UNIVERSIDADE FEDERAL DO CEARÁ

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO

MESTRADO ACADÊMICO EM ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO

RAIMUNDO FARRAPO PINTO JÚNIOR

RECONHECIMENTO DE GESTOS ESTÁTICOS UTILIZANDO REDES NEURAIS CONVOLUCIONAIS

SOBRAL - CE

2019

RAIMUNDO FARRAPO PINTO JÚNIOR

RECONHECIMENTO DE GESTOS ESTÁTICOS UTILIZANDO REDES NEURAIS CONVOLUCIONAIS

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica e de Computação da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Engenharia Elétrica e de Computação. Área de Concentração: Sistemas de Informação

Orientador: Prof. Dr. Iális Cavalcante de Paula Júnior

SOBRAL - CE

2019

RAIMUNDO FARRAPO PINTO JÚNIOR

RECONHECIMENTO DE GESTOS ESTÁTICOS UTILIZANDO REDES NEURAIS CONVOLUCIONAIS

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica e de Computação da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Engenharia Elétrica e de Computação. Área de Concentração: Sistemas de Informação

Aprovada em:

BANCA EXAMINADORA

Prof. Dr. Iális Cavalcante de Paula Júnior (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. Jarbas Joaci de Mesquita Sá Júnior
Universidade Federal do Ceará (UFC)

Prof. Dra. Elloá Barreto Guedes da Costa
Universidade do Estado do Amazonas (UEA)

Aos meus pais, Raimundo e Gracilda, meu irmão, Vitor, e minha amada, Larissa.

AGRADECIMENTOS

Agradeço a Deus, por sempre iluminar meus caminhos, me conduzir aos planos corretos e guiar meu coração.

Ao meu orientador Prof. Iális Cavalcante, pela sua dedicação, amizade e agregar inúmeros valores a minha vida acadêmica e profissional. O seu trabalho para o desenvolvimento dos alunos, pesquisas e do meio acadêmico é inspirador.

A minha amada Larissa, que conviveu durante toda a minha jornada acadêmica ao longo de 7 anos. Seus conselhos, conversas e apoio em momentos bons e difíceis foram essenciais para meu amadurecimento. Sou grato a Deus pela pessoa maravilhosa que você é e representa na minha vida.

Aos meus pais e ao meu irmão Vitor, pelo amor incondicional e por todo o esforço durante toda a minha estudantil e acadêmica. Eu possuo uma eterna gratidão pela dedicação de vocês.

Aos meus grandes amigos da UFC (Brena Lima, Danilo Oliveira, David Borges, Isaac Ben, João Rafael, Márcio Albuquerque, Saulo Cunha, Syllas Rangel e Robson Couto), pela forte amizade e os momentos inesquecíveis nos corredores e laboratórios.

Ao meu colega de trabalho e grande amigo Macário Martins, pelo apoio durante os momentos finais do mestrado.

A Fundação Cearense de Apoio ao Desenvolvimento Científico e Tecnológico (FUNCAP), pelo apoio financeiro durante o desenvolvimento desse trabalho.

Por fim, a Universidade Federal do Ceará (UFC), que foi um segundo lar ao longo de 7 anos (graduação e mestrado). Quero sempre que possível contribuir para o seu desenvolvimento.

"As moléculas da vida enchem o cosmos".

(Carl Sagan)

RESUMO

Os gestos humanos, além de complementarem a maneira como nos comunicamos e expressamos ideias, podem ser utilizados como ações específicas em um determinado meio de interação humano-máquina, como na manipulação de um computador diante de uma câmera. Para que esse processo funcione de modo satisfatório, é necessário seguir alguns passos, como capturar imagens, separar e definir os objetos de interesse, extrair informações relevantes e classificar dados conforme outros previamente conhecidos. Esta dissertação propõe um método de reconhecimento de gestos usando redes neurais convolucionais. O procedimento utiliza técnicas de processamento de imagens, como segmentação por cor, aplicação de filtros morfológicos e extratores de contornos e, para a extração de características e classificação, adota-se o uso de redes neurais convolucionais. As imagens utilizadas para a obtenção dos resultados são provenientes de duas bases de imagens principais com 24 gestos ASL, uma base própria construída durante o desenvolvimento deste trabalho e outra base já conhecida na literatura. Os resultados foram obtidos e comparados com diversas métricas de avaliação e demonstraram-se como promissores para a classificação de imagens de gestos estáticos. Esses resultado são comparados com outros trabalhos e arquiteturas de CNN conhecidas da literatura.

Palavras-chave: Processamento Digital de Imagens. Reconhecimento de Gestos. Redes Neurais Convolucionais.

ABSTRACT

Human gestures can complement the way we communicate and express ideas, they can be used as specific actions in a certain type of human-machine interaction, such as when manipulating a computer in front of a camera. For this process to work correctly, it is necessary to follow some steps, such as capturing images, separating and defining the objects of interest, extracting relevant information and classifying data. This dissertation proposes a method of gesture recognition using convolutional neural networks. The method uses image processing techniques, such as color segmentation, application of morphological filters and contour extractors and, for the extraction of characteristics and classification, the use of convolutional neural networks is adopted. The images used are obtained from two main image datasets with 24 ASL gestures. An own base built during the development of this work and another base already available in the literature. The results were obtained and compared with some evaluation metrics and proved to be promising for the classification of images of static gestures. These results are compared with other CNN architectures and another methods known from the literature.

Keywords: Digital Image Processing. Gesture Recognition. Convolutional Neural Network.

LISTA DE FIGURAS

Figura 1 – Exemplo de uma erosão, em que *a* é a imagem original, *b* o elemento estruturante e *c* o resultado 18
Figura 2 – Exemplo de uma dilatação, em que *a* é a imagem original, *b* o elemento estruturante e *c* o resultado 18
Figura 3 – Exemplo de uma aproximação poligonal, detalhes em verde 20
Figura 4 – Exemplo de uma MLP 21
Figura 5 – Exemplo de uma CNN e suas camadas. 22
Figura 6 – Exemplo de uma segmentação utilizando a técnica de subtração de *background* 24
Figura 7 – Exemplo de uma segmentação utilizando a técnica de segmentação por cor . 24
Figura 8 – Fluxograma da metodologia proposta 27
Figura 9 – Amostras dos gestos da base construída 29
Figura 10 – Amostras dos gestos da base (BARCZAK *et al.*, 2011) 30
Figura 11 – Amostras para o gesto A 30
Figura 12 – Amostras da base de imagens utilizada. 31
Figura 13 – Amostras das imagens utilizadas na camada de entrada da MLP 31
Figura 14 – Imagens das máscaras utilizadas para treino da MLP 32
Figura 15 – Imagens antes e depois da segmentação por cor, respectivamente 32
Figura 16 – Imagens antes e depois da operação de erosão, respectivamente 33
Figura 17 – Imagens antes e depois da operação de fechamento, respectivamente 33
Figura 18 – Imagem com o contorno gerado e a máscara final, respectivamente 34
Figura 19 – Imagens segmentadas com formas similares 35
Figura 20 – Imagens segmentadas com formas similares 35
Figura 21 – Gestos após a operação lógica AND com a imagens original da mão em cinza 36
Figura 22 – Gestos após a operação lógica AND com a imagens original da mão em cinza 36
Figura 23 – Taxas de acurácia e perda durante o treinamento e validação da CNN 1 . . . 39
Figura 24 – Taxas de acurácia e perda durante o treinamento e validação da CNN 3 . . . 39
Figura 25 – Taxas de acurácia e perda durante o treinamento e validação da InceptionV3 41
Figura 26 – Taxas de acurácia e perda durante o treinamento e validação da ResNet50 . 41

LISTA DE TABELAS

Tabela 1 – Matriz de confusão 24
Tabela 2 – Arquitetura das Rede Neural Convolucional (*Convolutional Neural Network*) (*CNN*) propostas 28
Tabela 3 – Resultados obtidos com as arquiteturas de CNN propostas 38
Tabela 4 – Resultados obtidos com as arquiteturas de CNN da literatura 40
Tabela 5 – Resultados obtidos para cada base de imagens individualmente, utilizando as arquiteturas propostas 42
Tabela 6 – Resultados obtidos para cada base de imagens individualmente, utilizando as arquiteturas disponíveis na literatura 42
Tabela 7 – Comparação dos resultados obtidos com trabalhos relacionados 43

LISTA DE ABREVIATURAS E SIGLAS

ASL	*American Sign Language*
CNN	Rede Neural Convolucional (*Convolutional Neural Network*)
FFCNN	*Feature Fusion-based Convolutional Neural Network*
KNN	*K-Nearest Neighbors*
LVQ	*Learning Vector Quantization*
MLP	Perceptron Multicamadas (*Multilayer Perceptron*)
SVM	Máquina de Vetor de Suporte (*Support Vector Machine*)
LIBRAS	Língua Brasileira de Sinais
PDI	Processamento Digital de Imagens
RNA	Rede Neural Artificial

SUMÁRIO

1	**INTRODUÇÃO**	12
1.1	**Objetivos**	15
1.1.1	***Objetivos Específicos***	15
1.2	**Organização**	15
2	**MATERIAIS E MÉTODOS**	17
2.1	**Operações Morfológicas**	17
2.2	**Extração de Contornos e Aproximação Poligonal**	19
2.3	**Aprendizagem profunda (*Deep learning*)**	20
2.3.1	***Redes neurais artificiais***	20
2.3.2	***Redes Neurais Convolucionais***	21
2.4	**Segmentação por Cor**	23
2.5	**Métricas de Avaliação**	24
2.6	**Considerações finais**	25
3	**METODOLOGIA**	26
3.1	**Considerações finais**	27
4	**EXPERIMENTOS**	29
4.1	**Considerações finais**	37
5	**RESULTADOS**	38
5.1	**Considerações finais**	43
6	**CONCLUSÃO**	44
6.1	**Perspectivas Futuras**	44
6.2	**Produções Bibliográficas**	45
	REFERÊNCIAS	46

1 INTRODUÇÃO

O ser humano é capaz de analisar, filtrar e reconhecer gestos corporais sem maiores dificuldades, esteja ele em diversas situações. A razão pela qual isso é possível deve-se à combinação dos sinais da visão humana com as interações sinápticas que ocorrem ao longo do desenvolvimento do cérebro, ou seja, seus estímulos durante o crescimento do ser humano (COUNCIL *et al.*, 2000). Os gestos humanos corporais, além de complementarem a maneira como nos comunicamos e expressamos ideias, podem ser utilizados para compor um idioma, por exemplo: a Língua Brasileira de Sinais (LIBRAS) (MACHADO, 2017) ou a *American Sign Language* (*ASL*) (INSTITUTE, 2019). Esses gestos podem ser utilizados como ações específicas em um determinado meio de interação humano-máquina, por exemplo na manipulação de um computador diante de uma câmera. Para que esse processo funcione de modo satisfatório é necessário seguir alguns passos, como: capturar imagens, separar e definir os objetos de interesse, extrair informações relevantes, classificar dados conforme outros previamente conhecidos e os demais possíveis passos (PISHARADY; SAERBECK, 2015).

A evolução da computação e a facilidade no acesso de novas tecnologias motivou o desenvolvimento de equipamentos nessa área de reconhecimento de gestos integrado com computadores e demais equipamentos, como *Kinect* e *Leep Motion*, que são exemplos de inovação em tecnologias de dispositivos de entrada e captura de dados (COHEN *et al.*, 2018) (BETANCOURT *et al.*, 2015) (RIOFRÍO *et al.*, 2017). Desse modo, esses dispositivos são capazes de capturar e interpretar gestos humanos, proporcionando uma nova possibilidade de interação humano-máquina (GORECKY *et al.*, 2014). As utilizações desses dispositivos estão presentes nas mais diversas áreas, como: Robótica, Medicina, tradução de língua de sinais, Computação Gráfica e Realidade Aumentada (TECHNAVIO, 2015) (LANGE *et al.*, 2012) (SHERMAN; CRAIG, 2018).

As metodologias presentes na literatura de reconhecimento de gestos são regularmente divididas em duas categorias, relacionando-se com os tipos de gestos utilizados, que podem ser: estáticos ou dinâmicos (PISHARADY; SAERBECK, 2015). Os gestos estáticos são definidos como aqueles que podem ser descritos somente em uma imagem (ou *frame* de um vídeo) para a extração de características e classificação. Para os gestos dinâmicos, um conjunto ou uma sequência de imagens é necessária para representar um único gesto durante o seu reconhecimento no classificador. Portanto, no uso de gestos estáticos um menor custo computacional é exigido por analisar imagens isoladamente durante a classificação, diferentemente dos gestos

dinâmicos no qual exige-se um elevado poder computacional devido ao elevado número de dados que costumam ser obtidos das inúmeras imagens.

Além da divisão em relação aos tipos dos gestos, os algoritmos classificadores também são caracterizados em dois grupos levando em consideração o seu tipo de aprendizado, que podem ser: supervisionado ou não-supervisionado. Na aprendizagem supervisionada, os dados são conhecidos e rotulados previamente, o que torna possível treinar um algoritmo com os dados de características e suas respectivas classes. Esse fato permite presumir a classe para uma nova amostra. Na aprendizagem não-supervisionada, os dados das amostras são conhecidos, porém a identificação ou quantidade de classes desses dados são desconhecidas. Dessa maneira, os algoritmos não-supervisionados, como, por exemplo, o *K-Means*, descrevem os dados conforme a organização dos mesmos.

