
CUSTO AUXILIADA POR SISTEMA DE VISÃO**

Dissertação de Mestrado apresentada ao Centro Universitário FEI, como parte dos requisitos necessários para obtenção do título de Mestre em Engenharia Elétrica.
Orientado pela Prof^a Dr^a Maria Claudia Ferrari de Castro.

São Bernardo do Campo

2020

Ficha catalográfica

Rigolin, Glauco.

Protótipo inteligente de prótese mioelétrica de mão de baixo custo auxiliada por sistema de visão. / Glauco Rigolin. São Bernardo do Campo, 2020.

121 p. : il.

Dissertação - Centro Universitário FEI.

Orientadora: Prof.^a Dra. Maria Claudia Ferrari de Castro.

1. Prótese. 2. Rede Neural. 3. Classificador de objetos. 4. EMG. I. Ferrari de Castro, Maria Claudia, orient. II. Título.

Elaborada pelo sistema de geração automática de ficha catalográfica da FEI com os dados fornecidos pelo(a) autor(a).



APRESENTAÇÃO DE DISSERTAÇÃO ATA DA BANCA EXAMINADORA

Mestrado

Programa de Pós-Graduação Stricto Sensu em Engenharia Elétrica

PGE-10

Aluno: Glauco Rigolin

Matrícula: 117307-9

Título do Trabalho: Protótipo de prótese de mão de baixo custo, auxiliada por classificador visual de objetos baseado em rede neural.

Área de Concentração: Inteligência Artificial Aplicada à Automação e Robótica

Orientador: Prof^a Dr^a Maria Claudia Ferrari de Castro

Data da realização da defesa: 07/04/2020

ORIGINAL ASSINADA

Avaliação da Banca Examinadora:

A necessidade de alteração do título da dissertação foi consenso entre os membros da banca, uma vez

que o atual não expressa a totalidade do trabalho desenvolvido. A sugestão para o novo título foi:

"Protótipo inteligente de prótese mioelétrica de mão de baixo custo auxiliada por sistema de visão".

São Bernardo do Campo, / / .

MEMBROS DA BANCA EXAMINADORA

Prof^a Dr^a Maria Claudia Ferrari de Castro

Ass.: _____

Prof. Dr. Carlos Eduardo Thomaz

Ass.: _____

Prof. Dr. Eric Rohmer

Ass.: _____

A Banca Julgadora acima-assinada atribuiu ao aluno o seguinte resultado:

APROVADO

REPROVADO

VERSÃO FINAL DA DISSERTAÇÃO

APROVO A VERSÃO FINAL DA DISSERTAÇÃO EM QUE
FORAM INCLUÍDAS AS RECOMENDAÇÕES DA BANCA
EXAMINADORA

Aprovação do Coordenador do Programa de Pós-graduação

Prof. Dr. Carlos Eduardo Thomaz

Dedico este trabalho à Deus, à minha esposa,
aos meus filhos, meus pais e meus amigos que
me incentivaram e apoiaram.

AGRADECIMENTOS

À minha orientadora, Profa. Dra. Maria Cláudia Ferrari de Castro, por acreditar no meu trabalho, pelo grande apoio e pela ajuda nas metas e objetivos necessários para o desenvolvimento deste trabalho.

A todos os professores, por todos os compartilhamentos de conhecimentos.

A todos os familiares, especialmente minha esposa e filhos que sempre me apoiaram.

Ao Centro Universitário FEI, por toda infraestrutura e disponibilidade de recursos essenciais no desenvolvimento deste trabalho.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001, a quem também agradeço.

RESUMO

Tendo em vista as necessidades de pessoas com amputação de membros superiores, é proposto, neste trabalho, o desenvolvimento de um protótipo de prótese de mão de baixo custo, operando em conjunto com uma rede neural para reconhecimento de objetos presentes no cotidiano para definir o padrão de preensão a ser executado, e de um sistema de Eletromiografia de superfície (sEMG) para comandar o acionamento dos motores da prótese. A prótese foi construída com material chamado de ácido polilático (PLA) e impressa em uma impressora Voolt 3D GI3. A rede neural convolucional *Visual Geometry Group* (VGG) foi escolhida para a classificação dos objetos e foi modificada para operar no microcomputador Raspberry Pi 3 (RPI3), que foi conectado a uma webcam. Foram utilizados os softwares Keras, como interface de programação de aplicativos, e TensorFlow, como software de computação numérica. Por meio de um banco de imagens de objetos de uso diário, a rede neural foi treinada para a classificação dos objetos, em função do tipo de preensão necessário para pegá-los, em cinco classes: Potência Punho Neutro, Potência Punho Pronado, Pinça de três pontos, Extensão do Indicador e Preensão Lateral. A imagem capturada do objeto que se deseja manusear é classificada pela rede neural e, por meio do sinal dos sensores de sEMG, o microcomputador Raspberry Pi comanda o fechamento e abertura da prótese para a correta manipulação do objeto. A prótese de mão proposta obteve 100% de acurácia para a preensão de objetos da classe potência com punho neutro, 99% de acurácia para a preensão de objetos da classe potência com punho pronado, 98% de acurácia para a preensão de objetos da classe pinça de três pontos, 99% de acurácia para a preensão de objetos da classe preensão lateral e 99% de acurácia para o uso da classe extensão do indicador.

Palavras-chave: Rede Neural. Classificação Visual. Prótese. EMG.

ABSTRACT

Considering the needs of persons with upper limb amputation, it is proposed, in this work, it is proposed the development of a low-cost hand prosthesis prototype, operating in conjunction with a neural network to recognize daily uses objects and to define the pattern of grip to be executed, and a Surface Electromyography (sEMG) system to command the drive of the motors of the prosthesis. The prosthesis was constructed with polylactic acid (PLA) material and printed on a Voolt 3D GI3 printer. The Visual Geometry Group (VGG) was chosen to classify the objects and was modified to operate on the Raspberry Pi 3 (RPI3) microcomputer, that was connected to a webcam. Keras software was used as an application programming interface and TensorFlow as numerical computing software. With the database of daily use objects, the neural network was trained to classify them, based on the grasp type needed, into five classes: Power Neutral, Power Pronated, Tripod Pinch, Indicator Extension and Lateral key hold. The captured image of the object to be handled is classified by the neural network and, through the signal of the sEMG sensors, the Raspberry Pi microcomputer commands the closure and opening of the prosthesis for the correct manipulation of the object.

The proposed hand prosthesis obtained a 98% accuracy for the grip of the class three-point tweezers, 100% accuracy for the grip of the class power with neutral wrist, 99% accuracy for the grip of the class power with pronated wrist, 99% accuracy for the grip of the class lateral key hold and 99% accuracy for the use of the indicator extension class.

Keywords: Neural Network. Visual Classification. Prosthesis. sEMG.

LISTA DE ILUSTRAÇÕES

Figura 1 – Estudo de preensão por meio da dimensão	17
Figura 2 – Quantidade de preensões importantes	18
Figura 3 – Redes neurais em cascata	18
Figura 4 – Robô com garra paralela	19
Figura 5 – Prótese comandada por cabos feita em impressora 3D	21
Quadro 1 – Comparação entre métodos	27
Figura 6 – Arquitetura funcional da prótese inteligente	23
Figura 7 – Representação das preensões da prótese inteligente	23
Figura 8 – Representação básica do mecanismo de controle motor e unidade motora	30
Figura 9 – Sinais de sEMG captados com 2 eletrodos, em modo diferencial simples	31
Figura 10 – Descrição das 16 camadas da rede VGG-16	33
Figura 11 – Exemplo da estrutura do tratamento de imagem da rede VGG16	33
Figura 12 – Exemplo de classificador K-NN	36
Figura 13 – Exemplo de classificador LDA	37
Figura 14 – Modelo da prótese de mão Kwawu 2	38
Figura 15 – Processo de impressão da prótese Kwawu 2	39
Figura 16 – Câmera USB e laser instalados na área palmar da prótese	39
Figura 17 – Algoritmo de comando da prótese	40
Figura 18 – Microprocessador Raspberry Pi 3 (RPI3)	41
Figura 19 – Sensor Muscular Myoware	42
Figura 20 – Sensor para medição do sEMG	42
Figura 21 – Arduino nano	42
Figura 22 – Prótese com servos e cabos de aço	43
Figura 23 – Exemplo de bateria de 11.1V com capacidade de 2200mAh	44
Figura 24 – Diagrama geral do sistema	44
Figura 25 – Imagens semelhantes de objetos nas 4 classes	51
Figura 26 – Objetos utilizados por Ghazaei et al. (2017)	52
Figura 27 – Objetos utilizados para comparação entre experimentos	53
Figura 28 – Objetos para validação dos tipos de preensão	53
Quadro 2 – Resultado da pesquisa de preensões com 14 voluntários	55
Figura 29 – Gráfico de treinamento e validação do primeiro experimento	58
Figura 30 – Gráfico de treinamento e validação do quarto experimento	64

Figura 31 – Gráfico de treinamento e validação do quinto experimento	66
Figura 32 – Gráfico de treinamento e validação do sexto experimento	68
Figura 33– Exemplos de imagens para classe pinça de três pontos	70
Figura 34 –Exemplos de imagens para classe preensão lateral	70
Figura 35 – Exemplos de imagens para classe potência neutro	71
Figura 36 – Exemplos de imagens para classe potência pronado	71
Figura 37 – Exemplos de imagens para classe teclado	72
Figura 38 – Resultado da função caixa de contorno - três pontos	72
Figura 39 – Resultado da função caixa de contorno – potência	73
Figura 40 – Resultado da função caixa de contorno – pinça	74
Figura 41 – Resultado da função caixa de contorno – potência	75
Figura 42 – Objeto classificado como potência neutro	76
Figura 43 – Imagens após filtro de convolução conv2d_1	77
Figura 44 – Imagens após Activation_1	77
Figura 45 – Imagens após Batch_Normalization_1	78
Figura 46 – Imagens após Max_pooling2d_1.....	78
Figura 47 – Imagens após Dropout_1	79
Figura 48 – Imagens após filtro de convolução conv2d_2	79
Figura 49 – Imagens após Activation_2	80
Figura 50 – Imagens após filtro de convolução conv2d_3.....	80
Figura 51 – Imagens após Activation_3	81
Figura 52 – Imagens após Max_pooling2d_2	81
Figura 53 – Imagens após filtro de convolução conv2d_4	82
Figura 54 – Imagens após Activation_4	83
Figura 55 – Imagens após filtro de convolução conv2d_5	84
Figura 56 – Imagens após Activation_5	85
Figura 57 – Mira laser no objeto a ser manuseado	87
Figura 58 – LEDs indicadores de classe	88
Figura 59 – Protótipo de prótese com preensão lateral	88
Figura 60 – Protótipo de prótese com preensão de potência	89
Figura 61 – Protótipo de prótese com preensão pinça de três pontos	89
Figura 62 – Protótipo de prótese como extensão do indicador	90
Figura 63 – Tempos do classificador neural	91

LISTA DE TABELAS

Tabela 1 – Matriz de confusão para resultado 1	54
Tabela 2 – Matriz de confusão para resultado 2	54
Tabela 3 – Matriz de confusão para experimento 1	58
Tabela 4 – Sensibilidade, especificidade e acurácia do experimento 1	59
Tabela 5 – Matriz de confusão para experimento 2 com KNN	60
Tabela 6 – Sensibilidade, especificidade e acurácia com KNN	60
Tabela 7 – Matriz de confusão para experimento 2 com LDA	60
Tabela 8 – Sensibilidade, especificidade e acurácia do com LDA	61
Tabela 9 – Matriz de confusão para experimento 3 com KNN	62
Tabela 10 – Sensibilidade, especificidade e acurácia com KNN	62
Tabela 11 – Matriz de confusão para experimento 3 com LDA	63
Tabela 12 – Sensibilidade, especificidade e acurácia com LDA	63
Tabela 13 – Matriz de confusão para experimento 4	64
Tabela 14 – Sensibilidade, especificidade e acurácia do experimento 4	65
Tabela 15 – Matriz de confusão para experimento 5	66
Tabela 16 – Sensibilidade, especificidade e acurácia do experimento 5	67
Tabela 17 – Matriz de confusão para experimento 6	68
Tabela 18 – Sensibilidade, especificidade e acurácia do experimento 6	69
Tabela 19 – Descrição dos tempos da prótese	91

SUMÁRIO

1	INTRODUÇÃO	12
2	OBJETIVO	15
2.1	OBJETIVO GERAL	15
2.2	OBJETIVOS ESPECÍFICOS	15
3	REVISÃO BIBLIOGRÁFICA	16
3.1	PREENSÃO DE OBJETOS	17
3.2	PRÓTESES ACIONADAS POR CABOS	20
3.3	PRÓTESES CONTROLADAS POR sEMG	21
3.4	PRÓTESES CONTROLADAS POR SISTEMAS HÍBRIDOS	24
3.5	COMPARAÇÃO ENTRE MÉTODOS	26
4	CONCEITOS FUNDAMENTAIS	29
4.1	ELETROMIOGRAFIA E SISTEMA NERVOSO	29
4.2	REDE NEURAL CONVOLUCIONAL	32
4.2.1	Rede neural convolucional modificada	34
4.3	CLASSIFICADOR KNN	35
4.4	CLASSIFICADOR LDA	36
5	PRÓTESE DE MÃO AUTOMÁTICA	38
5.1	ALGORITMO DE OPERAÇÃO DA PRÓTESE	40
5.2	SISTEMA COMPUTADORIZADO	40
5.3	INTERFACE PARA OS ELETRODOS DE SUPERFÍCIE	41
5.4	SISTEMA DE ACIONAMENTO DA PRÓTESE	43
5.5	ALIMENTAÇÃO ELÉTRICA DO SISTEMA	43
5.6	DIAGRAMA GERAL DO SISTEMA	44
6	MATERIAIS E MÉTODOS	45
6.1	METODOLOGIA DOS EXPERIMENTOS	46
6.2	BANCO DE IMAGENS PARA TREINAMENTO E OBJETOS DE TESTE	49
6.3	VALIDAÇÃO DO TIPO DE PREENSÃO	53
7	EXPERIMENTOS E RESULTADOS	57
7.1	EXPERIMENTO 1	57
7.2	EXPERIMENTO 2	59
7.3	EXPERIMENTO 3	61
7.4	EXPERIMENTO 4	63

7.5	EXPERIMENTO 5	65
7.6	EXPERIMENTO 6	67
7.7	VISUALIZAÇÃO DOS RESULTADOS DOS TESTES.....	69
8	VISUALIZAÇÃO DAS CAMADAS DA REDE NEURAL.....	76
9	TESTE PRÁTICO	87
9.1	ANÁLISE DE TEMPOS.....	90
10	DISCUSSÃO	92
11	CONCLUSÃO	98
	REFERÊNCIAS	99
	APÊNDICE A – LITERATURA COMPLEMENTAR	103
	APÊNDICE B – PROGRAMA PARA CONTAGEM DOS PULSOS DOS SENSORES DE sEMG.....	106
	APÊNDICE C – DIAGRAMA DE LIGAÇÃO	108
	APÊNDICE D – FOTOGRAFIAS DOS EXPERIMENTOS	110
	ANEXO A – PROGRAMA DA REDE NEURAL MODIFICADA, CONFORME ROSEBROCK (2018)	120

1 INTRODUÇÃO

A preocupação com a qualidade de vida e inclusão de indivíduos que tiveram um membro ou parte dele amputado, ou nasceram com problemas congênitos, tem crescido mundialmente. Segundo Cella e Vilagra (2011), “a qualidade de vida foi definida como a percepção do indivíduo sobre a sua posição na vida, no contexto da cultura e dos sistemas de valores nos quais ele vive, e em relação a seus objetivos, expectativas, padrões e preocupações, qualquer alteração que leve a mudanças na qualidade de vida, como a amputação, faz com que o indivíduo mude sua forma de interagir com a sociedade”.... A amputação pode, além de afetar a funcionalidade do membro, limitando e contribuindo para uma reorganização na forma de realização das atividades diárias, também reduzir a autoestima, levando o indivíduo ao isolamento e até mesmo à depressão.

A evolução dos métodos de reabilitação desses indivíduos ocorre em paralelo à evolução tecnológica. Existem próteses apenas para requisitos estéticos, que não se movem, outras com movimentos simples e comandadas por um sistema de tração de cabos, até as próteses mais modernas conhecidas como mioelétricas, comandadas através de um sinal neural de ativação muscular.

Atualmente, as próteses mioelétricas são feitas de material extremamente leve, com cobertura esteticamente similar à pele humana, são equipadas com processadores incorporados e estão conectadas a sinais musculares, programadas para interpretar e imitar movimentos naturais. Contudo, estas próteses apresentam elevado custo de aquisição e exigem treinamento por parte do usuário de forma a entender a relação comando e resposta da prótese.

Refutando a ideia de que próteses de baixo custo são fixas ou movidas por cabos, este estudo mostra que é possível construir próteses de baixo custo conectadas a microcomputadores embarcados e sistemas de classificação de imagens por meio de redes neurais, superando também as limitações de treinamento por parte do usuário.

A prótese de mão proposta, construída em impressora 3D, está equipada com uma câmera USB conectada a um microcomputador Raspberry Pi embarcado, e realiza 5 padrões de movimentos auxiliada por um classificador visual de objetos baseado em rede neural. A partir do reconhecimento do tipo de objeto, através da câmera na palma da prótese, o sistema executa um dos 5 padrões de preensão: potência com punho neutro, potência com punho pronado, pinça de três pontos, preensão lateral e extensão do dedo indicador.

A incorporação de um sistema de visão e de uma rede neural convolucional à prótese a torna inteligente, tornando-a capaz de sugerir o movimento de preensão a ser utilizado e que

deverá ser validado pelo usuário, ao mesmo tempo que facilita e agiliza o processo de aprendizado e utilização da prótese.

A preensão de potência corresponde ao movimento de fechar a mão envolvendo o objeto com a superfície palmar e pode ser utilizada para segurar objetos maiores como, por exemplo, copo, garrafa, ou caixa, e dependendo da localização do objeto requer uma posição de punho neutra ou pronada. A pinça de três pontos é um movimento que inclui a extensão do indicador e do dedo médio em oposição ao polegar e pode ser utilizada para segurar objetos pequenos e leves com as pontas dos dedos como, por exemplo, mouse, bala, borracha, etc. Já a preensão lateral é aquela em que o objeto é seguro entre a face lateral do indicador flexionado e o polegar como, por exemplo, chave, caneta e papel. E por fim, a extensão do dedo indicador que, apesar de não ser uma preensão, é muito útil para o uso de teclado, tablets e celulares.

Este trabalho está estruturado na seguinte forma:

- a) Objetivo: apresenta-se a proposta deste trabalho, quais as suas metas geral e específicas;
- b) Revisão bibliográfica: estudos sobre os tipos de preensões humanas são apresentados e suas importâncias, são relatados os trabalhos existentes sobre controle e comando de próteses, partindo dos modelos acionados por cabos, passando por modelos elétricos comandados por eletromiografia e por fim, próteses auxiliadas por sistemas de visão e redes neurais;
- c) Conceitos fundamentais: são encontrados os fundamentos teóricos do projeto, como: eletromiografia; redes neurais; classificadores;
- d) Prótese de mão inteligente: a proposta da nova prótese é descrita, considerando o processo construtivo, sistema de processamento, tipos de acionamento e sistema de visão;
- e) Materiais e métodos: estão descritos em detalhes os 6 experimentos de validação do projeto;
- f) Resultados dos experimentos: são apontados os resultados de cada um dos experimentos;
- g) Visualização das camadas da rede neural: é apresentado uma visão das imagens ao passar pelas camadas da rede neural utilizada;
- h) Testes práticos: é apresentado o resultado da rede neural ao ser embarcada no protótipo da prótese;
- i) Discussão: os resultados são discutidos e comparados com os da literatura

- j) Conclusões: são apresentadas as considerações finais destacando as contribuições deste trabalho e propostas de melhorias.

2 OBJETIVO

São descritos neste capítulo os objetivos gerais e os objetivos específicos deste projeto.

2.1 OBJETIVO GERAL

Este projeto visa a construção de um protótipo inteligente de prótese mioelétrica de mão de baixo custo, com um sistema de visão incorporado de forma a identificar a forma do objeto a ser pego e definir autonomamente a preensão a ser executada. Através de um sistema de eletromiografia de superfície o usuário indica sua intenção de pegar o objeto, momento este em que a prótese executa o padrão de movimento identificado. Foram definidos 5 padrões utilizados na realização das atividades do dia a dia, sendo potência com punho neutro, potência com punho pronado, pinça de três pontos, preensão lateral e extensão do indicador.

2.2 OBJETIVOS ESPECÍFICOS

- a) desenvolvimento de protótipo de prótese em impressora 3D
- b) especificação e montagem da estrutura elétrica da prótese
- c) definição de experimentos como situações de teste
- d) treinamento e teste da rede neural em cada experimento
- e) validação da rede neural embarcada

3 REVISÃO BIBLIOGRÁFICA

São descritos neste capítulo os recentes estudos sobre projetos de próteses, métodos de preensão para garras robóticas, próteses controladas por sEMG, próteses auxiliadas por sistema de visão, próteses auxiliadas por sistema de visão e redes neurais e próteses auxiliadas por sistema cognitivo.

Cordella et al. (2016) analisaram os trabalhos e as pesquisas relativas a próteses de membros superiores, buscando identificar pontos críticos sob o ponto de vista dos usuários, relacionar os motivos do abandono da utilização das próteses e promover uma lista de atributos e aspectos para satisfazer os usuários de próteses. Com relação aos tipos de próteses, as prioridades relativas ao projeto são:

- a) próteses passivas: custo, durabilidade, funcionalidade, aparência, conforto.
- b) acionadas pelo corpo: aparência, custo, durabilidade, conforto, funcionalidade.
- c) mioelétricas: custo, durabilidade, aparência, funcionalidade, conforto.

Com relação à funcionalidade das próteses, as prioridades são:

- a) passivas: linguagem corporal, aparência, manipulação, estabilidade, preensão.
- b) ativas: linguagem corporal, preensão, manipulação, estabilidade, aparência.

Com relação aos recursos das próteses, as características a serem melhoradas são:

- a) próteses cosméticas: aparência, controle de temperatura, material da luva, habilidade e escorregamento, melhora do movimento do punho, menos chamativa, peso.
- b) próteses mioelétricas: movimentação do polegar, escorregamento, força de preensão, velocidade, aparência, controle de temperatura, precisão, peso, ruído, *feedback*.

Com relação aos tipos de preensão, os requisitos dos usuários de próteses são:

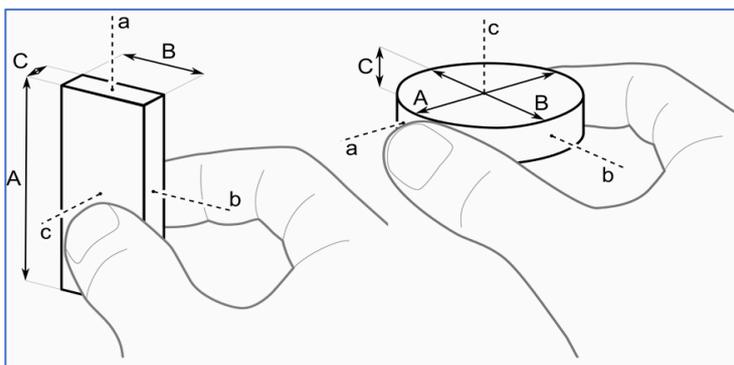
- a) tipos de preensão: lateral, pinça, gancho, esfera, potência, mão aberta, centralizado.
- b) posição descanso: mão aberta.
- c) movimentos: rotação, deslizamento, translação, indicador, empurrar moeda.

O estudo também mostra que as necessidades dos usuários de prótese divergem para fatores como: sexo, nível de educação, situação civil, etnia, ocupação antes e depois da amputação, hobbies e esportes.

3.1 PREENSÃO DE OBJETOS

No trabalho de Feix et al. (2013) foram analisados o comportamento de preensões humanas em 2 grupos de pessoas: 2 donas de casa e 2 mecânicos. Uma câmera foi colocada em cada participante para registrar os tipos de preensões utilizadas durante o dia de trabalho. Os vídeos com as imagens das pessoas foram analisados e cada preensão identificada por 2 pessoas. Um exemplo do método baseado nas dimensões dos objetos pode ser visto na Figura 1 ilustrando que objetos com formas diferentes possuem diferentes locais de preensão. O paralelogramo mostrado na Figura 1 pode ser manuseado pela dimensão A, B ou C, enquanto o cilindro pode ser manuseado pelas dimensões C e A/B. Em alguns casos o tamanho do objeto determina o tipo de preensão, pois depende da anatomia da mão humana, limitada em cerca de 15 cm entre polegar e indicador. Sete características do objeto a ser pego foram levadas em consideração para os testes: massa, formato, tamanho, posição, abertura da mão, rigidez e forma arredondada. O estudo também levou em consideração a força de preensão influenciada pela massa do objeto, mas fatores como posição do ponto de preensão, centro de gravidade e atrito não foram considerados. O estudo revelou a existência de 5 preensões mais relevantes para o uso cotidiano, sendo o tipo potência médio (1) capaz de manusear até objetos com 400g e 4,5 cm e preensão lateral (2) capaz de manusear até 150g e 2 cm, considerados muito relevantes e frequentes, seguido das preensões polegar e 2 dedos, capaz de manusear até 105g e 1cm, potência esférica e pinça de três pontos conforme pode ser visto na Figura 2. Este trabalho tem importante contribuição para reabilitação de pessoas e aprendizado de preensões de robôs.

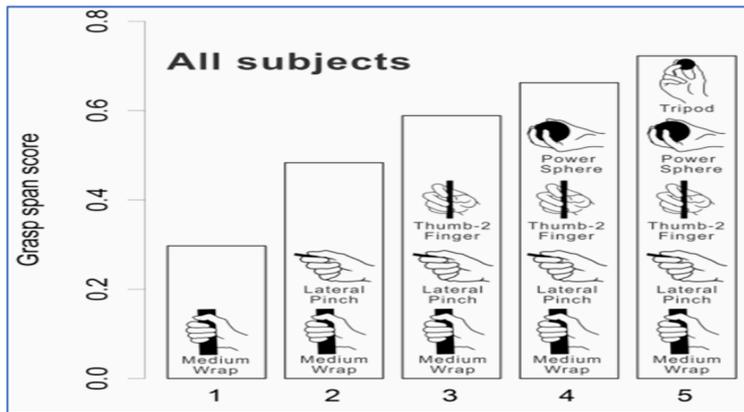
Figura 1 – Estudo de preensão por meio da dimensão



Fonte: Feix et al. 2013

Legenda: Exemplo de preensão levando em consideração a dimensão do objeto

Figura 2 – Quantidade de preensões importantes

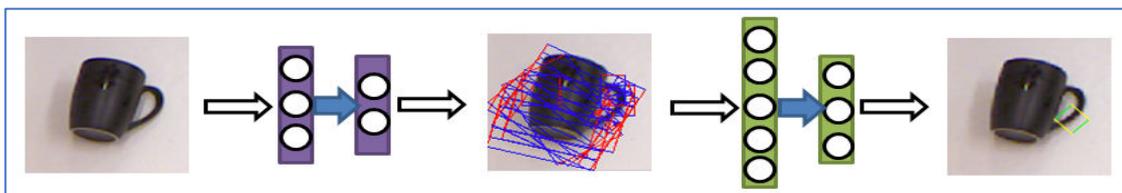


Fonte: Feix et al. 2013

Legenda: A figura mostra que a preensão média (1) e lateral (2) estão presentes no maior número de situações diárias

No trabalho de Lenz et al. (2013) foi apresentado um algoritmo para preensão com garras de robôs baseado em uma única câmera RGB com sensor de profundidade. A proposta é a utilização de duas redes neurais em cascata. A primeira rede neural de duas camadas escondidas é responsável pela determinação dos retângulos externos à imagem, representando todas as possibilidades de preensão, e a segunda rede neural auto-encoder, mais robusta, para determinar o retângulo que melhor representa o método de preensão. A Figura 3 mostra as redes neurais em cascata. E no caso mostrado, a forma de preensão escolhida foi pegar a xícara pela alça. O trabalho foi desenvolvido para robôs com garras de duas pás paralelas, conforme pode ser visto na Figura 4, mas pode ser adaptado para outros modelos de garras. Para os experimentos uma câmera de profundidade kinect foi montada na “cabeça” de dois robôs, o PR2 e o Baxter. O kinect envia uma imagem RGB-D com resolução 640x480. Foram utilizados 100 objetos divididos em 5 categorias: objetos de cozinha, objetos de laboratório, frascos, brinquedos, outros. A acurácia para o robô PR2 foi de 89% e para o robô Baxter 84%.

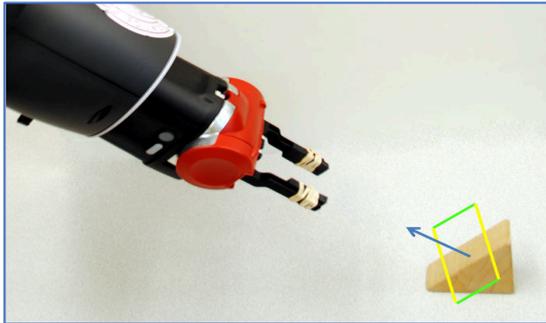
Figura 3 – Redes neurais em cascata.



Fonte: Lenz et al. 2013

Legenda: A figura exemplifica a utilização da cascata de redes neurais para busca do ponto de preensão

Figura 4 – Robô com garra paralela



Fonte: Lenz et al. 2013

Legenda: A figura exemplifica o retângulo externo à peça e o modo de preensão para robô de garras paralelas

Bohg et al. (2014) analisaram os métodos de preensão por meios analíticos, medições dos objetos ou contatos, observando as dificuldades de preensão relativas ao formato, textura, rigidez, força de atrito e erros dos próprios sensores de medição. Eles analisaram preensões humanas, preensões em garras robóticas, preensões em simuladores 2D e 3D e também métodos de preensão por análise de dados, onde o tipo de preensão é baseado em formulações analíticas, construindo uma superfície envoltória convexa sobre o objeto. Esse método calcula a força necessária para o fechamento da garra por meio do raio da maior esfera contida na envoltória do objeto. Em função de problemas de estabilidade e equilíbrio do modelo matemático com a realidade, foi sugerido o aprendizado de máquina, incluindo sistemas de visão estéreo e projetos com sensores de profundidade.

Bohg et al. (2014) identificaram quatro áreas para desenvolvimento em manipulações de objetos por meio de garras robóticas: segmentação dos objetos, aprendizado da preensão, plano de manipulação autônoma e execução robusta.

No trabalho de Li et al. (2016) foram analisados os modos de preensão humanas através de câmeras instaladas no plano de primeira pessoa, observando os tipos de preensão de pessoas exercendo atividades. Estas análises são importantes para a ciência da computação, neurociência, interação humano robô e o aprendizado na área de robótica. As imagens passaram por uma rede neural convolucional para classificação, por uma rede neural de regiões (R-CNN) para localização das mãos humanas e posteriormente por um sistema de segmentação para encontrar as diferenças entre os tipos de preensões. Foram utilizados os bancos de dados de imagens de pessoas manuseando objetos, Yale Human Grasping (YHG) e UT Grasp Dataset (UTG). O classificador e detector multiclases por CNN e R-CNN obteve acurácia de 96.85%

para UTG e 63.64% de acurácia para YHG. As preensões do YHG reconhecidas foram: preensão lateral, potência média, polegar e 3 dedos, pinça, pinça lateral de três pontos, polegar e 3 dedos, polegar e 4 dedos. As preensões reconhecidas para UTG foram: disco de precisão, extensão paralela, extensão tipo, esfera grande, potência média, pinça, pinça de 3 pontos, polegar e 3 dedos, polegar e 4 dedos, adução do polegar, extensão do indicador, pinça lateral, polegar e 2 dedos, três pontos lateral, diâmetro grande, anel, ferramenta leve.

3.2 PRÓTESES ACIONADAS POR CABOS

As próteses de membros superiores podem ser comandadas por sistema de tração de cabos ou elétricas, sendo o acionamento pneumático ou elétrico. Segundo Geenthanjali (2016) e Reaz (2016), as vantagens das próteses comandadas por cabos estão relacionadas ao baixo custo e facilidade de reparo, mas como desvantagens existem as questões estéticas e as dificuldades de operação e limitação de movimentos. Como exemplo de próteses de mãos de baixo custo, pode se indicar o programa Mão3D da Unifesp que tem como objetivo protetizar e reabilitar crianças e adultos, acometidos de malformação ou amputação de braços, mãos ou dedos, com próteses feitas por impressão 3D. A Figura 5 mostra uma prótese comandada por cabos feita em impressora 3D; os fios de nylon conectam o suporte da prótese aos dedos através do sistema de balanço e através do movimento do punho ou cotovelo os fios de nylon tracionam todos os dedos da prótese fazendo o movimento de preensão de potência. O retorno da prótese à posição de repouso é feito através de molas ou elásticos.

Figura 5- Prótese comandada por cabos feita em impressora 3D



Fonte: Thingiverse, 2019

Legenda: A figura mostra uma prótese de mão acionada pelo movimento do punho, que movimenta os dedos por meio da tração dos fios de nylon.

3.3 PRÓTESES CONTROLADAS POR sEMG

As próteses elétricas possuem melhor aparência, e quantidade de movimentos; entretanto são caras, pesadas e de difícil reparo. As próteses elétricas avançaram em tecnologia, podendo ser comandadas por pressão, sensores de deformação e eletromiografia (EMG). Em geral, próteses elétricas estão disponíveis em formato de ganchos, pinças, mãos ou algum dispositivo especial. Os materiais dos ganchos podem ser: alumínio; titânio; aço inox. As mãos artificiais apresentam aparência similar às mãos naturais e são geralmente controladas por sinais de sEMG, reagindo às intenções do usuário.

Segundo Khezri e Jahed (2007), uma possibilidade para classificação do sinal de sEMG é a utilização de uma lógica neuro *fuzzy* adaptativa, integrada com um sistema de aprendizado em tempo real. O sistema proposto foi utilizado para classificar 6 movimentos: mão aberta e fechada; pinça; flexão do polegar; flexão e extensão do punho. Foram utilizadas duas características dos sinais de sEMG, tempo e frequência, passando um PCA para redução de dimensionalidade. O sistema alcançou uma acurácia média de 96.67% para o correto reconhecimento dos padrões.

Algumas técnicas vêm sendo desenvolvidas para melhorar a estabilidade e acurácia dos classificadores dos sinais de sEMG para o reconhecimento de padrões. Segundo Chen et al. (2013) a inclusão de autocorreção para os classificadores LDA e também para o classificador

de análise de discriminante quadráticos (QDA) elevaram a acurácia de classificação, atingindo 94.43% para o LDA e 97.58% para o QDA. O sistema proposto por Chen et al. (2013) incluiu 10 classes de movimentos para mão e punho: punho pronado, punho supinado, mão fechada, mão aberta, flexão radial do punho, flexão ulnar do punho, extensão de punho, flexão de punho, dedos abertos, potência.

Redes neurais também vêm sendo estudadas como classificadores de sinais de EMG; segundo Pulliam et al. (2013) a utilização de uma rede neural com atraso pode ser usada para detectar e classificar a trajetória desejada de um braço. Em seu experimento foram utilizados sensores de EMG em 7 músculos, incluindo um sensor intramuscular no músculo braquial. O resultado obtido para este sistema de previsão de movimento de pronação e supinação do antebraço atingiu coeficiente de correlação $R^2 = 0.81$.

Buscando melhorar a acuracidade dos classificadores de sinais de sEMG, Amsüss et al. (2014) discutiram sobre a utilização de uma rede neural artificial, com parâmetros ajustáveis, para validar os dados vindos de um classificador LDA em função de sua probabilidade de acerto. A rede neural mantém o seu último estado classificado válido caso o valor proveniente do LDA seja rejeitado. O tempo para alteração de classe é de cerca de 250ms. O sistema de Amsüss et al. (2014) foi projetado para classificar 8 movimentos: pronação do antebraço, supinação do antebraço, extensão de punho, flexão de punho, mão aberta, mão fechada e nenhum movimento. Este sistema alcançou acurácia de 92.2%.

Com o objetivo de verificar a existência de um conjunto ótimo de gestos para cada tipo de amputação, Al-Timemy e Khushaba (2016) experimentaram 2 métodos de extração de características dos sinais de sEMG: Domínio do tempo no espectro de potência (TD-PSD), Domínio do tempo (TD) e 2 classificadores para os gestos: LDA e Random Forest (RF) para um grupo de 4 amputados. O experimento mostrou que os gestos de pronação, supinação e flexão do polegar apresentaram acurácia maior que 97.5% em todos os experimentos e para todos os 4 voluntários. A conclusão foi que para cada tipo de amputação é necessário um estudo detalhado do modelo de extração de características do sinal de sEMG e dos classificadores para a determinação do conjunto ótimo de gestos.

Geng et al. (2016) propuseram a utilização de uma matriz de 8 colunas e 16 linhas de sensores de sEMG para reconhecimento de gestos e posturas. Esta matriz de sensores sEMG pode ser tratada como uma imagem e a classificação de 8 gestos pode ser feita utilizando uma rede neural convolucional. Este projeto alcançou acurácia de 99.5% em 150 imagens de sinais instantâneos de sEMG. Os gestos reconhecidos foram: mão fechada, mão aberta, dedos afastados, indicador, dois dedos, três dedos, quatro dedos e polegar para cima.

Para a utilização dos sinais de sEMG, Geenthanjali (2016) descreve os modelos de controle:

- a) liga-desliga: é baseado no nível de amplitude do sinal de sEMG, comparado a um limiar, escolhendo a direção de controle da prótese. Este sistema de controle apresenta apenas 2 graus de liberdade e os movimentos são feitos com velocidade constante.
- b) controle proporcional: leva em consideração a variação da intensidade do sinal de sEMG, aplicando uma tensão proporcional ao motor da prótese.
- c) controle direto: utiliza controle similar ao proporcional, utilizando sinais do sEMG para controle independente dos dedos da prótese. Melhores resultados são obtidos com sensores de EMG intramusculares.
- d) controle de máquina de estado finito: para o sistema de controle por poses, as poses das mãos são definidas como estados e transições para o controle de estado de máquina. Este controle é aplicável para um número fixo de gestos.
- e) controle por reconhecimento de padrões: características, como amplitude, frequência e tempo são extraídas do sinal para classificação e controle da prótese. A transformada de Fourier é utilizada para extrair padrões do sinal de sEMG no domínio da frequência. Wavelets têm sido utilizadas para a decomposição do sinal no domínio da frequência e do tempo, bem como algoritmos genéticos, análise de componentes principais (PCA), análise de discriminante linear (LDA), lógica fuzzy e redes neurais (NN). A vantagem das redes neurais é a capacidade de aprender as relações lineares e não lineares dos dados dos sinais de sEMG.
- f) esquema de controle de postura: o sinal de sEMG é mapeado de acordo com as componentes principais. Este mapeamento é transformado em ângulos que representam a postura a ser atingida pela prótese.
- g) esquema de controle por regressão: a regressão linear controla simultaneamente vários ângulos e juntas da prótese, criando um modelo de variação.

Xavier et al. (2016) propuseram a fabricação de uma prótese de membro superior baseada no projeto Inmoov e controlada por 2 canais de sEMG. Os sinais de sEMG passaram por um processamento prévio para filtragem e obtenção do valor médio e servem como entrada da placa de controle arduino mega. Os movimentos da mão foram mimetizados através de uma luva com sensores resistivos e cada movimento foi armazenado em um cartão de memória. A

contagem dos pulsos dos sinais de sEMG foram relacionados com os movimentos previamente gravados na memória que comandam os movimentos da prótese.

Fajardo et al. (2017) propuseram uma prótese de baixo custo, *open source* e feita em impressora 3D. A prótese foi equipada com 6 motores controles internos à prótese e realiza 6 movimentos do polegar humano. A prótese é comandada por sEMG por um sistema finito de estados de máquina. Cinco (5) movimentos foram implementados: Potência; Gancho; Pinça; Lateral; Extensão do Indicador.

Na mesma linha da busca por melhorias da acuracidade dos classificadores de sinais de sEMG, Zhai et al. (2017) apresentaram a proposta de inclusão de uma rede neural de convolução autocalibrada para classificar os movimentos das mãos, utilizando um espectrograma de 12 canais de sensores sEMG de superfície como entrada. Foram considerados dez padrões de movimentos para o teste, como: ferramenta, preensão lateral, potência, mandril, pinça, flexão de punho, extensão de punho, pronação do antebraço, supinação do antebraço e mão aberta. O sistema proposto alcançou acurácia de 88.42% em testes com pessoas hígdas e 73.31% em testes com pessoas amputadas. O sistema utilizado para os testes foi uma GPU NVIDIA.

Li et al. (2018) propuseram um sistema de controle de força para a preensão pinça de três pontos em próteses. O método consiste na utilização do sensor *MYO armband* para coletar sinais de sEMG. O método de PCA foi aplicado ao sinal de sEMG para redução de dimensionalidade e uma rede neural profunda (DNN) de três camadas foi utilizada para treinamento e classificação da força a ser utilizada na preensão do objeto em 8 níveis, de 0 a 40N. Lógica Fuzzy foi utilizada para controlar a força na prótese. Um sinal vibratório foi introduzido como *feedback* no braço do usuário para indicar a força utilizada. O sistema alcançou acurácia de 95% na classificação da força a ser aplicada no objeto.

3.4 PRÓTESES CONTROLADAS POR SISTEMAS HÍBRIDOS

Um sistema de visão em conjunto com um sistema de medição de distância do objeto a ser manuseado foi proposto por Dosen et al. (2010). Com as informações da dimensão do objeto na imagem e da distância do objeto relativa à prótese, o sistema procura encontrar regras pré-definidas de seleção dos padrões de preensão para os objetos. O sistema de sEMG funciona como gatilho para abrir e fechar a prótese. Um total de 9 padrões de preensão de objetos foram conseguidos, sendo estes: potência de objetos grandes, potência de objetos médios, potência de objetos pequenos, preensão lateral de objetos grandes, preensão lateral de objetos médios,

preensão lateral de objetos pequenos, pinça de três pontos para objetos grandes, pinça de três pontos para objetos médios, pinça de três pontos para objetos pequenos. Este sistema alcançou uma acurácia média de 84%. O processamento foi feito em um computador pessoal equipado com um sistema de aquisição de dados NI-DAQ 6062E.

De Andrade et al. (2017) propuseram uma prótese com a utilização conjunta de sinais de sEMG, sinais de detecção de movimentos e a identificação de objetos com *Radio Frequency Identification* (RFID). Foram realizados 2 experimentos. No primeiro experimento, o sistema foi equipado com o sensor de sEMG e detector de movimento (IMU). O sinal de sEMG foi responsável por detectar as contrações musculares e o sinal do detector de movimento foi responsável por informar a orientação do braço. A combinação de número de contrações e orientações do braço determinavam o tipo de preensão do simulador virtual de prótese (v-rep). No segundo experimento, o sistema foi equipado com sEMG e sinal de RFID. Etiquetas codificadas por RFID foram colocadas nos objetos a serem manuseados e informavam ao sistema o padrão de preensão. A combinação do sinal de RFID e dos sinais de sEMG determinavam o momento da preensão, cancelamento do comando ou nova leitura do objeto. O sistema equipado com RFID é de fácil utilização, sendo a calibração e *feedback* fundamentais para sua utilização. Foram apontados como vantagem a facilidade de aprendizado, mas como desvantagem a utilização restrita de objetos equipados com RFID.

DeGol et al. (2017) propuseram a inclusão de um sistema de visão e classificação de objetos baseada em uma rede neural de convolução VGG-16 de 16 camadas em uma prótese controlada por sEMG. Foram utilizadas 1035 imagens de 280 objetos, retiradas do banco de imagens DeepGrasping, do banco ImageNet e de fotografias tiradas pelos autores para o treinamento da rede neural. A rede foi treinada para classificar objetos em 5 classes: potência, pinça, pinça de três pontos, ferramenta, preensão lateral. O sistema alcançou uma acurácia média de reconhecimento de 93.2%. O sistema utilizou uma GPU Tegra da NVIDIA para processamento das imagens.

Andrade et al. (2017) propuseram uma prótese com sistema de sEMG e visão incorporados. A imagem capturada por uma câmera embarcada é processada em um servidor externo através dos algoritmos de reconhecimento de imagem Tensorflow e Inception-v3 e o resultado, com a sugestão de preensão, retorna para a unidade de processamento local, um Raspberry Pi 3 (RPI3). A proposta é a substituição futura do RPI por um celular android. Com a sugestão do tipo de preensão, o usuário pode aceitar ou cancelar o resultado da preensão. Caso aceito, a preensão resultante vai para o sistema simulador V-rep, que conta com 14 possibilidades de padrões de preensões: mão relaxada, indicador ativo, ferramenta, polegar

abduzido, flexão do indicador, gancho, preensão lateral, utilização com mouse de computador, palma aberta, pinça, potência, abertura pinça de precisão, fechamento de pinça de precisão e pinça de três pontos. Os comandos do sEMG são: fechamento de mão, flexão de punho, extensão de punho. O usuário tem a opção de rejeitar o padrão fornecido, até que o controlador apresente a preensão desejada.

Outro estudo apresenta a inclusão de um sistema visual em próteses para melhorar o controle e a destreza dos usuários. Segundo Ghazaei et al. (2017), um sistema de visão e classificação de objetos baseada em uma rede neural de convolução de duas camadas foi adicionada a uma prótese controlada por dois canais de eletromiografia. A rede neural foi treinada para a classificação de objetos em quatro classes: pinça, pinça de três pontos, potência com punho neutro e potência com punho pronado. Para a verificação do sistema, o projeto foi testado por dois voluntários amputados. O sistema alcançou uma acurácia média de reconhecimento de classes de 88%. Este sistema é controlado pela medição de tempo dos sinais de extensão e flexão do músculo do punho, um sinal de 300ms de flexão do músculo do punho tira uma fotografia do objeto, enquanto um sinal de 3s de extensão do músculo do punho reinicia o sistema e abre a prótese. O sistema utilizou um computador pessoal para os testes.

Como prótese de mercado, pode-se mencionar a I-limb, da Touch Bionics (2019), que possibilita o controle dos dedos separadamente ou em conjunto. Esta prótese é comandada por meio de 2 métodos: aplicativo ou sEMG. A rotação do polegar acontece automaticamente, trocando os padrões de preensão lateral e opositor. O usuário pode escolher entre 24 tipos de preensões pré-definidos e mais 12 que ele próprio pode programar, sendo que destes, apenas 4 padrões de preensão (por vez) podem ser associados a padrões específicos de sinais de EMG. A prótese i-limb Quantum é equipada com o sistema de melhoria da força de preensão e de um sistema que evita escorregamento dos objetos.

3.5 COMPARAÇÃO ENTRE OS MÉTODOS

O Quadro 1 mostra o comparativo entre os modelos de controle de próteses apresentados na revisão bibliográfica, com relação aos recursos utilizados como sEMG, sistema de visão, sistema de medição, tipos de classificadores, quantidade de preensões e portabilidade do sistema.

Quadro 1 – Comparação entre os métodos

Projeto	Utilização	Metodologia	Recursos	Equipamentos	Quantidade de classes	Portabilidade	Acurácia média
Khezri e Jahed (2007)	Prótese	EMG	EMG + Fuzzy + PCA	CPU+ MatLab	6	NÃO	96.7%
Chen et al (2013)	Prótese	EMG	EMG + LDA + QDA	CPU	10	NÃO	97.6%
Pulliam et al (2013)	Prótese	EMG	EMG + RN	CPU	2	NÃO	81.0%
Amsüss et al (2014)	Prótese	EMG	EMG + RN + LDA	CPU	8	NÃO	92.2%
Geng et al (2016)	Prótese	Matriz de EMG	EMG + CNN	GPU	8	NÃO	99.5%
Zhai et al (2017)	Prótese	12 canais	EMG + PCA + CNN	GPU	10	NÃO	88.4%
Dosen at al (2010)	Prótese	Visão + medição	EMG+visão+medição	CPU+ MatLab	9	NÃO	84.0%
Ghazaei et al (2017)	Prótese	Visão + EMG	EMG + CNN	CPU	4	NÃO	88.0%
DeGol et al (2017)	Prótese	Visão + EMG	EMG + CNN	GPU	5	NÃO	93.2%
Andrade et al (2017)	Prótese	Visão + EMG	EMG+ CNN(sequencia)	RPI + SERVER	14	SIMULADOR	

Fonte: Autor

Legenda: Quadro comparativo dos trabalhos apresentados, métodos e resultados

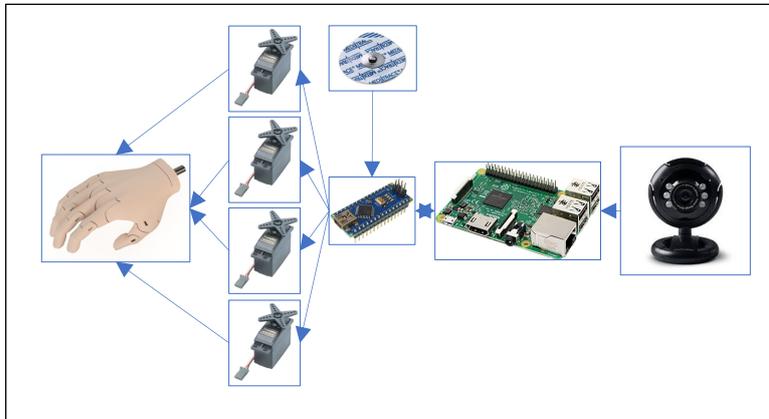
Neste capítulo foram apresentados os modelos de controle e comando das próteses, partindo de próteses comandada por cabos, próteses controladas por sistema de sEMG e próteses híbridas que são comandadas pela combinação de sEMG associados a sistemas de visão ou sistemas de etiquetas (RFID).

Apesar dos resultados promissores dos sistemas de reconhecimento de padrões de sEMG, com o objetivo de reconhecer diretamente a intenção de movimento do usuário, ainda não foram atingidos os níveis de exigência do mercado. A maioria dos trabalhos não chegou a 98% de acuracidade. Apenas Geng et al. (2016), com a utilização de um sistema de sEMG de alta densidade, superou 99%, porém sem apresentar algum padrão de preensão. Por esse motivo, a utilização do sEMG continua restrito a comandos simples, possibilitando a definição de apenas 4 padrões de movimento. Os sistemas híbridos que associam o sEMG a outros sistemas se apresentam como alternativa, possibilitando aumentar o número de padrões de preensão oferecidos e com isso maior praticidade e autonomia ao usuário. A maioria das propostas introduziram um sistema de visão, segundo o qual o padrão de preensão é definido a partir da imagem do objeto a ser pego. Contudo, o processamento das imagens foi feito em computador (CPU) e a melhor taxa de acerto foi de 93,2%, obtida por DeGol et al. (2017), para 5 classes: potência, pinça, pinça de três pontos, ferramenta e preensão lateral, a partir do uso de uma rede neural de convolução VGG-16 de 16 camadas utilizada. Por sua vez, a prótese comercial I-limb resolveu a limitação de padrões de preensão criando um aplicativo a partir do qual o usuário pode escolher o movimento que quer executar.

Dentro deste contexto, um novo modelo de prótese inteligente é proposto, comandado por sEMG e auxiliado por um sistema de visão computacional totalmente embarcado, conforme pode ser visto na Figura 6, oferecendo 5 modos de preensão: Potência Punho Neutro, Potência

Punho Pronado, Pinça de três pontos, Prensão Lateral e Extensão do Indicador, conforme visto na Figura 7. Este modelo inteligente facilita e agiliza o processo de aprendizado e utilização da prótese.

Figura 6 – Arquitetura funcional da prótese inteligente



Fonte: Autor

Legenda: Ilustração da arquitetura de controle e comando da prótese

Figura 7 - Representação das preensões da prótese inteligente



Fonte: Shutterstock.com, 2020

Legenda: Definições dos padrões de preensão da prótese inteligente

4 CONCEITOS FUNDAMENTAIS

São descritos neste capítulo uma síntese dos estudos sobre sinais de eletromiografia de superfície (sEMG), da rede neural VGG-16, da rede modificada de 5 camadas (SmallerVGG) e dos classificadores KNN e LDA, técnicas utilizadas no desenvolvimento deste trabalho. Os classificadores KNN e LDA serão utilizados como técnicas de comparação com a rede neural.

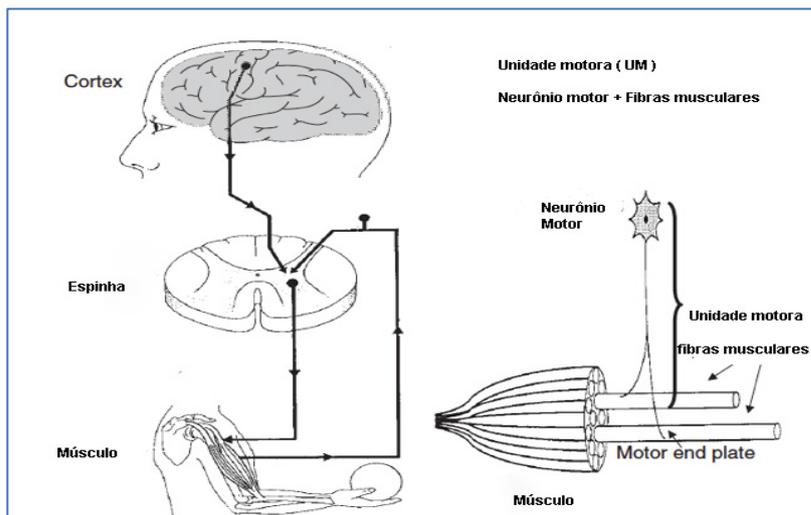
4.1 ELETROMIOGRAFIA E SISTEMA NERVOSO

Segundo Merletti e Parker (2004), o sistema nervoso central é organizado em hierarquia. A programação motora acontece no córtex pré motor, na área motora suplementar e em outras áreas do córtex. Sinais vindos destas áreas excitam ou inibem vários neurônios do córtex primário, que por sua vez apresentam forte influência nos interneurônios e neurônios motores da medula. Esta ligação promove controle direto das atividades musculares, conforme pode ser visualizado na Figura 8.

Uma unidade motora (MU) é composta por um único moto-neurônio alfa da medula e as fibras musculares por este inervadas. O número de fibras musculares inervadas por uma MU pode variar de 100 até 1000 ou mais, bem como pode variar sua capacidade de geração de força muscular.

Segundo Merletti e Parker (2004), durante as contrações voluntárias de músculos, a força é modulada pela combinação de unidades motoras e por mudanças na frequência de ativação muscular; quanto maior o número de unidades motoras e maior sua frequência de descarga elétrica, maior será a força muscular. Existe uma relação direta entre o sinal de eletromiografia e a força muscular.

Figura 8 - Representação básica do mecanismo de controle motor e unidade motora



Fonte: Merletti, 2004

Legenda: Fluxo dos sinais motores partindo do córtex até músculos.

De acordo com Konrad (2005) a excitação das fibras musculares através do controle neural representa o maior fator na fisiologia muscular. Este fenômeno pode ser explicado pelo modelo de membranas semipermeáveis descrevendo a propriedade elétrica do sarcolema. O equilíbrio iônico entre o interior e exterior das células do músculo em repouso na membrana muscular é aproximadamente de -80mV até -90mV . Esta diferença de potencial é mantida pelos processos fisiológicos, chamados de bombas de íons, que carregam negativamente o interior da célula, comparado ao meio externo. A ativação de um moto neurônio alfa, vindo do sistema nervoso central, resulta em uma excitação ao longo do nervo motor. Depois de liberar substâncias transmissoras, um potencial é formado nos músculos inervados por esta unidade motora. As características difusoras desta membrana de fibra muscular são rapidamente modificadas e íons Sódio Na^+ fluem para dentro. Isto causa a despolarização da membrana que é imediatamente restaurada pela saída de íons K^+ e ativa o mecanismo de bomba de íons, a repolarização. Se um certo nível é excedido com a entrada de íons de Sódio Na^+ , a despolarização gera um potencial de ação alterando de -80mV para $+30\text{mV}$. Isso é um impulso elétrico que é imediatamente restaurado pela fase de repolarização seguido por um período de hiperpolarização. Iniciando pela unidade motora, o potencial de ação se espalha pelas fibras musculares.

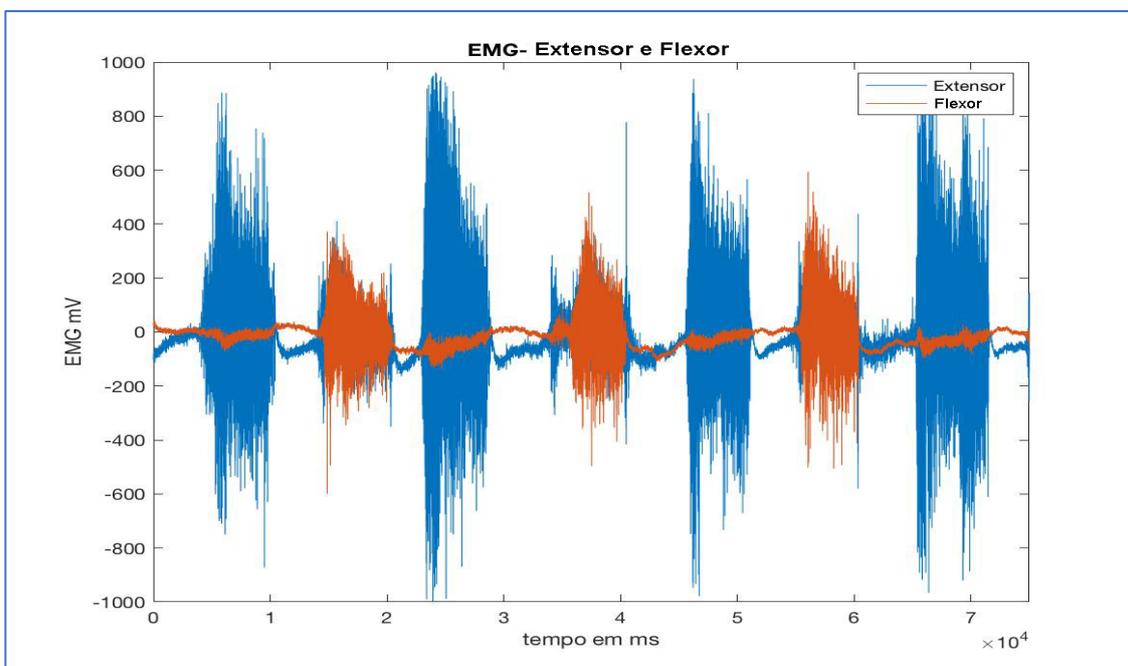
O sinal de sEMG é baseado nos ciclos de despolarização e repolarização criando uma onda que se propaga ao longo da superfície da fibra muscular. O que se vê no sinal de sEMG é

a resultante da superposição dos sinais de ativação das unidades motoras. Um sinal de sEMG não tratado fica dentro do range ± 5000 microvolts com frequência entre 6 e 500 Hz, mostrando uma potência maior de sinal entre 20 e 150 Hz. Alguns fatores podem influenciar o sinal de sEMG, como: características da pele, interferência entre sinais dos músculos, alterações na posição dos eletrodos, ruídos elétricos externos, características dos eletrodos e amplificadores do sinal. (KONRAD, 2005)

Segundo Merletti e Parker (2004), a interface entre pele e eletrodo é bem complexa, pois o sistema apresenta resistência e capacitância dependente da frequência do sinal, bem como a introdução de um diferencial de potencial formado entre o metal do eletrodo e o eletrólito. O eletrodo de prata / cloreto de prata (Ag / AgCl) é o mais utilizado por apresentar impedância resistiva no intervalo de frequência do sEMG, enquanto outros metais apresentam impedância capacitiva introduzindo um filtro adicional.

A Figura 9 mostra a relação de amplitudes e tempo dos sinais de sEMG coletados no braço do autor, para os movimentos dos músculos extensor radial longo do carpo e flexor ulnar do carpo. Após o pré-processamento (amplificação e filtragem analógica), filtros digitais de frequência, detectores de nível de tensão e contadores de pulso podem ser utilizados para o processamento dos sinais provenientes dos eletrodos de sEMG.

Figura 9 – Sinais de sEMG captados com 2 eletrodos, em modo diferencial simples.



Fonte: Autor

Legenda: Exemplo de sinal de eletromiografia de superfície e sua propagação temporal nos músculos extensor radial longo do carpo e flexor ulnar do carpo.

4.2 REDE NEURAL CONVOLUCIONAL

Segundo Araújo et al. (2017), as redes neurais convolucionais (CNNs) pertencem a uma categoria de algoritmos baseados em redes neurais artificiais que utilizam a convolução em pelo menos uma de suas camadas. Segundo Araújo et al. (2019) as CNNs provaram sua eficiência em tarefas de reconhecimento de imagens e vídeos, necessitando, porém, de grande quantidade de amostras rotuladas para o aprendizado. As CNNs são formadas por sequências de camadas e cada uma destas possui uma função específica na propagação do sinal de entrada. Suas principais camadas são: *Convolução* e *Pooling*. As camadas convolucionais são responsáveis por extrair atributos dos volumes de entrada, enquanto as camadas de *pooling* são responsáveis por reduzir a dimensionalidade do volume resultante. As demais camadas são totalmente conectadas propagando o sinal por meio da multiplicação ponto a ponto e do uso de uma função de ativação, sendo responsáveis pela classificação da imagem.

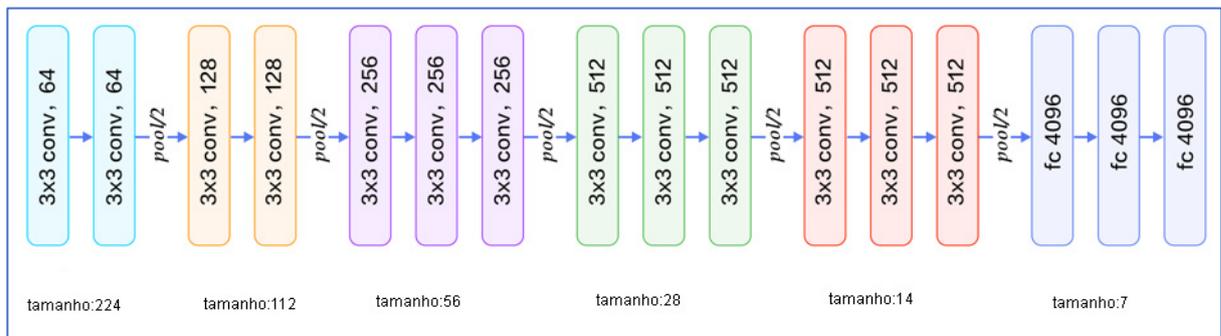
Keras (2019) é uma Interface de Programação de Aplicativos (API) de alto nível para redes neurais, escrita em Python e capaz de operar no topo de aplicações como TensorFlow ou Theano, focando em velocidade de experimentações.

TensorFlow (2019) é uma biblioteca de software de código aberto para computação numérica que usa gráficos de fluxo de dados. Os nodos no gráfico representam operações matemáticas e as arestas representam as matrizes ou tensores de dados multidimensionais que se comunicam com os nodos. A arquitetura flexível permite a implementação de aplicações de computação a uma ou mais Unidades Centrais de Processamento (CPUs) ou Unidades Gráficas de Processamento (GPUs) em um computador, servidor ou dispositivo móvel usando uma única API. O TensorFlow foi desenvolvido por pesquisadores e engenheiros da Google Brain Team, no departamento de pesquisas de inteligência de máquina do Google, com a finalidade de realizar pesquisas sobre redes neurais profundas e aprendizado de máquina. No entanto, devido à característica abrangente do sistema, esta biblioteca também pode ser aplicada a vários outros domínios.

Segundo Simonyan e Zisserman (2015), a rede Visual Geometry Group de Oxford (VGG), vista na Figura 10, têm por características módulos ou matrizes de convolução de tamanho 3x3, módulos de Max *Pooling* de tamanho 2x2 e camadas de saída totalmente conectadas. As matrizes de convolução 3x3 reduzem o custo computacional em comparação às matrizes de convolução 5x5 e 7x7. A função softmax normaliza entre 0 e 1 as saídas de um problema multiclasse, gerando um vetor de probabilidades para os possíveis resultados da classificação.

Os parâmetros dos filtros das camadas de convolução são adaptáveis. Conforme o filtro de convolução, matriz ou *kernel* percorre a imagem de entrada da camada na largura e altura, o valor computado do produto ponto a ponto gera um mapa com as características extraídas pelo filtro. As características destacadas pelos filtros podem ser: cores, bordas horizontais, bordas verticais, texturas, etc.