Na literatura, podemos encontrar inúmeras metodologias de reconhecimento de gestos utilizando aprendizado supervisionado e/ou não-supervisionado. Podemos citar alguns exemplos, como: redes neurais (YANG; TABB, 2002) (ELSOUD; ELNASER, 2009) (NGUYEN *et al.*, 2015), *CNN* (OYEDOTUN; KHASHMAN, 2017), Máquina de Vetor de Suporte (*Support Vector Machine*) (*SVM*) (HUANGA; CHANGA, 2011) (OTINIANO-RODRÍGUEZ, 2012), *nearest neighbors* (BERGH *et al.*, 2011), grafos (TRIESCH; MALSBURG, 2001), *distributed locally linear embeddings* (GE *et al.*, 2006) e outros (PISHARADY; SAERBECK, 2015).

Em (OTINIANO-RODRÍGUEZ, 2012), são apresentadas duas metodologias de reconhecimento de gestos que utilizam o *SVM* como classificador e as características são extraídas utilizando os momentos de Hu e os de Zernike. A base de imagens utilizada contém 2040 imagens e é composta por 24 gestos estáticos em *ASL*. Em seus resultados é apresentado que a metologia que utiliza as características dos momentos de Zernike possui maior acurácia em comparação aos momentos de Hu.

Em (ELSOUD; ELNASER, 2009), um método de reconhecimento de gestos utilizando um tipo especial de rede neural denominado *Learning Vector Quantization* (*LVQ*) é apresentado. As características utilizadas são extraídas utilizando a Transformada e Projeção Discreta do Cosseno. Os resultados são obtidos utilizando uma base de imagens de gestos estáticos e apresentam variação de iluminação, rotação, translação e cores nas imagens.

Em (NGUYEN *et al.*, 2015), é apresentada uma metodologia que detecta as regiões de mãos nas imagens e efetua os pré-processamentos necessários para determinar as características. Essas mesmas são extraídas através do uso de autovetores e autovalores e que a partir desses

dados é realizada a Análise de Componentes Principais (*Principal Component Analysis* - PCA) para a seleção dos melhores atributos. Por fim, os dados são classificados utilizando uma rede neural Perceptron Multicamadas (*Multilayer Perceptron*) (*MLP*).

Em (OYEDOTUN; KHASHMAN, 2017) é utilizado *Deep Learning* para reconhecimento de gestos estáticos *ASL*. Sua metodologia apresenta o uso de diversas técnicas de processamento de imagens para extrair as melhores formas de todas as imagens da base utilizada. Após esse passo de pré-processamento dos dados, as imagens segmentadas com as formas geradas são utilizadas e comparadas em duas técnicas de classificação, que são as redes neurais convolucionais e o *stacked denoising autoencoder*. Esses dois métodos utilizam a própria imagem como atributo de entrada dos classificadores.

Em (CHEVTCHENKO *et al.*, 2018) é apresentada uma metodologia que combina o uso de redes neurais convolucionais com outras técnicas de extração de características conhecidas da literatura, que são: características de Gabor, momentos de Zernike, momentos de Hu e descritores baseados no contorno. Assim, os autores propõem o que foi denominado por Redes Neurais Convolucionais baseadas em fusão de características *Feature Fusion-based Convolutional Neural Network* (*FFCNN*), para classificar imagens estáticas de gestos *ASL* binarizadas e em tons de cinza. Além disso, é apresentada uma proposta para reconhecimento dos gestos em tempo real.

Em (RANGA *et al.*, 2018) uma metodologia de extração de características de imagens coloridas utilizando o filtro híbrido *Discrete Wavelet Transform-Gabor* é apresentada. Os resultados são obtidos a partir do uso de diversos classificadores, como: Floresta Aleatória, *SVM*, *K-Nearest Neighbors* (*KNN*) e *CNN*, essa última apresentou taxas de acertos superiores às outras técnicas.

Neste trabalho, a metodologia proposta nos capítulos seguintes utiliza técnicas de processamento de imagens, como: segmentação por cor, aplicação de filtros morfológicos e extratores de contornos, e para a extração de características e classificação adota-se o uso de redes neurais convolucionais. As imagens utilizadas para a obtenção dos resultados são proveniente de duas bases de imagens principais com 24 gestos *ASL*, uma base própria construída durante o desenvolvimento deste trabalho (PINTO JÚNIOR; DE PAULA JÚNIOR, 2019) e outra base já conhecida na literatura (BARCZAK *et al.*, 2011). Adicionalmente, uma terceira base de imagens foi utilizada (MOESLUND'S, 2002), para fins de comparação com os resultados adquiridos das duas bases principais. Desse modo, é possível demonstrar a robustez de arquiteturas simples de

redes neurais convolucionais juntamente com um conjunto de técnicas de Processamento Digital de Imagens (PDI), pois os resultados obtidos são promissores para a classificação de imagens de gestos estáticos. Esses resultados apresentados são comparados com diversas arquiteturas de *CNN* conhecidas da literatura e outras metodologias.

1.1 Objetivos

O objetivo desta dissertação está no uso de um conjunto de técnicas de processamento de imagens, que proporcionam em imagens de gestos de mãos a remoção de informações não relevantes e no realce das características de suas formas e informações de intensidade dos conjuntos de pixels que geram essa forma.

1.1.1 Objetivos Específicos

- Propor uma metodologia de pré-processamento de imagens capaz de remover informações não relevantes para o reconhecimento de gestos, além de remover ruídos e realçar as regiões de mãos e seus detalhes nas imagens;
- Propor arquiteturas de *CNN* que possuam um menor custo computacional e apresentem valores satisfatórios de acurácia para as bases de imagens apresentadas;
- Comparar e discutir os resultados obtidos das arquiteturas propostas com outras arquiteturas complexas conhecidas na literatura e outros trabalhos que apresentam metodologias alternativas de reconhecimento de gestos.

1.2 Organização

Os capítulos desta dissertação estão divididos como apresentado abaixo:

- **Capítulo 2:** apresentação de uma resenha das técnicas utilizadas de processamento de imagens e de Inteligência Computacional;
- **Capítulo 3:** descreve-se a metodologia proposta, os passos adotados na etapa de pré-processamento das imagens dos gestos e os algoritmos classificadores adotados e suas arquiteturas;
- **Capítulo 4:** apresentação dos experimentos realizados durante a aplicação da metodologia proposta, desde a confecção da base de imagens própria ao ambiente utilizado para o treinamento dos classificadores;

- **Capítulo 5:** são discutidos os resultados obtidos pelas arquiteturas de *CNN* propostas e comenta-se a comparação desses resultados com os dados obtidos de arquiteturas conhecidas da literatura e outros trabalhos;
- **Capítulo 6:** apresenta uma discussão com os objetivos alcançados, suas contribuições e as perspectivas dos trabalhos futuros.

2 MATERIAIS E MÉTODOS

Este capítulo discute as técnicas de processamento de imagens e classificação de dados utilizadas neste trabalho. Para realizar o treinamento de determinados classificadores é necessário extrair dados que contenham atributos característicos relevantes. Porém, ao lidar com imagens é importante extrair características somente das regiões de interesse. Técnicas de segmentação, filtros e operações morfológicas podem ser aplicados para aprimorar os detalhes dessas características relevantes para uma aplicação específica. Através do uso de redes neurais convolucionais, não se faz necessário extrair vetores de características numéricos das imagens, mas devido a entrada desse tipo de rede ser a própria imagem é relevante que as regiões de interesse das imagens sejam realçadas. Com essas considerações, uma bom pré-processamento deve separar as características importantes da imagem dos ruídos e das regiões de não-interesse.

2.1 Operações Morfológicas

Ao utilizar segmentadores simples, as imagens binarizadas geradas podem apresentar elementos distorcidos por ruídos ou textura e imperfeições na estrutura e forma da imagem. Para contornar esses problemas é comum utilizar filtros morfológicos, pois eles apresentam-se como uma forte ferramenta para extração de componentes de imagens, de modo que são úteis na representação e descrição de formas de uma região. Esses filtros não são de uso exclusivamente em imagens binarizadas, eles também podem ser aplicados nas imagens em tons de cinza (GONZALEZ; WOODS, 2006).

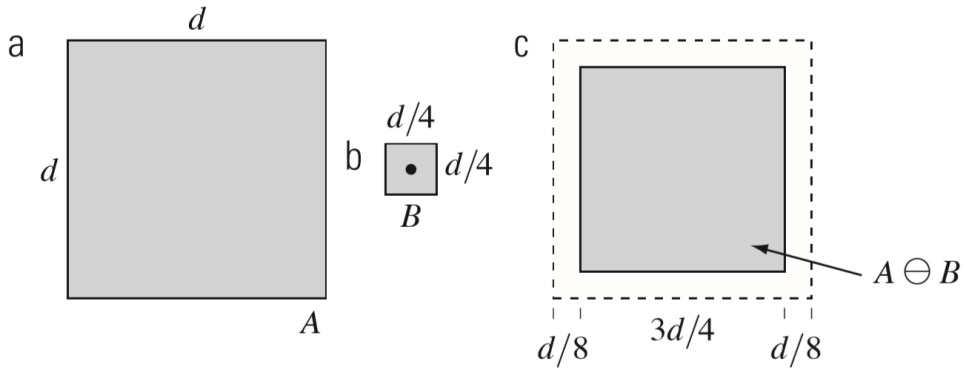
O uso de filtros morfológicos consiste em um conjunto de operações não lineares relacionados à geometria do objeto, onde as operações dependem das posições dos pixels e não dos seus valores. Por isso, são na maioria das vezes utilizados em imagens binárias. A teoria dos filtros morfológicos é baseada na matemática morfológica que utiliza a teoria de conjuntos em sua definição.

As operações morfológicas são determinadas a partir da análise da vizinhança de um ponto central, o conjunto que representa essa vizinhança é denominado de elemento estruturante. Os elementos estruturantes costumam ser matrizes binárias em que seus elementos podem ser definidos por qualquer geometria, como: linha, quadrado, círculo, prisma e outros. Essas operações podem ser definidas por duas elementares que são a erosão e a dilatação.

Na erosão, uma imagem A inicial ao ser processada por um elemento estruturante B,

para que um pixel inicial de $A_{x,y}$ permaneça na imagem, todo o elemento estruturante deve estar contido no ponto (X,Y), como podemos exemplificar na imagem da Figura 1. Matematicamente, podemos definir por: $A \ominus B = \{z \in E | B_z \subseteq A\}$.

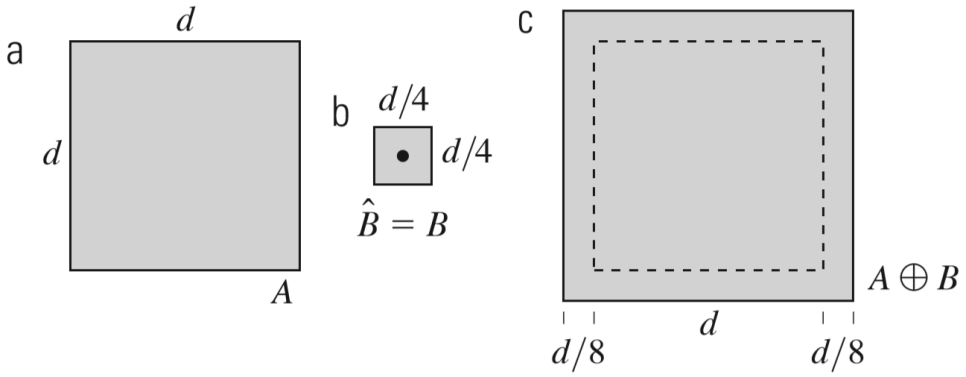
Figura 1 – Exemplo de uma erosão, em que *a* é a imagem original, *b* o elemento estruturante e *c* o resultado



Fonte: Gonzalez e Woods (2006).

Na dilatação, uma imagem A inicial ao ser dilatada por um elemento estruturante B, quando quanlquer ponto desse elemento estruturante estiver contido no ponto (X,Y) de A, logo irá ocorrer uma interseção com A, como podemos exemplificar na imagem da Figura 2. Matematicamente, podemos definir por: $A \oplus B = \bigcup_{b \in B} A_b$.

Figura 2 – Exemplo de uma dilatação, em que *a* é a imagem original, *b* o elemento estruturante e *c* o resultado



Fonte: retirado de Gonzalez e Woods (2006).

Além dessas operações, existem outras operações morfológicas também importantes, como a abertura e o fechamento, que são resultados das combinações do uso das operações elementares. A abertura pode ser definida como uma erosão de A por B, seguido de uma dilatação por B, conforme a equação: $A \circ B = (A \ominus B) \oplus B$. O fechamento pode ser definido como uma dilatação de A por B, seguido de uma erosão por B, conforme a equação: $A \bullet B = (A \oplus B) \ominus B$.

Todas essas operações atuam em conjunto com diversos elementos estruturantes e, portanto, cada operação resulta em uma nova imagem. Sua aplicação é comum em etapas pós-segmentação e em pré-processamento de imagens.

2.2 Extração de Contornos e Aproximação Poligonal

Uma das características que podem ser extraídas de uma imagem é o contorno dos objetos nela presente. Esse contorno pode ser útil em muitas aplicações, seja para classificação baseada na geometria dos objetos ou para contabilizar a quantidade de determinados objetos em uma imagem.

A técnica descrita por (SUZUKI *et al.*, 1985) apresenta um método para geração de contornos, no qual para cada elemento da imagem, conjuntos de pontos são gerados envolvendo todas as regiões de contorno dos objetos. Assim, ao aplicar essa técnica em uma imagem qualquer, serão extraídos todos os contornos dos possíveis objetos da imagem. Percebe-se que existe uma dependência da qualidade dos contornos gerados com o processamento da imagem antes de sua aplicação, pois uma imagem com ruídos ou objetos não-relevantes resultará em contornos imprecisos.