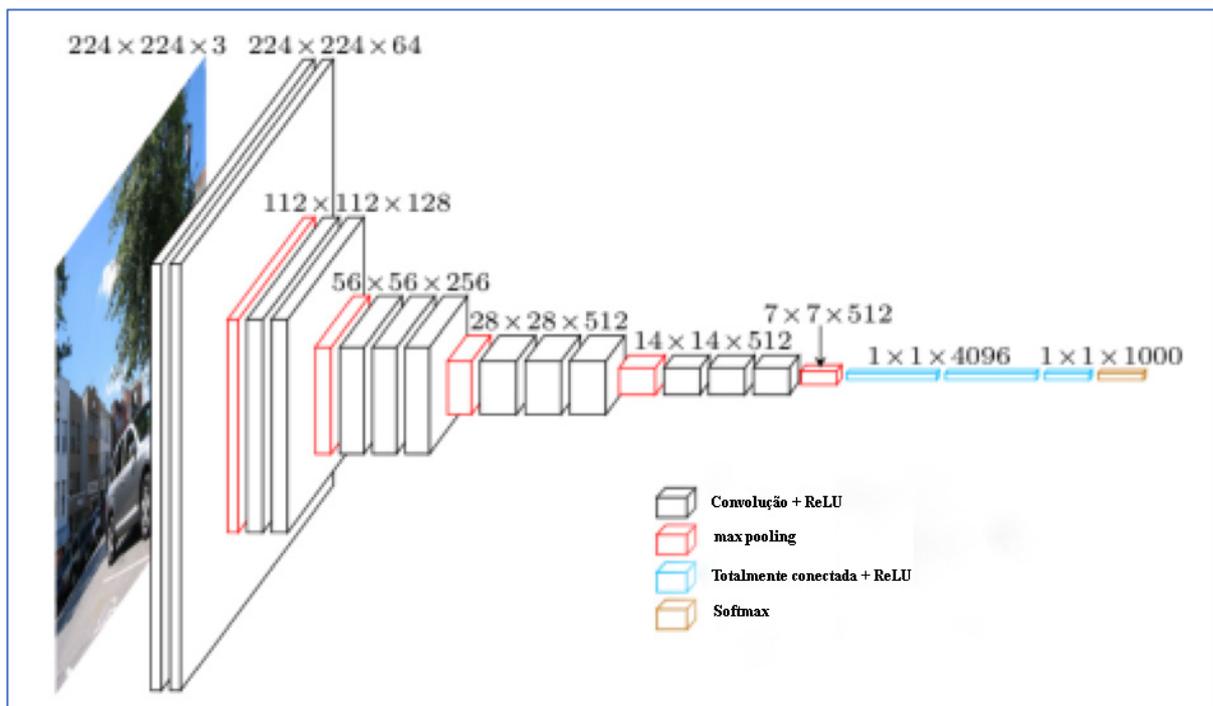
Figura 10 – Descrição das 16 camadas da rede VGG -16



Fonte: Autor “adaptado de” Quora, 2019

Legenda: Figura ilustra as etapas de convolução e *pooling* entre as diversas camadas da rede neural. A camada de convolução extrai características enquanto a camada *pooling* reduz a dimensionalidade da imagem.

Figura 11 – Exemplo da estrutura do tratamento de imagem da rede VGG16



Fonte: Autor “adaptado de” Quora, 2019

Legenda: A figura exemplifica a alteração da dimensionalidade e profundidade da imagem ao passar pelos filtros de convolução, *pooling*, camada totalmente conectada e softmax.

A Figura 11 mostra uma imagem de entrada da rede VGG16 de dimensões 224x224x3 e suas 16 camadas, mostrando a alteração da dimensionalidade e profundidade da imagem ao passar pelos filtros de convolução, *pooling*, camada totalmente conectada e *softmax*. As camadas são:

- a) convolução com 64 filtros;
- b) convolução com 64 filtros e max *pooling*;
- c) convolução com 128 filtros;
- d) convolução com 128 filtros e max *pooling*;
- e) convolução com 256 filtros;
- f) convolução com 256 filtros;
- g) convolução com 256 filtros e max *pooling*;
- h) convolução com 512 filtros;
- i) convolução com 512 filtros;
- j) convolução com 512 filtros e max *pooling*;
- k) convolução com 512 filtros;
- l) convolução com 512 filtros;
- m) convolução com 512 filtros e max *pooling*;
- n) rede totalmente conectada com 4096 neurônios;
- o) rede totalmente conectada com 4096 neurônios;
- p) saída com Softmax e 1000 neurônios.

4.2.1 Rede neural convolucional modificada

Segundo Rosebrock (2018), a rede modificada (SmallerVGG) é uma rede neural convolucional com apenas 5 funções de convolução. A imagem de entrada da rede, com dimensões 96 x 96 pixels x 3 canais, passa por um filtro de convolução ou matriz de 3 x 3, seguido por uma função linear retificada (*RELU*) de ativação e uma função de normalização (*BATCH NORMALIZATION*). A primeira camada *POOLING* da rede utiliza uma matriz 3 x 3 para redução da dimensionalidade da imagem de 96 x 96 pixels para 32 x 32 pixels. Nas camadas consecutivas da rede, a dimensão do filtro de convolução é alterado de 32 para 64, e por fim de 64 para 128. Em todas as camadas intermediárias é aplicada a função *DROPOUT* que desconecta 25% dos neurônios da camada para reduzir o *overfitting*. A camada final da rede é totalmente conectada através da função *DENSE* que utiliza uma função de ativação

retificada linear e em seguida passa por uma função *SOFTMAX* para retornar o valor da probabilidade da classificação de cada classe.

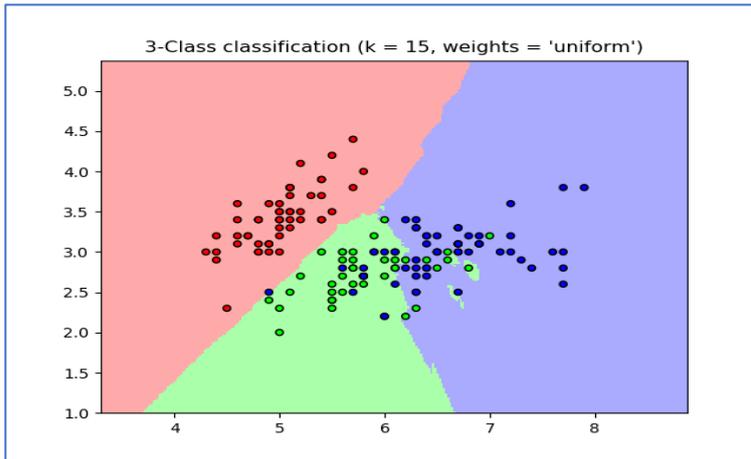
O algoritmo modificado, contendo as funções básicas dos filtros de convolução, *relu* e *pooling*, pode ser encontrado no ANEXO A.

4.3 CLASSIFICADOR KNN

Segundo Russell e Norvig (2013), o classificador KNN, chamado de k-vizinhos mais próximos, é um sistema de aprendizagem supervisionada a partir do treinamento das classes. A expressão “mais próximos” implica em uma métrica de distância. Busca-se características semelhantes em K elementos mais próximos do ponto de pesquisa. Para evitar empates utiliza-se k como um número ímpar. Comumente, atribui-se a distância euclidiana ou a distância de Manhattan. Caso os valores de atributos sejam booleanos, o número de atributos em que dois pontos diferem é chamado de distância de Hamming. A dimensão numérica dos atributos define a escala da distância, parâmetros com dimensões diferentes afetam a medida da distância e neste caso, para eliminar o erro utiliza-se a normalização dos dados. O classificador KNN funciona muito bem para espaços de dimensão baixa com abundância de dados, mas à medida que o número de dimensão cresce, os vizinhos mais próximos deixam de estar muito próximos. Este problema é conhecido como a maldição da dimensionalidade.

De acordo com Scikit-learn KNN (2019), o princípio do método dos K vizinhos mais próximos é encontrar em um número pré-definido de amostras de treinamento, um novo ponto mais próximo e predizer sua classe usando uma métrica matemática como a distância euclidiana. O ponto será atribuído à classe que estiver à menor distância. Um exemplo gráfico do classificador KNN pode ser visto na Figura 12.

Figura 12 – Exemplo de classificador KNN



Fonte: SCIKIT-LEARN, 2019

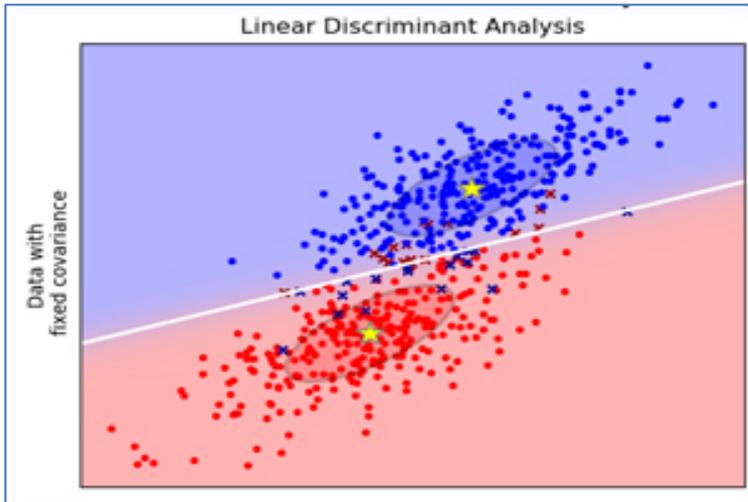
Legenda: Gráfico mostra um classificador de 3 classes knn com k=15

4.4 CLASSIFICADOR LDA

Segundo Fisher (1936, apud XAVIER et al. 2012), a proposta do classificador de análise linear discriminante (LDA), também conhecido como método de Fisher, é encontrar o hiperplano de maior separação entre amostras. O hiperplano leva em consideração o treinamento das classes. O método baseia-se na redução da dispersão das amostras em relação a classe pertencente e na maximização da distância média entre as classes. O principal objetivo do método LDA é encontrar a matriz de projeção que maximiza a razão entre o determinante da matriz de dispersão inter-classes e o determinante da matriz de espalhamento intra-classes.

Um exemplo gráfico do classificador LDA pode ser visto na Figura 13, onde um hiperplano separa 2 classes definidas.

Figura 13 – Exemplo de classificador LDA



Fonte: SCIKIT-LEARN, 2019

Legenda: Gráfico mostra um hiperplano separador de 2 classes

5 PRÓTESE DE MÃO INTELIGENTE

A prótese foi construída em uma impressora 3D em material poliláctico (PLA) e é composta por: um sistema de visão computadorizado responsável por captar a imagem do objeto que o usuário deseja pegar, uma rede neural convolucional para classificar este objeto de acordo com 5 padrões de prensão, um sistema de sEMG para detectar a intenção do usuário em pegar o objeto, e uma placa de desenvolvimento Arduino Nano que comanda o fechamento e abertura da prótese por meio de servo motores, conforme visto na Figura 6.

O modelo de prótese de mão utilizado foi idealizado por Buchanan (2018), conforme Figura 14. O processo de impressão pode ser visualizado na Figura 15. O projeto Kwawu Arm foi encontrado nos sites enablingthefuture.org e thingiverse.com, sob licença da Creative Commons Attribution 4.0 International Licence. (<https://creativecommons.org/licenses/by/4.0/>)

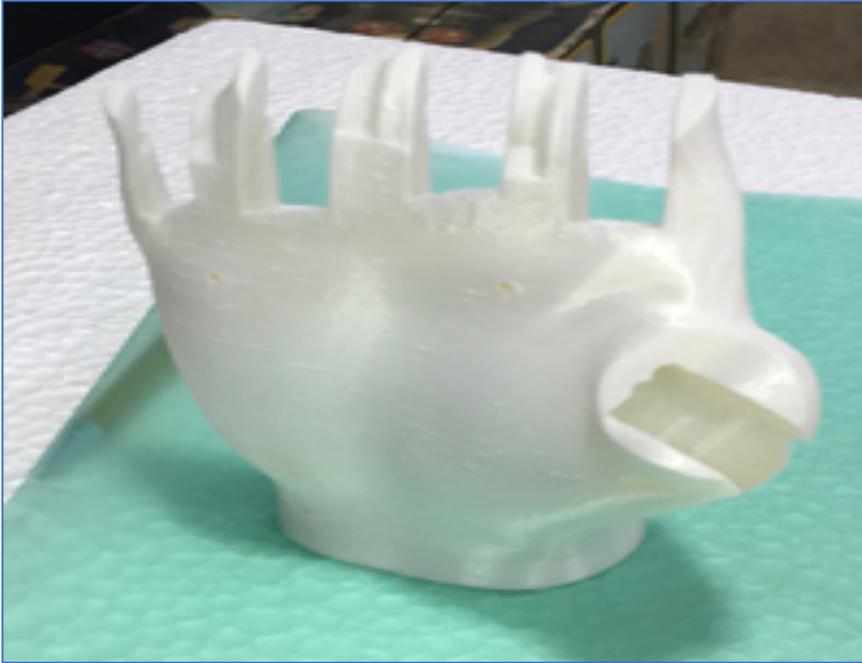
Figura 14 – Modelo da prótese de mão Kwawu 2



Fonte: Thingiverse, 2019

Legenda: Imagem da prótese de mão Kwawu 2

Figura 15– Processo de impressão da prótese Kwawu 2



Fonte: Autor

Legenda: Processo de impressão em PLA da prótese de mão Kwawu 2

A prótese foi modificada para a instalação de uma câmera USB, modelo APP-TECH de 16 Mega pixels e de um indicador laser na área palmar, conforme Figura 16. O laser tem a função de indicar ao usuário o objeto que se deseja manusear. Foram instalados LEDs no dorso da prótese com a finalidade de informar ao usuário qual a classe proposta pelo classificador neural.

Figura 16 – Câmera USB e laser instalados na área palmar da prótese



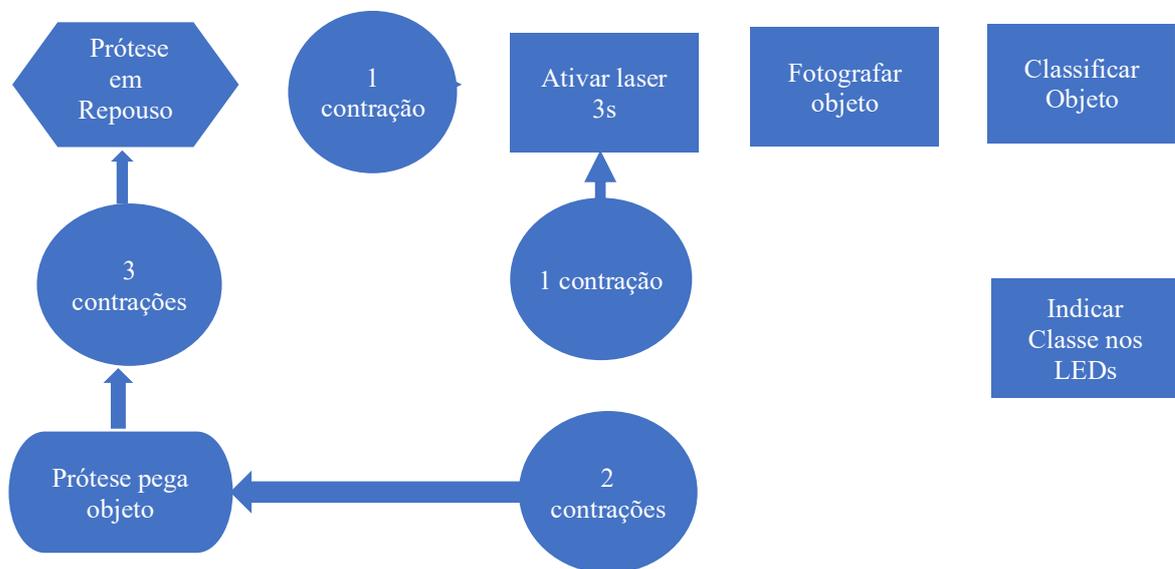
Fonte: Autor

Legenda: Imagem do protótipo de mão proposto

5.1 ALGORITMO DE OPERAÇÃO DA PRÓTESE

Partindo da prótese em repouso, o sistema de controle, ao detectar o sinal de 1 contração, ativa o sistema de mira a laser por 3 segundos. Decorrido o tempo de 3 segundos para o usuário identificar visualmente o objeto que deseja manusear, o sistema, após 500ms, fotografa o objeto e inicia a classificação pela rede neural de 5 camadas. O padrão escolhido pela rede neural é apresentado nos indicadores LEDs no dorso da prótese. O usuário tem a opção de, com 1 contração reiniciar o processo de classificação ou com 2 contrações comandar a prótese para segurar o objeto a ser manuseado. Nesta condição, o usuário comanda a prótese para liberar o objeto com 3 contrações, retornando a prótese para a condição inicial. O diagrama de controle pode ser visto na Figura 17.

Figura 17 – Algoritmo de comando da prótese



Fonte: Autor

Legenda: Fluxograma de decisão do protótipo da prótese de mão

5.2 SISTEMA COMPUTADORIZADO

Com foco em baixo custo e simplicidade, foi utilizado um microcomputador Raspberry Pi (RPI3) visto na Figura 18, operando com o sistema Raspbian.

Para a plataforma de visão computacional foi utilizado o software Open CV e para o programa de gerenciamento foi utilizado o software Python. Para a interface com os mecanismos da prótese, o Arduino Nano foi utilizado pelo baixo custo e praticidade de

programação em seu ambiente de programação IDE básico. Todos os softwares utilizados são livres para utilização não comercial.

Figura 18– Microprocessador Raspberry Pi 3 (RPI3)



Fonte: Raspberry Pi, 2019

Legenda: A ilustração exemplifica a placa Raspberry Pi 3

5.3 INTERFACE PARA ELETROMIOGRAFIA DE SUPERFÍCIE

Foi utilizada a placa de captura de sinais Myoware, *Muscle Sensor* AT-04-001 que pode ser vista na Figura 19. Foram utilizados os sensores descartáveis de sEMG da empresa Meditrace, conforme visto na Figura 20. A placa possui um ajuste de ganho para o sinal do sEMG, possibilitando customizar para cada usuário.

O sinal de saída SIG da placa Myoware, foi conectado à entrada analógica do arduino para detecção e tratamento do sinal. O algoritmo do arduino compara o sinal de entrada analógico com um valor de limiar ajustável por software e caso o valor adquirido seja maior que este limiar, o algoritmo incrementa um contador e inicializa um temporizador. O contador é novamente incrementado pela presença de outro pulso vindo da entrada analógica e continua com o algoritmo até o final do temporizador utilizando o valor do contador. A placa arduino pode ser vista conforme Figura 21.

Os comandos de movimento da prótese e do sistema de visão são efetuados considerando a contagem dos pulsos vindos dos sensores de sEMG, cada contração muscular gera um pulso. O valor do contador de pulsos retorna a zero após 2 segundos. O programa construído pelo autor pode ser visto no APÊNDICE B.

Figura 19– Sensor Muscular Myoware



Fonte: Myoware, 2019

Legenda: Ilustração da placa de sensor muscular Myoware

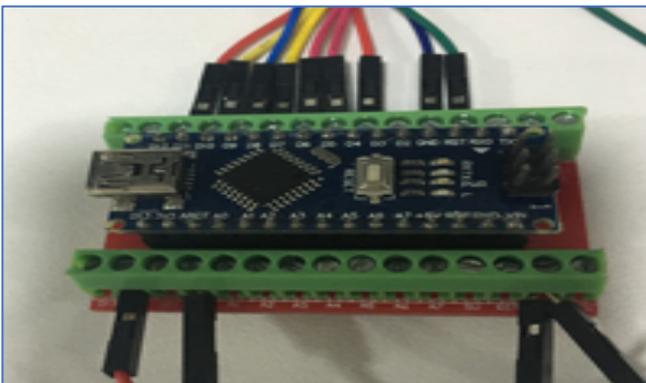
Figura 20 – Sensor para medição do sEMG



Fonte: Covidien, 2019

Legenda: Ilustração do sensor de sEMG

Figura 21 – Arduino Nano



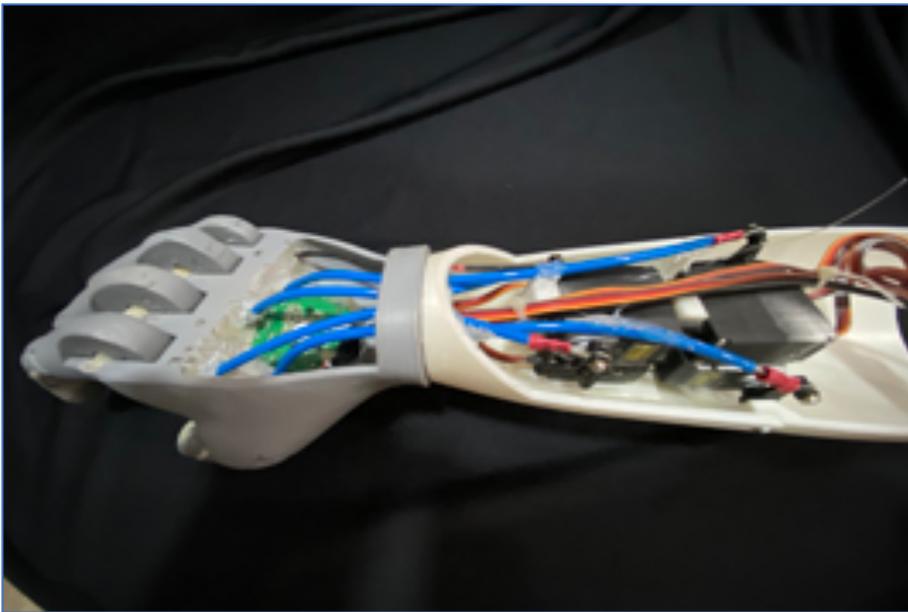
Fonte: Autor

Legenda: Ilustração da placa de controle arduino nano

5.4 SISTEMA DE ACIONAMENTO DA PRÓTESE

Foram utilizados 5 servos motores Tower Pro MG995, com engrenagem metálica, conectados às saídas digitais e comandados eletronicamente pelo Arduino. Um dos motores é responsável pela movimentação do dedo anelar e mínimo. Pode ser visto na Figura 22 os servos conectados aos dedos por meio de cabos de aço. Para a suave movimentação dos cabos de aço, guias de PVC foram adicionadas.

Figura 22 – Prótese com Servos e cabos de aço.



Fonte: Autor

Legenda: Ilustração do sistema de comando com cabos da prótese

5.5 ALIMENTAÇÃO ELÉTRICA DO SISTEMA

Uma bateria de polímeros de lítio (Li-Po) com tensão de 11.1V e capacidade de 2200mAh foi utilizada; um exemplo pode ser visto na Figura 23. Um regulador abaixador de tensão para 5V-3A foi utilizado para alimentação de todo sistema de controle e comando.

Figura 23 – Exemplo de Bateria de 11,1 V com capacidade de 2200mAh



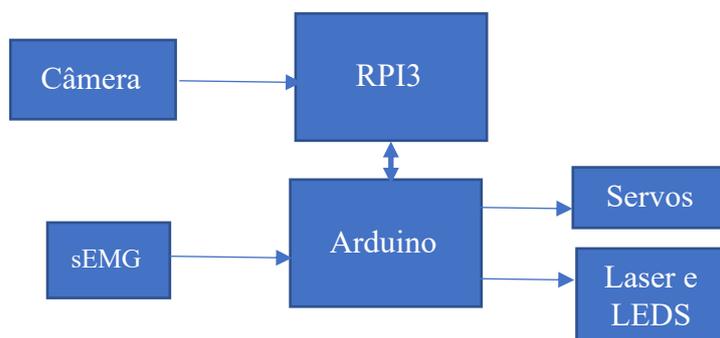
Fonte: Autor

Legenda: Ilustração da bateria a ser utilizada para portabilidade da prótese

5.6 DIAGRAMA GERAL DO SISTEMA

A figura 24 mostra o diagrama básico de interligação dos elementos de entrada e saída do protótipo da prótese de mão.

Figura 24– Diagrama geral do sistema



Fonte: Autor

Legenda: Diagrama básico de interligação dos elementos do sistema de controle e atuadores da prótese

O diagrama elétrico de ligação entre o Arduino e o Raspberry PI 3 pode ser encontrado no ANEXO B.

6 MATERIAIS E MÉTODOS

Neste capítulo são descritos os equipamentos, métodos, sistemas utilizados no desenvolvimento dos experimentos de classificação de imagens, uma descrição do protótipo de mão desenvolvido e algoritmo de funcionamento do protótipo. Todos os experimentos visam validar o classificador de imagens com rede neural convolucional de 5 camadas proposto, o SmallerVGG, definido no item 4.2.1 deste trabalho.

Todos os seis (6) experimentos, nas fases de treinamento e validação, foram conduzidos em um computador Mac mini 2012, nas mesmas condições, e o arquivo gerado pelo treinamento da rede neural foi transferido para o sistema Raspberry Pi para utilização nos testes práticos.

O algoritmo da rede neural converte as imagens para tons de cinza e as redimensiona para 96x96 pixels. A utilização das imagens para o treinamento acontece de forma aleatória e sem repetição. Dois vetores foram criados para a entrada no algoritmo da rede neural; o primeiro formado com os valores da intensidade dos pixels da imagem (96x96), que efetivamente passa pelas camadas da rede neural e outro com os nomes das classes.

Nos experimentos com rede neural, todos utilizaram a rede modificada SmallerVGG.

Como condição de contorno dos projetos, tem-se para o experimento 1, 2 e 3, imagens com o fundo branco e nível de iluminação natural do ambiente, para os experimentos 4, 5 e 6, tem-se imagens com fundo preto e nível de iluminação normal do ambiente.

Para todos os experimentos, apenas 1 objeto por vez foi apresentado aos classificadores. A apresentação de vários objetos ao mesmo tempo e segmentação de imagem não fizeram parte do escopo deste estudo.

Para a tabela contendo os valores de sensibilidade, especificidade e acurácia, utilizamos as seguintes definições, conforme Makhtar et al. (2011).

- a) a sensibilidade (Se) de um teste é a razão entre os verdadeiros positivos (TP) que são identificados corretamente e a soma de verdadeiros positivos e falsos negativos (FN); neste caso, a probabilidade de o teste classificar corretamente o padrão de prensão para manipular o objeto da imagem capturada;

$$Se = TP / (TP + FN)$$

b) a especificidade (Sp) de um teste, por outro lado, é a razão entre verdadeiros negativos (TN) identificados corretamente e a soma dos falsos positivos (FP) e verdadeiros negativos; sugere como o teste é bom para identificar as outras classes;

$$Sp = TN / (TN + FP)$$

c) a acurácia (ACC) é a razão entre a soma dos resultados verdadeiros positivos e verdadeiros negativos com o total de amostra; este mede o grau de veracidade de um teste em determinada condição.

$$ACC = (TP + TN) / (TP + FP + FN + TN)$$

A acurácia média (ACCmed) é a soma das acurácias dividido pelo número de classes.

$$ACCmed = \text{soma (Acci)} / ni$$

6.1 METODOLOGIA DOS EXPERIMENTOS

Para o experimento 1 a rede neural de cinco camadas foi treinada com um banco de imagens contendo 836 imagens para o padrão de classificação pinça, 858 imagens para o padrão de classificação potência e 823 imagens para o padrão de classificação pinça de três pontos. Imagens com fundo branco foram utilizadas para treinamento, validação e testes práticos com objetos. Parâmetros de treinamento da SmallerVGG: *Epochs* = 100, Erro = 0.001, tamanho de lote = 32, 80% das imagens para treinamento e 20% para validação cruzada. Treinamento realizado em 13 horas. Os testes foram feitos com 266 imagens não presentes no banco anterior, sendo 84 para o padrão pinça, 98 para o padrão potência e 84 para pinça de três pontos.

Para o experimento 2 o classificador por vizinhos mais próximo (KNN) e o classificador por análise linear discriminante (LDA), foram treinados com o mesmo banco de imagens do experimento 1, Para este experimento não foram utilizadas características discriminantes dos objetos, obedecendo a mesma estrutura de dados da entrada das imagens na rede neural. Parâmetros de treinamento dos classificadores LDA e KNN: 75% das imagens para treinamento e 25% para validação cruzada. Treinamento realizado em 2 minutos. Para o classificador KNN foi utilizado o número K=5 como parâmetro. Para o classificador LDA foi utilizado a

configuração padrão do módulo scikit-learn LDA. Os testes foram feitos com as mesmas imagens utilizadas nos testes do experimento 1.

Para o experimento 3 o classificador por vizinho mais próximo (KNN) e classificador por análise linear discriminante (LDA) foram treinados com um banco de imagens contendo 7 imagens para o padrão de classificação pinça, 4 imagens para o padrão de classificação potência e 7 imagens para o padrão de classificação pinça de três pontos. Imagens com fundo branco utilizadas para treinamento, validação e testes. Para este experimento foram definidas características discriminantes dos objetos. Foi escolhido, pelo autor, apenas 2 parâmetros: área em pixels, e relação comprimento e largura em pixels. Parâmetros de treinamento dos classificadores LDA e KNN: 75% das imagens para treinamento e 25% para validação cruzada. Treinamento realizado em 2 minutos. Para o classificador KNN foi utilizado o número $K=5$ como parâmetro. Para o classificador LDA foi utilizada a configuração padrão do módulo scikit-learn LDA. Já para os testes foram utilizadas 114 imagens, sendo 84 para o padrão pinça, 12 para o padrão potência e 18 para pinça de três pontos. Este experimento foi realizado para comprovar o aumento da acurácia, especificidade e sensibilidade quando selecionado parâmetros discriminantes para a entrada dos classificadores KNN e LDA, sendo necessário e suficiente um banco de imagens reduzido.

Para o experimento 4 a rede neural de cinco camadas foi treinada com um banco de imagens original da Newcastle Grasp library (NGL) acrescido do banco de imagens original da Amsterdam library of object images (ALOI), sendo 7.633 imagens para o padrão de classificação pinça, 11.811 imagens para o padrão de classificação pinça de três pontos, 8.778 imagens para o padrão de classificação potência com posição neutra de punho (neutro) e 11.305 imagens para o padrão de classificação potência com rotação de punho (pronado). Imagens com fundo preto utilizadas para treinamento, validação e testes práticos com objetos. Parâmetros de treinamento da SmallerVGG: *Epochs* = 100, Erro = 0.001, tamanho de lote = 32, 80% das imagens para treinamento e 20% para validação cruzada. Treinamento realizado em 4 dias. Os testes foram realizados com 182 imagens, não presentes no banco de treinamento, sendo 42 para o padrão pinça, 42 para o padrão pinça de três pontos, 42 para potência neutro e 56 para potência pronado. Este experimento tem o objetivo de comparar o resultado da classificação da rede SmallerVGG com o experimento de Ghazaei et al (2017).

Para o experimento 5 a rede neural de cinco camadas foi treinada com um banco de imagens modificado da NGL e da ALOI, sendo 6.901 imagens para o padrão de classificação pinça de três pontos, 8.778 imagens para o padrão de classificação potência com posição de punho neutro e 15.552 imagens para o padrão de classificação potência com posição de punho

pronado, 2.189 imagens para classificação de preensão lateral. Imagens com fundo preto utilizadas para treinamento, validação e testes práticos com objetos utilizando a rede neural de 5 camadas. Parâmetros de treinamento da SmallerVGG: *Epochs* = 100, Erro = 0.001, tamanho de lote = 32, 80% das imagens para treinamento e 20% para validação cruzada. Treinamento realizado em 4 dias. Foi incluído o movimento de preensão lateral por sua relevância, conforme Feix et al. (2013). Os testes foram realizados com 182 imagens, sendo 70 para o padrão pinça de três pontos, 42 para potência com posição de punho neutro, 56 para potência com posição de punho pronado e 14 para preensão lateral.

Para o experimento 6 a rede neural de cinco camadas foi treinada com um banco de imagens modificado da NGL e ALOI, sendo 6.901 imagens para o padrão de classificação pinça de três pontos, 8.778 imagens para o padrão de classificação potência com posição de punho neutro e 15.552 imagens para o padrão de classificação potência com posição de punho pronado, 2.189 imagens para classificação de preensão lateral, 3.211 imagens para classificação de teclado de computador, 2.809 imagens para o classificador de teclado musical e 2.337 imagens para o classificador de tablets. Imagens com fundo preto utilizadas para treinamento, validação e testes práticos com objetos utilizando a rede neural de 5 camadas. Parâmetros de treinamento da SmallerVGG: *Epochs* = 60, Erro = 0.001, tamanho de lote = 32, 80% das imagens para treinamento e 20% para validação cruzada. Treinamento realizado em 3 dias. As classificações provenientes de teclado de computador, teclado musical e tablet resultam no movimento de extensão do indicador. Os testes foram realizados com 182 imagens, sendo 70 para o padrão pinça de três pontos, 42 para potência com posição de punho neutro, 42 para potência com posição de punho pronado, 14 para preensão lateral e 14 para extensão do indicador.

Resumidamente, os experimentos 1, 2 e 3 visam comparar a acurácia de classificação da rede neural SmallerVGG com os classificadores KNN e LDA com e sem a utilização de características discriminantes, na classificação de objetos em 3 classes; pinça, pinça de três pontos e potência. Enquanto os experimentos 4 e 5 visam comparar a acurácia de classificação da rede neural SmallerVGG com o classificador proposto por Ghazaei et al. (2017), utilizando o banco de imagens eles proposto e posteriormente adaptado pelo autor. E finalmente, o experimento 6 consolida o classificador SmallerVGG em um classificador visual de 7 classes para acionar 4 movimentos do protótipo da prótese de mão proposto.