Para solucionar esses casos de contornos ruidosos, é possível utilizar técnicas de aproximação poligonal. Essas técnicas costumam gerar polígonos semelhantes aos conjuntos de pontos apresentados inicialmente. A técnica descrita em (RAMER, 1972) gera inicialmente um número mínimo de vértices que acomodam-se sobre o conjunto de pontos (por exemplo, um contorno) apresentado. Essa técnica possui um parâmetro ε que define a distância máxima entre os pontos da curva original aos pontos do polígono. A partir do ε, serão eliminados recursivamente todos os pontos de contorno cuja distância de curva do contorno esteja acima do limiar ε determinado. Assim, quanto menor for o parâmetro ε, mais semelhante será o polígono gerado ao conjuntos de dados (contorno) apresentado. Na Figura 3, são apresentadas três imagens, uma inicial e outras duas imagens com o contorno gerado em verde com valores de ε diferentes, para a segunda imagem o valor corresponde a 10% do comprimento do arco e na terceira imagem a 1%.

Figura 3 – Exemplo de uma aproximação poligonal, detalhes em verde

Fonte: retirado de Mordvintsev (2013).

2.3 Aprendizagem profunda (*Deep learning*)

Tópico da área de Aprendizagem de Máquina (*Machine learning*), a Aprendizagem Profunda (*Deep Learning*) é definida como um conjunto de algoritmos baseados em redes neurais que extraem padrões abstratos de dados da entrada à medida em que esses dados se propagam pelas camadas da rede através de técnicas de *backpropagation*. Esses dados propagados são resultados das informações das taxas de acerto à medida que ocorrem as etapas de aprendizagem.

Alguns modelos de *Deep Learning* são capazes de executar a extração de características a partir de dados brutos, como: imagens, texto ou som, e selecionam automaticamente características úteis para o processo de aprendizado. As técnicas de aprendizagem profunda possuem diversas aplicabilidades nos mais diversos setores, como: reconhecimento facial e de voz, reconhecimento e tradução de textos ou detecção de doenças a partir de imagens médicas.

A inspiração dessas técnicas é o cérebro humano, no qual a informação é processada e extraída camada por camada dos conjuntos de neurônios. Alguns exemplos de algoritmos de Aprendizagem Profunda são as Redes Neurais Artificiais Profundas e Redes Neurais Convolucionais.

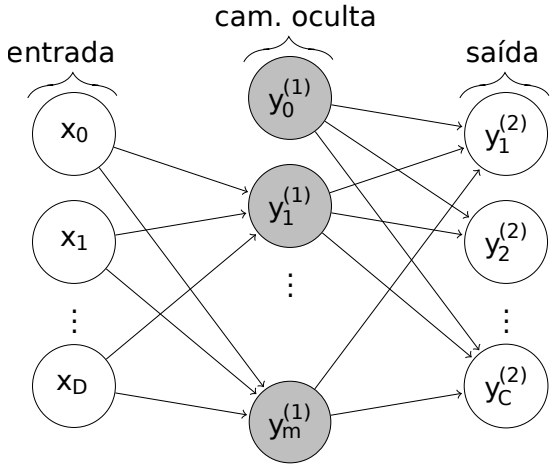
2.3.1 Redes neurais artificiais

As Rede Neural Artificial (RNA) são modelos amplamente utilizados em tarefas de classificação e em inúmeras áreas do conhecimento. Em uma RNA, o objeto que será classificado é apresentado à rede através da ativação de neurônios artificiais na camada de entrada. Essas ativações são processadas nas camadas internas e o resultado converge como um padrão na camada de saída.

O elemento básico de uma rede neural é o perceptron simples, porém ele sozinho somente é capaz de resolver problemas linearmente separáveis. Para resolver problemas não linearmente separáveis, é necessário utilizar uma rede neural *MLP*. A *MLP* consiste em uma

rede cuja arquitetura é composta, por uma camada de entrada, várias camadas intermediárias ocultas e uma camada de saída, como pode ser visto na Figura 4.

Figura 4 – Exemplo de uma MLP



Fonte: adaptado de Stutz (2014).

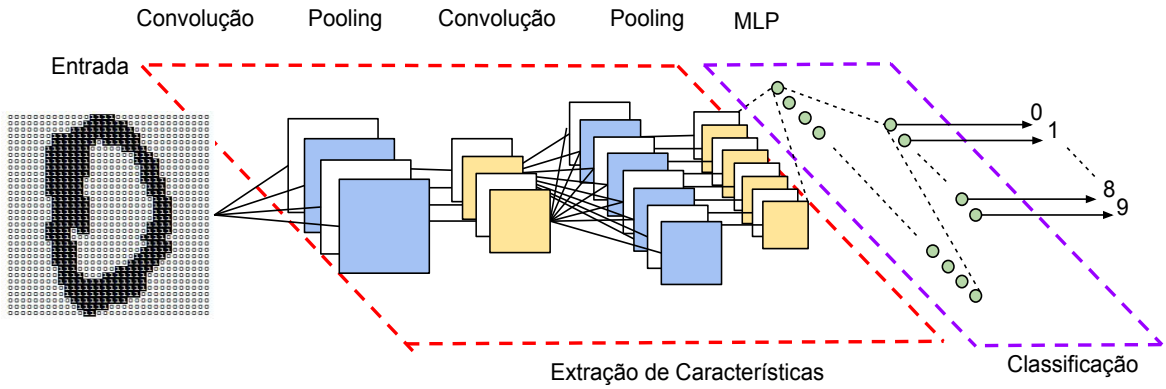
O treinamento de uma rede neural consiste em ajustar os valores dos pesos das conexões que interligam os perceptrons entre as camadas, de modo que esse valor modifica as sinapses artificiais que ativam ou não os neurônios para a solução de um problema específico. Para treinar uma *MLP*, um dos métodos mais comuns é usar o algoritmo *backpropagation*, que modifica os pesos das conexões das camadas ocultas à medida que o erro é propagado na direção contrária à de saída da rede, com o objetivo de adaptar a rede neural à resolução do problema. Logo, uma rede treinada para uma tarefa de classificação é determinada pelos valores fixados dos pesos de suas conexões internas (HAYKIN, 2007).

2.3.2 Redes Neurais Convolucionais

As redes neurais convolucionais (*Convolutional Neural Network*) *CNN* são amplamente utilizadas na literatura nas mais diversas áreas como, por exemplo, classificação de imagens e reconhecimento e detecção de objetos (VARGAS *et al.*, 2016). Sua estrutura é resumidamente composta por dois blocos: extração de características e classificação. O bloco de extração de características é composto por dois tipos de camadas: convolução e *pooling*. O bloco de classificação pode ser composto por uma rede neural *MLP*, como mostrado na Figura 5. A arquitetura da *CNN* deve ser definida de acordo com a aplicação e é geralmente composta por uma sequência de camadas de convolução e *pooling*. Além disso, é preciso definir o número de neurônios em cada camada, tamanho da janela de convolução e a escolha de suas funções de

ativação. Na literatura já foram definidas algumas arquiteturas de *CNN* como: LeNet (LECUN *et al.*, 1998), InceptionResNetV2 (SZEGEDY *et al.*, 2017), InceptionV3 (SZEGEDY *et al.*, 2016), VGG-16 (SIMONYAN; ZISSERMAN, 2014), VGG-19 (SIMONYAN; ZISSERMAN, 2014), ResNet50 (HE *et al.*, 2016) e DenseNet201 (HUANG *et al.*, 2017).

Figura 5 – Exemplo de uma CNN e suas camadas.



Fonte: adaptado de Vargas *et al.* (2016).

No contexto de classificação de imagens, a entrada inicial de uma *CNN* é uma imagem (matriz bi-dimensional) de seu modelo de cores qualquer, podendo ser, por exemplo: em tons de cinza ou RGB. Na camada de convolução, cada neurônio está associado a uma janela (*kernel*) que é convoluída com a imagem de entrada durante o treinamento e classificação da *CNN*. Esta janela de convolução é composta pelos pesos de cada neurônio associado. A saída dessa camada de convolução é um conjunto de N imagens, uma para cada um dos N neurônios. Por causa da convolução, essas novas imagens geradas podem conter valores negativos. Para evitar esse problema é possível utilizar algumas funções lineares, um exemplo é a função linear retificada (ReLU) que substitui valores negativos por zero. As saídas dessa camada são conhecidas como mapas de características.

Depois de uma camada de convolução, é comum aplicar uma camada de *pooling*. Sua importância deve-se ao fato de que a camada pode reduzir a dimensionalidade dos mapas de características e, por consequência, promover uma redução no tempo de treinamento da rede. Algumas arquiteturas conhecidas possuem uma configuração de alternância entre camadas convolução e *pooling*, mas isso não é uma regra. A GoogLeNet (SZEGEDY *et al.*, 2015), por exemplo, tem cinco camadas de convolução seguidas por uma camada de *pooling*. Após toda a sequência de camadas de convolução e *pooling*, é necessário utilizar um algoritmo classificador

diante de todas as características extraídas. Portanto, é possível utilizar uma rede neural *MLP* como camada final que executa a classificação com base nos dados da etapa anterior.

Devido ao seu grande número de camadas e aplicações bem-sucedidas, as *CNN* são uma das técnicas preferidas do aprendizado profundo. Sua arquitetura permite a extração automática de diversas características da imagem, como bordas, círculos, linhas e texturas. Essas características extraídas são cada vez mais otimizadas em camadas adicionais. É importante enfatizar que os valores das janelas aplicadas nas camadas de convolução são o resultado da retropropagação durante o treinamento da *CNN*.

2.4 Segmentação por Cor

O processo de segmentação subdivide uma imagem em regiões (GONZALEZ; WOODS, 2006), de modo que é possível destacar regiões que contêm características de interesse. Portanto, podem ser implementados diversos algoritmos segmentadores com o foco em um tipo específico de dado a ser destacado, como: separação de cores, texturas, pontos, linhas, descontinuidades, bordas, entre outros. Logo, o processo de segmentação varia de acordo com a problemática a ser solucionada.

Quando se apresenta a problemática de reconhecimento de gestos, toda a região de fundo (*background*) de uma imagem contendo um gesto não é de interesse, portanto somente o conjunto de *pixels* com a presença da mão humana deve ser mantido. Um método para essa segmentação é a implementação da segmentação por remoção de fundo, na qual coletam-se amostras de imagens do ambiente e em seguida são adicionados objetos na cena. Desse modo, os *pixels* das novas imagens obtidas são comparados com as imagens do cenário e, onde houver diferença de valores, esses *pixels* serão mantidos (OPENCV, 2018). Apesar de ser um bom método, esse tipo de segmentação é bastante suscetível à variação de iluminação do ambiente, o que provoca falhas no resultado final. Na Figura 6 é apresentado um exemplo desse tipo de segmentação.

Um outro método é a segmentação por cores, no qual é possível dividir as imagens em regiões que possuam tons de cores previamente definidos. No caso do problema apresentado de reconhecimento de gestos, os tons de cores a serem segmentados são semelhantes aos tons de pele humana. Para que isso ocorra, é possível treinar uma rede *MLP* que aprenda os tons de cores de pele e, posteriormente, identifique quais dos *pixels* da imagem pertencem aos conjuntos de cores de pele ou não. Esse método apresenta uma maior robustez, pois só depende de que os

Figura 6 – Exemplo de uma segmentação utilizando a técnica de subtração de *background*

Fonte: retirado de (Derawi *et al.*, 2010)

conjuntos de cores sejam corretamente definidos. Uma vez que seja apresentado à rede neural um bom conjunto de cores, o resultado será satisfatório sem depender de mudança de iluminação ou *background* do ambiente. Na Figura 7 é apresentado um exemplo desse tipo de segmentação.

Figura 7 – Exemplo de uma segmentação utilizando a técnica de segmentação por cor

Fonte: retirado de (PHUNG *et al.*, 2005).

2.5 Métricas de Avaliação

As métricas de avaliação são utilizadas para medir a qualidade de um modelo de classificação. Para calcular essas métricas é necessário entender a definição de matriz de confusão, que indica os acertos e erros de um modelo em comparação com o valor real (GOUTTE; GAUSSIER, 2005). Na Tabela 1, é apresentado um modelo de matriz de confusão.

Tabela 1 – Matriz de confusão

		Valor Real	
		Verdade	Falso
Valor Previsto	Verdade	Verdadeiro Positivo (VP)	Falso Positivo (FP)
	Falso	Falso Negativo (FN)	Verdadeiro Negativo (VN)

Fonte: elaborado pelo autor (2019).

A acurácia apresenta uma performance geral do classificador, seu valor é a razão do número de acertos de um conjunto de amostras por sua quantidade que foram classificadas.

A precisão aponta dentre a classificação de um tipo de classe quantas realmente estão corretas. Seu valor é definido pela seguinte equação:

$$\text{Precisão} = \frac{VP}{VP+FP} \tag{2.1}$$

A revocação aponta dentre as situações possíveis de classificação de uma determinada classe como valor esperado, quantas foram definidas corretamente. Sua definição é apresentada a seguir:

$$\text{Revocação} = \frac{VP}{VP+FN} \tag{2.2}$$

O *F1-Score* é uma média harmônica que envolve a precisão e a revocação, representada a seguir:

$$F1 = 2 \cdot \frac{\text{precisão} \cdot \text{revocação}}{\text{precisão} + \text{revocação}} \tag{2.3}$$

2.6 Considerações finais

Neste capítulo foram apresentados conceitos básicos que serão utilizados neste trabalho. Abordaram-se conceitos de operações morfológicas em imagens, extração de contornos e aproximação poligonal, definição de aprendizagem profunda, redes neurais artificiais, redes neurais convolucionais e segmentação por cor. A compreensão desses conceitos possibilita o entendimento da metodologia que será proposta no capítulo seguinte.