6.2 BANCO DE IMAGENS PARA TREINAMENTO E OBJETOS DE TESTE

Todas as imagens utilizadas para treinamento dos experimentos podem ser encontradas em: <https://www.dropbox.com/sh/flzrme07g4bphu9/AAB4ghJSO3ay18uFFqJEkhkea?dl=0>

Até o presente momento, os artigos acadêmicos pesquisados, relativos aos padrões de preensão de objetos, estão relacionados a preensões de garras robóticas ou estudos da mecânica da mão humana. Por este motivo, a escolha das classes de objetos para os classificadores utilizados nos experimentos 1, 2 e 3 foi baseada no julgamento do autor e validado por meio de pesquisa com voluntários (item 6.3). Para a escolha das classes de preensão do banco de dados, cada objeto foi manuseado pelo autor por meio dos três tipos de preensão e escolhido o que melhor apresentou preensão do objeto.

Para o banco de imagens dos experimentos 1, 2 e 3 foram utilizadas imagens disponibilizadas pela Universidade de Cornell, extraídas do endereço: http://pr.cs.cornell.edu/grasping/rect_data/data.php em conjunto com imagens coletadas pelo autor. As imagens coletadas pelo autor foram capturas com fundo branco, webcam posicionada à 20 cm sobre os objetos e à 50 cm de distância do objeto, como pode ser visto no apêndice D. Como indicativo, o nível de iluminação de 130 Lux foi encontrado no ambiente durante a aquisição das fotos para o treinamento. Para aumentar o tamanho do banco de imagens, os objetos foram fotografados mais de uma vez em diversos ângulos e posições. Foram utilizadas 836 imagens para o padrão de classificação pinça, 858 imagens para o padrão de classificação potência e 823 imagens para o padrão de classificação pinça de três pontos.

Da Universidade de Cornell (2009) foram utilizadas as imagens dos bancos de imagem: data01, data02, data03, data04, data05, data06, data07, data08, data09, data10. Cada imagem foi analisada utilizando o mesmo critério do autor referente ao tipo de preensão utilizado no uso diário. Nem todas as imagens foram utilizadas, apenas as imagens de objetos semelhantes às imagens obtidas pelo autor.

O terceiro experimento tem o intuito de avaliar os classificadores utilizando características discriminantes como entrada para o algoritmo. Foram utilizadas sete imagens para a classe pinça, quatro imagens para a classe potência e sete imagens para a classe pinça de três pontos retiradas do banco de imagens do experimento 1. Para a classe pinça foram selecionados: LEDs, transistores, parafusos, capacitores. Para a classe potência foram selecionados: chave de fenda, controle remoto, caixa, desodorante. Para a classe pinça de três pontos foram selecionados: tampa de caneta, escova de dentes, caneta.

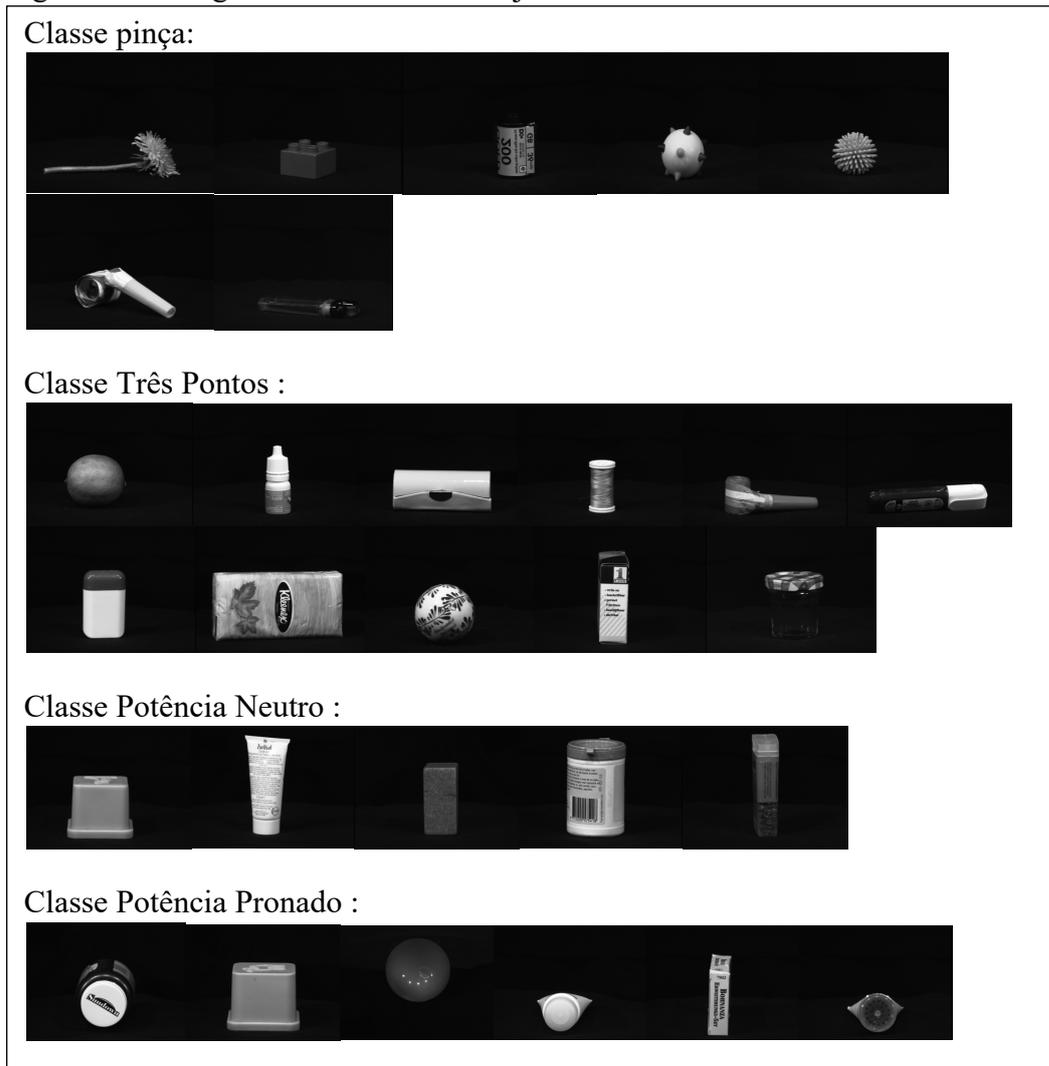
Para determinar as características discriminantes, foi utilizada a função python e open CV, caixa de contorno do objeto (`cv2.ContourArea`), retornando com a informação, em pixels, do lado maior e lado menor do objeto detectado na imagem. Com estas informações, duas características discriminantes foram calculadas como entrada dos classificadores: a área em pixels da caixa de contorno, a razão lado maior pelo lado menor.

Para o experimento 4 foi utilizado o banco de imagens original da Newcastle Grasp library (NGL) acrescido do banco de imagens original da Amsterdam library of object images (ALOI), sendo 7.633 imagens para o padrão de classificação pinça, 11.811 imagens para o padrão de classificação pinça de três pontos, 8.778 imagens para o padrão de classificação potência com posição neutra de punho (neutro) e 11.305 imagens para o padrão de classificação potência com posição rotacionada de punho (pronado). Todas as imagens deste experimento apresentam fundo preto.

A referência encontrada no material suplementar de Ghazaei et al. (2017) em stacks.iop.org/JNE/14/036025/mmedia foi utilizada para a classificação do banco de imagens em quatro classes: pinça, potência com punho neutro, potência com punho pronado e pinça de três pontos.

Em função dos resultados obtidos no experimento 4, foram feitas alterações no banco de imagens para o experimento 5, eliminando imagens de formatos semelhantes encontrados em mais de uma classe e acrescentando a classe prensão lateral. Exemplos destas imagens semelhantes podem ser vistos na Figura 25. Esse banco de imagens modificado da NGL e da ALOI, contém 6.901 imagens para o padrão de classificação pinça de três pontos, 8.778 imagens para o padrão de classificação potência com posição de punho neutro e 15.552 imagens para o padrão de classificação potência com posição de punho pronado, 2.189 imagens para classificação de prensão lateral. Todas as imagens deste experimento apresentam fundo preto.

Figura 25 – Imagens semelhantes de objetos nas 4 classes.



Fonte: Autor

Legenda: Imagens semelhantes encontradas em mais de uma classe do banco de imagens ALOI

Para o experimento 6 foi utilizado o banco de imagens modificado da NGL e ALOI, sendo 6.901 imagens para o padrão de classificação pinça de três pontos , 8.778 imagens para o padrão de classificação potência com posição de punho neutro e 15.552 imagens para o padrão de classificação potência com posição de punho pronado, 2.189 imagens para classificação de prensão lateral, 3.211 imagens para classificação de teclado de computador, 2.809 imagens para o classificador de teclado musical e 2.337 imagens para o classificador de tabletes. Todas as imagens deste experimento apresentam fundo preto.

Para a comparação dos experimentos 1, 2 e 3, foi utilizado um conjunto de objetos semelhantes ao testado por Ghazaei et al. (2017) para a fase de testes. Este conjunto é formado por 24 objetos, acrescidos de: 14 imagens de teclados de computador; 7 imagens de pilhas; 7 imagens de capacitores; 7 imagens de porcas; 7 imagens de botões eletrônicos; 7 imagens de

A matriz de confusão para o Resultado 1 pode ser visto na Tabela 1 e a matriz de confusão para o Resultado 2 pode ser visto na Tabela 2. A matriz de confusão mostra o resultado prático do experimento em relação ao resultado teórico esperado.

O resultado da pesquisa indicando o tipo de prensão mais utilizado, bem como o tipo de prensão esperado pelo autor, pode ser visto no Quadro 2.

Tabela 1 – Matriz de confusão para o Resultado 1

		Resultado 1			
		Pinça	Potência	Três pontos	Total
Esperado	Pinça	4	0	1	5
	Potência	0	18	0	18
	Três pontos	1	2	13	16

Fonte: Autor

Tabela 2 – Matriz de confusão para o Resultado 2

		Resultado 2			
		Pinça	Potência	Três pontos	Total
Esperado	Pinça	5	0	0	5
	Potência	0	16	2	18
	Três pontos	4	0	12	16

Fonte: Autor

Quadro 2 - Resultado da pesquisa de preensões com 14 voluntários

Objetos	Autor	Resultado 1	Resultado 2
Controle Remoto TV	potência	potência	potência
Controle remoto Game	potência	potência	potência
Celular	potência	potência	potência
Óculos	três pontos	três pontos	pinça
Caneta	três pontos	três pontos	pinça
Lápis	três pontos	três pontos	pinça
Escova de dentes	três pontos	três pontos	três pontos
Pasta de dentes	três pontos	três pontos	três pontos
Bola ping pong	três pontos	três pontos	três pontos
Garfo	três pontos	três pontos	três pontos
Colher	três pontos	três pontos	três pontos
Faca	três pontos	três pontos	três pontos
Desodorante	potência	potência	potência
Cartão memória	pinça	pinça	pinça
Lanterna	potência	potência	três pontos
Pano	potência	potência	potência
Tinta spray	potência	potência	potência
Vassoura de mesa	potência	potência	potência
Mouse	potência	potência	potência
Capacitor grande	potência	potência	potência
Caixa de alarme	potência	potência	potência
Tinta 100mL	três pontos	três pontos	três pontos
Fonte 5V	potência	potência	três pontos
Bateria Lithium	três pontos	três pontos	três pontos
Chave de fenda pequena	três pontos	potência	três pontos
Fita crepe	três pontos	três pontos	três pontos
Fita isolante	Três pontos	pinça	pinça
Isqueiro deitado	três pontos	três pontos	três pontos
Alicate de bico	potência	potência	potência
Alicate universal	potência	potência	potência
Grampeador	potência	potência	potência
Tesoura média	três pontos	potência	três pontos
Cópo de água	potência	potência	potência
Secador de cabelos	potência	potência	potência
Parafusadeira	potência	potência	potência
Pafafuso pequeno	pinça	pinça	pinça
Porca	pinça	pinça	pinça
Parafuco com porca	pinça	três pontos	pinça
Carvilha	pinça	pinça	pinça

Fonte: Autor

Legenda: Resultado da pesquisa indicando os tipos de preensão com maior ocorrência

É importante mencionar que, durante a primeira seção de testes, alguns voluntários utilizaram um padrão de preensão diferente, preparando e manipulando os objetos para utilização. Não se pode dizer que existem padrões de preensões certos ou errados para cada objeto, mas as preensões utilizadas diferentes de preensão pinça, pinça de três pontos e potência foram desconsiderados para este experimento por terem apresentado baixíssima frequência de ocorrência.

Por meio da análise das matrizes de confusão, foi adotado o padrão de preensão sugerido pelo autor, para os experimentos 1, 2 e 3.

7 EXPERIMENTOS E RESULTADOS

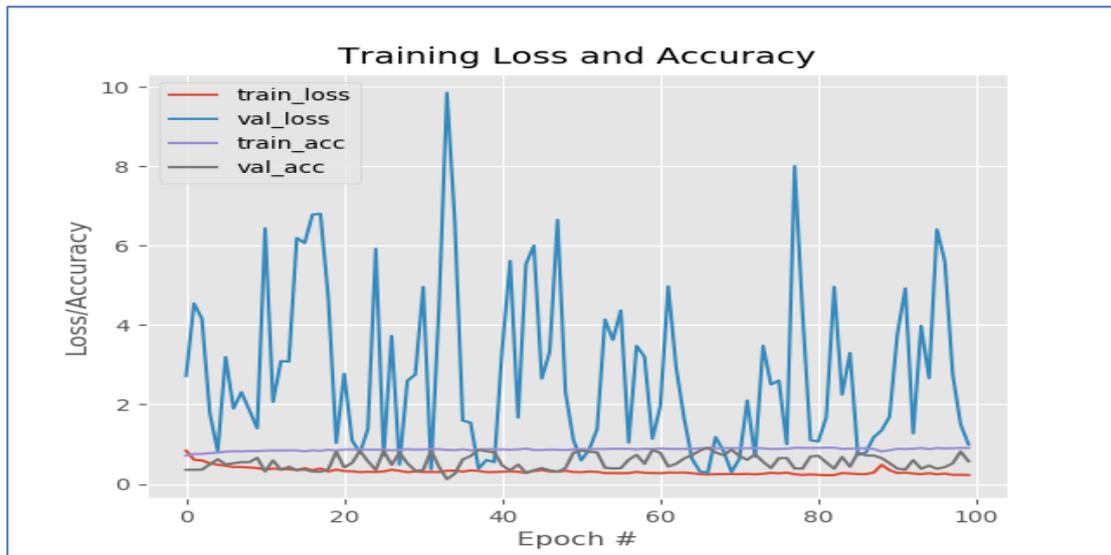
Por razões de desempenho computacional, todos os experimentos foram realizados em um computador Mac mini 2012, operando com os mesmos algoritmos instalados no microcomputador RPI3. O computador Mac mini 2012 apresentou condições de trabalhar com os 6 experimentos ao mesmo tempo, logo, cada imagem proveniente da câmera pode ser utilizada ao mesmo tempo nos 6 experimentos.

7.1 EXPERIMENTO 1

A rede neural de cinco camadas foi treinada com um banco de imagens contendo 836 imagens para o padrão de classificação pinça, 858 imagens para o padrão de classificação potência e 823 imagens para o padrão de classificação pinça de três pontos. Imagens com fundo branco utilizadas para treinamento, validação e testes práticos com objetos. Parâmetros de treinamento da SmallerVGG: *Epochs* = 100, Erro = 0.001, tamanho de lote = 32, 80% das imagens para treinamento e 20% para validação cruzada. Treinamento realizado em 13 horas. Os testes foram feitos com 266 imagens não presentes no banco anterior, sendo 84 para o padrão pinça, 98 para o padrão potência e 84 para pinça de três pontos.

O gráfico da Figura 29 é formado por 4 valores: acurácia de treinamento, perda no treinamento; acurácia da validação; perda na validação. A escala do eixo das ordenadas foi construída para atender aos valores obtidos nos erros durante a validação, enquanto os valores de acurácia variam entre 0 a 1, ou 0 a 100%. Observa-se, na Figura 29, grande variação dos valores de erro durante a validação cruzada da rede neural (*val_loss*).

Figura 29 – Gráfico de treinamento e validação do primeiro experimento.



Fonte: Autor

Legenda: Gráfico com os valores acurácia e de perda do treinamento da rede neural

A Tabela 3 mostra a matriz de confusão para testes com objetos no experimento 1.

Tabela 3 – Matriz de confusão para o experimento 1

		Resultado REDE 1			
		Pinça	Potência	Três pontos	Total
Esperado	Pinça	76	0	8	84
	Potência	0	80	18	98
	Três pontos	0	1	83	84
Total		76	81	109	266

Fonte: Autor

A Tabela 4 mostra os valores de sensibilidade, especificidade e acurácia para o experimento 1.

Tabela 4 –Sensibilidade, especificidade e acurácia do experimento 1

	Resultado REDE 1		
	Sensibilidade	Especificidade	Acurácia
Pinça	0.90	1.00	0.97
Potência	0.82	0.99	0.93
Três pontos	0.99	0.86	0.90
Total REDE 1	0.90	0.95	0.93

Fonte: Autor

Mesmo com um elevado erro durante o treinamento da rede neural, ocasionado pela quantidade de objetos em cada classe, o experimento 1 obteve 93% de acurácia.

7.2 EXPERIMENTO 2

O classificador por vizinho mais próximo (KNN) e o classificador por análise linear discriminante (LDA) foram treinados com um banco de imagens igual ao do experimento 1. Para este experimento não foram utilizadas características discriminantes dos objetos, obedecendo a mesma estrutura de dados da entrada das imagens na rede neural. Parâmetros de treinamento dos classificadores LDA e KNN: 75% das imagens para treinamento e 25% para validação cruzada. Treinamento realizado em 2 minutos. Para o classificador KNN foi utilizado o número $K=5$ como parâmetro. Para o classificador LDA foi utilizado a configuração padrão do módulo scikit-learn LDA. O banco de imagens utilizado para testes também foi o mesmo utilizado no experimento 1.

Através da função `modelxxx.score(treinamento, teste)` foi obtido para a validação:

- a) 100% de precisão para o classificador KNN;
- b) 88.5% de precisão para o classificador LDA.

A Tabela 5 mostra a matriz de confusão para testes com objetos no experimento 2, com classificador KNN.

Tabela 5 – Matriz de confusão para o experimento 2 com KNN.

		Resultado KNN			
		Pinça	Potência	Três pontos	Total
Esperado	Pinça	70	0	14	84
	Potência	0	43	55	98
	Três pontos	0	17	67	84
Total		70	60	136	266

Fonte: Autor

A Tabela 6 mostra os valores de sensibilidade, especificidade e acurácia para o experimento 2, com classificador KNN.

Tabela 6 – Sensibilidade, especificidade e acurácia com KNN

	Resultado KNN		
	Sensibilidade	Especificidade	Acurácia
Pinça	1.00	1.00	0.95
Potência	0.72	0.90	0.73
Três pontos	0.49	0.66	0.75
Total KNN	0.68	0.84	0.81

Fonte: Autor

A Tabela 7 mostra a matriz de confusão para testes com objetos no experimento 2, com classificador LDA.

Tabela 7 – Matriz de confusão para o experimento 2, com classificador LDA

		Resultado LDA			
		Pinça	Potência	Três pontos	Total
Esperado	Pinça	0	84	0	84
	Potência	0	50	48	98
	Três pontos	0	69	15	84
Total		0	203	63	266

Fonte: Autor

A Tabela 8 mostra os valores de sensibilidade, especificidade e acurácia para o experimento 2, com classificador LDA.

Tabela 8 –Sensibilidade, especificidade e acurácia com LDA

	Resultado LDA		
	Sensibilidade	Especificidade	Acurácia
Pinça	0.00	1.00	0.68
Potência	0.25	0.09	0.24
Três pontos	0.24	0.74	0.56
Total LDA	0.24	0.62	0.50

Fonte: Autor

O experimento 2 demonstrou que os classificadores KNN e LDA tem sua acurácia afetada pela falta de características discriminantes, obtendo uma acurácia de 81% e 50% respectivamente.

7.3 EXPERIMENTO 3

O classificador por vizinho mais próximo (KNN) e o classificador por análise das componentes discriminantes (LDA) foram treinados com um banco de imagens contendo 7 imagens para o padrão de classificação pinça, 4 imagens para o padrão de classificação potência e 7 imagens para o padrão de classificação pinça de três pontos. Imagens com fundo branco utilizadas para treinamento, validação e testes. Para este experimento foram definidas características discriminantes dos objetos. Foi escolhido, pelo autor, apenas 2 parâmetros: área em pixels, e relação comprimento e largura em pixels. Parâmetros de treinamento dos classificadores LDA e KNN: 75% das imagens para treinamento e 25% para validação cruzada. Treinamento realizado em 2 minutos. Para o classificador KNN foi utilizado o número $K=5$ como parâmetro. Para o classificador LDA foi utilizado a configuração padrão do módulo scikit-learn LDA. Já para os testes foram utilizadas 114 imagens, sendo 84 para o padrão pinça, 12 para o padrão potência e 18 para pinça de três pontos. Este experimento foi realizado para comprovar o aumento da acurácia, especificidade e sensibilidade quando selecionado parâmetros discriminantes para a entrada dos classificadores KNN e LDA, sendo necessário e suficiente um banco de imagens reduzido.

A Tabela 9 mostra a matriz de confusão para testes com objetos no experimento 3, com classificador KNN.

Tabela 9 – Matriz de confusão para o experimento 3 com KNN.

		Resultado KNN Discriminante			
		Pinça	Potência	Três pontos	Total
Esperado	Pinça	83	0	1	84
	Potência	0	9	3	12
	Três pontos	1	0	17	18
Total		84	9	21	114

Fonte: Autor

A Tabela 10 mostra os valores de sensibilidade, especificidade e acurácia para o experimento 3, com classificador KNN.

Tabela 10 – Sensibilidade, especificidade e acurácia com KNN

	Resultado KNN Discriminante		
	Sensibilidade	Especificidade	Acurácia
Pinça	0.99	0.97	0.98
Potência	1.00	1.00	0.97
Três Pontos	0.81	0.96	0.96
Total KNN	0.96	0.98	0.97

Fonte: Autor

A Tabela 11 mostra a matriz de confusão para o experimento 3, com classificador LDA.

Tabela 11 – Matriz de confusão para o experimento 3 com LDA.

		Resultado LDA Discriminante			
		Pinça	Potência	Três pontos	Total
Esperado	Pinça	74	0	10	84
	Potência	0	9	3	12
	Três pontos	3	0	15	18
	Total	77	9	28	114

Fonte: Autor

A Tabela 12 mostra os valores de sensibilidade, especificidade e acurácia para o experimento 3, com classificador LDA.

Tabela 12 – Sensibilidade, especificidade e acurácia com LDA

	Resultado LDA Discriminante		
	Sensibilidade	Especificidade	Acurácia
Pinça	0.88	0.90	0.88
Potência	0.75	1.00	0.96
Três Pontos	0.83	0.86	0.85
Total LDA	0.86	0.93	0.90

Fonte: Autor

O experimento 3 confirma o aumento da acurácia dos classificadores KNN e LDA ao se utilizar características discriminantes, obtendo 97% e 90% de acurácia respectivamente.

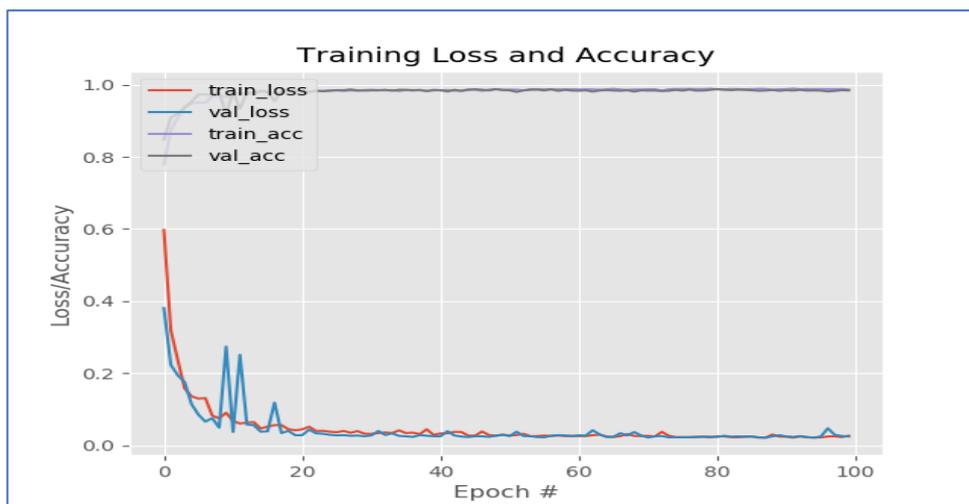
7.4 EXPERIMENTO 4

A rede neural de cinco camadas foi treinada com um banco de imagens original da Newcastle Grasp library (NGL) acrescido do banco de imagens original da Amsterdam library of object images (ALOI), sendo 7.633 imagens para o padrão de classificação pinça, 11.811 imagens para o padrão de classificação pinça de três pontos, 8.778 imagens para o padrão de classificação potência com posição neutra de punho (neutro) e 11.305 imagens para o padrão de classificação potência com posição rotacionada de punho (pronado). Imagens com fundo preto utilizadas para treinamento, validação e testes práticos. Parâmetros de treinamento da

SmallerVGG: *Epochs* = 100, Erro = 0.001, tamanho de lote = 32, 80% das imagens para treinamento e 20% para validação cruzada. Treinamento realizado em 4 dias. Os testes foram realizados com 182 imagens, não presentes no banco de treinamento, sendo 42 para o padrão pinça, 42 para o padrão pinça de três pontos, 42 para potência neutro e 56 para potência pronado.

Observa-se, na Figura 30, uma rápida convergência para o treinamento da rede neural, mesmo apresentado variação no valor de erro da validação cruzada (*val_loss*), sendo que a escala no eixo das ordenadas atende tanto aos valores obtidos para os erros no treinamento e validação, quanto aos valores de acurácia que variam de 0 a 1, ou de 0 a 100%.

Figura 30 – Gráfico de treinamento e validação do quarto experimento.



Fonte: Autor

Legenda: Gráfico com os valores acuracidade e de perda do treinamento da rede neural

A Tabela 13 mostra a matriz de confusão para testes com objetos no experimento 4.

Tabela 13 – Matriz de confusão para o experimento 4

		Resultado Newcastle				Total
		Pinça	Três pontos	P.Neutro	P.Pronado	
Esperado	Pinça	41	0	0	1	42
	Três pontos	41	1	0	0	42
	P.Neutro	0	0	31	11	42
	P.Pronado	3	2	0	51	56
Total		85	3	31	63	182

Fonte: Autor

A Tabela 14 mostra os valores de sensibilidade, especificidade e acurácia para o experimento 4.

Tabela 14 – Sensibilidade, especificidade e acurácia do experimento 4

	Resultado Newcastle		
	Sensibilidade	Especificidade	Acurácia
Pinça	0.98	0.69	0.75
Três pontos	0.02	0.99	0.76
P.Neutro	0.74	1.00	0.94
P.Pronado	0.91	0.90	0.91
Total NC	0.68	0.89	0.84

Fonte: Autor

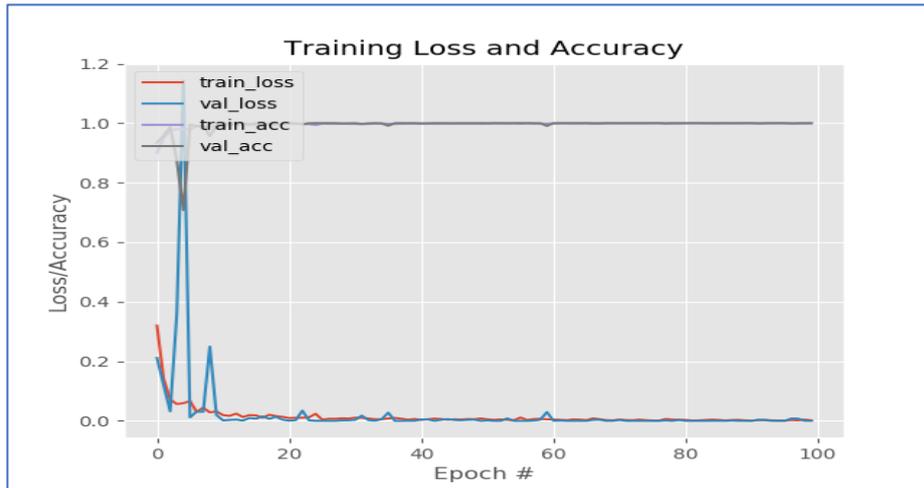
No experimento 4, utilizando o banco de imagens de Ghazaei et al. (2017) para treinamento, a acurácia ficou em 84%. Isto se deve ao fato de o banco de imagens utilizado para o treinamento da rede neural conter objetos semelhantes em diferentes classes.

7.5 EXPERIMENTO 5

A rede neural de cinco camadas foi treinada com um banco de imagens modificado da NGL e da ALOI, sendo 6.901 imagens para o padrão de classificação pinça de três pontos, 8.778 imagens para o padrão de classificação potência com posição de punho neutro e 15.552 imagens para o padrão de classificação potência com posição de punho pronado, 2.189 imagens para classificação de preensão lateral. Imagens com fundo preto utilizadas para treinamento, validação e testes práticos com a rede neural de 5 camadas. Parâmetros de treinamento da SmallerVGG: *Epochs* = 100, Erro = 0.001, tamanho de lote = 32, 80% das imagens para treinamento e 20% para validação cruzada. Treinamento realizado em 4 dias. Os testes foram realizados com 182 imagens, sendo 70 para o padrão pinça de três pontos, 42 para potência neutro e 56 para potência pronado e 14 para preensão lateral.

Observa-se, na Figura 31, uma rápida convergência para o treinamento da rede neural, mesmo apresentado variação no valor de erro da validação cruzada (*val_loss*),

Figura 31 – Gráfico de treinamento e validação do quinto experimento.



Fonte: Autor

Legenda: Gráfico com os valores acuracidade e de perda do treinamento da rede neural

A Tabela 15 mostra a matriz de confusão para testes com objetos no experimento 5.

Tabela 15 – Matriz de confusão para o experimento 5

		Resultado Newcastle Modificada				
		Três Pontos	P.Neutro	P.Pronado	Lateral	Total
Esperado	Três Pontos	68	0	0	2	70
	P.Neutro	0	41	1	0	42
	P.Pronado	0	0	54	2	56
	Lateral	3	0	0	11	14
Total		71	41	55	15	182

Fonte: Autor

A Tabela 16 mostra os valores de sensibilidade, especificidade e acurácia para o experimento 5.

Tabela 16 – Sensibilidade, especificidade e acurácia do experimento 5

	Resultado Newcastle Modificada		
	Sensibilidade	Especificidade	Acurácia
Três Pontos	0.97	0.97	0.97
P.Neutro	0.98	1.00	0.99
P.Pronado	0.96	0.99	0.98
Lateral	0.79	0.98	0.96
Total NC	0.96	0.99	0.98

Fonte: Autor

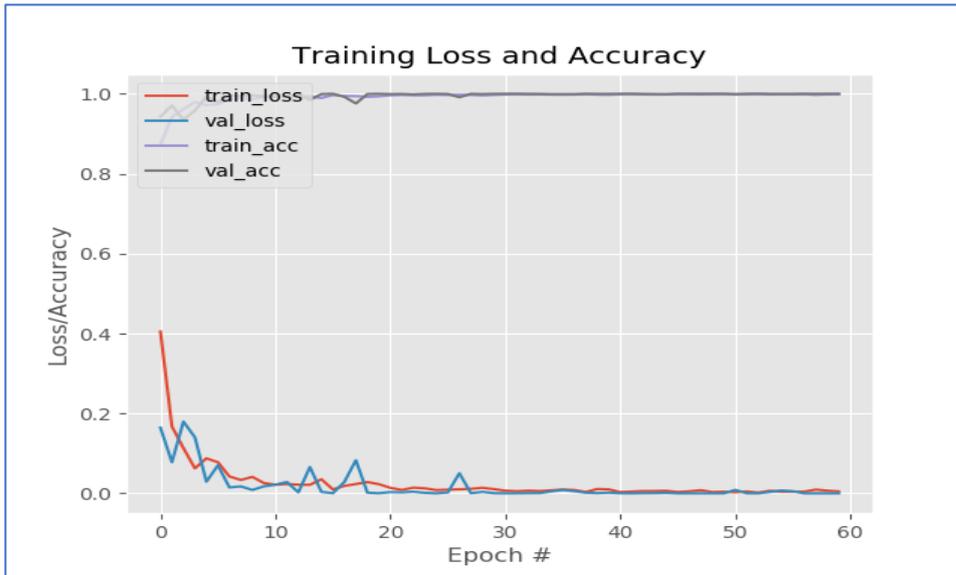
O experimento 5 apresentou uma acurácia de 98%. Neste experimento foram eliminadas as imagens de objetos semelhantes no banco de treinamento e foi acrescentado a classe de preensão lateral.

7.6 EXPERIMENTO 6

A rede neural de cinco camadas foi treinada com um banco de imagens modificado da NGL e ALOI, sendo 6.901 imagens para o padrão de classificação pinça de três pontos, 8.778 imagens para o padrão de classificação potência com posição de punho neutro e 15.552 imagens para o padrão de classificação potência com posição de punho pronado, 2.189 imagens para classificação de preensão lateral, 3.211 imagens para classificação de teclado de computador, 2.809 imagens para o classificador de teclado musical e 2.337 imagens para o classificador de tablets. Imagens com fundo preto utilizadas para treinamento, validação e testes práticos com a rede neural de 5 camada. Parâmetros de treinamento da SmallerVGG: *Epochs* = 60, Erro = 0.001, tamanho de lote = 32, 80% das imagens para treinamento e 20% para validação cruzada. Treinamento realizado em 3 dias. Para os testes foram utilizadas 182 imagens, sendo 70 para o padrão pinça de três pontos, 42 para potência neutro e 42 para potência pronado, 14 para preensão lateral e 14 para extensão do indicador.

Observa-se, na Figura 32, uma rápida convergência para o treinamento da rede neural e uma pequena variação no valor de erro da validação cruzada (*val_loss*),

Figura 32 – Gráfico de treinamento e validação do sexto experimento.



Fonte: Autor

Legenda: Gráfico com os valores acuracidade e de perda do treinamento da rede neural

A Tabela 17 mostra a matriz de confusão para testes com objetos no experimento 6.

Tabela 17 – Matriz de confusão para o experimento 6

		Resultado PROPOSTA					Total
		Três Pontos	P.Neutro	P.Pronado	Lateral	Indicador	
Esperado	Três Pontos	68	0	0	2	0	70
	P.Neutro	0	42	0	0	0	42
	P.Pronado	0	0	41	0	1	42
	Lateral	1	0	0	12	1	14
	Indicador	0	0	0	0	14	14
Total		69	42	41	14	16	182

Fonte: Autor

A Tabela 18 mostra os valores de sensibilidade, especificidade e acurácia para o experimento 6.

Tabela 18 – Sensibilidade, especificidade e acurácia do experimento 6

	Resultado PROPOSTA		
	Sensibilidade	Especificidade	Acurácia
Três Pontos	0.97	0.99	0.98
P.Neutro	1.00	1.00	1.00
P.Pronado	0.98	1.00	0.99
Lateral	0.86	0.99	0.98
Indicador	1.00	0.99	0.99
Total Proposta	0.97	0.99	0.99

Fonte: Autor

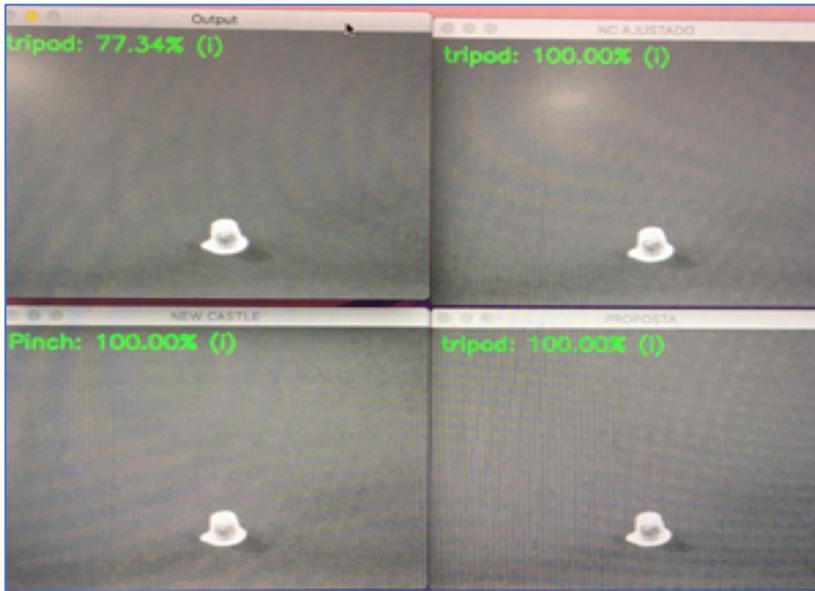
A proposta de classificador neural, no experimento 6, obteve acurácia de 99%, incluindo as classes teclado, piano e tablet para o movimento do indicador.

7.7 VISUALIZAÇÃO DOS RESULTADOS DOS TESTES.

Nos experimentos, a rede neural foi treinada com bancos diferentes o que justifica as diferenças encontradas nos exemplos das classificações.

A Figura 33 mostra os resultados dos experimentos 1, 4, 5 e 6 tendo como entrada da rede a imagem de uma cápsula de café.

Figura 33 – Exemplo de objeto para classe pinça de três pontos

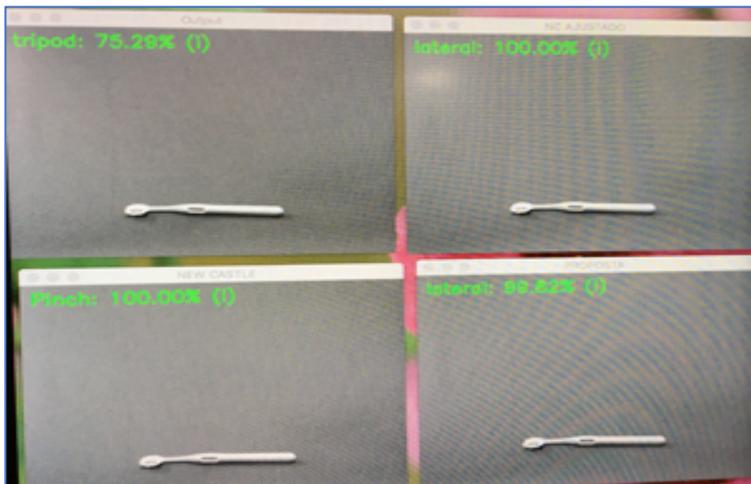


Fonte: Autor

Legenda: Experimento 1, Experimento 5
Experimento 4, Experimento 6

A Figura 34 mostra o resultado dos experimentos 1, 4, 5 e 6 tendo como entrada da rede a imagem de uma escova de dentes.

Figura 34 – Exemplo de objeto para classe prensão lateral

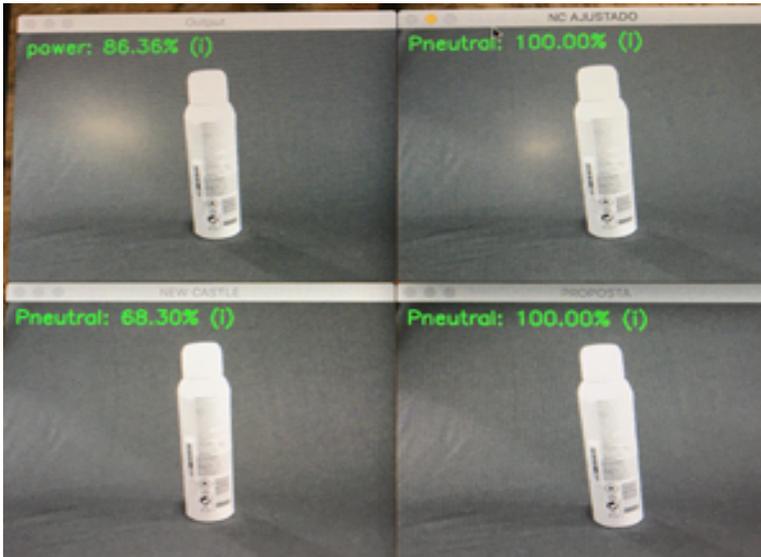


Fonte: Autor

Legenda: Experimento 1, Experimento 5
Experimento 4, Experimento 6

A Figura 35 mostra o resultado dos experimentos 1, 4, 5 e 6 tendo como entrada da rede a imagem de um desodorante.

Figura 35 – Exemplo de objeto para classe potência neutro

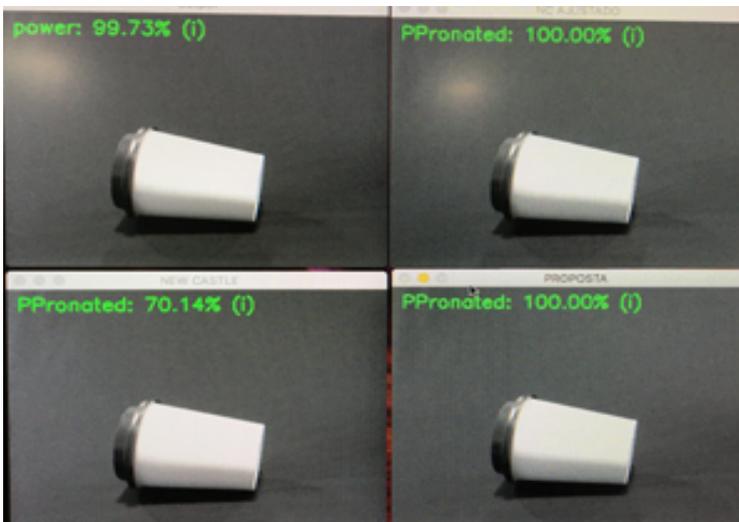


Fonte: Autor

Legenda: Experimento 1, Experimento 5
Experimento 4, Experimento 6

A Figura 36 mostra o resultado dos experimentos 1, 4, 5 e 6 tendo como entrada da rede a imagem de um copo de café.

Figura 36 – Exemplo de objeto classe potência pronado

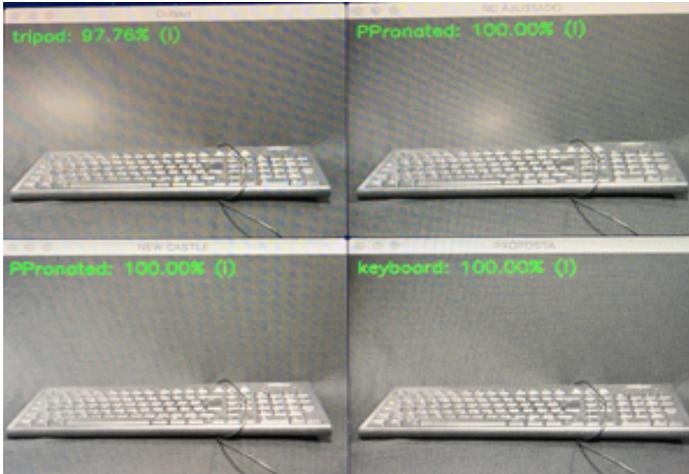


Fonte: Autor

Legenda: Experimento 1, Experimento 5
Experimento 4, Experimento 6

A Figura 37 mostra o resultado dos experimentos 1, 4, 5 e 6 tendo como entrada da rede a imagem de um teclado de computador.

Figura 37 – Exemplo de objeto classe teclado

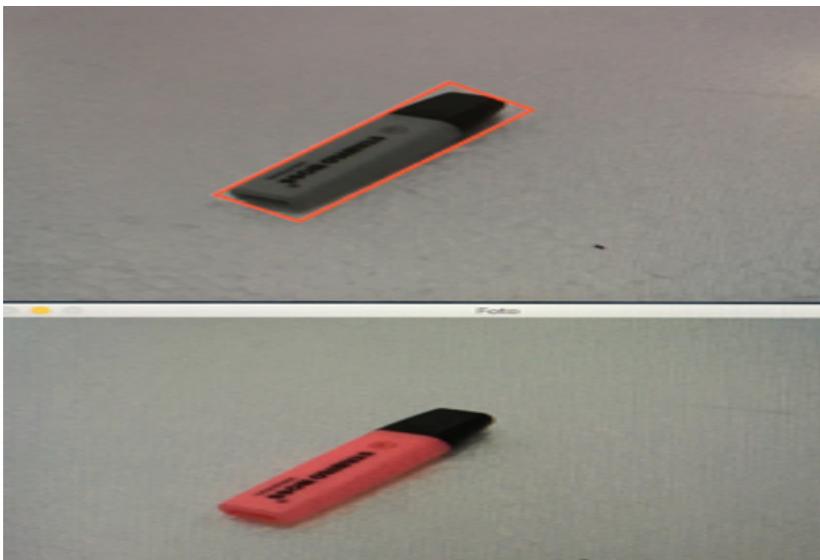


Fonte: Autor

Legenda: Experimento 1, Experimento 5
Experimento 4, Experimento 6

Pode ser visto, na Figura 38, um exemplo de resultado da função caixa de contorno para o objeto caneta marca texto e a imagem original da entrada do classificador, classificado como pinça de três pontos.

Figura 38 - Resultado da função caixa de contorno - três pontos

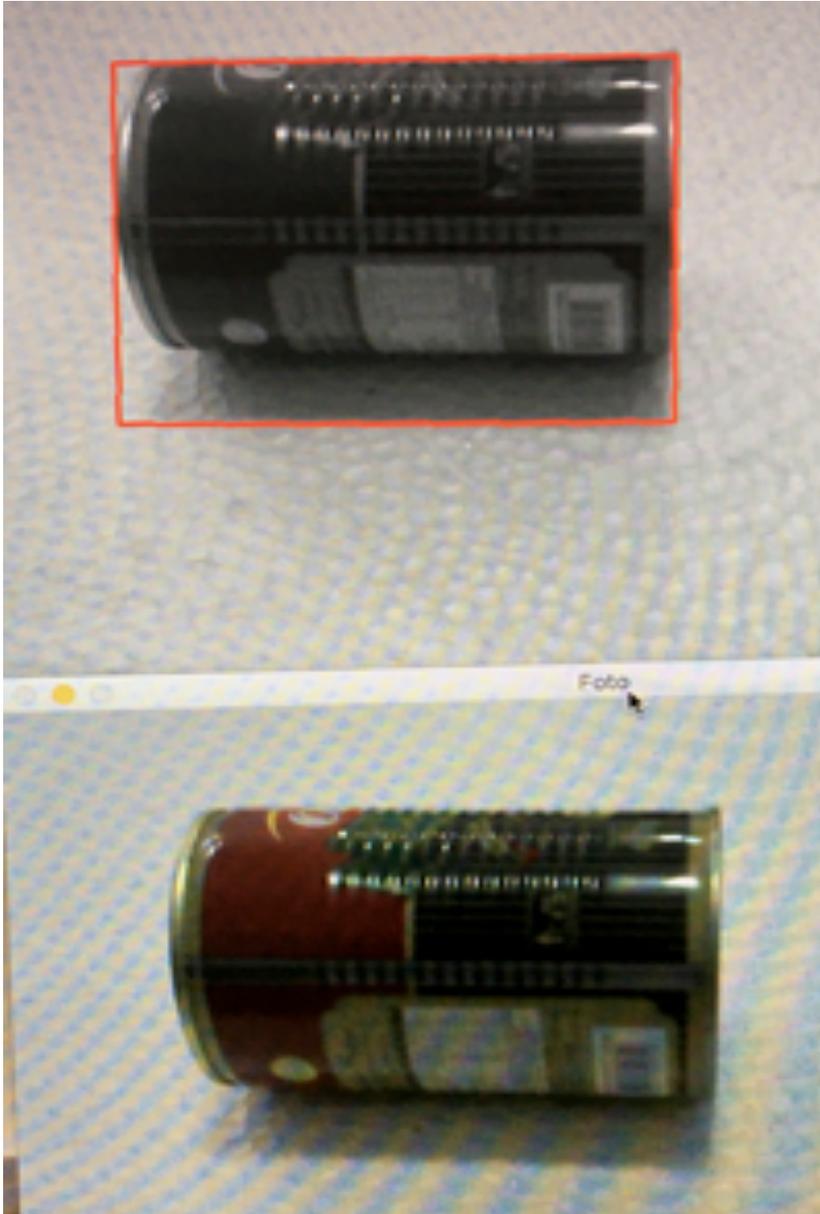


Fonte: Autor

Legenda: Resultado da aplicação função caixa de contorno e imagem original

Pode ser visto, na Figura 39, um exemplo de resultado da função caixa de contorno para o objeto lata de milho e a imagem original da entrada do classificador, classificado como potência.

Figura 39 - Resultado da função caixa de contorno - potência

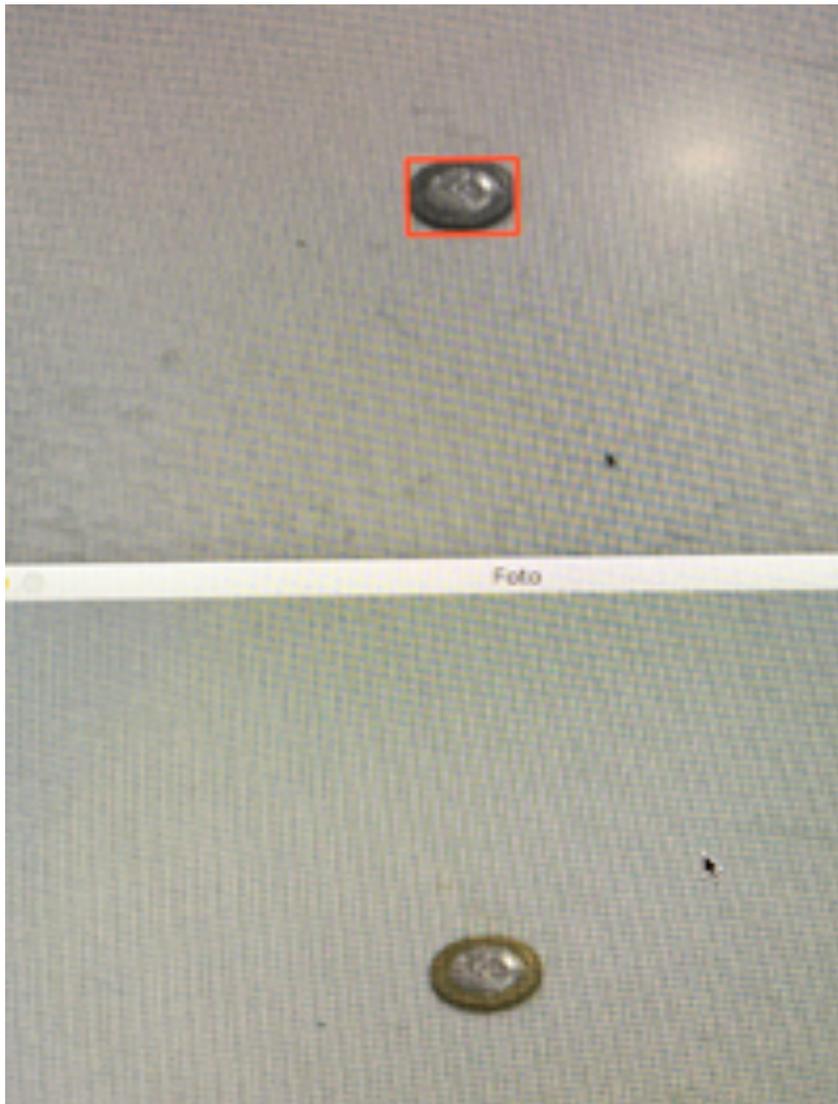


Fonte: Autor

Legenda: Resultado da aplicação função caixa de contorno e imagem original

Pode ser visto, na Figura 40, um exemplo de resultado da função caixa de contorno para o objeto moeda de R\$1,00 e a imagem original da entrada do classificador, classificado como pinça.

Figura 40 - Resultado da função caixa de contorno - pinça

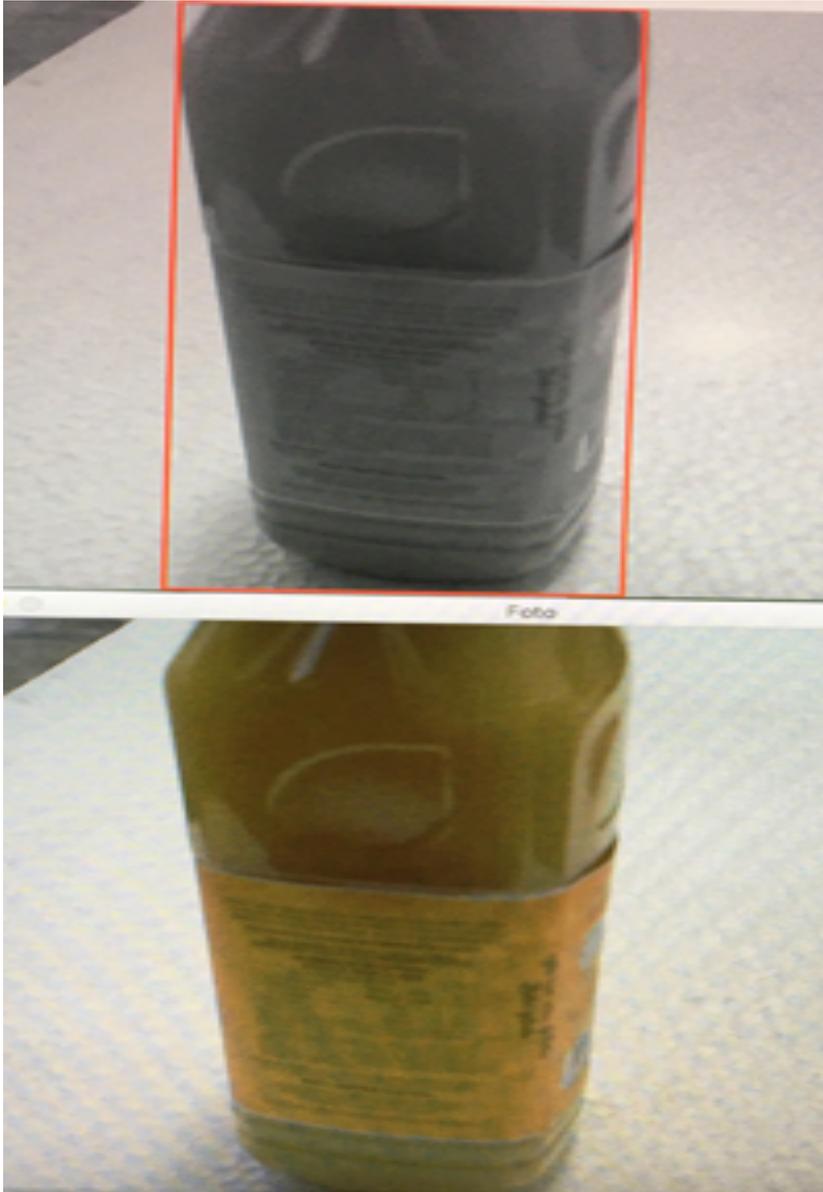


Fonte: Autor

Legenda: Resultado da aplicação função caixa de contorno e imagem original

Pode ser visto, na Figura 41, um exemplo de resultado da função caixa de contorno para o objeto garrafa de suco e a imagem original da entrada do classificador, classificado como potência.

Figura 41 - Resultado da função caixa de contorno - potência



Fonte: Autor

Legenda: Resultado da aplicação função caixa de contorno e imagem original

8 VISUALIZAÇÃO DAS CAMADAS DA REDE NEURAL.

A visualização das camadas da rede SmallerVGG é apenas informativa, não tendo como objetivo a análise do funcionamento interno da rede neural.

Utilizando um copo de café classificado como potência neutro, com probabilidade de acerto de 100% pela rede neural, conforme visto na Figura 42, foram retiradas as imagens resultantes dos filtros de convolução, funções pooling, relu, softmax e *batch normalization* das camadas da rede neural SmallerVGG.

Todas as funções da rede neural modificada SmallerVGG, podem ser encontradas no ANEXO A.

Figura 42 – Objeto Classificado como potência neutro

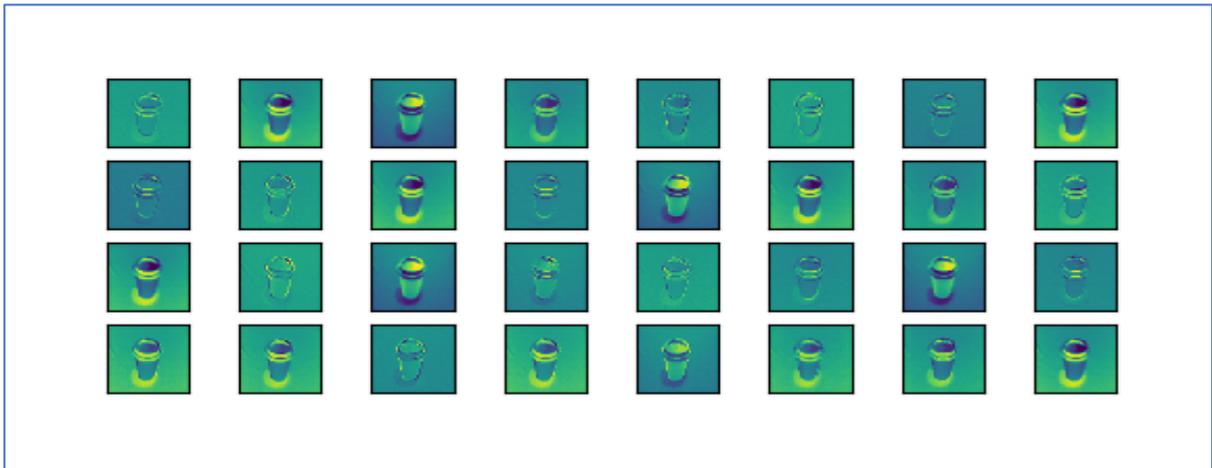


Fonte: Autor

Legenda: Resultado da classificação da rede neural modificada

A Figura 43 mostra o resultado do primeiro filtro de convolução da rede neural, conv2d_1, resultando em um conjunto de 32 imagens.

Figura 43 - Imagens após filtro de convolução conv2d_1

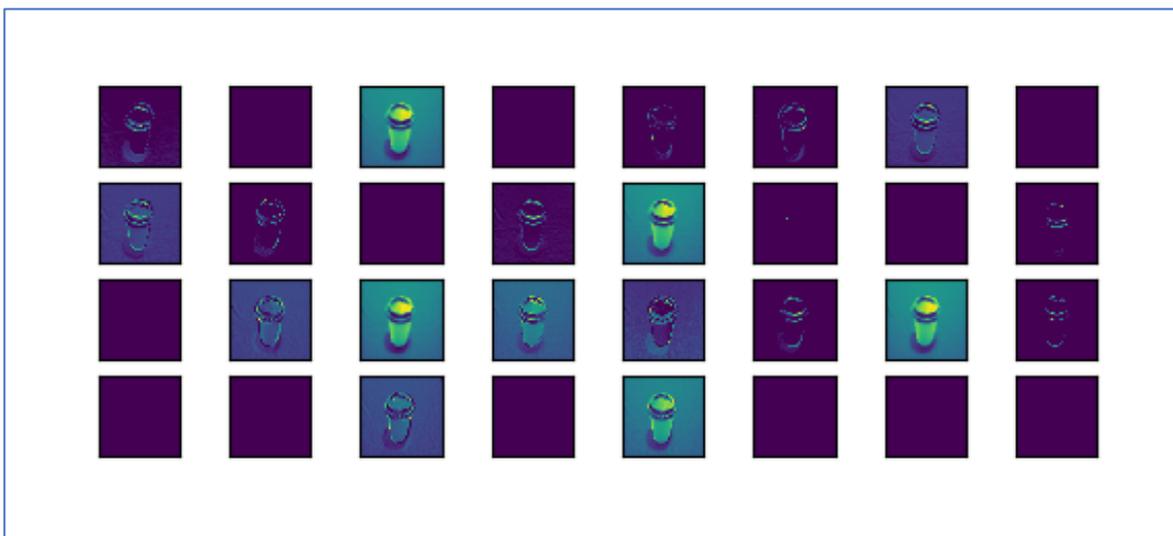


Fonte: Autor

Legenda: Apresentação do resultado após o filtro de convolução conv2d_1

A Figura 44 mostra o resultado da função de ativação, Activation_1 da rede neural.

Figura 44 - Imagens após Activation_1



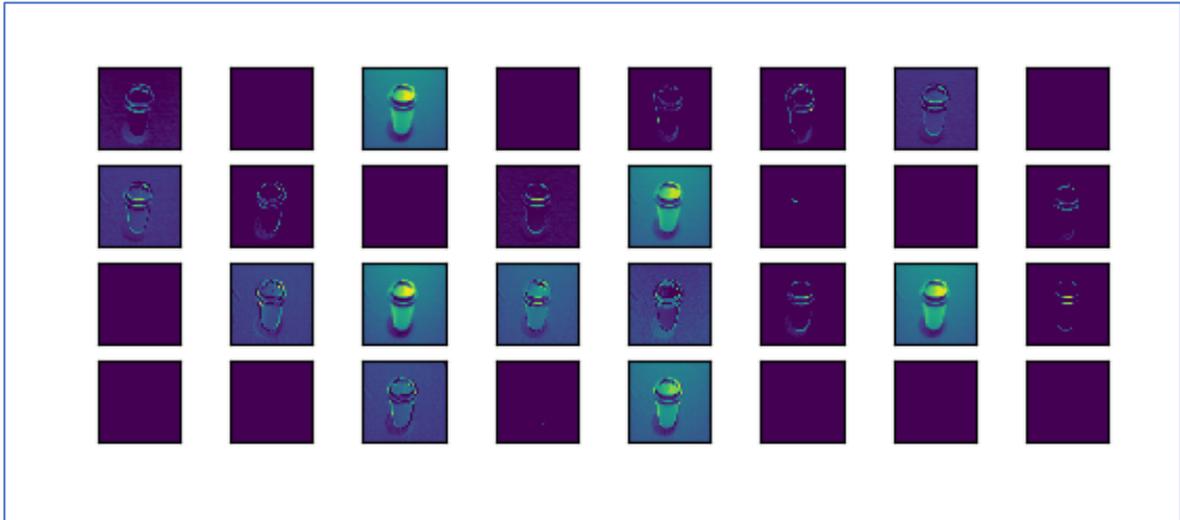
Fonte: Autor

Legenda: Apresentação do resultado após activation_1

A Figura 45 mostra o resultado da função de normalização, Batch_normalization_1.

Não serão mostrados os demais resultados das funções de normalização, pois esta função não altera o resultado das imagens.

Figura 45 - Imagens após Batch_normalization_1

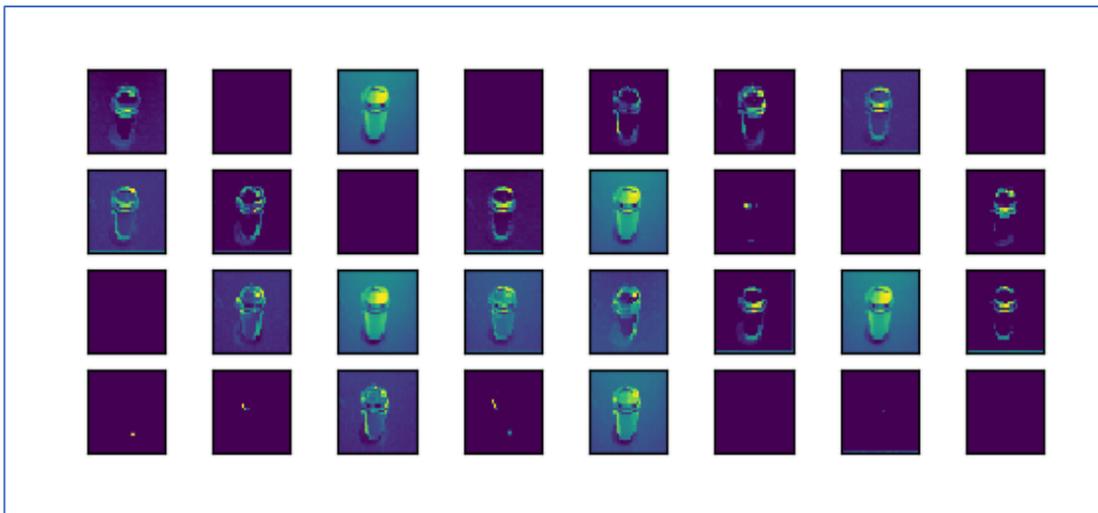


Fonte: Autor

Legenda: Apresentação do resultado após batch_normalization_1

A Figura 46 mostra o resultado da função Max_pooling2d_1.

Figura 46 - Imagens após Max_pooling2d_1



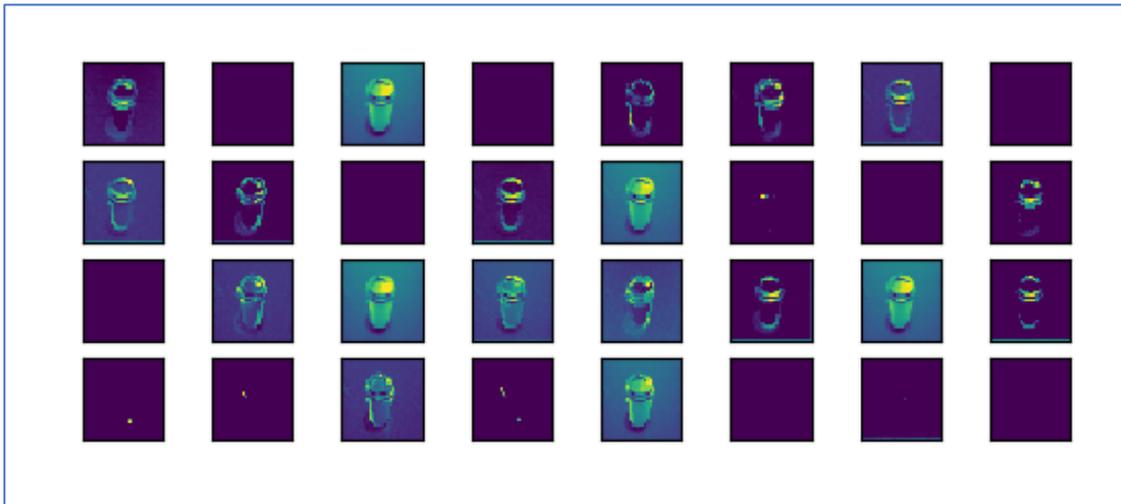
Fonte: Autor

Legenda: Apresentação do resultado após max_pooling2d_1

A Figura 47 mostra o resultado da função dropout_1.