3 METODOLOGIA

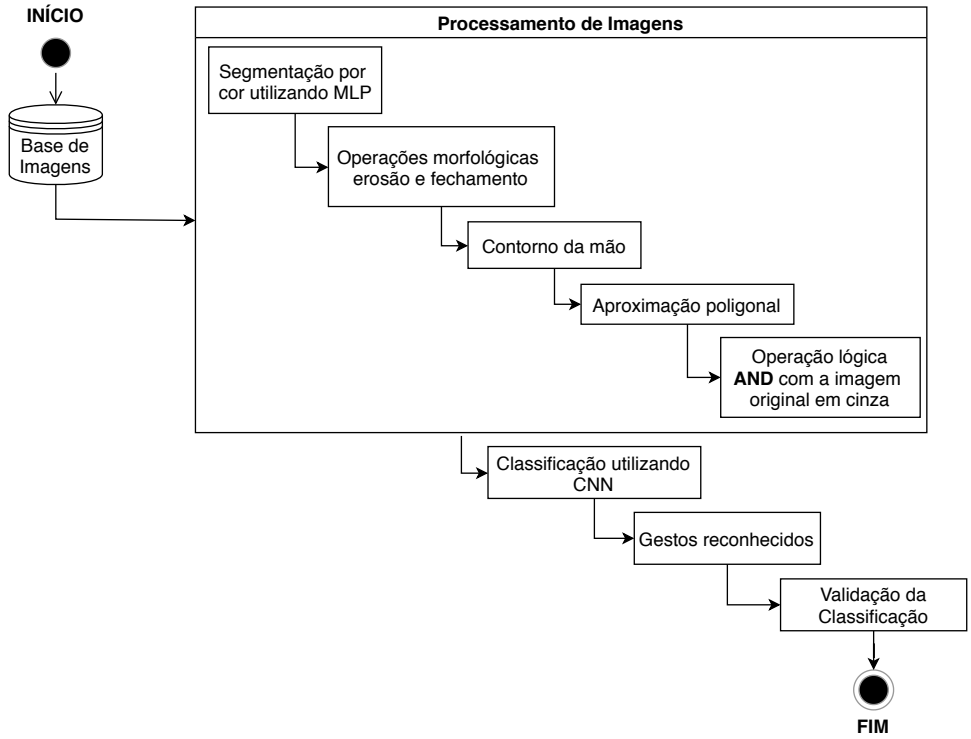
Com o uso das técnicas de PDI apresentadas, é possível desenvolver uma metodologia de reconhecimento de gestos, como apresentada na Figura 8. Inicialmente, as imagens estão presentes em uma base e, em sequência, são aplicadas diversas técnicas de PDI (presentes no bloco chamado de *Processamento de Imagens* da Figura 8). Primeiro, é necessário que as imagens passem por um processo de segmentação por cor. Desse modo, somente as regiões com cores semelhantes aos tons de pele humana permanecerão nas imagens. Após isso, operações lógicas de erosão e dilatação são aplicadas para eliminação de furos e buracos e na redução de ruídos nas regiões segmentadas.

Para complementar a redução de ruído, é aplicado um algoritmo detector de contornos em imagens, que gera o contorno da região da mão presente na imagem. Devido ao fato de esse contorno gerado ainda estar suscetível ao ruído, é utilizada uma técnica de aproximação poligonal para remoção dos ruídos presentes no contorno gerado da região da mão. Com esta região definida através do polígono final, é realizada uma operação lógica ***AND*** com a imagem original em escala de cinza. Desse modo, somente a região com a presença da mão juntamente com as informações que podem ser extraídas presentes no interior da palma da mão e dos dedos em escala de cinza são utilizadas para classificação. Na etapa de classificação, essas imagens são utilizadas como parâmetro de entrada nas *CNN* e a saída dessas redes são as classes dos gestos para cada imagem. A validação dos resultados obtidos é realizada utilizando técnicas de validação cruzada. Por fim, a comparação dos resultados é analisada.

As imagens presentes na base mencionada foram adquiridas através de uma aplicação desenvolvida para essa finalidade. Com o seu auxílio, foi possível capturar os gestos de diversos colaboradores sem dificuldades, pois sempre foi apresentado um exemplo do gesto que deve ser capturado ao lado da imagem da mão do colaborador.

Durante a etapa de classificação, os resultados são obtidos através de 4 arquiteturas de *CNN* propostas que foram projetadas com o foco em um número reduzido de camadas de profundidade, cujo intuito é reduzir o custo computacional. As arquiteturas das *CNN* propostas estão disponíveis na Tabela 2, essas arquiteturas são *fully connected* com um *max pooling* global nas camadas de *pooling*, para regularização dos valores obtidos em cada camada utiliza-se a função ReLU. Para complementar esses resultados, foram realizadas também classificações com outras arquiteturas já conhecidas e robustas da literatura, das quais podemos citar: LeNet (LECUN *et al.*, 1998), InceptionResNetV2 (SZEGEDY *et al.*, 2017), InceptionV3 (SZEGEDY *et*

Figura 8 – Fluxograma da metodologia proposta



Fonte: elaborado pelo autor (2019).

al., 2016), VGG-16 (SIMONYAN; ZISSERMAN, 2014), VGG-19 (SIMONYAN; ZISSERMAN, 2014), ResNet50 (HE *et al.*, 2016) e DenseNet201 (HUANG *et al.*, 2017).

Para a validação dos resultados obtidos, adota-se a metodologia de validação cruzada. Nela, é utilizado o *hold-out*, com parâmetros definidos em 75% da base de imagens para treino e os 25% restantes para teste. Essa divisão é realizada após uma permutação entre todas as imagens e os resultados finais são obtidos após a média de 10 execuções.

3.1 Considerações finais

Nesse capítulo foi abordada a metodologia da pesquisa desenvolvida, assim como descreveu-se cada passo necessário e de que modo foram realizados os experimentos, bem como os classificadores e as técnicas de validação utilizadas.

No capítulo seguinte, são explanados os experimentos realizados a partir da metodologia proposta e serão discutidos os caminhos necessários que levaram a sua construção.

Tabela 2 – Arquitetura das *CNN* propostas

Profundidade	CNN1	CNN2	CNN3	CNN4
1	Convolução (5x5)	Convolução (5x5)	Convolução (5x5)	Convolução (5x5)
2	*Max Pooling (2x2)*	*Max Pooling (2x2)*	Convolução (7x7)	Convolução (7x7)
3	Convolução (5x5)	Convolução (7x7)	*Max Pooling (2x2)*	Convolução (9x9)
4	*Max Pooling (2x2)*	*Max Pooling (2x2)*	Convolução (5x5)	*Max Pooling (2x2)*
5		Convolução (5x5)	Convolução (7x7)	Convolução (5x5)
6		Convolução (7x7)	*Max Pooling (2x2)*	Convolução (7x7)
7		*Max Pooling (2x2)*	Convolução (5x5)	Convolução (9x9)
8			Convolução (7x7)	*Max Pooling (2x2)*
9			Convolução (9x9)	Convolução (5x5)
10			*Max Pooling (2x2)*	Convolução (7x7)
11				Convolução (9x9)
12				*Max Pooling (2x2)*

Fonte: elaborado pelo autor (2019).

4 EXPERIMENTOS

Conforme a metodologia apresentada no capítulo anterior, foram desenvolvidos os experimentos necessários que proporcionaram a aquisição dos resultados. Inicialmente, elaborou-se a confecção de uma base de imagens com os 24 gestos estáticos que compõem o alfabeto *ASL*, com exceção dos gestos das letras J e Z, que são gestos dinâmicos e, por isso, não foram capturados. A aquisição dessas imagens contou com a participação de 8 pessoas de diferentes etnias, em um ambiente controlado de laboratório com a presença de luz artificial e sem um *background* fixo e regular.

Durante a captura desses gestos, foi utilizada uma câmera Logitech Brio, com a resolução definida em 1920x1080 *pixels*. Assim, foram capturados vídeos com a duração de 5 (cinco) segundos para cada um dos 24 gestos. A partir de cada um desses vídeos, foram extraídos 5 *frames* com a resolução de 400x400 *pixels* para compor as imagens de cada gesto por pessoa, compondo a base com um total de 960 amostras. Para aumentar a quantidade e a variedade de amostras da base própria criada, foram aplicadas rotações de 20^{o} no sentido horário e anti-horário, e uma mudança de escala com uma redução de 15% de altura das imagens. Desse modo, a base própria de imagens é composta por um total de 3840 amostras. Na Figura 9, é possível conferir algumas das amostras de imagens coletadas que compõem a base própria produzida.

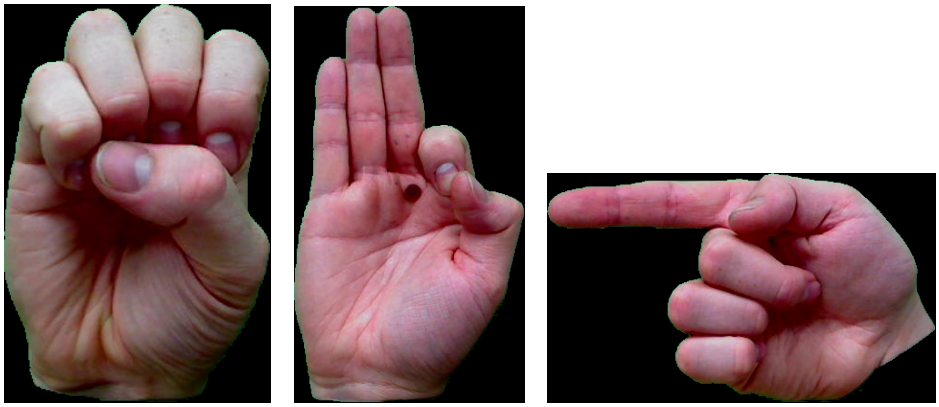
Figura 9 – Amostras dos gestos da base construída

Fonte: elaborado pelo autor (2018).

Para aumentar a variedade da base de imagens de modo que se eleve o número de diferentes amostras, foi utilizada a combinação da base própria com outra base disponível na literatura, encontrada em (BARCZAK *et al.*, 2011), que também foi elaborada utilizando os mesmos gestos estáticos *ASL*. Essa outra base (BARCZAK *et al.*, 2011) é constituída originalmente

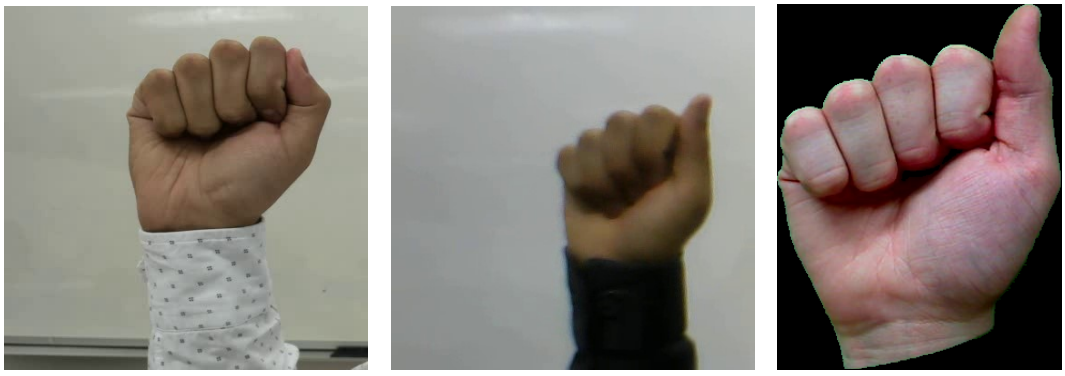
de 1815 amostras. No entanto, adotaram-se as mesmas técnicas de translação e mudança de escala que na base própria para elevar esse número, totalizando 7260 imagens na outra base, como pode ser visto na Figura 10. A união de ambas as bases proporcionou um conjunto de imagens final de 11000 amostras constituída de 24 gestos, com uma enorme variedade de formas de mãos e tons de pele. Algumas variedades para o gesto A estão presentes na Figura 11. Na Figura 12 são apresentadas algumas imagens de diferentes gestos.

Figura 10 – Amostras dos gestos da base (BARCZAK *et al.*, 2011)



Fonte: retirado de (BARCZAK *et al.*, 2011).

Figura 11 – Amostras para o gesto A

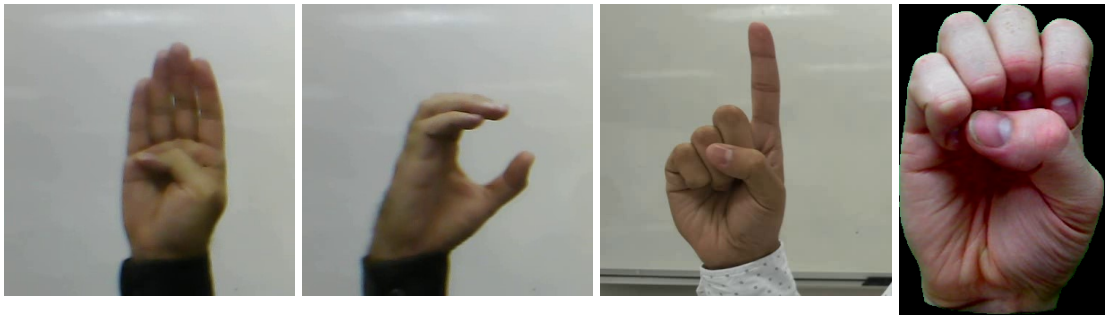


Fonte: elaborado pelo autor (2018) e (BARCZAK *et al.*, 2011).

Conforme a metodologia apresentada na Figura 8, é necessário aplicar nas imagens da base uma etapa de segmentação. Esse processo é necessário devido ao fato de que as imagens possuem diversas informações, como o cenário ao fundo, de que a etapa de classificação necessita apenas de informações extraídas dos gestos formados pelas mãos nas imagens. Para garantir a remoção do cenário (*background*), desenvolveu-se uma rede neural *MLP* para atuar como um segmentador de cor de pele. Essa *MLP* foi definida com uma arquitetura que possui duas camadas ocultas, com 5 e 10 neurônios, respectivamente. A camada de entrada possui 3

Figura 12 – Amostras da base de imagens utilizada.

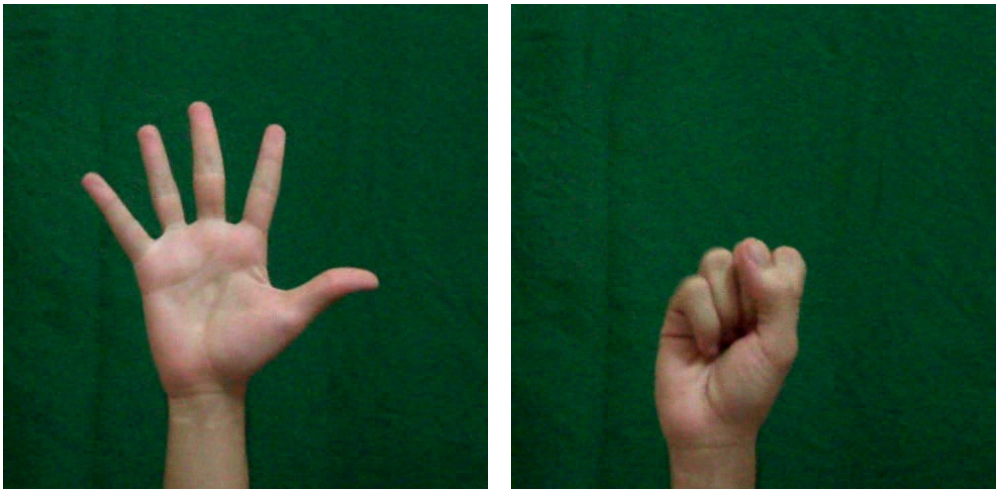
(a) Gesto B (b) Gesto C (c) Gesto D (d) Gesto E



Fonte: elaborado pelo autor (2018) e (BARCZAK *et al.*, 2011).

neurônios, referindo-se aos tons de cores da paleta *RBG*. A camada de saída possui somente 1 neurônio, proporcionando à rede uma decisão binária: 0 para não-pele e 1 para pele. Para o treinamento dessa rede, foram usadas 5 imagens de mãos de diferentes etnias sem *background* e os valores dos *pixels* dessas imagens foram usados como parâmetros de entrada. Para a camada de saída, usou-se uma máscara das imagens usadas como saída da rede, essa máscara nos *pixels* que possuem cores de pele atribui-se o valor 1 e os *pixels* de *backgroud* o valor 0. Desse modo, associa-se para os conjuntos de valores *RGB* que representam tons de pele humana a classe 1 e, caso contrário, o valor 0. Na Figura 13 podemos ver algumas das imagens usadas como entrada da rede e na Figura 14 as máscaras usadas para definir as classes.

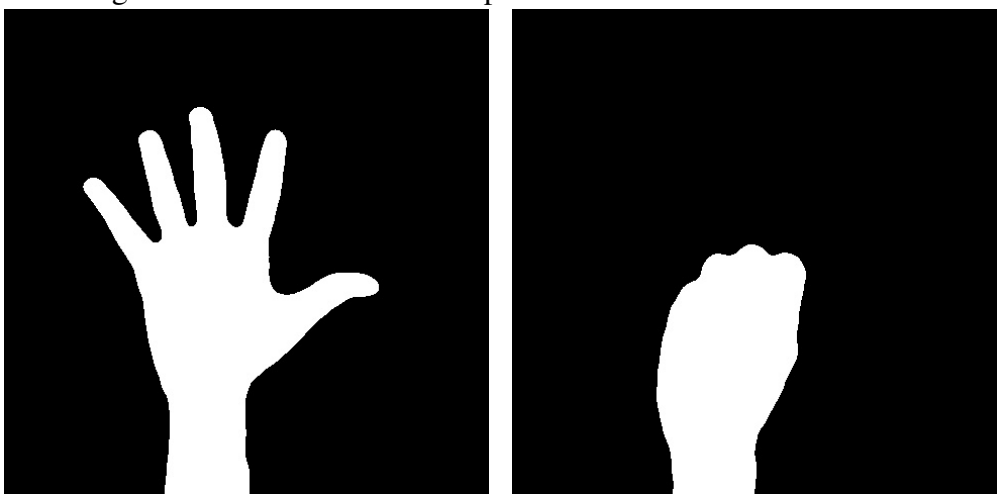
Figura 13 – Amostras das imagens utilizadas na camada de entrada da MLP



Fonte: elaborado pelo autor (2016).

Foram adotadas 10 épocas para o treinamento do segmentador. Após esse momento, todas as imagens de mãos da base foram segmentadas. Dessa forma, foram geradas as máscaras das regiões propícias a serem pele em comparação à imagem original. Observou-se que em

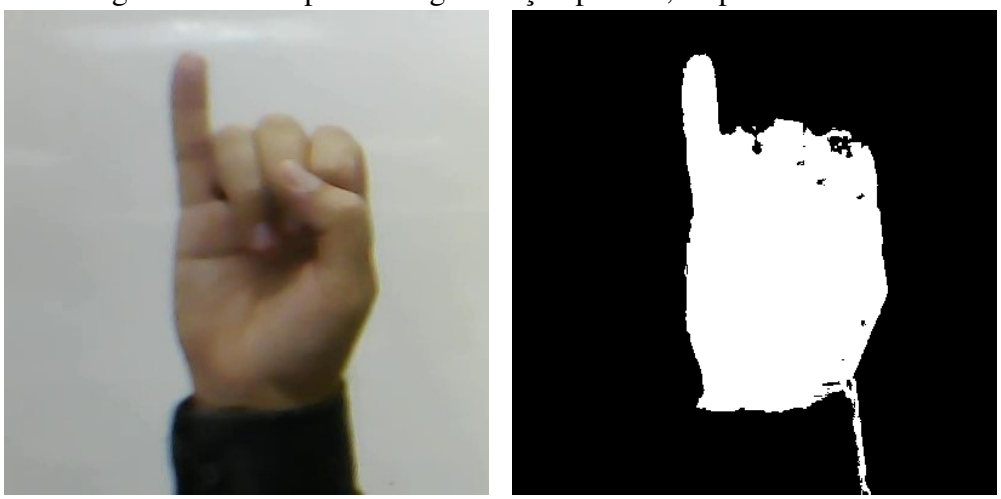
Figura 14 – Imagens das máscaras utilizadas para treino da MLP



Fonte: elaborado pelo autor (2016).

diversas imagens existiam ruídos nas máscaras, como: linhas, buracos e bordas sinuosas, que são ruídos e podem ser visualizados na Figura 15.

Figura 15 – Imagens antes e depois da segmentação por cor, respectivamente

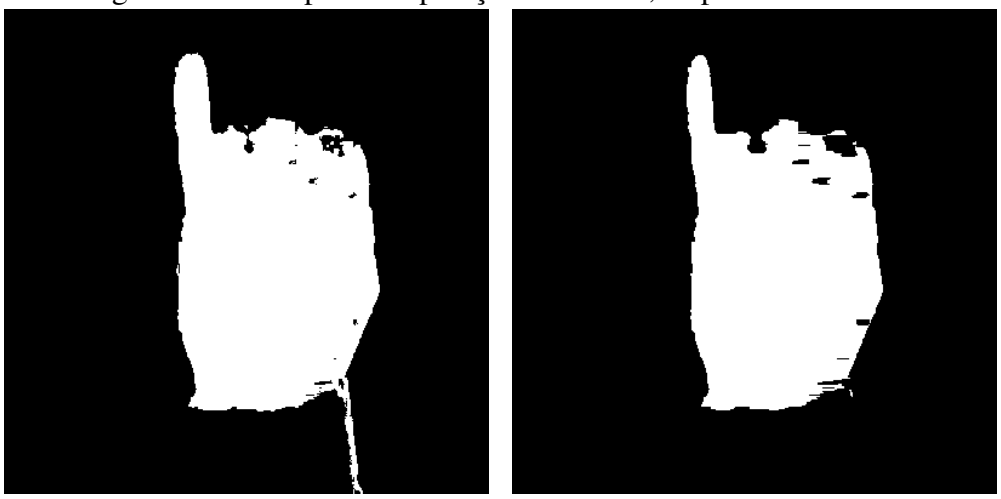


Fonte: elaborado pelo autor (2019).

Para corrigir esses problemas, analisou-se que os filtros morfológicos podem remover e corrigir os elementos ruidosos das imagens. Logo, para remover linhas presentes nas imagens, utilizou-se uma operação de erosão com um elemento estruturante do tipo linha (vetor) com um tamanho de 9 *pixels*. Na Figura 16 pode-se ver o resultado dessa operação, no qual diversos elementos foram removidos.

Apesar do bom resultado, o uso da erosão provocou o aumento no tamanho das regiões de descontinuidades nas bordas e buracos no interior das imagens. Uma alternativa foi usar um outro filtro morfológico, com a operação de fechamento com um elemento estruturante

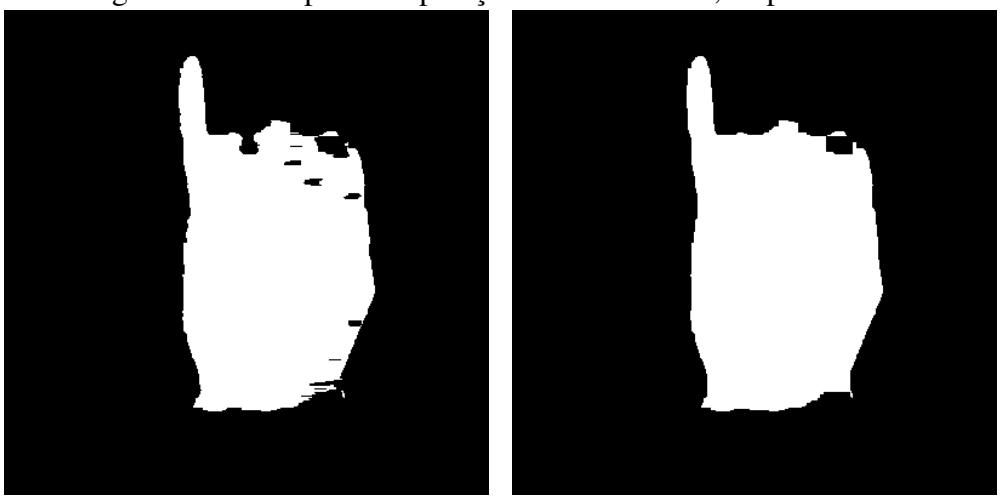
Figura 16 – Imagens antes e depois da operação de erosão, respectivamente



Fonte: elaborado pelo autor (2019).

a partir do formato quadrático de tamanho 13x13 *pixels*. Na Figura 17(b) nota-se que diversos furos da região da mão na máscara foram preenchidos.

Figura 17 – Imagens antes e depois da operação de fechamento, respectivamente



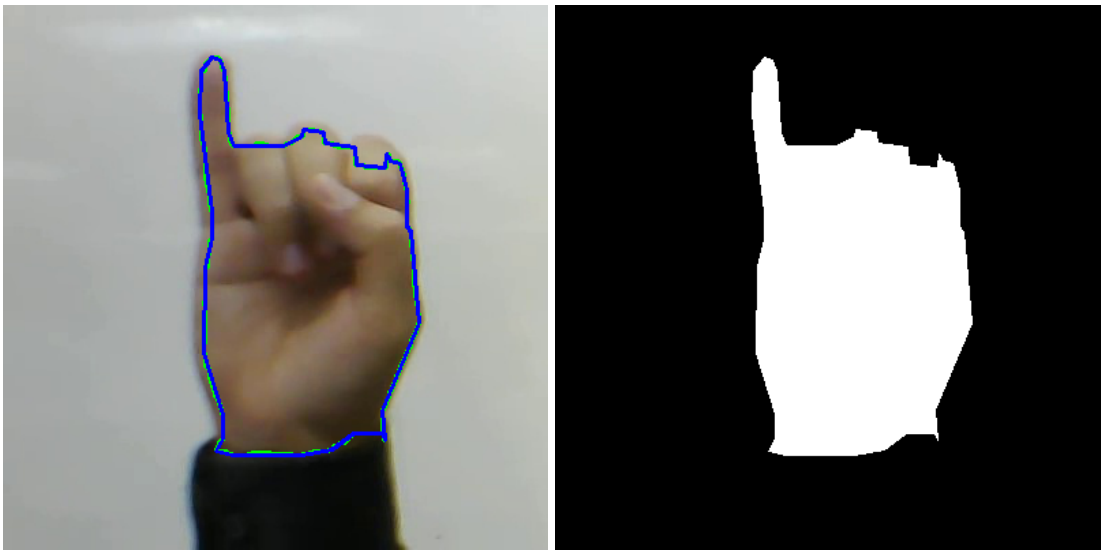
Fonte: elaborado pelo autor (2019).

Apesar dos bons resultados obtidos após a aplicação das operações morfológicas, um pequeno grupo de imagens ainda apresentou buracos da região da palma da mão e ruídos nas bordas. Para corrigir isso, foi adotado o uso da aproximação poligonal através de um contorno gerado.