Não serão mostrados os demais resultados das funções drop out.

Figura 47 - Imagens após dropout_1

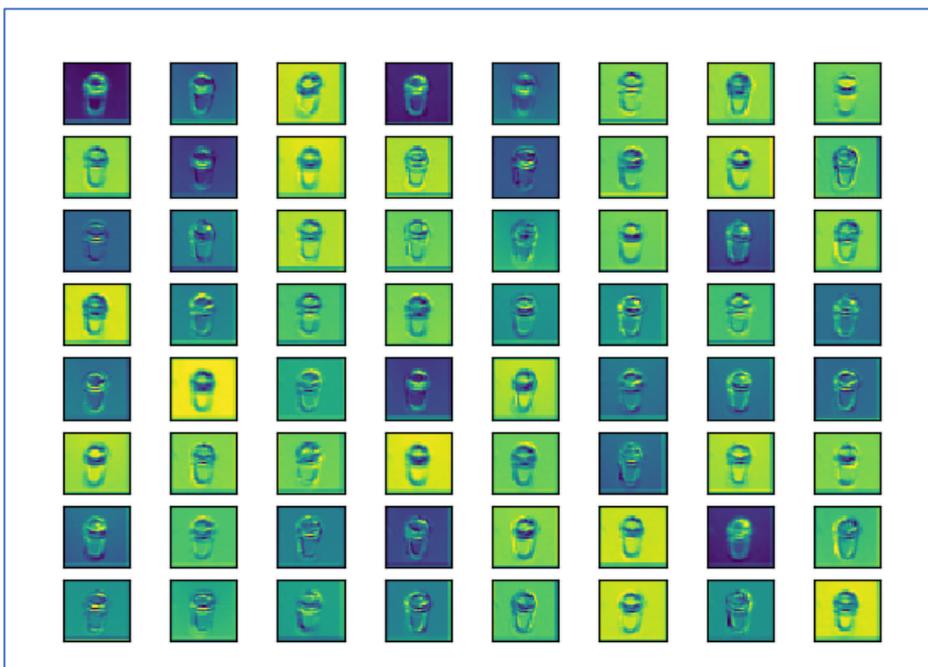


Fonte: Autor

Legenda: Apresentação do resultado após dropout_1

A Figura 48 mostra o resultado do segundo filtro de convolução, função conv2d_2, resultando em um conjunto de 64 imagens.

Figura 48 - Imagens após filtro de convolução conv2d_2

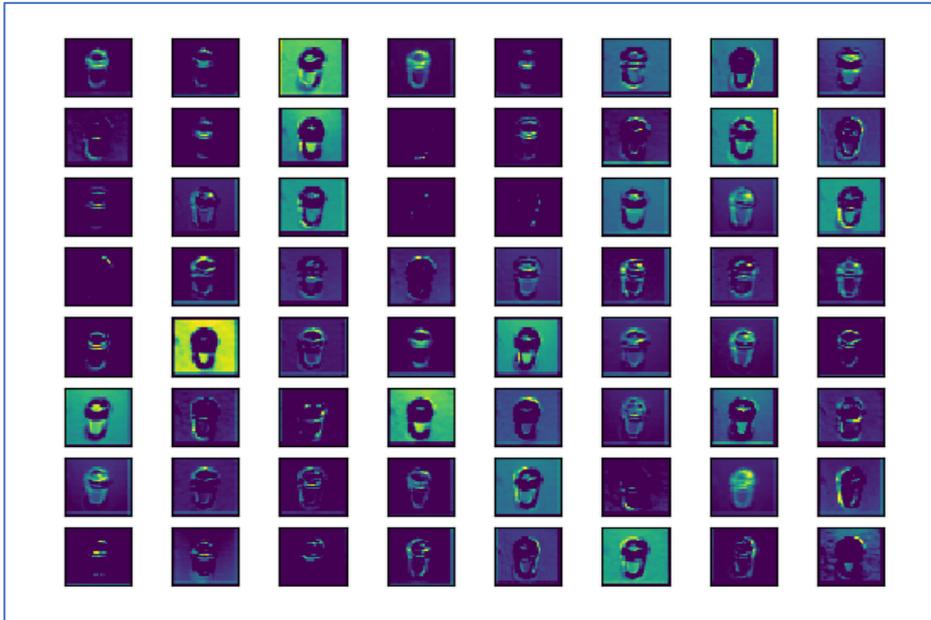


Fonte: Autor

Legenda: Apresentação do resultado após o filtro de convolução conv2d_2

A Figura 49 mostra o resultado da função `activation_2`.

Figura 49 - Imagens após `activation_2`

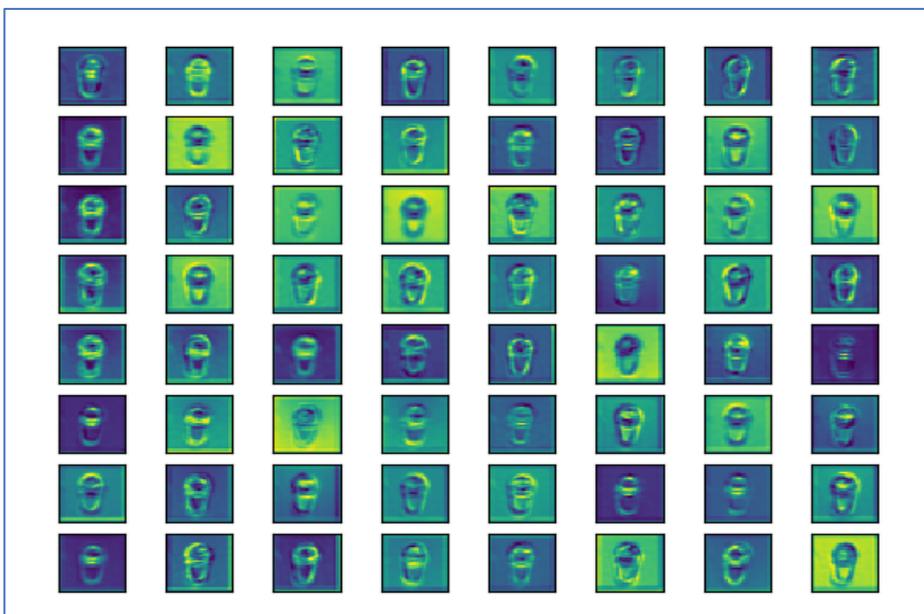


Fonte: Autor

Legenda: Apresentação do resultado após `activation_2`

A Figura 50 mostra o resultado da terceira convolução, `conv2d_3`.

Figura 50 - Imagens após filtro de convolução `conv2d_3`

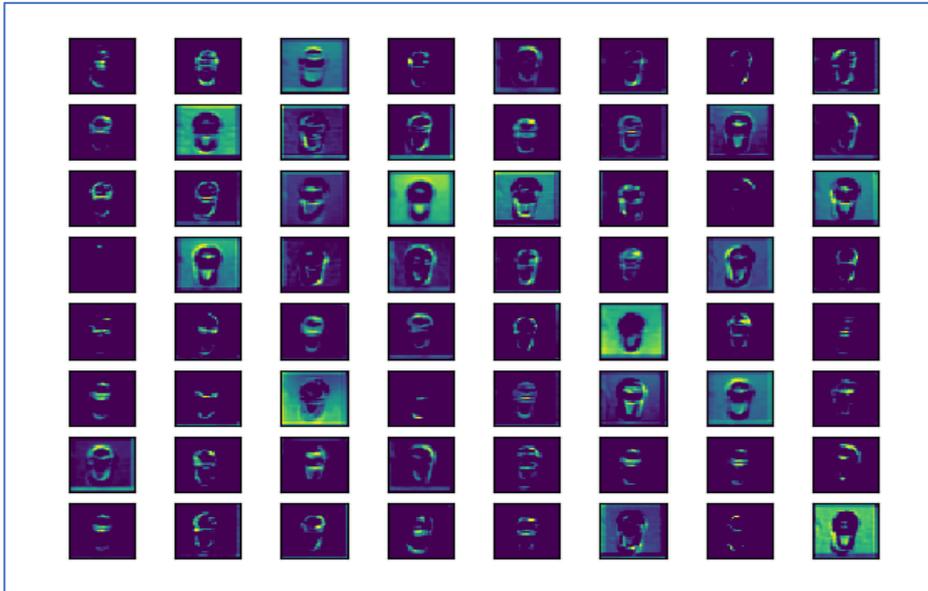


Fonte: Autor

Legenda: Apresentação do resultado após o filtro de convolução `conv2d_3`

A Figura 51 mostra o resultado da função `activation_3`.

Figura 51 - Imagens após `activation_3`

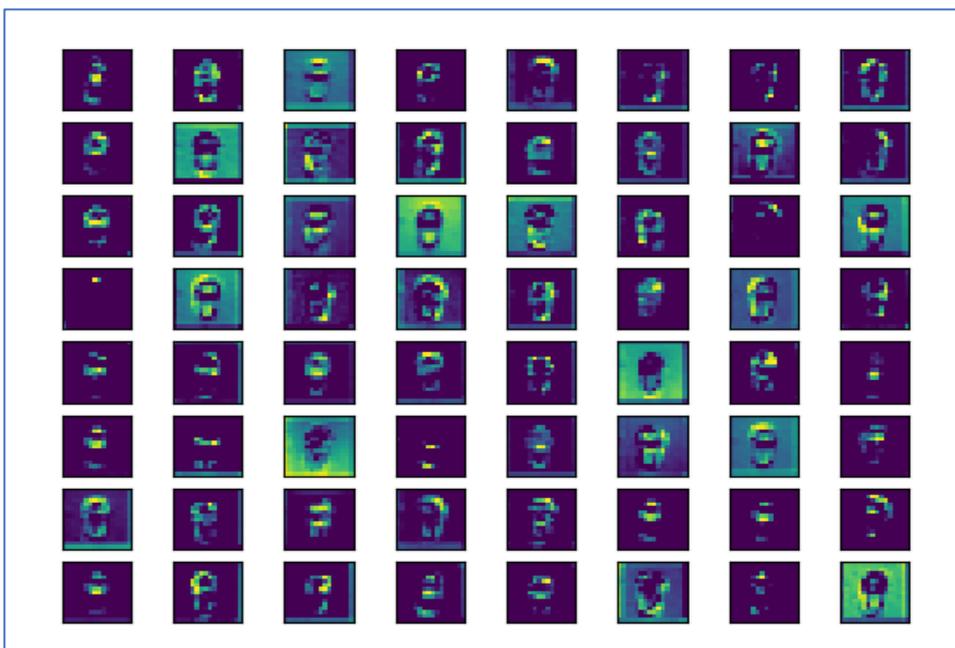


Fonte: Autor

Legenda: Apresentação do resultado após `activation_3`

A Figura 52 mostra o resultado da função `max_poolin2d_2`.

Figura 52 - Imagens após `max_poolin2d_2`

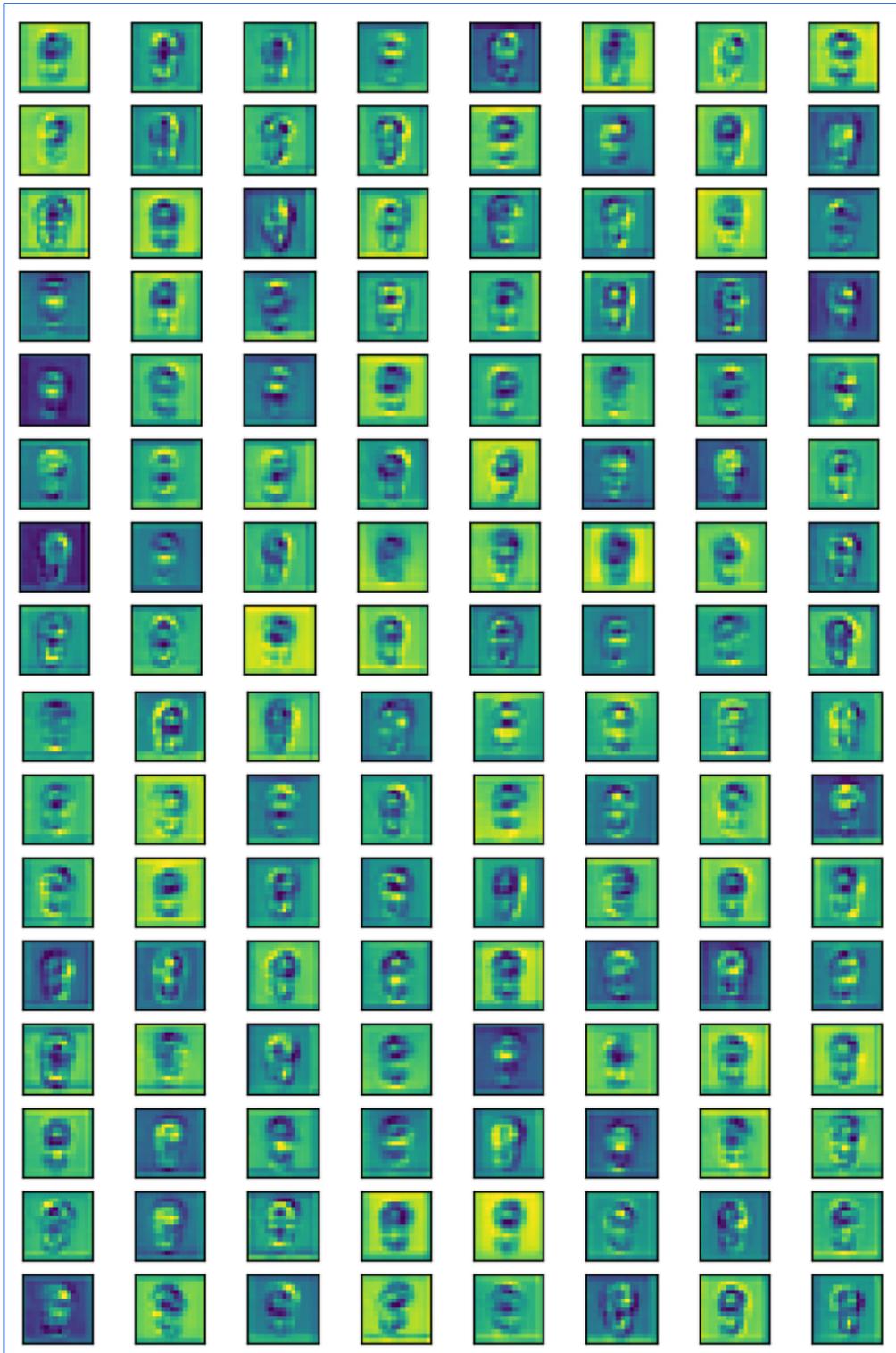


Fonte: Autor

Legenda: Apresentação do resultado após `max_poolin2d_2`

A Figura 53 mostra o resultado da quarta convolução, conv2d_4, resultando em um conjunto de 128 imagens.

Figura 53 - Imagens após filtro de convolução conv2d_4

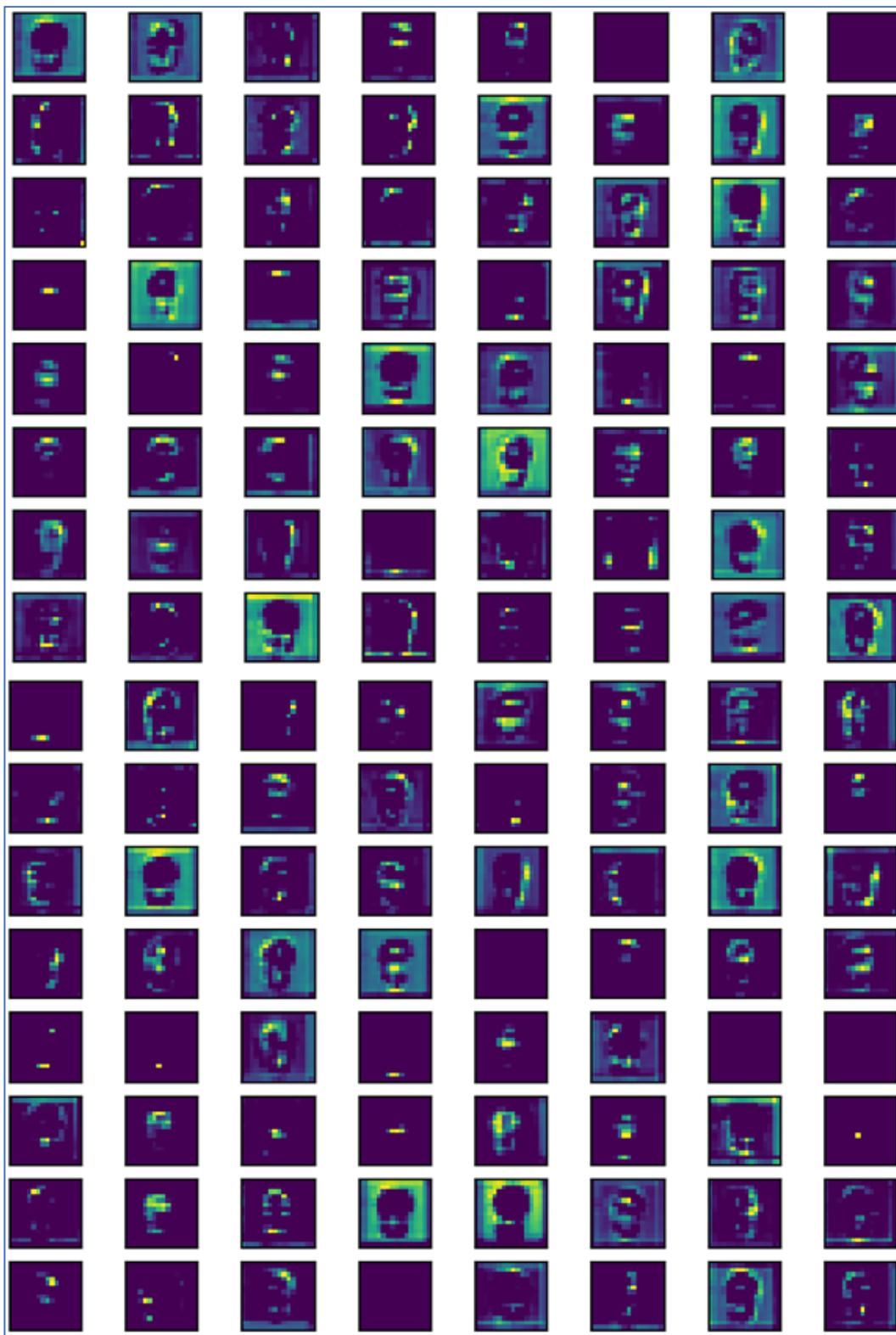


Fonte: Autor

Legenda: Apresentação do resultado após o filtro de convolução conv2d_4

A Figura 54 mostra o resultado da função activation_4.

Figura 54 - Imagens após função activation_4

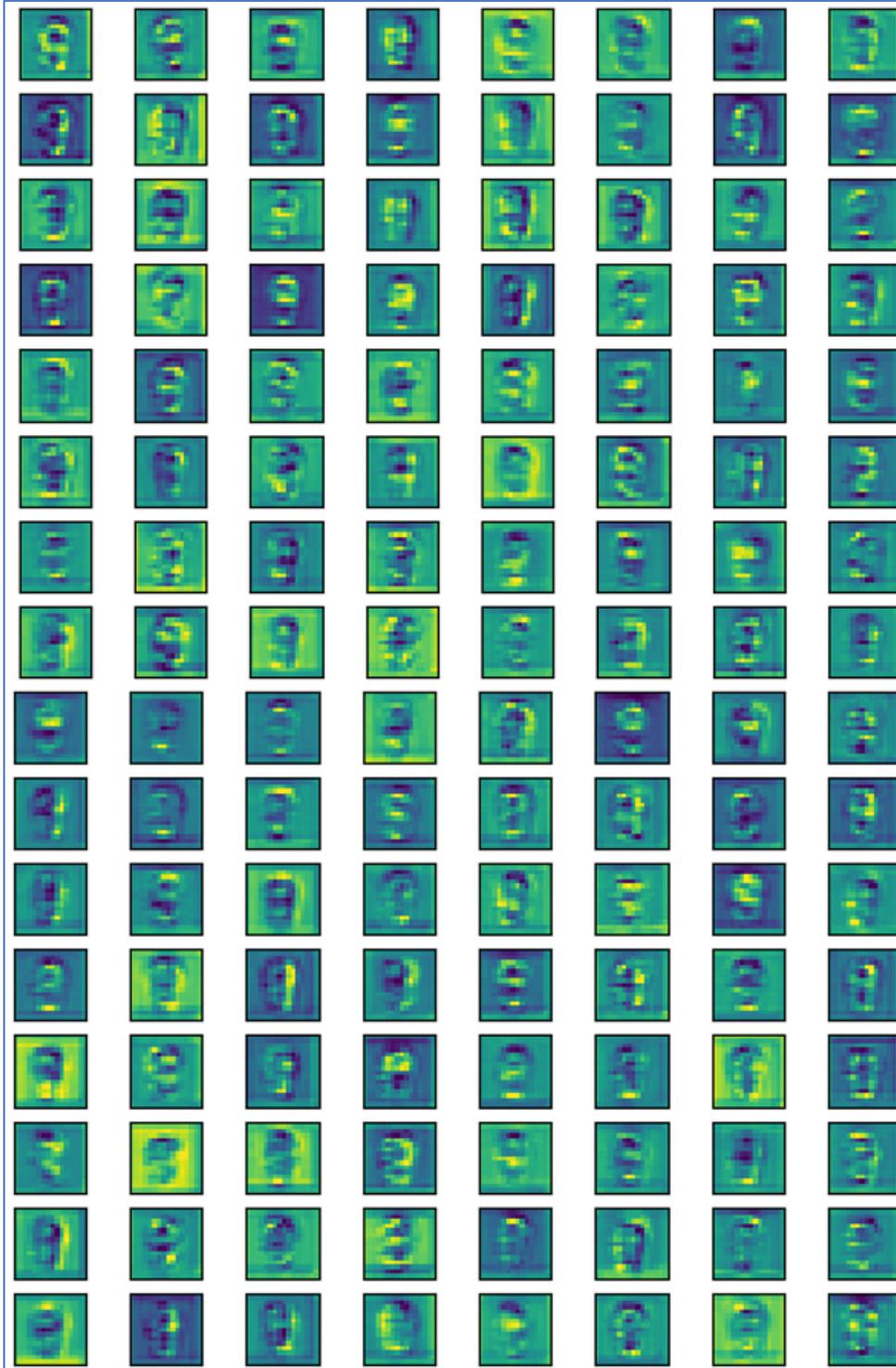


Fonte: Autor

Legenda: Apresentação do resultado após activation_4

A Figura 55 mostra o resultado da quinta convolução, conv2d_5.

Figura 55 - Imagens após quinta convolução conv2d_5

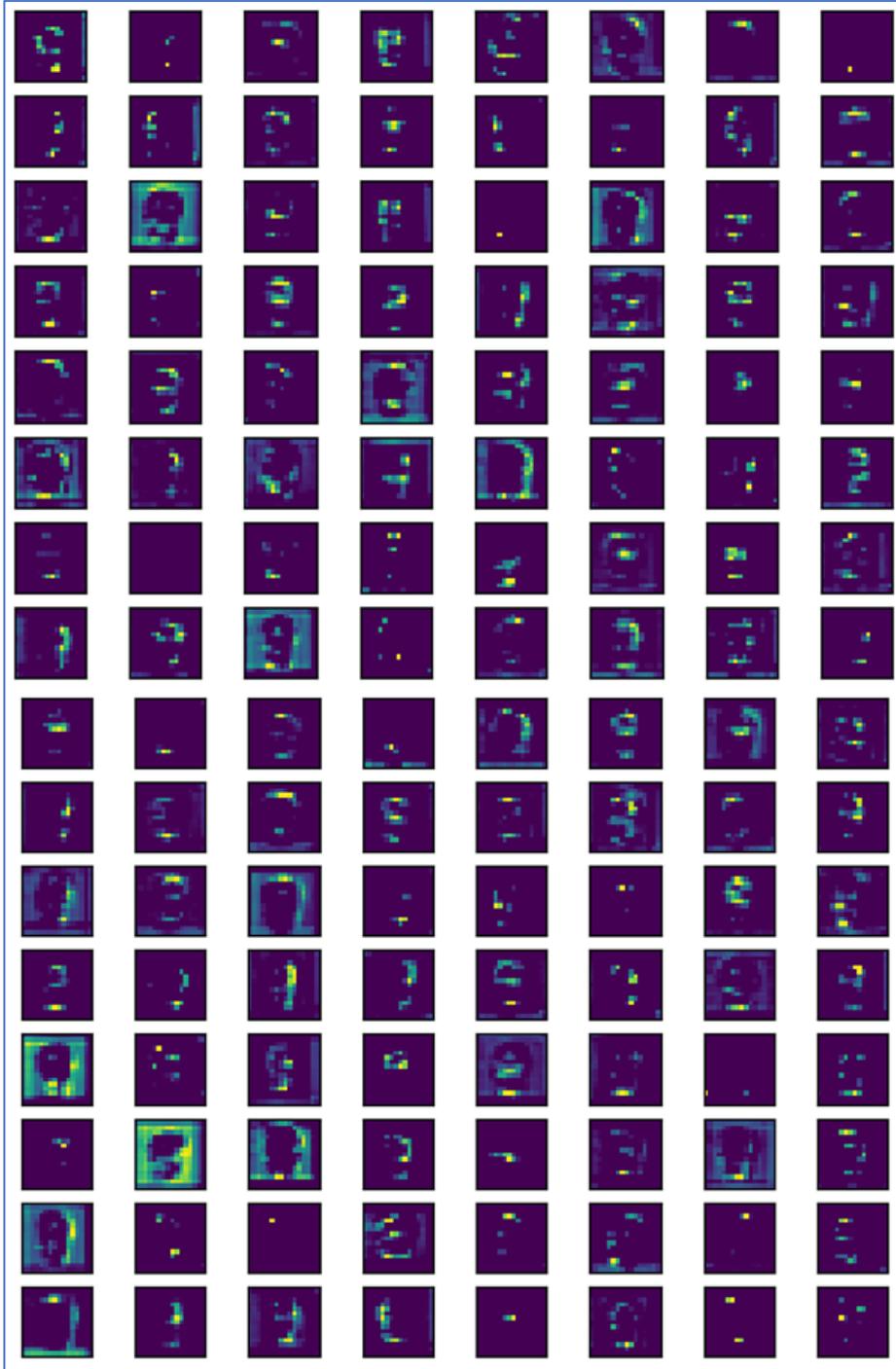


Fonte: Autor

Legenda: Apresentação do resultado após o filtro de convolução conv2d_5

A Figura 56 mostra o resultado da função activation_5.

Figura 56 - Imagens após função activation_5



Fonte: Autor

Legenda: Apresentação do resultado após activation_5

Os resultados das funções Flatten_1, Dense_1, Activation_6 podem ser encontrados em <https://www.dropbox.com/sh/flzrme07g4bphu9/AAB4ghJSO3ay18uFFqJEkhkea?dl=0>.

O resultado da função Flatten_1 é um vetor com 8192 números. O resultado da função Dense_1 é um vetor com 1024 números. O resultado da função Activation_6 é um vetor com 1024 números. O resultado da função Batch Normalization_6 é um vetor com 1024 números. O resultado da função Dropout_4 é um vetor com 1024 números.

O resultado da função Dense_2 pode ser visto no vetor abaixo:

```
9.097628593444824219e-01  2.485757827758789062e+01  -6.624767303466796875e+00  -
1.051636981964111328e+01  9.443134069442749023e-01  4.954936206340789795e-01  -
6.087395668029785156e+00
```

O resultado da função Activation_7, é o resultado da classificação da rede neural, e pode ser visto no vetor abaixo:

```
1.837337384015882980e-13  1.0000000000000000e+00  4.836954946817184631e-13
6.990849414113819993e-13  1.749820427443538051e-11  1.052806508639037020e-09
9.079636863326343010e-14
```

Os nomes das classes da rede neural, após o treinamento, estão em um vetor chamado Label, que foram selecionados pela pasta onde se encontram as imagens de cada classe.

```
Label = [PPronated; Pneutral; keyboard; lateral; piano; tablet; tripod]
```

A correlação entre o resultado da função Activation_7 e o vetor Label determina a classificação e a probabilidade de acerto do objeto pela rede neural.

No caso apresentado, a probabilidade foi de 1, isto é, 100% para a classe Potência Neutro.

9 TESTE PRÁTICO

O protótipo de prótese apresentado não apresenta rotação de punho, os objetos classificados como potência neutro e potência pronado são classificados como potência pelo algoritmo de preensão, da mesma forma, as classes tablet, teclado informática e teclado musical são classificados como extensão do indicador pelo algoritmo de preensão.

Após a transferência do arquivo de treinamento da rede neural para o Raspberry PI 3, foram feitos testes com as classes potência, indicador, lateral e três pontos. Esses testes foram feitos com o autor segurando a prótese, e sendo ele o provedor do sinal de sEMG.

Pode ser visto na Figura 57 o momento que, após uma contração muscular, o ponto de mira laser atinge o objeto.

Figura 57 – Mira laser no objeto a ser manuseado

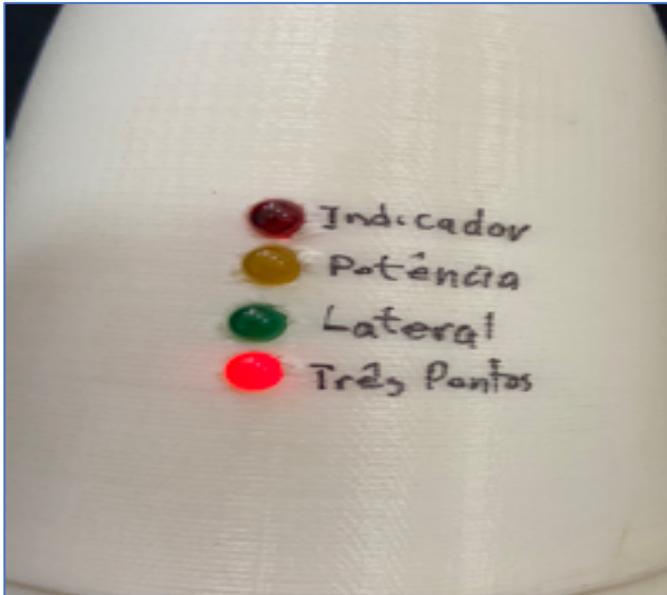


Fonte: Autor

Legenda: A imagem mostra a escolha do objeto pelo laser

Na Figura 58 pode ser visto os LEDs indicadores de classe instalados no dorso da prótese, que estão informando ao usuário a classificação pinça de três pontos.

Figura 58 – LEDs indicadores de classes



Fonte: Autor

Legenda: Indicadores de classe colocados na prótese

A Figura 59 mostra a prótese com atuação na preensão lateral.

Figura 59 – Protótipo de prótese com preensão lateral



Fonte: Autor

Legenda: Prótese executando o movimento de preensão lateral

A Figura 60 mostra a prótese com atuação na preensão de potência.

Figura 60 – Protótipo de prótese com
preensão potência



Fonte: Autor

Legenda: Prótese executando o movimento de
preensão potência

A Figura 61 mostra a prótese com atuação na pinça de três pontos.

Figura 61 – Protótipo de prótese com
preensão pinça de três pontos



Fonte: Autor

Legenda: Prótese executando o movimento de
preensão pinça de três pontos

A Figura 62 mostra a prótese atuando com extensão do indicador.

Figura 62 - Protótipo de prótese como extensão do indicador



Fonte: Autor

Legenda: Prótese executando o movimento de
extensão do indicador

9.1 ANÁLISE DE TEMPOS

A Tabela 19 mostra a indicação aproximada de tempo, em segundos, desde o início da primeira contração até o objeto estar seguro. Para a abertura da prótese o sistema precisa de 2 s adicionais. Deve-se ressaltar que, tendo em vista que os testes foram feitos com o autor segurando a prótese, e sendo ele o provedor do sinal de sEMG para os comandos, as temporizações propostas são baseadas na experiência e habilidade deste, segundo as condições de realização do teste.