Na imagem da máscara gerada após a segmentação e a aplicação de filtros morfológicos, somente a região da mão está presente na imagem. Logo, ao aplicar o algoritmo de geração de contornos (RAMER, 1972), somente é gerado um único contorno que envolve a região da mão presente na imagem. A partir desse contorno, é possível aplicar um algoritmo

de aproximação poligonal com um parâmetro ε de valor 2. Esse pequeno valor atribuído a ε permite que o polígono gerado seja semelhante ao contorno original, porém com uma distância máxima de 2 *pixels* do ponto no polígono ao contorno original. Assim, adotamos a região do interior do polígono como a máscara para a imagem da mão. Isso permite que todos os ruídos presentes nas bordas e buracos no interior da região da mão sejam removidos. Na Figura 18(a), podemos ver o contorno original da mão em verde e o polígono gerado em azul. Na Figura 18(b) vemos a máscara final gerada.

Figura 18 – Imagem com o contorno gerado e a máscara final, respectivamente
(a) Contorno original em verde e aproximação poligonal em azul (b) Máscara final



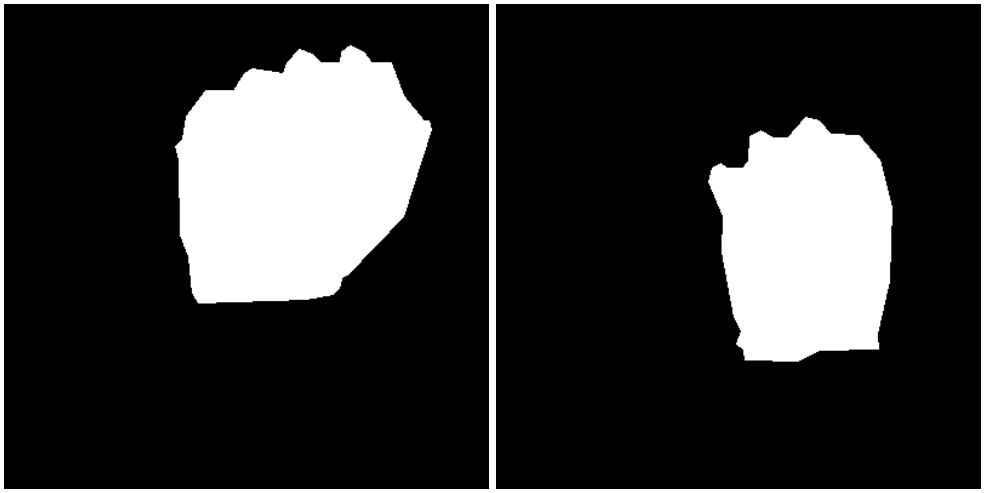
Fonte: elaborado pelo autor (2019).

Após esses passos, todas as imagens segmentadas de todas as classes não apresentaram mais ruídos ou deformações. Porém, observou-se uma grande similaridade em sua forma para alguns gestos. Nas Figuras 19 e 20 é possível notar essa similaridade entre os gestos A e E e para os gestos S e T. A partir dessas figuras, é possível chegar à conclusão que utilizá-las nesse estado para treinar uma *CNN* irá proporcionar classificações erradas e falsos positivos para esses gestos altamente similares, pois na *CNN* a imagem é o atributo de entrada.

A estratégia adotada para evitar esse problema de similaridade foi transformar as imagens originais da escala *RGB* para tons de cinza e aplicar uma operação lógica *AND* da imagem original (em cinza) com a máscara gerada. Nesse caso, como a máscara é uma imagem binarizada, a imagem final conterá somente a região da mão em cinza e todo o fundo da imagem será removido. Assim, todas as informações e características da palma da mão e dos dedos

Figura 19 – Imagens segmentadas com formas similares

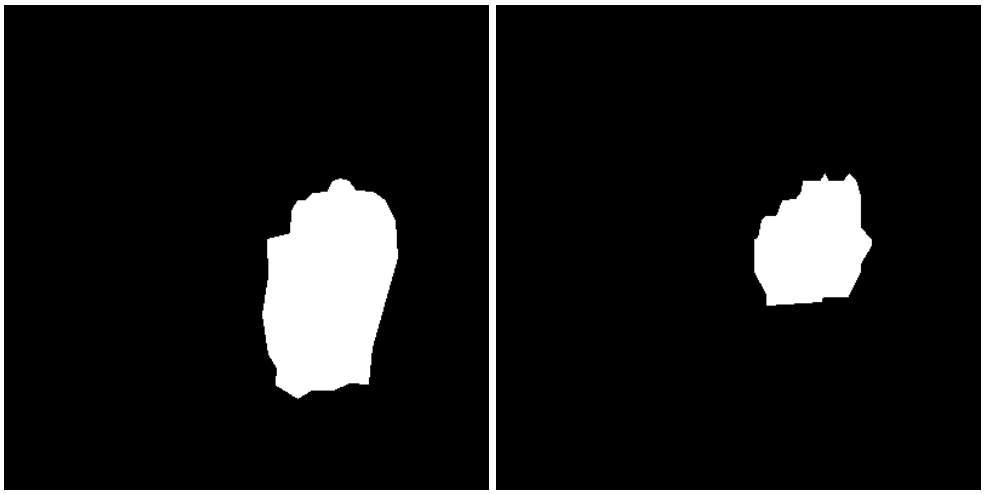
(a) gesto A (b) gesto E



Fonte: elaborado pelo autor (2019).

Figura 20 – Imagens segmentadas com formas similares

(a) gesto S (b) gesto T



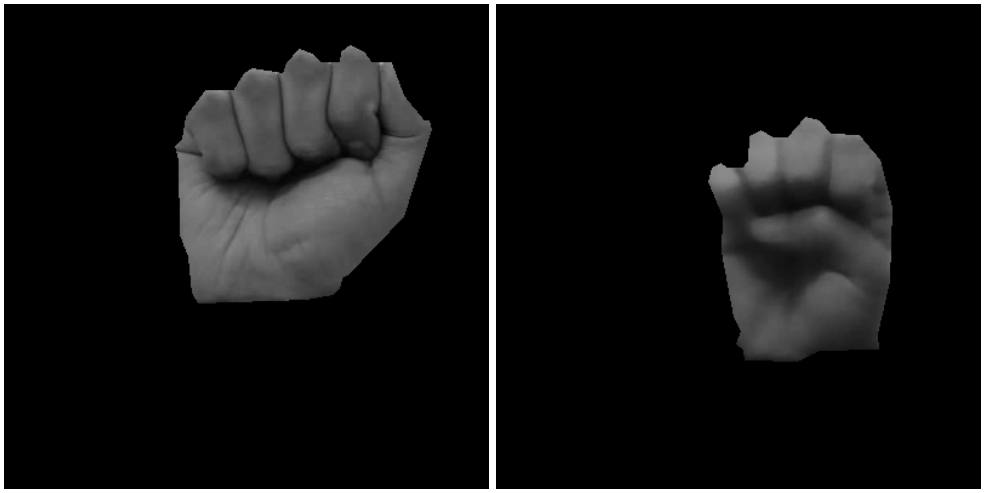
Fonte: elaborado pelo autor (2019).

estarão presentes na imagem final, juntamente com as informações das formas dos gestos. Na Figura 19 notamos que apesar da forma da mão nos gestos A e E serem similares, a posição do polegar é uma informação que os diferencia. Na Figura 20 notam-se as diferenças para os gestos S e T.

Após todos os passos da metodologia para os ajustes e correções na base de imagens, esses dados estão prontos para serem utilizados em um classificador de gestos estáticos. Desse modo, organizou-se em 4 arquiteturas de *CNN* que serão treinadas e testadas. Essas arquiteturas foram planejadas inspiradas na arquitetura clássica *LeNet* (LECUN *et al.*, 1998), e apresentam uma simplicidade em suas estruturas e demandam de um menor custo computacional durante a

Figura 21 – Gestos após a operação lógica AND com a imagens original da mão em cinza

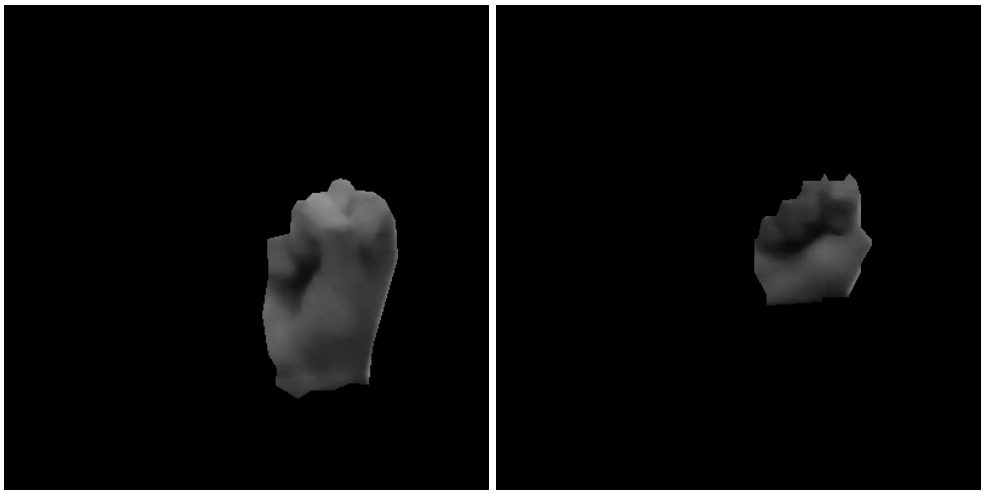
(a) gesto A (b) gesto E



Fonte: elaborado pelo autor (2019).

Figura 22 – Gestos após a operação lógica AND com a imagens original da mão em cinza

(a) gesto S (b) gesto T



Fonte: elaborado pelo autor (2019).

fase de treinamento.

Em cada neurônio das camadas de convolução e *pooling* das arquiteturas propostas, utilizou-se a função de ativação *Rectified Linear Units* (ReLU). Para classificar as características extraídas pelas camadas da *CNN* que geram muitos dados, foi adotada uma rede *MLP* com 400 e 800 neurônios em suas camadas ocultas, utilizando a função de ativação ReLU. Na camada de saída da *MLP* foram adotados 24 neurônios, referentes aos 24 gestos estáticos utilizando a função de ativação *softmax*.

Para que seja possível comparar melhor a robustez das arquiteturas de *CNN* propostas, as imagens foram treinadas e testadas com outras arquiteturas já conhecidas da literatura, que

são: LeNet (LECUN *et al.*, 1998), InceptionResNetV2 (SZEGEDY *et al.*, 2017), InceptionV3 (SZEGEDY *et al.*, 2016), VGG-16 (SIMONYAN; ZISSERMAN, 2014), VGG-19 (SIMONYAN; ZISSERMAN, 2014), ResNet50 (HE *et al.*, 2016) e DenseNet201 (HUANG *et al.*, 2017). Todas as arquiteturas, propostas e da literatura, foram submetidas às mesmas condições de treinamento e validação. Em todos os experimentos foi utilizado o método de validação cruzada *holdout*. Portanto, foi adotada para a base uma divisão de 75% das amostras para treino e 25% para teste, como definido pela literatura em (KOHAVI, 1995). Desses 75% de dados de treino, 5% foram reservados para validação durante o treinamento. A divisão dos dados da base ocorreu de maneira igualitária para cada classe. Dessa maneira, nenhuma classe apresenta vantagem por possuir um maior número de amostras. Os resultados obtidos são uma média de 10 rodadas de execuções para cada arquitetura de *CNN*, e em cada rodada os dados de treino e testes são permutados aleatoriamente, garantindo que não haja repetição dos grupos de imagens. As métricas finais do *holdout* são obtidas a partir da média dos resultados das 10 rodadas. Essas métricas foram: acurácia, precisão, revocação e *F1-Score*.

Para a execução das rotinas de treinamento e teste, utilizou-se um servidor com a seguinte configuração: Intel (R) Core i7-6800K @ 3.40GHz CPU, 64 GB de memória RAM e 2 placas de vídeo NVIDIA Titan Xp. A necessidade de uma máquina com essa capacidade de processamento deve-se ao fato de a CNN gerar muitos dados durante a extração das características das imagens. O tamanho das imagens com resolução 400x400 *pixels* eleva o número de dados e o uso da GPU diminui drasticamente o tempo necessário para o treinamento.

4.1 Considerações finais

Neste capítulo foram abordados com detalhes os passos apresentados na metodologia, como: descrição da base utilizada, problemas encontrados durante a segmentação e as técnicas utilizadas para corrigir as imagens, apresentação da classificação das imagens e descrição do ambiente utilizado para executar as simulações.

No capítulo seguinte serão expostos os resultados obtidos para cada arquitetura e discutido esses resultados com outros trabalhos da literatura.

5 RESULTADOS

Os resultados obtidos para as arquiteturas propostas estão disponíveis na Tabela 3. Desse modo, são apresentados os resultados com as seguintes métricas adotadas: acurácia, precisão, revocação e média F1 *score*. As arquiteturas propostas apresentam bons resultados, com taxas de acerto superiores a 96%. Para a CNN 1, com somente duas camadas de convolução, atingiu-se uma acurácia de 94,7%. Para as demais arquiteturas de CNN (2, 3 e 4) com mais camadas de convolução, as taxas superaram 96%.

Tabela 3 – Resultados obtidos com as arquiteturas de CNN propostas

	Acurácia	**Precisão**	**Revocação**	**Média F1**
CNN 1	94.71%	94.77%	94.7%	94.7%
CNN 2	96.5%	96.54%	96.49%	96.49%
CNN 3	96.83%	96.86%	96.82%	96.82%
CNN 4	96.83%	96.86%	96.82%	96.82%

Fonte: elaborado pelo autor (2019).

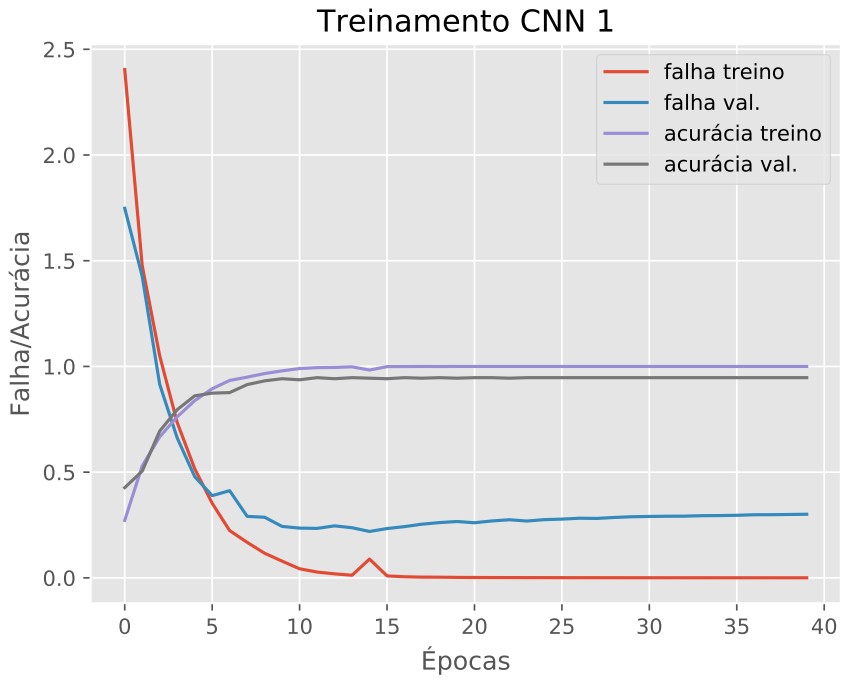
Fica claro com os resultados obtidos que a partir de 3 camadas de convolução juntamente com as camadas de *pooling* modificar esse tipo de arquitetura de rede neural não eleva a capacidade de extração de características e classificação da rede para esse problema de classificação. Fazer isso não implicará em um aumento significante na acurácia da rede, ao contrário, elevará os custos computacionais durante a etapa de treinamento, exigindo um maior uso de CPU e memória ou uso de GPU. Para complementar as métricas obtidas, foram gerados gráficos relacionando às taxas de acurácia e perda dos dados de treino e validação durante cada época do treinamento das arquiteturas. Esses gráficos utilizados neste capítulo possuem a seguinte legenda:

- *train_loss* - perda durante o treinamento, em vermelho;
- *val_loss* - perda durante a validação, em azul;
- *train_acc* - acurácia durante o treinamento, em lilás;
- *val_acc* - acurácia durante a validação, em cinza.

A Figura 23 mostra os passos de treino para a CNN 1 e a Figura 24 para a CNN 3. Comparando essas imagens, nota-se que um maior número de camadas de convolução e *pooling* proporcionam uma redução no número de épocas necessárias para a rede neural convergir, permitindo assim que a rede extraia informações das imagens de modo mais veloz e eficaz. Analisando as últimas figuras, nota-se que na Figura 23 a CNN 1 converge a partir da época 16,

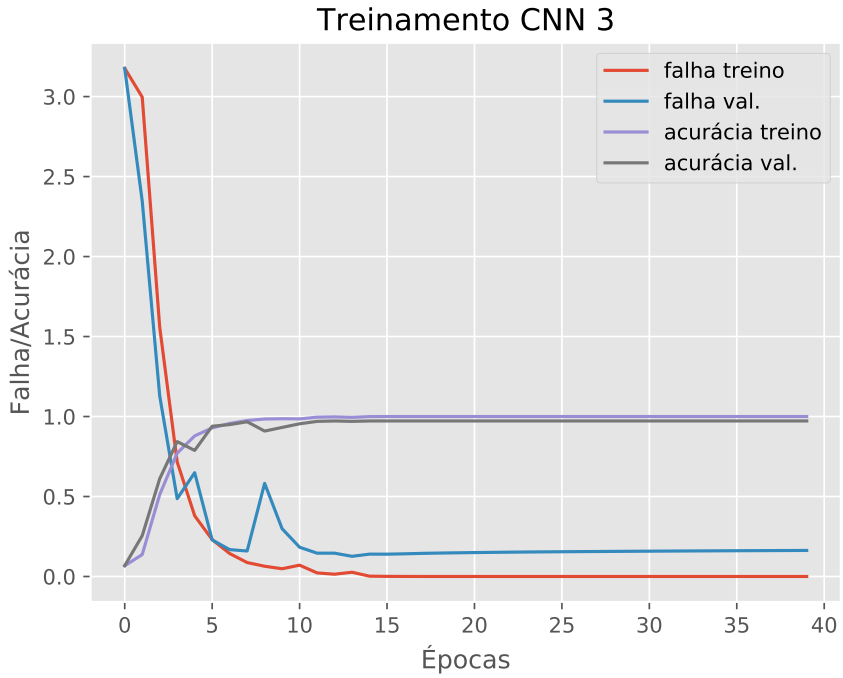
enquanto na Figura 24 a CNN 3 converge a partir da época 7.

Figura 23 – Taxas de acurácia e perda durante o treinamento e validação da CNN 1



Fonte: elaborado pelo autor (2019).

Figura 24 – Taxas de acurácia e perda durante o treinamento e validação da CNN 3



Fonte: elaborado pelo autor (2019).

Além desses resultados para as arquiteturas propostas, foram obtidos também os resultados paras as arquiteturas da literatura LeNet, InceptionResNetV2, InceptionV3, VGG-16, VGG-19, ResNet50 e DenseNet201. Os dados obtidos estão presentes na Tabela 4, nota-se que essas arquiteturas apresentaram métricas com valores superiores a 99%. Apesar do resultado, essas arquiteturas apresentam um dos casos, uma rede com mais de 200 camadas de profundidade.

Tabela 4 – Resultados obtidos com as arquiteturas de CNN da literatura

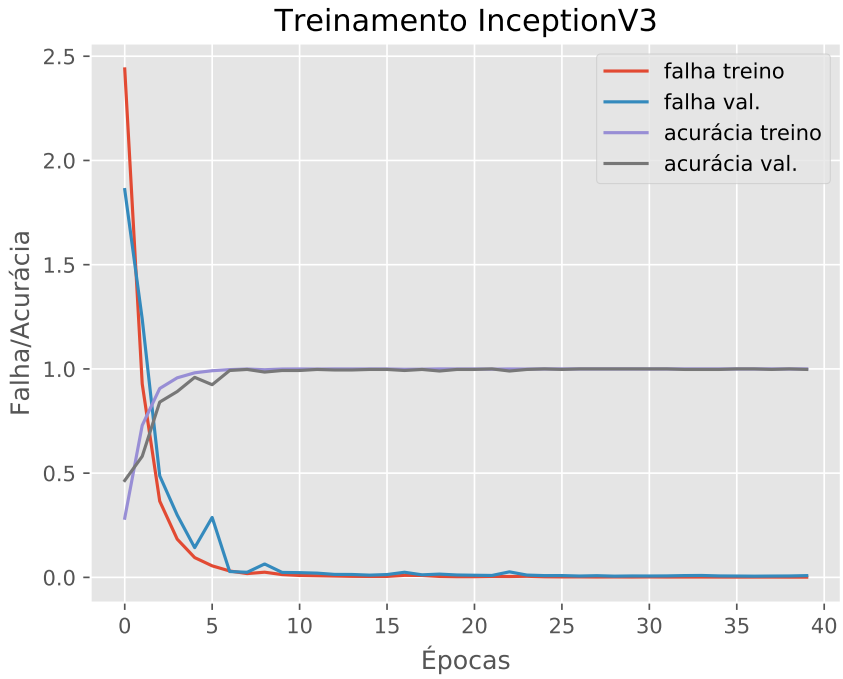
	Acurácia	**Precisão**	**Revocação**	**Média F1**
InceptionV3	99.46%	99.46%	99.45%	99.45%
Inception ResNetV2	99.62%	99.62%	99.62%	99.62%
ResNet50	97%	97.07%	96.99%	97%
Dense Net201	98.38%	98.47%	98.37%	98.37%
VGG16	97.54%	97.57%	97.54%	97.54%
VGG19	97.29%	97.32%	97.29%	97.3%
LeNet	93.54%	93.61%	93.53%	93.54%

Fonte: elaborado pelo autor (2019).

Desse modo, observa-se novamente que com o maior número de camadas, a extração de características ocorre rapidamente e apresenta um número muito menor de épocas necessárias para convergência. Assim, é possível verificar na Figura 25 para a InceptionV3 que essa rede converge na época 5, e na Figura 26 a rede ResNet50 converge na época 7, em ambos os casos os gráficos se apresentam constantes. Apesar do bom resultado, a aquisição desses dados para a base de imagens utilizada requisita o uso de uma GPU obrigatoriamente, e é necessário um tempo consideravelmente superior às demais arquiteturas propostas para as etapas treinamento e teste.

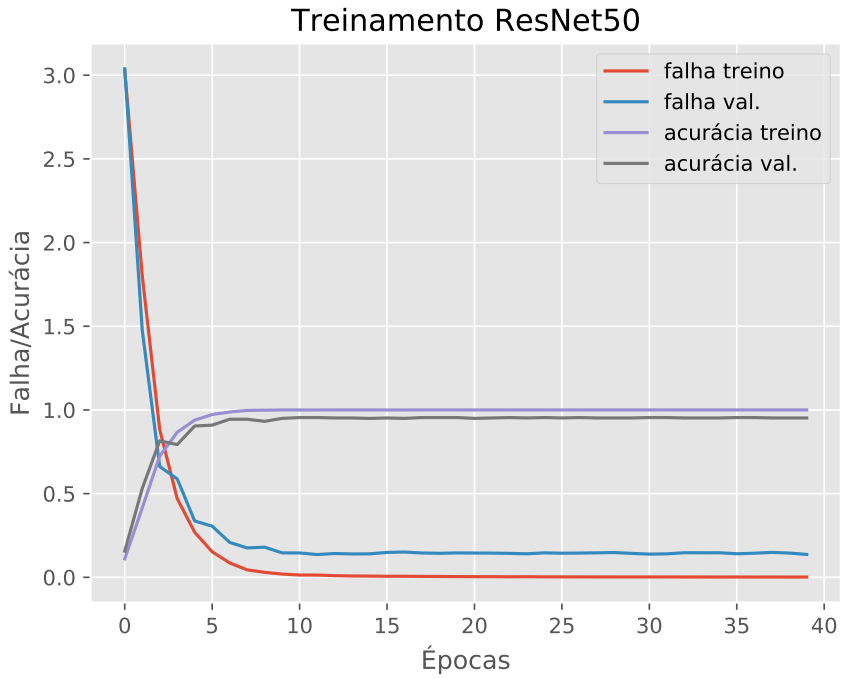
Os resultados apresentados anteriormente, na Tabela 4, utilizaram as duas bases (própria e (BARCZAK *et al.*, 2011)) em conjunto, criando um cenário o mais diferente possível entre as imagens, proporcionando uma maior variedade de informações. Além desses resultados, outras simulações ocorreram, porém utilizando as bases de imagens individualmente, das quais foram: própria e outras duas disponíveis da literatura (BARCZAK *et al.*, 2011) e (MOESLUND'S, 2002). Esses novos resultados estão dispostos nas Tabelas 5 para as arquiteturas de CNN propostas e 6 para as arquiteturas disponíveis na literatura. Analisando esses resultados, constata-se que as *CNN* são capazes de extrair as características e classificar os padrões obtidos suficientemente para alcançar as taxas de acurácia próximas a 100%, para quase todos os tipos de arquiteturas e bases de imagens. Esse comportamento das *CNN* notado motivou o uso das duas bases de imagens em conjunto, base de imagens própria e a (BARCZAK *et al.*, 2011), pois assim

Figura 25 – Taxas de acurácia e perda durante o treinamento e validação da InceptionV3



Fonte: elaborado pelo autor (2019).

Figura 26 – Taxas de acurácia e perda durante o treinamento e validação da ResNet50



Fonte: elaborado pelo autor (2019).

é possível demonstrar a robustez da metodologia proposta independente das imagens utilizadas.

Além de comparar os resultados obtidos pela metodologia proposta com outras bases e arquiteturas da literatura, também foi realizada uma comparação com outros trabalhos

Tabela 5 – Resultados obtidos para cada base de imagens individualmente, utilizando as arquiteturas propostas

	Base própria	**(BARCZAK *et al.*, 2011)**	**(MOESLUND'S, 2002)**
CNN 1	97.53%	98.09%	99.61%
CNN 2	97.74%	98.99%	99.41%
CNN 3	98.32%	98.99%	99.61%
CNN 4	98.24%	99.40%	99.41%

Fonte: elaborado pelo autor (2019).