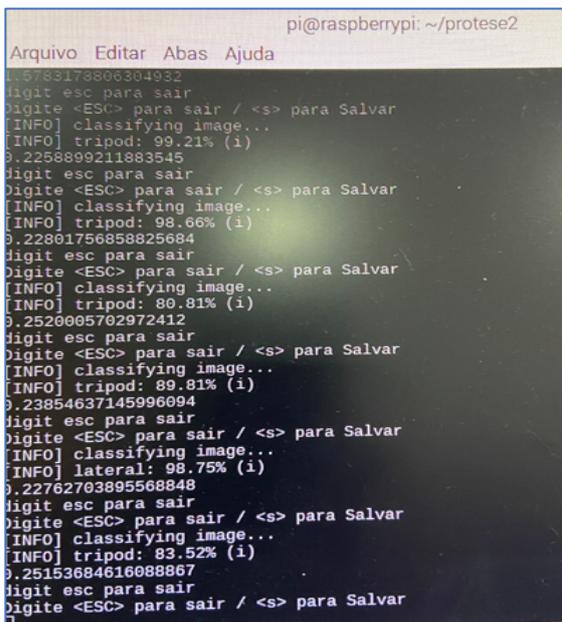
Tabela 19– Descrição dos tempos da prótese

Evento	Início (s)	Fim (s)
Repouso	0	0
Contração	0	2
Mira laser	2	5
Apaga laser	5	5.5
Fotografia objeto	5.5	5.5
Classificação padrões	5.5	5.75
Apresenta resultado	5.75	5.75
Comando do usuário	5.75	7.75
Movimento dos servos	5.75	6.75
Objeto seguro		6.75
Total		6.75

Fonte: Autor

A Figura 63 exemplifica os tempos no classificador com rede neural, desde o comando para obter a fotografia até sua classificação.

Figura 63 – Tempos do classificador neural



```

pi@raspberrypi: ~/protese2
Arquivo Editar Abas Ajuda
0.5783178806304932
Digite <ESC> para sair
Digite <ESC> para sair / <s> para Salvar
[INFO] classifying image...
[INFO] tripod: 99.21% (i)
0.2258899211883545
Digite esc para sair
Digite <ESC> para sair / <s> para Salvar
[INFO] classifying image...
[INFO] tripod: 98.66% (i)
0.22801756858825684
Digite esc para sair
Digite <ESC> para sair / <s> para Salvar
[INFO] classifying image...
[INFO] tripod: 80.81% (i)
0.2520005702972412
Digite esc para sair
Digite <ESC> para sair / <s> para Salvar
[INFO] classifying image...
[INFO] tripod: 89.81% (i)
0.23854637145996094
Digite esc para sair
Digite <ESC> para sair / <s> para Salvar
[INFO] classifying image...
[INFO] lateral: 98.75% (i)
0.22762703895568848
Digite esc para sair
Digite <ESC> para sair / <s> para Salvar
[INFO] classifying image...
[INFO] tripod: 83.52% (i)
0.25153684616088867
Digite esc para sair
Digite <ESC> para sair / <s> para Salvar

```

Fonte: Autor

Legenda: Medição do tempo decorrido durante
classificação

10 DISCUSSÃO

Embora existam estudos sobre preensões de garras robóticas ou mesmo estudos dos movimentos da mão humana em atividades diárias, não foi encontrada na literatura a descrição de nenhuma prótese de mão que utilize a real intenção do usuário para pegar e manusear um determinado objeto, com exceção da I-limb cuja escolha do usuário se dá através de um aplicativo, já que por sEMG o padrão do sinal é pré-programado. Durante o processo de pesquisa e validação dos tipos de preensão foi confirmado que não existe uma forma correta de de preensão para um certo objeto e sim possibilidades ou maneira mais usual para pegá-lo, diante de um determinado propósito de utilização do objeto, resultante de uma maior frequência de ocorrência. O conceito de possibilidades de preensão de um objeto foi aplicado pelo autor para a determinação e escolha das preensões utilizadas.

Como condição de contorno dos experimentos, tem-se para os experimentos 1, 2 e 3, o fundo branco e nível de iluminação natural do ambiente e para os experimentos 4, 5 e 6, tem-se fundo preto e nível de iluminação normal do ambiente. Para todos os experimentos, apenas 1 objeto por vez foi apresentado aos classificadores. A apresentação de vários objetos ao mesmo tempo e segmentação de imagem não fizeram parte do escopo deste estudo, apesar de ser uma situação mais realista do cotidiano.

Os eletrodos de sEMG utilizados nos testes práticos foram colocados no braço do autor para comandar a prótese e não foram efetuados testes com voluntários amputados.

A rede neural de 5 camadas do experimento 1 foi treinada com objetos classificados pelo autor. Não foi utilizado, na entrada da rede, nenhum tipo de filtro, pré-tratamento ou algoritmo para se extrair características discriminante dos objetos contidos nas imagens.

Os objetos de testes, semelhantes aos utilizados por Ghazaei et al. (2017), não foram classificados como pinça no experimento 1. Foram adicionados objetos ao experimento 1 para completar o conjunto de classes válidos do experimento. Este fato é decorrente das diferenças de atributos das imagens utilizadas para o treinamento da rede neural e para os testes.

O gráfico de treinamento da rede neural do experimento 1, Figura 27, demonstrou baixa acurácia para o treinamento e validação cruzada, fato que pode ser explicado pela quantidade e tipos de objetos utilizados no banco de imagens utilizado para treinamento, bem como a classificação proposta. Isto mostra a dependência do resultado à base de imagens e associação com os padrões de preensão estabelecida.

Para a comparação da rede neural do experimento 1, o experimento 2 foi realizado com os classificadores KNN e LDA, utilizando o mesmo critério adotado para o experimento 1, ou

seja, imagens provenientes da webcam em fundo branco, sem filtro ou pré-tratamento das imagens e sem algoritmo de extração de características discriminantes dos objetos contidos nas imagens.

Em decorrência dos resultados obtidos no experimento 2, o experimento 3 foi conduzido com o objetivo de validar os algoritmos utilizados e analisar o resultado dos classificadores KNN e LDA utilizando características discriminantes de imagens como entrada dos classificadores. Com as alterações da sistemática de treinamento e classificação das imagens, o experimento 3 não será utilizado para comparação.

A acurácia do experimento 1, com a rede neural de 5 camadas, foi de 93%, enquanto a acurácia do experimento 2 para o classificador KNN foi de 81% e a acurácia para o classificador LDA foi de 50%.

Os valores de acurácia dos classificadores KNN e LDA são explicados pelo fato da imagem de entrada nos classificadores, tanto na fase de treinamento como na fase de testes, não ter passado por nenhum tipo de pré-processamento para extração de características discriminantes. Grandes quantidades de parâmetros de entrada para os classificadores KNN e LDA não são bem classificados, apresentando erros elevados, o que a literatura chama de maldição da dimensionalidade.

O experimento 3 demonstra a melhora significativa da acurácia ao se utilizar 2 parâmetros discriminantes dos objetos nos classificadores KNN e LDA, obtendo uma acurácia de 97% para o classificador KNN e 91% de acurácia para o classificador LDA.

Os classificadores KNN e LDA apresentam tempo de treinamento e tempo de resposta de classificação menor do que os tempos da rede neural. O tempo de classificação para estes classificadores está em torno dos 120ms contra 250ms da rede neural SmallerVGG.

Por outro lado, uma das vantagens da utilização da rede neural sobre os classificadores KNN e LDA está no fato de não ser necessário passar as imagens por uma etapa de pré-processamento para encontrar características discriminantes de cada objeto ou classe, já que a própria rede neural faz esse processo internamente. Outra vantagem da rede neural está na facilidade de se aumentar o número de preensões ou classes, apenas modificando o banco de imagens e treinando novamente a rede neural. Existe a possibilidade de se otimizar os resultados da rede neural através de ajustes ou *tunning* da rede.

Comparando os valores de sensibilidade, observa-se que a rede neural do experimento 1 apresentou maior eficácia em classificar corretamente os objetos que pertencem a determinada classe do que os classificadores KNN e LDA, apresentando uma menor quantidade de resultados falso-negativos.

Em se tratando dos valores de especificidade, a rede neural do experimento 1 apresentou maior eficácia em classificar corretamente os objetos que não pertencem a determinada classe, apresentando uma menor quantidade de resultados falso-positivos.

O experimento 4 foi conduzido com a rede neural de 5 camadas treinada com a base original de imagens de Ghazaei et al. (2017). Entretanto, notou-se a inconsistência da classificação do banco de imagens utilizada para o treinamento das classes, pois o banco apresentava imagens semelhantes em diversas classes.

A rede neural de 5 camadas proposta no experimento 4 foi treinada com o mesmo banco de imagens de Ghazaei et al. (2017) e alcançou uma acurácia total de 84%, enquanto o trabalho de Ghazaei et al. (2017) obteve acurácia de 88%, conforme Quadro 1.

Essa acurácia de 84% pode ser justificada pelas imagens utilizadas por Ghazaei et al. (2017) para o treinamento da rede neural, as quais se repetiam no bando de imagem. Um exemplo claro é um tubo de pomada que ao ser fotografado frontalmente, olhando a tampa, apresenta o mesmo formato que uma bola de tênis de mesa.

A rede proposta por Ghazaei et al. (2017) utilizou apenas 2 camadas, não sendo capaz de identificar objetos ou características mais específicas, o que justifica a acurácia de 88% com seu banco de imagens, mesmo com objetos semelhantes em mais de uma classe.

Eliminando as inconsistências encontradas no banco de imagens original de Ghazaei et al. (2017), um novo banco de imagens foi elaborado para a execução do experimento 5. Este novo banco de imagens foi acrescido da classe de preensão lateral em função de sua importância, conforme Feix et al. (2013). O experimento 5 obteve 98% de acurácia, conforme Tabela 17.

No experimento 6, a rede neural de 5 camadas proposta treinada com o banco de imagens de Ghazaei et al. (2017) modificado pelo autor, com a inclusão da classe preensão lateral e classe indicador, obteve 99% de acurácia, conforme Tabela 18.

Pode-se dizer que para o experimento 6 a rede neural foi treinada de forma mista, isto é, para as classes potência neutro, potência pronado, três pontos e lateral a rede foi treinada para reconhecer padrões, enquanto para as classes teclado informática, teclado musical e tablet, a rede foi treinada para reconhecer objetos. Mesmo com apenas 5 camadas convolucionais, a rede SmallerVGG apresentou elevada acurácia para reconhecer estes objetos e padrões, conforme Tabela 18.

Os gráficos de treinamentos, Figura 30, Figura 31 e Figura 32, demonstraram maior acurácia no treinamento e validação cruzada nos experimentos 5 e 6 do que no experimento 4, sinalizando inconsistências com o banco de imagens de Ghazaei et al. (2017).

Comparando os valores de sensibilidade, observa-se que os experimentos 5 e 6 apresentaram maior eficácia em classificar corretamente os objetos que pertencem a determinada classe do que o experimento 4, isto é, as redes neurais dos experimentos 5 e 6 foram treinadas com um banco de imagens mais consistente e apresentaram uma menor quantidade de resultados falso-negativos. O mesmo pode ser visto nos valores de especificidade, indicando que as redes neurais dos experimentos 5 e 6 foram treinadas com um banco de imagens mais consistente e apresentaram uma menor quantidade de resultados falso-positivos.

espaço .

Os resultados foram obtidos em ambiente controlado, com a distância e altura da prótese fixa em relação ao objeto. Modificações da distância e altura utilizadas alteram o resultado do classificador. Estes erros estão relacionados ao modelo de classificador que utiliza padrões de imagens para classificação do tipo de preensão.

Observou-se também, que a correta classificação dos objetos pela rede neural depende dos objetos escolhidos para a composição do banco de imagens de treinamento, bem como a associação feita com os padrões de preensão, visto que os objetos de testes utilizados no experimento 4 não foram classificados corretamente no experimento 1.

A utilização de um micro computador RPI 3 e não um computador com GPU para análise em tempo real, conforme Dosen et al. (2010), associado ao fato de que a prótese foi confeccionada em impressoras 3D e não foi utilizada uma prótese i-limb Ultra TM, conforme Ghazaei et al. (2017), tornaram o projeto em um protótipo de baixo custo.

O protótipo de mão apresentado tem todo o sistema de controle e comando embarcado à prótese, tornando-a portátil, diferentemente das propostas de Ghazaei et al. (2017), Dosen et al. (2010), DeGol et al. (2017) e Andrade et al. (2017) nos quais o processamento foi feito em unidade a parte como um PC padrão Intel ou NVIDIA Tegra com GPU, tendo inclusive apresentado acurácia superior as apresentadas por todos eles.

A utilização de um arduino nano para comandar os servos motores pode ser descartada, tendo o Raspberry Pi condições de gerenciar esta tarefa. Essa alteração reduz a quantidade de fios e cabos internos e aumenta o espaço disponível dentro do corpo do protótipo da prótese. Mesmo com um maior espaço interno, o corpo da prótese deve ser ajustado para cada usuário.

O protótipo de prótese de mão proposto teve como foco a aparência mais natural, incorporando uma discreta webcam na palma da mão, diferentemente das propostas de Ghazaei et al. (2017) e Dosen et al. (2010) que utilizaram uma webcam externa à prótese.

Em decorrência da escolha do autor pela valorização estética da prótese proposta, a utilização interna de cabos de aço para seu acionamento deixa os movimentos dos dedos limitados. Uma alternativa poderia ser a utilização de motores para acionamento individuais dos dedos conforme Fajardo et al. (2017), entretanto, deve se levar em consideração o espaço para a instalação da câmera na palma da prótese.

Para a utilização de objetos com preensão lateral, o objeto deve ser primeiramente posicionado na prótese para posterior preensão; esta limitação se deve à ausência do movimento do polegar em 2 planos.

Outra limitação ocorre com o movimento de punho, que não foi implementado em função dos cabos de aços passarem do braço para a mão através do punho.

A placa de sensor Myoware possui ajuste de ganho do sinal proveniente do sEMG e o software da placa do arduino possui um ajuste eletrônico do limiar para contador de pulsos. Embora simples, esses dois ajustes possibilitam customização necessária em função do posicionamento dos sensores de sEMG e das habilidades e condições físicas de cada usuário.

A utilização de sensores mioelétricos descartáveis pode gerar desconforto ao usuário, a utilização de um modelo compatível com o myo armband pode solucionar este inconveniente.

Como o sistema funciona com contagem de pulsos, os eletrodos de sEMG podem ser substituídos por chaves, micros *switches*, sensores eletrônicos de deformação ou pressão, potenciômetros, levando em consideração o tipo de amputação.

Mesmo com o tempo de classificação de aproximadamente de 250 ms, o tempo total, do momento da primeira contração do usuário até a preensão do objeto, foi de cerca de 6,75s. Os tempos propostos foram baseados em experimentações realizadas com o autor. Reduções nos tempos da contagem das contrações ou mesmo a adoção de um novo algoritmo para a contagem dos pulsos, associados à redução do tempo de estabilização do laser estabelecido em cerca de 2s e redução no tempo para iniciar a captura da imagem podem reduzir significativamente o tempo total para a preensão do objeto. Estes tempos devem ser ajustados levando em consideração as habilidades de cada usuário, podendo chegar a cerca de 3s.

A análise das imagens das camadas da rede neural SmallerVGG não faz parte do escopo deste projeto. Contudo, testes com as imagens podem ser feitos para se extrair os filtros aplicados em cada camada da rede neural. Pelas figuras 43 a 56, mesmo com a baixa resolução das imagens, existem imagens que parecem ter passado por filtros de contorno, enquanto outras imagens parecem ter passado por filtros de máxima e outras por filtros de textura.

A utilização de um computador MAC propiciou a execução simultânea dos 6 experimentos. O treinamento da rede neural neste computador e sua posterior utilização,

contornou o problema da falta de capacidade de processamento do RPI3, tornando o sistema portátil. Para o perfeito funcionamento entre as plataformas, foram utilizadas mesmas versões de softwares e bibliotecas, não gerando nenhuma inconsistência aos experimentos.

11 CONCLUSÃO

O novo sistema proposto obteve uma acurácia de 98% para a preensão de objetos da classe pinça de três pontos, 100% de acurácia para a preensão de objetos da classe potência punho neutro, 99% de acurácia para a preensão de objetos da classe potência punho pronado, 98% de acurácia para a preensão de objetos da classe preensão lateral e 99% de acurácia para a preensão de objetos da classe indicador, como visto na Tabela 18.

Comparando com outros trabalhos da literatura que utilizaram sistema de visão para definir o padrão de preensão para a prótese, o sistema proposto com rede neural de 5 camadas alcançou acurácia de 99.0 %, enquanto a proposta de Dosen et al. (2010) apresentou acurácia de 93.2%, a proposta de DeGol et al. (2017) apresentou acurácia de 93.2% e a proposta de Ghazaei et al. (2017) apresentou acurácia de 88.0%, conforme visto no Quadro 1.

Este trabalho demonstrou que a utilização de um sistema de visão para auxiliar a a definição do padrão de preensão e manipulação de objetos é promissor e que estudos nesta área devem ser continuados.

Como contribuições deste trabalho pode-se destacar:

- Adição de novas classes e novos movimentos comparando com próteses existentes;
- Utilização de um sistema totalmente embarcado;
- Utilização de tecnologia embarcada de baixo custo em uma prótese de baixo custo, garantindo acessibilidade à grande parte da população usuária;
- Reduzido tempo de treinamento por parte do usuário, considerando que o sistema embarcado utiliza contagem de pulsos do sinal de sEMG.
- Disponibilidade do banco de imagens utilizado.

Como contribuição ao meio acadêmico ficam todos os arquivos, bancos de imagens, algoritmos testados no RPI e programas do arduino disponíveis para consulta e utilização em: <https://www.dropbox.com/sh/flzrme07g4bpnu9/AAB4ghJSO3ay18uFFqJEkhkea?dl=0>

Para a continuidade dos trabalhos propõe-se: o aprimoramento da prótese para movimentação do polegar, extensão e rotação do punho; o aprimoramento do tempo de resposta dos movimentos da prótese; o aprimoramento na motorização dos dedos; a inclusão de novas classes e movimentos; o treinamento da rede neural em condições reais em que os objetos estejam em condições de uso; e testes e validação dos resultados com amputados.

REFERÊNCIAS

- AL-TIMEMY, A. H.; KHUSHABA, R. N. Selecting the optimal movement subset with different pattern recognition-based EMG control algorithms. 978-1-4577-0220-4/16/\$31.00. IEEE. 2016.
- ANDRADE, D. T. G.; ISHIKAWA, A.; MUNOZ, A. D.; ROHMER, E. **A hybrid approach for the actuation of upper limb prostheses based on computer vision.** IEEE 978-1-5386-0956-9/17/431.00 2017.
- AMSÜSS, S.; GOEBEL, P. M.; JIANG, N.; GRAIMANN, L. P.; FARINA, D. Self-correcting pattern recognition system of surface EMG signal for upper limb prosthesis control. **IEEE Transactions on Biomedical Engineering**, Vol. 61, n. 4. abr. 2014.
- ARAUJO, F. H. D.; CARNEIRO, A. C.; SILVA, R. R.V.; MEDEIROS, F. N. S.; USHIZIMA, D. M. Redes neurais convolucionais com tensorflow: teoria e prática. In: ESCOLA REGIONAL DO PIAUÍ, 3., jun. 2017, Picos. **Anais... Picos (PI)**: SBC, 2017. p. 382-406.
- ARDUINO. **Manual do Myoware Muscle Sensor.** 2019. Disponível em: <http://www.advancertechnologies.com/p/myoware.html>. Acesso em 20 mar. 2019
- BOHG, J.; MORALES, A.; ASFOUR, T.; KRAGIC, D. **Data-driven grasp synthesis. A survey.** IEEE Transactions on robotics. vol. 30, n. 2, p.289-309, abr. 2014.
- BUCHANAN, J. **Printing and Assembly of Kwawu 2.0 Prosthetic Hand.** Jan. 2018. Disponível em: <http://enablingthefuture.org/e-nable-device-specific-badges>. Acesso em 4 maio 2019.
- CELLA, C. C.; VILAGRA, J. M. O. **Análise da qualidade de vida de pacientes amputados transfemoral e transtibial com sensação fantasma.** In:FACULDADE ASSIS GURGACZ. FIEP BULLETIN, v. 81, Special Edition. Article II. 2011. ISSN 2412-2688. Disponível em: <http://www.fiepbulletin.net/index.php/fiepbulletin/article/view/438>. Acesso em: 4 maio 2019.
- CHEN, X.; ZHANG, D.; ZHU, X. Application of self-enhancing classification method to electromyography pattern recognition for multifunctional prosthesis control. **Journal of Neuroengineering and Rehabilitation**, v.10, n. 44, p. 1-13, maio 2013. PMID: 23634939.
- CORDELLA, F.; CIANCIO, A. L.; SACCHETTI, R.; DAVALLI, A.; CUTTI, A. G.; GUGLIELMELLI, E.; ZOLLO, L. Literature review on needs of upper limb prosthesis users. **Frontiers in neuroscience.** doi.10.3389/fnins.2016.00209. vol. 10, n. 209, maio 2016.
- CORNELL UNIVERSITY. **Learning to Grasp.** 2009. Disponível em: http://pr.cs.cornell.edu/grasping/rect_data/data.php. Acesso em: 20 mar. 2019.
- ANDRADE, D. T. G.; RIBAS NETO, A.; PEREIRA, G. M.; ROHMER, E. **Human prosthetic interaction: Integration of several techniques.** ISSN 2175 8905. Porto Alegre. Out 2017.

DE LUCA, C. J. **Surface electromyography: detection and recording**. [S.l.]: DelSys, 2002.

DEGOL, JOSEPH.; AKHTAR, A.; MANJA, B.; BRETI, T. Automatic grasp selection using a camera in a hand prosthesis. **HHS Public Access**. p. 431 – 434. fev, 2017. PMID: 28261002.

DOSEN, STRAHUNJA.; CIPRIANI, C.; KOSTIC, M.; CONTROZZI, M.; CARROZZA, M. C. Cognitive vision system for control of dexterous prosthetic hands: experimental evaluation. **Journal of Neuroengineering and Rehabilitation**. v. 7, n. 42, p. 7-42, ago. 2010. PMID: 20731834.

FAJARDO, J.; FERMAN, V.; LEMUS, A.; ROHMER, E. An affordable open-source multifunctional upper-limb prosthesis with intrinsic actuation. **IEEE Xplore**. mar. 2017. DOI 10.1109/ARSO.2017.8025206

FEIX, T.; BULLOCK, I. M.; DOLLAR, A. Analysis of human grasping behavior: object characteristics and grasp type. **IEEE Transactions on Haptics**. p.1939-1412. DOI 10.1109/TOH.2014.2326871.

FISHER, R. A. **The use of multiple measurements in taxonomic problems**. Annals of Eugenics. p. 179-188, set. 1936.

GEETHANJALI, P. Myoelectric control of prosthetic hands: state-of-art review. **Medical Devices: Evidence and Research**, v. 9, p. 247-255. july, 2016. PMID: 27555799

GENG, W.; DU, Y.; JIN, W.; WEI, W.; HU, Y.; LI, J. Gesture recognition by instantaneous surface EMG images. **Scientific Reports**. 14p, nov. 2016. PMID: 27845347.

GEUSEBROEK, J.M.; BURGOUTS, G.J.; SMEULDERS, A. W. M. The Amsterdam library of object images. **Int. J. Comput Vision**, v. 61, n. 1, p. 103-112, jan. 2005.

GHAZAEI, G.; ALAMEER, A.; DEGENAAR, P.; MORGAN, G.; NAZARPOUR, K. Deep learning-based artificial vision for grasp classification in myoelectric hands. **Journal of Neural Engineering** 14 036025, 18 p, maio, 2017.

I-LIMB. **Manual da prótese i-limb**. Touch bionics. 2019. Disponível em: <http://touchbionics.com/products/active-prostheses/i-limb-quantum>. Acesso em 20 mar. 2019.

KHEZRI, M.; JAHED, M. Real-time intelligent pattern recognition algorithm for surface EMG signals. **BioMed Eng Online**. v. 6, n. 45, p. 6 – 45, dec. 2007. PMID 18053184.

KOESTER, J. Membrane potencial, In: KANDEL, E. R.; SCHWARTZ, J. H.; JESSELL, T. M. **Principles of neural science**, 3 ed., p-33-94, New York: Elsevier, 1991.

KONRAD, P. **The ABC of EMG**. A practical introduction to kinesiological electromyography. USA. Noraxon INC. 2005.

KUIKEN, T. A.; LI, G.; LOCK, B. A.; LIPSCHUTZ, R. D.; MILLER, L. A.; STUBBLEFIELD, K. A.; ENGLEHART, K. Targeted muscle reinnervation for real-time

myoelectric control of multifunction artificial arms. **JAMA**. p. 619-628. feb. 2009. PMID: 19211469.

KERAS. **Manual do aplicativo Keras**. 2019. Disponível em <https://keras.io>. Acesso em: 20 mar 2019.

LENZ, I.; LEE, H.; SAXENA, A. Deep learning for detecting robotic grasps. **Sage journals**. v. 34, p. 705-724. april, 2015. DOI:10.1177/2F0278364914549607.

LI, C.; REN, J.; HUANG, H.; WANG, B.; ZHU, Y.; HU, H. PCA and deep learning based myoelectric grasping control of prosthetic hand. **Biomed eng online**. v. 17, aug, 2018. PMID: PMC6080221.

LI, Y.; Zhang, Y.; Qiao, H.; CHEN, K.; XI, X. **Grasp type understanding-Classification, localization and clustering**. In: WORLD CONGRESS ON INTELLIGENT CONTROL AND AUTOMATION (WCICA). 12, Jun. 2016. Guilin. China.

MAKHTAR, M.; NEAGU, D. C.; RIDLEY, M. J.; Comparing Multi-class Classifiers: On the similarity of confusion matrices for redictive toxology applications. **IDEAL**. LNCS 6936. p. 252-261. Springer-Verlag Berlim Heidelberg. 2011.

MEDITRACE. **Manual do sensor Meditrace. Covidien**. 2019. Disponível em: <https://cdn.awsli.com.br/463/463473/arquivos/Portfolio%20Novo%20-%20Meditrace.pdf>. Acesso em: 20 mar. 2019.

MERLETTI, R.; HERMENS, H. J. Detection and conditioning of the surface EMG signal. MERLETTI, R.; PARKER, P.; **Electromyography: physiology, engineering, and noninvasive applications**. Nova Jersey: Wiley-Interscience. p.107-128. 2004.

NANDA, Y. What is the VGG neural network? Ago. 2018. **QUORA**. Disponível em: <https://www.quora.com/What-is-the-VGG-neural-network>. Acesso em: 10 fev. 2019

PULLIAM, C. L.; LAMBRECHT, J. M.; KIRSCH, R. F. EMG-based neural network control of trashumeral prostheses. **J Rehabil Res Dev**. p. 739 -754.fev. 2013. PMID: 21938659.

RASPBERRY. **Manual do Raspberry Pi**. 2019. Disponível em: <https://www.raspberrypi.org/products>. Acesso em: 20 mar. 2019.

REAZ, M.B.I.; HUSSAIN, M.S.; YASIN, M. Techniques of EMG signal analysis, detection, processing, classification and applications. **Biol. Proced. Online**. Faculty of Engineering, Multimedia University. Malásia. p.11-35. 2006.

ROBINSON, A. J. Physiology of muscle and nerve, In: ROBINSON, A. J.; SNYDER-MACKLER, L. **Clinical electrophysiology: electrotherapy and electrophysiologic testing**, 2 ed., p. 81-119, Williams & Wilkins, Baltimore, 1995.

RUSSELL, S.; NORVIG, P. **Inteligência Artificial**. Tradução da terceira edição. Rio de Janeiro: Elsevier, 2013.

SCIKIT-LEARN KNN. **Manual do Aplicativo KNN. 2019.** Disponível em: <https://scikit-learn.org/stable/modules/knn>. Acesso em: 18 jul. 2019.

SCIKIT-LEARN LDA. **Manual do Aplicativo LDA. 2019.** Disponível em: https://scikit-learn.org/stable/modules/lda_qda.html#lda-qda. Acesso em: 18 jul. 2019.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large scale image recognition. In: INTERNATIONAL CONFERENCE ON LEARNING REPRESENTATIONS, may 2015, San Diego. **Anais eletrônicos...** San Diego: ICLR, 2015. Disponível em: <https://arxiv.org/pdf/1409.1556.pdf>. Acesso em: 4 jul. 2018.

TENSORFLOW. **Manual do aplicativo Tensorflow. 2019.** Disponível em: <https://www.tensorflow.org>. Acesso em: 10 fev. 2019.

THINGVERSE. Design Community for 3D printing things. 2019. Disponível em: <https://www.thingiverse.com/about/>. Acesso em: 10 maio. 2019.

XAVIER, A. C.; SATO, J. R.; GIRALDI, G. A.; RODRIGUES, P. S. S.; THOMAZ, C. E. Classificação e extração de características discriminantes de imagens 2D de ultrassonografia mamaria. In: NEVES, L. A. P.; NETO, H. V.; GONZAGA, A.; **Avanços em Visão Computacional**. cap 4, p. 65-84, Curitiba: Omnipax, 2012. ISBN 978-85-64619-09-8.

XAVIER, R. T.; BORIS, F. A.; CASTRO, F. R.; BLANCO, J. E.; GIOVANINI, R. M.; SANCHES, M. A. A.; CARVALHO, A. A. Prótese de membro superior com movimento pré-definidos pelo usuário. In: CONGRESSO BRASILEIRO DE ENGENHARIA BIOMÉDICA XXV, out, 2016, Foz do Iguaçu. Anais... Foz do Iguaçu: CBEB 2016, p. 856. ISSN: 2359-3164.

ZHAI, X.; JELFS, B.; CHAN, R.H.M.; TIN, C. Self-Recalibrating surface EMG pattern recognition for neuroprosthesis control based on convolutional neural network. **Front Neurosci**. v. 11, n. 379, 17 p, july. 2017. PMID. 28744189.

APÊNDICE A – LITERATURA COMPLEMENTAR

APÊNDICE A – LITERATURA COMPLEMENTAR

BACKES, A. R.; SÁ JUNIOR, J. J. M. **Introdução à visão computacional usando MATLAB**. Rio de Janeiro: Alta books. 2016.

BARELLI, F. **Introdução à visão computacional: Uma abordagem prática com Python e OpenCV**. São Paulo: Casa do Código. 2018.

BARRY, P. **Use a cabeça! Python**. Tradução PINTO, R. A. Rio de Janeiro: Alta Books. 2012.

GRAETZ, F. M. **How to visualize convolutional features in 40 lines of code**. Disponível em: <https://towardsdatascience.com/how-to-visualize-convolutional-features-in-40-lines-of-code-70b7d87b0030>. Acesso em: jun.2019.

MACHINE LEARNING. **Image Classification using Python and Scikit-learn**. Disponível em: <https://gogul.dev/software/image-classification-python>. Acesso em: 20 de jun. 2019.

MACHINE LEARNING MASTERY. **How to visualize filters and features maps in convolutional neural networks**. Disponível em: <https://machinelearningmastery.com/how-to-visualize-filters-and-feature-maps-in-convolutional-neural-networks/>. Acesso em: ago. 2019.

MOLLOY, D. **Exploring raspberry pi: Interfacing to the real world with embedded linux**. Indiana: Wiley. 2016.

MONK, S. **Programando o raspberry pi: Primeiros passos com Python**. São Paulo: Novatec. 2013.

MORITANI, T.; STEGEMAN, R.; MERLETTI, R. Basic physiology and biophysics of EMG signal generation. In: MERLETTI, R.; PARKER, P.; **Electromyography: physiology, engineering, and noninvasive applications**. Nova Jersey: Wiley-Interscience. 2004. p. 1-20.

MUELLER, J. P. **Começando a programar em Python para leigos**. Tradução CONVERSO, G. M. Rio de Janeiro: Alta Books. 2016.

NORRIS, D. J. **Beginning artificial intelligence with the raspberry pi**. Nova Iorque: Apress. 2017.

PAJANKAR, A. **Raspberry pi image processing programming: develop real-life examples with Python, Pillow and SciPy**. Nova Iorque: Apress. 2017.

RICARDSON, M.; WALLACE, S. **Primeiros passos com o raspberry pi**. Tradução MAEDA, P. T. São Paulo: Novatec. 2013.

ROSEBROCK, A. **Convolutions with OpenCV and Python**. jul. 2016. Disponível em: <https://www.pyimagesearch.com/2016/07/25/convolutions-with-opencv-and-python/>. Acesso em: 6 out. 2018.

ROSEBROCK, A. **How to create a deep learning dataset using Google images**. set. 2018. Disponível em: <https://www.pyimagesearch.com/2017/12/04/how-to-create-a-deep-learning-dataset-using-google-images/>. Acesso em: 6 out. 2018.

ROSEBROCK, A. **Imagenet classification with Python and Keras**. ago. 2016. Disponível em: <https://www.pyimagesearch.com/2016/08/10/imagenet-classification-with-python-and-keras/>. Acesso em: 10 fev. 2019.

ROSEBROCK, A. **Install OpenCV 4 on macOS**. ago. 2018. Disponível em: <https://www.pyimagesearch.com/2018/08/17/install-opencv-4-on-macos/>. Acesso em: 7 out. 2018.

ROSEBROCK, A. **Keras and deep learning on the raspberry pi**. dez. 2017. Disponível em: <https://www.pyimagesearch.com/2017/12/18/keras-deep-learning-raspberry-pi/>. Acesso em: 27 out. 2018.

ROSEBROCK, A. **Keras and convolutional neural networks: CNNs**. abr. 2018. Disponível em: <https://www.pyimagesearch.com/2018/04/16/keras-and-convolutional-neural-networks-cnns/>. Acesso em: 17 fev. 2019.

ROSEBROCK, A. **Keras tutorial: How to get started with Keras, deep learning, and Python**. set. 2018. Disponível em: <https://www.pyimagesearch.com/2018/09/10/keras-tutorial-how-to-get-started-with-keras-deep-learning-and-python/>. Acesso em: 10 fev. 2019.

ROSEBROCK, A. **Lenet-convolutional neural networking Python**. ago. 2016. Disponível em: <https://www.pyimagesearch.com/2016/08/01/lenet-convolutional-neural-network-in-python/>. Acesso em: 10 fev. 2019.

ROSEBROCK, A. **Optimizing OpenCV on the raspberry pi**. out. 2017. Disponível em: <https://www.pyimagesearch.com/2017/10/09/optimizing-opencv-on-the-raspberry-pi/>. Acesso em: 4 jul. 2018.

ROSEBROCK, A. **Real-time object detection with deep learning an OpenCV**. set. 2017. Disponível em: https://www.pyimagesearch.com/2017/09/18/real-time-object-detection-with-deep-learning-and-opencv/?__s=vdxpc7ru8meodqt4bf9q. Acesso em: 4 jul. 2018.

ROSEBROCK, A. **Unifying picamera and cv2.VideoCapture into a single class with OpenCV**. jan. 2016. Disponível em: <https://www.pyimagesearch.com/2016/01/04/unifying-picamera-and-cv2-videocapture-into-a-single-class-with-opencv/>. Acesso em: 4 jul. 2018.

SEBELIUS, F.; AXELSSON, M.; DANIELSEN, N.; SCHOUENBORG, J.; LAURELL, T. **Real-time control of a virtual hand**. Department of Electrical Measurements, Lund Institute of Technology. Suécia. [ca. 2010]

UPTON, E.; HAFACREE, G. **Raspberry pi: Manual do usuário**. Tradução PASCHOA, C. R. São Paulo: Novatec. 2013

**APÊNDICE B – PROGRAMA PARA CONTAGEM DOS PULSOS DOS SENSORES
DE sEMG**

APÊNDICE B – PROGRAMA PARA CONTAGEM DOS PULSOS DOS SENSORES DE sEMG

```
int currentVoltage = analogRead(A0);

if (currentVoltage > voltageThreshold){
  flag = HIGH;
  else {flag = LOW; }

  if(flag ==HIGH)

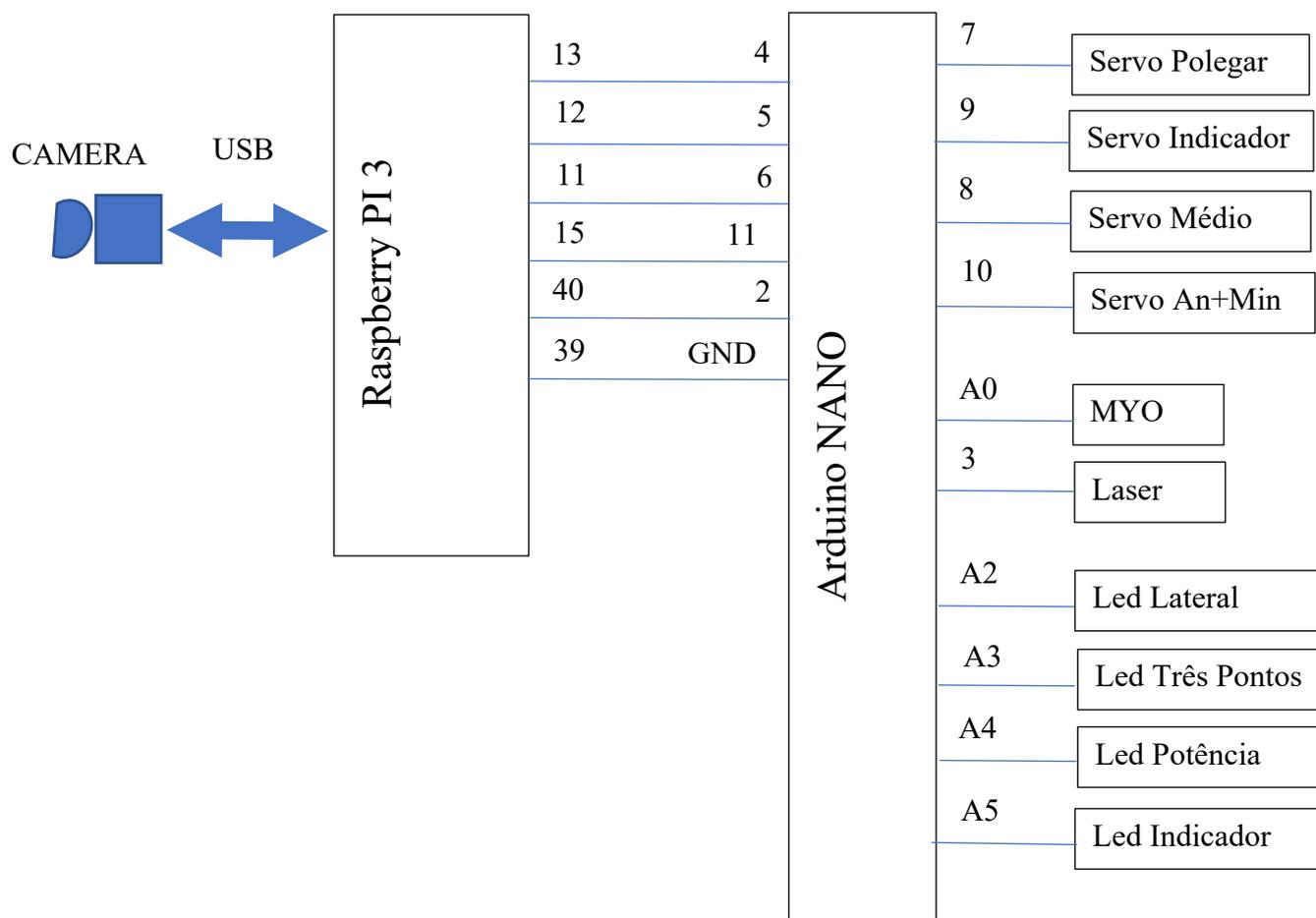
  digitalWrite(onboardLED, HIGH);
  delay(100);
  if ( aux == LOW)
  contador = contador + 1 ;
  delay(1000);

  timerSafety.onTimeout(3000);
  aux = HIGH;

  else
  aux = LOW;
  if (timerSafety.onTimeout(3000)) {
  digitalWrite(onboardLED, LOW);
  contador =0;
```

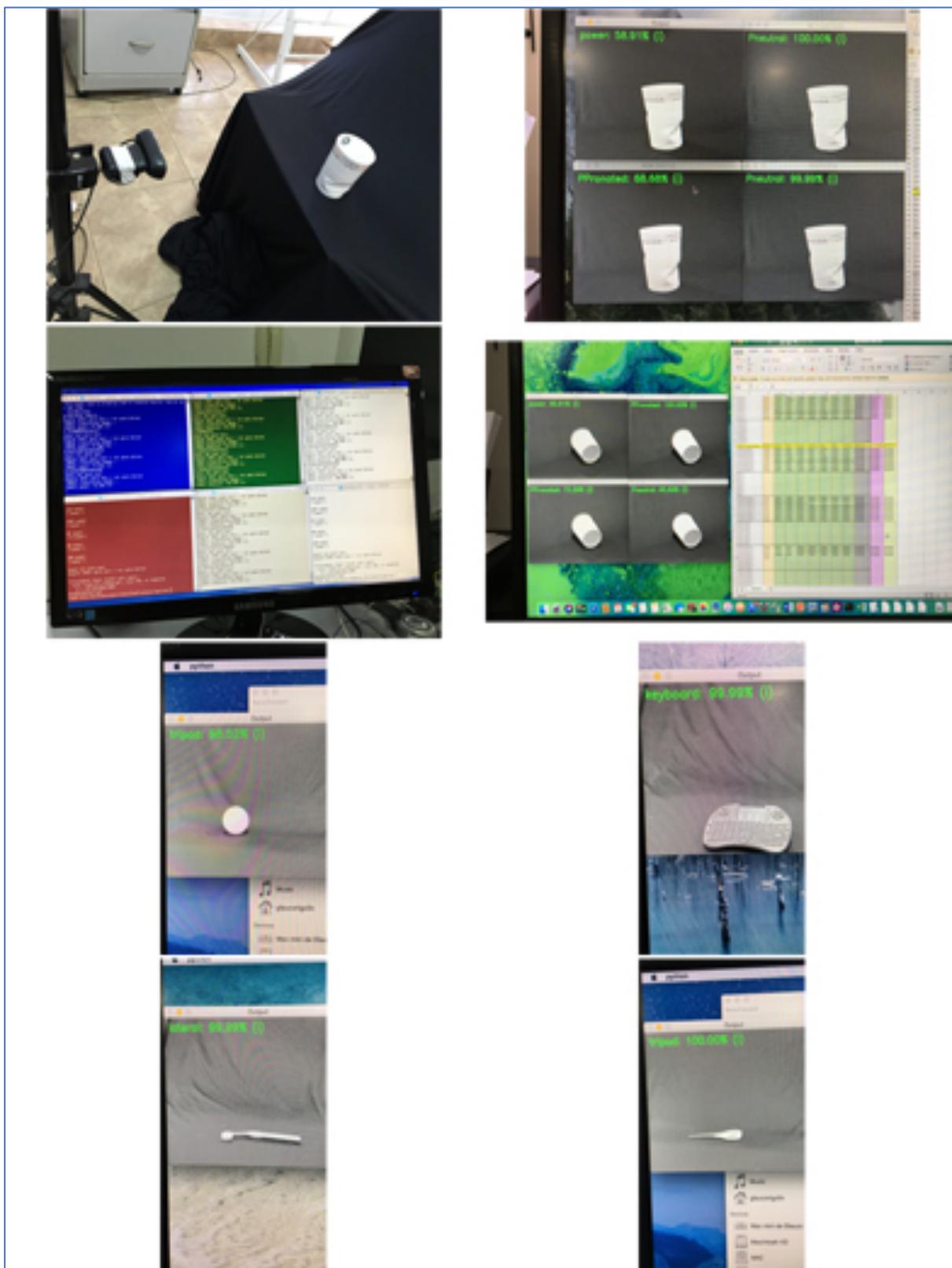
APÊNDICE C – DIAGRAMA DE LIGAÇÃO

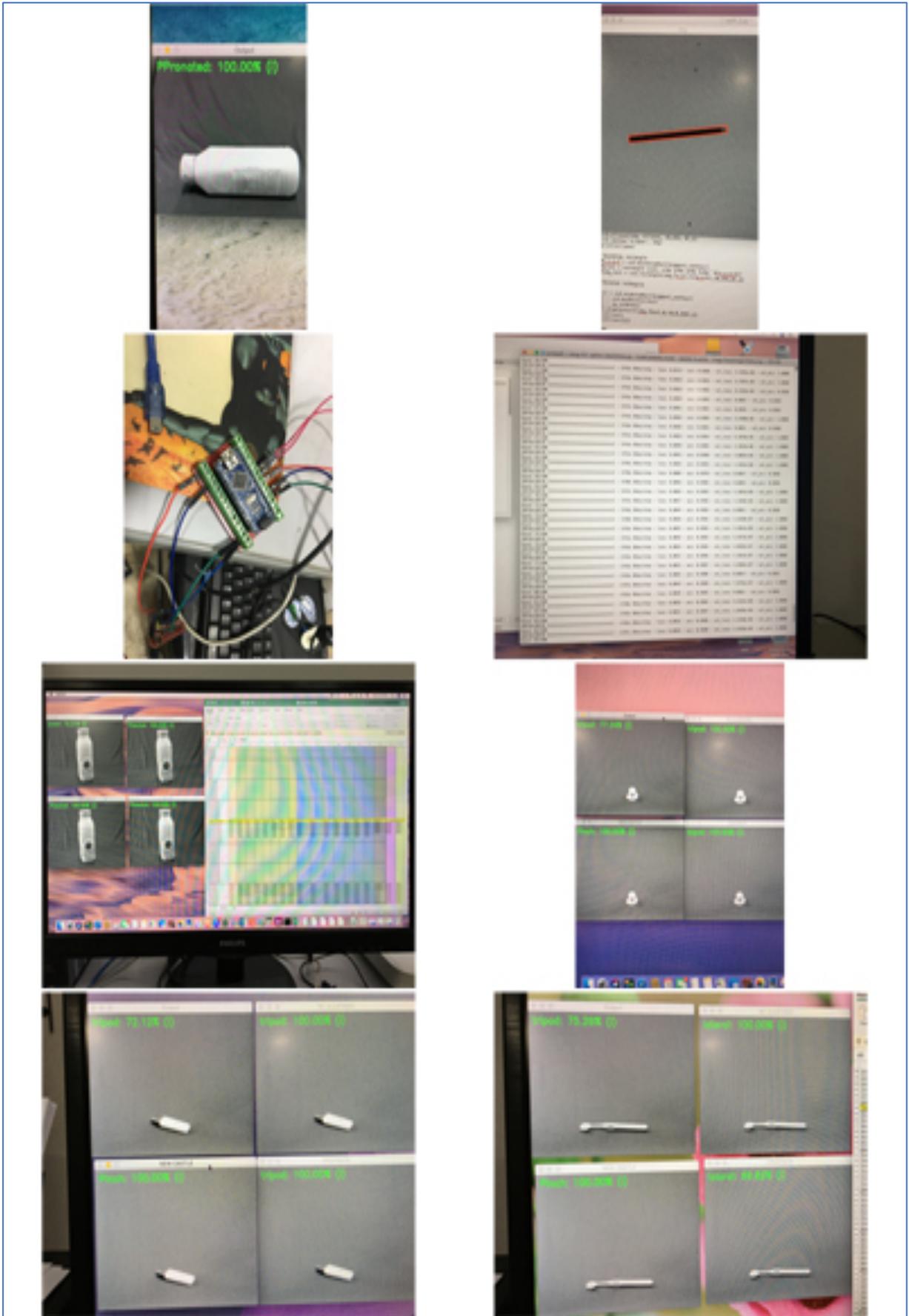
APÊNDICE C – DIAGRAMA DE LIGAÇÃO

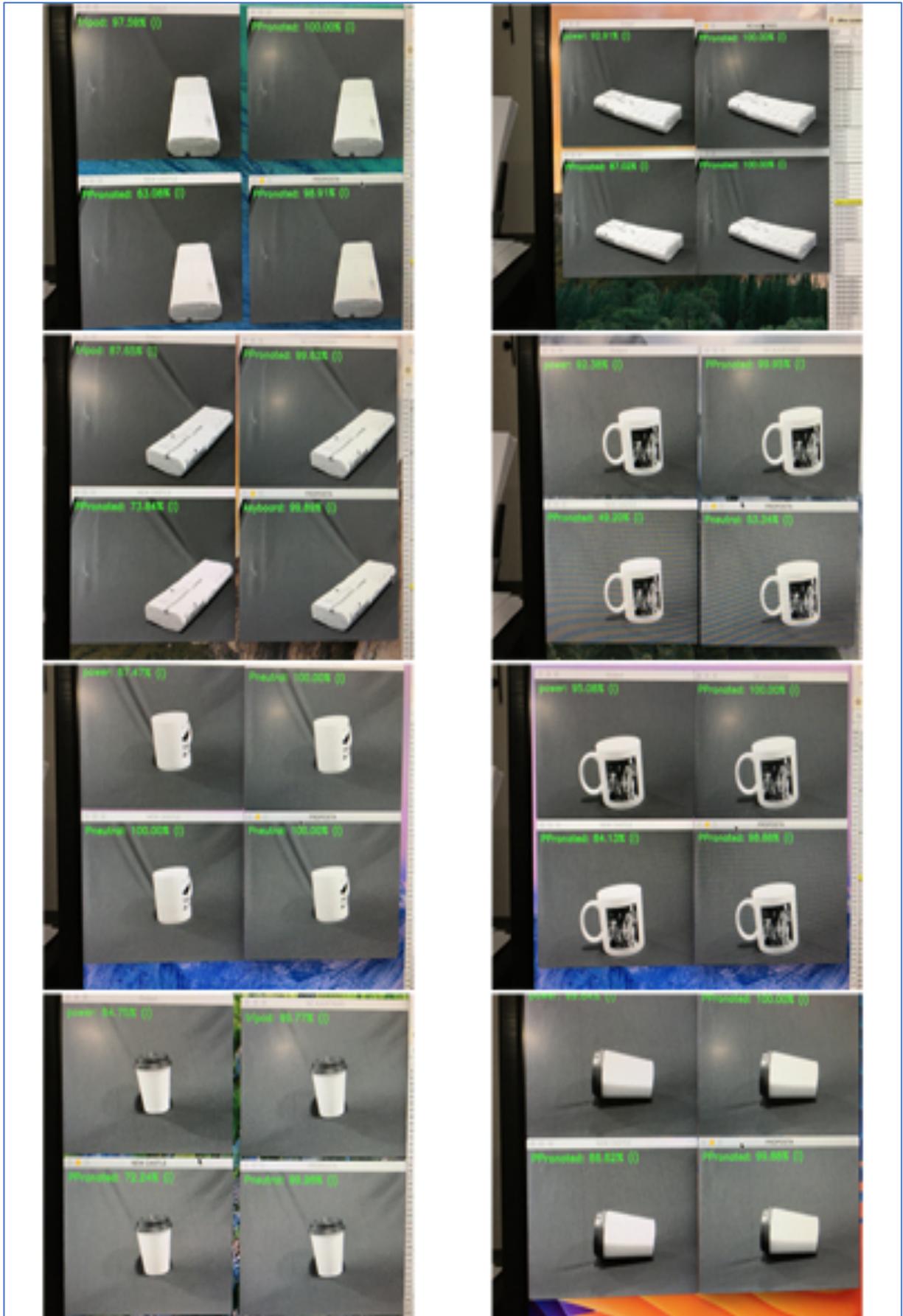


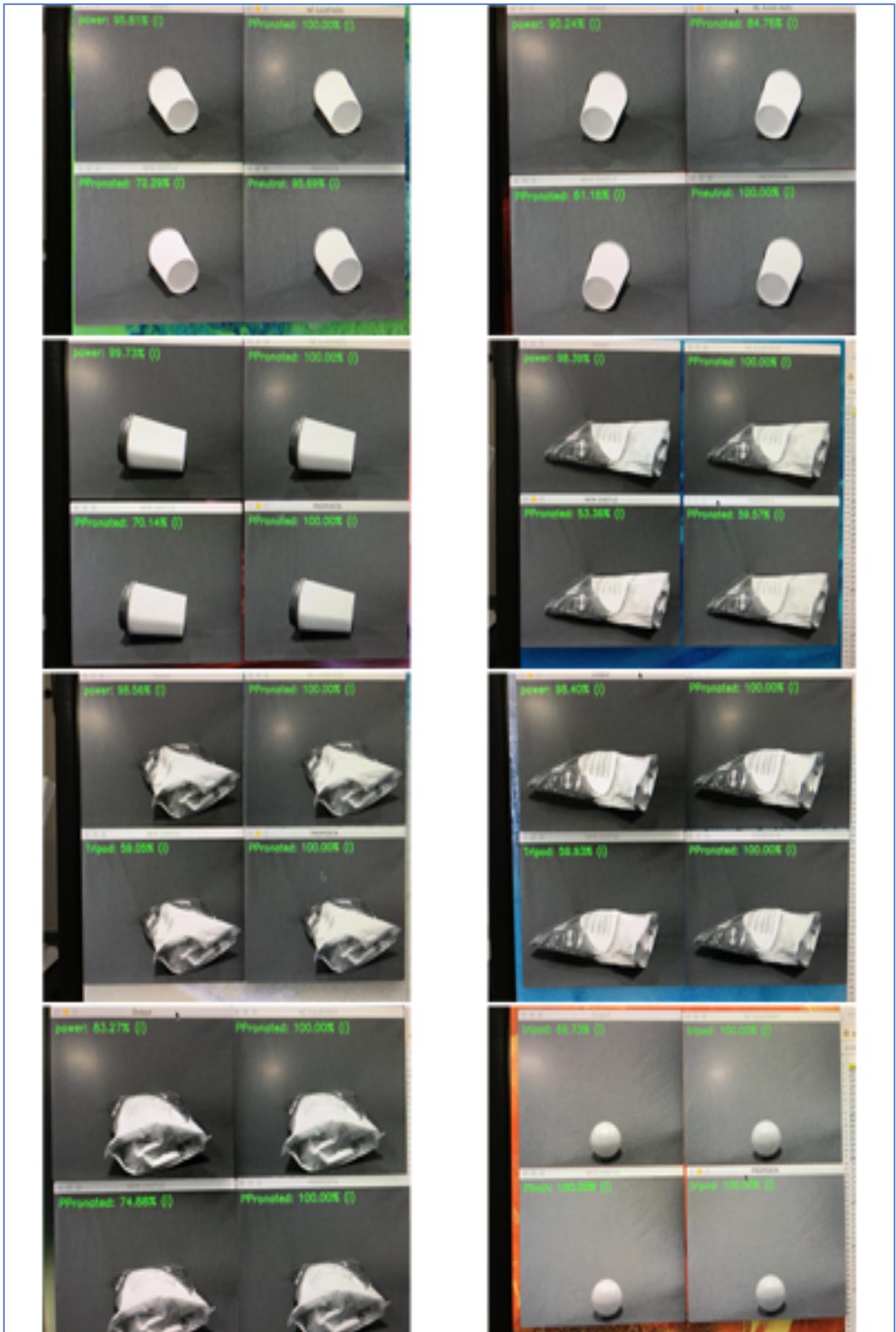
APÊNDICE D – FOTOGRAFIAS DOS EXPERIMENTOS

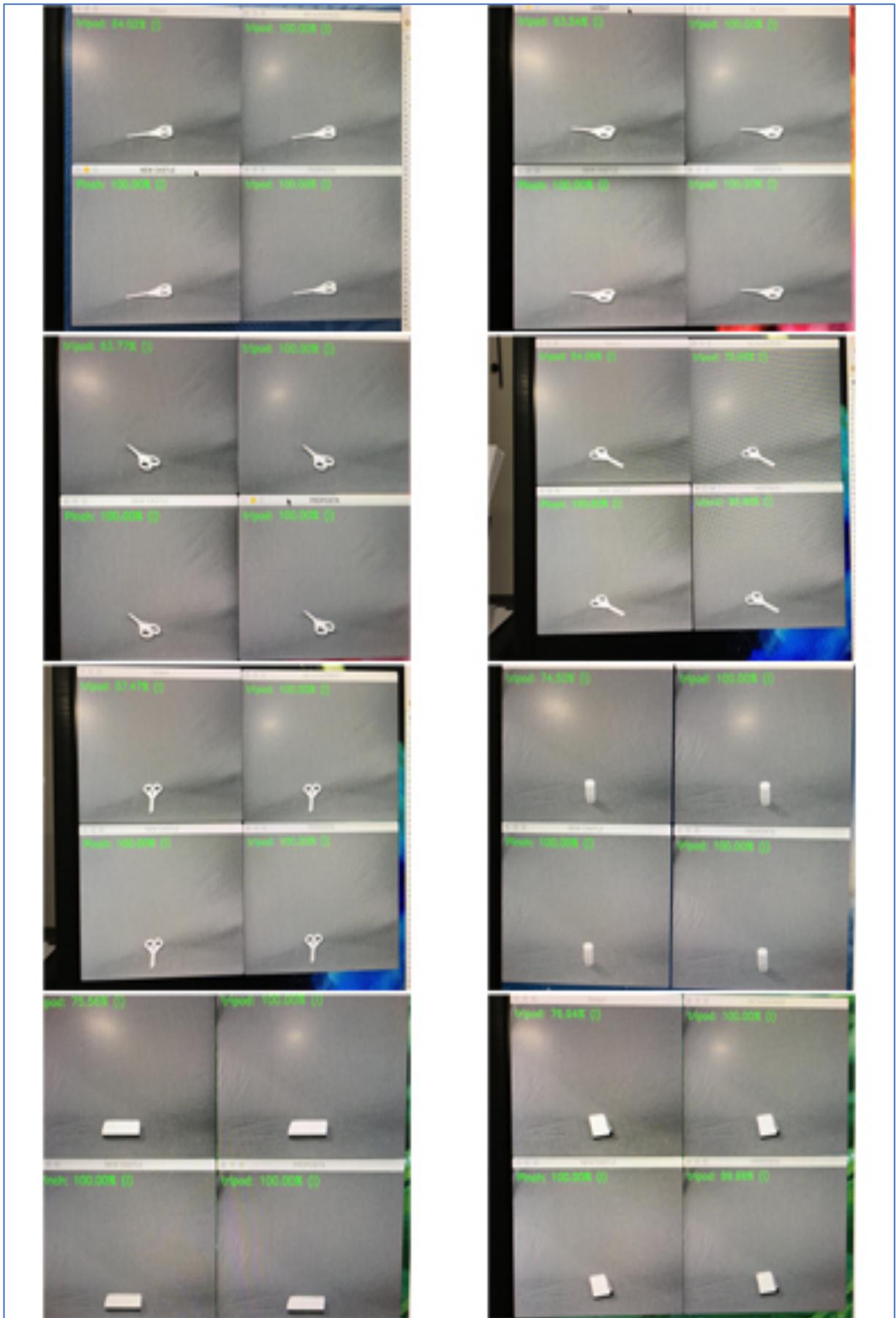
APÊNDICE D – FOTOGRAFIA DOS EXPERIMENTOS

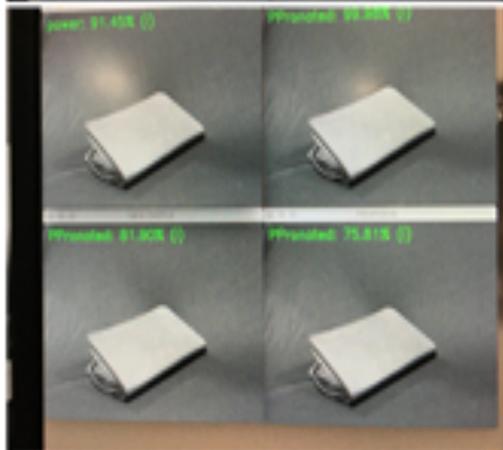
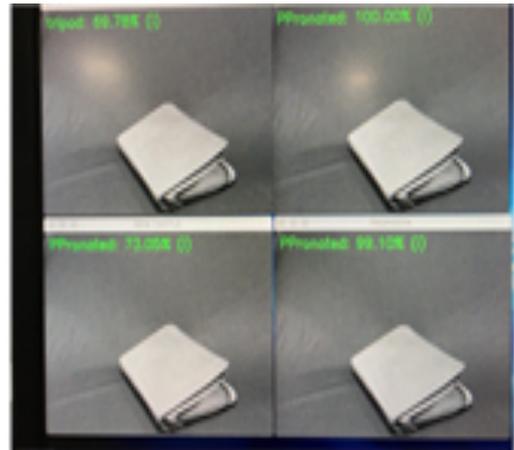


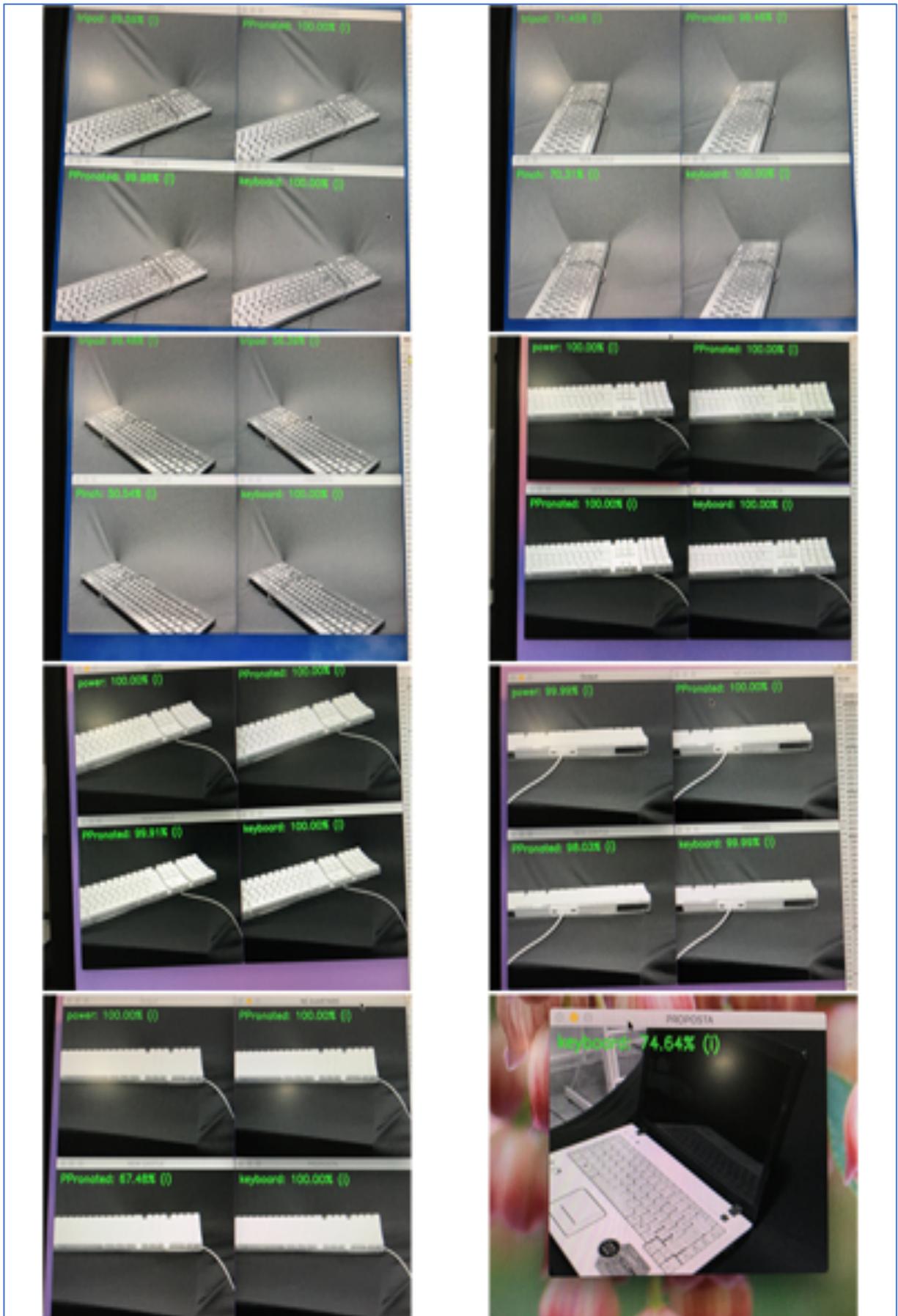




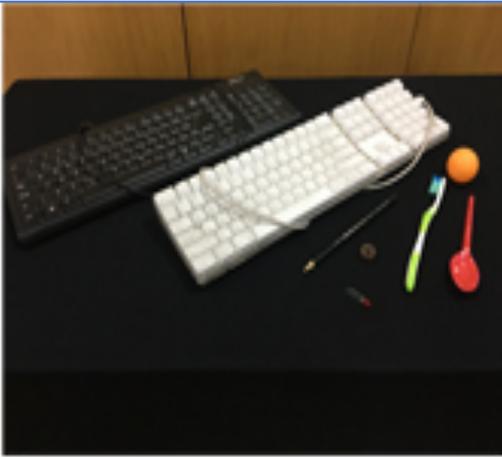












**ANEXO A – PROGRAMA DA REDE NEURAL MODIFICADA, CONFORME
ROSEBROCK (2018).**

ANEXO A – PROGRAMA DA REDE NEURAL MODIFICADA, CONFORME ROSEBROCK (2018).

```
model.add(Conv2D(32, (3, 3), padding="same",
input_shape=inputShape))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(MaxPooling2D(pool_size=(3, 3)))
model.add(Dropout(0.25))
model.add(Conv2D(64, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(Conv2D(64, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Conv2D(128, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(Conv2D(128, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Flatten())
model.add(Dense(1024))
model.add(Activation("relu"))
model.add(BatchNormalization())
model.add(Dropout(0.5))
model.add(Dense(classes))
model.add(Activation("softmax"))
```