Tabela 6 – Resultados obtidos para cada base de imagens individualmente, utilizando as arquiteturas disponíveis na literatura

	Base própria	**(BARCZAK *et al.*, 2011)**	**(MOESLUND'S, 2002)**
InceptionV3	98.44%	99.23%	99.41%
Inception ResNetV2	99.35%	99.74%	99.80%
ResNet50	98.89%	99.11%	98.24%
Dense Net201	99.27%	98.89%	99.02%
VGG16	98.10%	99.58%	99.61%
VGG19	98.32%	99.53%	99.61%
LeNet	86.15%	98.15%	97.65%

Fonte: elaborado pelo autor (2019).

relacionados que possuem uma metodologia semelhante de classificação e utilizam os mesmos gestos definidos pela *ASL*. No trabalho (OTINIANO-RODRÍGUEZ, 2012), foram utilizados os momentos de Zernike para extração de características e para classificação adotou-se o uso de um *SVM*. Nos trabalhos (ELSOUD; ELNASER, 2009) e (NGUYEN *et al.*, 2015), também foram utilizaram momentos de Zernike, mas adotou-se o uso de redes neurais *MLP* para a classificação. Por fim, nos trabalhos (OYEDOTUN; KHASHMAN, 2017), (CHEVTCHENKO *et al.*, 2018) e (RANGA *et al.*, 2018) diferentes arquiteturas de *CNN* foram utilizadas para treinamento e classificação. Nota-se que o uso da *CNN* juntamente com técnicas de processamento de imagens eficiente produz um excelente desempenho, conforme a metodologia proposta. Esse desempenho pode ser visto na Tabela 7, que demonstra as taxas de acerto obtidas são superiores aos trabalhos relacionados utilizados nessa comparação.

Os resultados obtidos com a metodologia proposta foram superiores a outras metodologia presentes na literatura: similares e as que utilizam o mesmo método de classificação de gestos, como em (OYEDOTUN; KHASHMAN, 2017). Desse modo, a acurácia de 96,83% mostra a robustez da metodologia apresentada, a importância de preparar e trabalhar as imagens antes de um processo de classificação e a necessidade de estudar e analisar os tipos de arquiteturas de redes neurais convolucionais para obter o melhor desempenho com os menores custos. Durante o uso somente da base de dados (BARCZAK *et al.*, 2011), obtiveram-se taxas

Tabela 7 – Comparação dos resultados obtidos com trabalhos relacionados

Metodologias	**Acurácia (%)**
Metodologia Proposta – base combinada	96.83%
Metodologia Proposta – base (MOESLUND'S, 2002)	99.4%
(OTINIANO-RODRÍGUEZ, 2012)	96.27%
(NGUYEN *et al.*, 2015)	94.3%
(ELSOUD; ELNASER, 2009)	90.83%
(OYEDOTUN; KHASHMAN, 2017)	91.33%
(CHEVTCHENKO *et al.*, 2018)	98.06%
(RANGA *et al.*, 2018)	97.01%

Fonte: elaborado pelo autor (2019).

de acurácia superiores aos trabalhos (CHEVTCHENKO *et al.*, 2018) e (RANGA *et al.*, 2018) que utilizam a mesma base, como pode ser visto na Tabela **??**.

Na metodologia proposta com o uso da base de imagens combinada (base de imagens própria e (BARCZAK *et al.*, 2011)) cujo objetivo é aumentar a diversidade de mãos, obteve-se uma precisão um pouco menor. Porém, apesar dessa redução, pode-se demonstrar que a metodologia apresentada é capaz de adaptar-se a uma variedade maior de imagens de gestos e, por consequência, apresentar bons resultados em aplicações em tempo real.

5.1 Considerações finais

Este capítulo discutiu os resultados obtidos para a metodologia proposta, onde apresentam-se gráficos de treinamento comparando diversas arquiteturas de *CNN* e tabelas com os resultados. Os resultados adquiridos são discutidos com outras arquiteturas e com outros trabalhos da literatura.

No capítulo seguinte serão abordadas as conclusões com o trabalho exposto e as perspectivas futuras para as quais a metodologia apresentada abre possibilidades.

6 CONCLUSÃO

Em reconhecimento de gestos, um dos grandes problemas é lidar com regiões de não-interesse das imagens e a extração de características relevantes para o algoritmo classificador. Pode-se afirmar que o fundo da imagem e os elementos de cenário compõem as regiões de não-interesse de uma imagem para reconhecimento de gestos. Na metodologia apresentada neste trabalho, o uso de redes neurais como segmentador de cores para remoção do fundo das imagens apresentou resultados satisfatórios, apesar dos ruídos presentes nas regiões de bordas e furos no interior de regiões das imagens binarizadas geradas. Esse problema pôde ser contornado utilizando operações morfológicas e a geração de contornos combinada com uma aproximação poligonal. Apresentando assim, bons resultados como um método para separar a região da mão do fundo da imagem e a remoção dos ruídos. Essa etapa é importante, pois remove objetos de imagem que não são interessantes para o método de classificação, permitindo que redes neurais convolucionais possam extrair os recursos de gestos mais relevantes por meio de suas camadas de convolução e *pooling* e, portanto, para aumentar a precisão da rede.

A proposta de realizar uma operação lógica com as imagens binarizadas das formas dos gestos e as imagens originais em cinza possibilitou a extração de características relevantes da região da palma da mão e dos dedos. Em virtude disso, as arquiteturas propostas de *CNN* alcançaram altas taxas de acertos com um custo computacional relativamente baixo, devido ao reduzido número de camadas de convolução. Dessa forma, os resultados obtidos foram superiores às metodologias mencionadas em trabalhos relacionados, apontando a relevância do método apresentado. Além disso, as arquiteturas propostas alcançaram resultados muito semelhantes às arquiteturas já definidas pela literatura, embora sejam muito mais simples e com menor custo computacional. Isso é possível devido à metodologia de processamento de imagem proposta, na qual todas as informações desnecessárias foram removidas das imagens, permitindo uma extração eficiente de características relevantes pela *CNN*.

6.1 Perspectivas Futuras

A metodologia proposta de processamento de imagens e as arquiteturas simples de *CNN* permitem a possibilidade de futuros trabalhos que investiguem o uso e a eficácia do método para a implementação de reconhecimento de gestos em dispositivos embarcados com limitações de hardware (HALFACREE; UPTON, 2012). Além disso, este trabalho aborda apenas casos de

gestos presentes em imagens estáticas, sem técnicas para a detecção e rastreamento das mãos e o tratamento de oclusão de mãos em reconhecimento utilizando vídeos. Permitindo assim, a oportunidade de novos trabalhos para o reconhecimento de gestos dinâmicos e a possibilidade do reconhecimento de gestos em tempo real de aquisição de imagens. Logo, faz-se necessário o estudos de novas técnicas de pré-processamento de dados e imagens, investigando outros métodos de segmentação de cores (XU *et al.*, 2017) (LEI *et al.*, 2016) (ZUO *et al.*, 2017) e arquiteturas de *Deep Learning* (NIE *et al.*, 2018) (HASSAN; MAHMOOD, 2018).

6.2 Produções Bibliográficas

Artigo publicado em periódico:

- **Raimundo F. Pinto Jr.**, Carlos D. B. Borges, Antônio M. A. Almeida, e Iális C. Paula, Jr., **"Static Hand Gesture Recognition Based on Convolutional Neural Networks"**, *Journal of Electrical and Computer Engineering*, vol. 2019, DOI: 10.1155/2019/4167890, 2019.

REFERÊNCIAS

BARCZAK, A. L. C.; REYES, N. H.; ABASTILLAS, M.; PICCIO, A.; SUSNJAK, T. A new 2d static hand gesture colour image dataset for asl gestures. **Research Letters in the Information and Mathematical Sciences**, v. 15, p. 12–20, 2011.

BERGH, M. V. den; CARTON, D.; NIJS, R. D.; MITSOU, N.; LANDSIEDEL, C.; KUEHNLENZ, K.; WOLLHERR, D.; GOOL, L. V.; BUSS, M. Real-time 3d hand gesture interaction with a robot for understanding directions from humans. In: **2011 RO-MAN**. [S.l.: s.n.], 2011. p. 357–362. ISSN 1944-9445.

BETANCOURT, A.; MORERIO, P.; REGAZZONI, C. S.; RAUTERBERG, M. The evolution of first person vision methods: A survey. **Circuits and Systems for Video Technology, IEEE Transactions on**, IEEE, v. 25, n. 5, p. 744–760, 2015.

CHEVTCHENKO, S. F.; VALE, R. F.; MACARIO, V.; CORDEIRO, F. R. A convolutional neural network with feature fusion for real-time hand posture recognition. **Applied Soft Computing**, v. 73, p. 748 – 766, 2018. ISSN 1568-4946.

COHEN, M. W.; ZIKRI, N. B.; VELKOVICH, A. Recognition of continuous sign language alphabet using leap motion controller. In: **2018 11th International Conference on Human System Interaction (HSI)**. [S.l.: s.n.], 2018. p. 193–199.

COUNCIL, N.; EDUCATION, D.; BEHAVIORAL, C. Board on; PRACTICE, C. **How People Learn: Brain, Mind, Experience, and School: Expanded Edition**. [S.l.]: National Academies Press, 2000. (National Research Council). ISBN 9780309070362.

Derawi, M. O.; Bours, P.; Holien, K. Improved cycle detection for accelerometer based gait authentication. In: **2010 Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing**. [S.l.: s.n.], 2010. p. 312–317.

ELSOUD, A. S. T. A.; ELNASER, O. A. Lvq for hand gesture recognition based on dct and projection features. **Journal of Electrical Engineering**, v. 60, n. 4, p. 204–208, 2009.

GE, S. S.; YANG, Y.; LEE, T. H. Hand gesture recognition and tracking based on distributed locally linear embedding. In: **2006 IEEE Conference on Robotics, Automation and Mechatronics**. [S.l.: s.n.], 2006. p. 1–6. ISSN 2158-2181.

GONZALEZ, R. C.; WOODS, R. E. **Digital Image Processing (3rd Edition)**. [S.l.]: Prentice-Hall, Inc., 2006.

GORECKY, D.; SCHMITT, M.; LOSKYLL, M.; ZÜHLKE, D. Human-machine-interaction in the industry 4.0 era. In: IEEE. **2014 12th IEEE international conference on industrial informatics (INDIN)**. [S.l.], 2014. p. 289–294.

GOUTTE, C.; GAUSSIER, E. A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. In: LOSADA, D. E.; FERNÁNDEZ-LUNA, J. M. (Ed.). **Advances in Information Retrieval**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005. p. 345–359. ISBN 978-3-540-31865-1.

HALFACREE, G.; UPTON, E. **Raspberry Pi user guide**. [S.l.]: John Wiley & Sons, 2012.

HASSAN, A.; MAHMOOD, A. Convolutional recurrent deep learning model for sentence classification. **IEEE Access**, IEEE, v. 6, p. 13949–13957, 2018.

HAYKIN, S. **Redes Neurais: Princípios e Prática**. [S.l.]: Artmed, 2007. ISBN 9788577800865.

HE, K.; ZHANG, X.; REN, S.; SUN, J. Deep residual learning for image recognition. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2016. p. 770–778.

HUANG, G.; LIU, Z.; MAATEN, L. V. D.; WEINBERGER, K. Q. Densely connected convolutional networks. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2017. p. 4700–4708.

HUANGA, W.-C. H. D.-Y.; CHANGA, S.-H. Gabor filter-based hand-pose angle estimation for hand gesture recognition under varying illumination. **Expert Systems with Applications**, Elsevier, v. 38, n. 5, p. 6031–6042, 2011.

INSTITUTE, N. **American Sign Language**. 2019. <https://www.nidcd.nih.gov/health/american-sign-language>. Acessado: 11/11/2019.

KOHAVI, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: **Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2**. [S.l.]: Morgan Kaufmann Publishers Inc., 1995. (IJCAI'95), p. 1137–1143. ISBN 1-55860-363-8.

LANGE, B.; KOENIG, S.; MCCONNELL, E.; CHANG, C.-Y.; JUANG, R.; SUMA, E.; BOLAS, M.; RIZZO, A. Interactive game-based rehabilitation using the microsoft kinect. In: IEEE. **2012 IEEE Virtual Reality Workshops (VRW)**. [S.l.], 2012. p. 171–172.

LECUN, Y.; BOTTOU, L.; BENGIO, Y.; HAFFNER, P. Gradient-based learning applied to document recognition. **Proceedings of the IEEE**, v. 86, n. 11, p. 2278–2324, Nov 1998.

LEI, Y.; YUAN, W.; WANG, H.; WENHU, Y.; BO, W. A skin segmentation algorithm based on stacked autoencoders. **IEEE Transactions on Multimedia**, IEEE, v. 19, n. 4, p. 740–749, 2016.

MACHADO, F. M. Á. **Conceitos abstratos: escolhas interpretativas de português para Libras**. [S.l.]: Appris Editora e Livraria Eireli-ME, 2017.

MOESLUND'S, T. **Gesture Recognition Database**. 2002. <http://www-prima.inrialpes.fr/FGnet/data/12-MoeslundGesture/database.html>. Acessado: 25/11/2018.

MORDVINTSEV, A. **Contour Features**. 2013. <https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_contours/py_contour_features/py_contour_features.html>. Acessado: 08/11/2019.

NGUYEN, T.-N.; HUYNH, H.-H.; MEUNIER, J. Static hand gesture recognition using principal component analysis combined with artificial neural network. **Jounal of Automation and Control Engineering**, v. 3, n. 1, p. 40–45, 2015.

NIE, S.; ZHENG, M.; JI, Q. The deep regression bayesian network and its applications: Probabilistic deep learning for computer vision. **IEEE Signal Processing Magazine**, IEEE, v. 35, n. 1, p. 101–111, 2018.

OPENCV. **How to Use Background Subtraction Methods**. 2018. <https://docs.opencv.org/master/d1/dc5/tutorial_background_subtraction.html>. Acessado: 10/11/2019.

OTINIANO-RODRÍGUEZ, G. C.-C. e. D. M. K. C. Hu and zernike moments for sign language recognition. **Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV)**, 2012.

OYEDOTUN, O. K.; KHASHMAN, A. Deep learning in vision-based static hand gesture recognition. **Neural Computing and Applications**, v. 28, n. 12, p. 3941–3951, Dec 2017. ISSN 1433-3058.

PHUNG, S. L.; BOUZERDOUM, A.; CHAI, D. Skin segmentation using color pixel classification: analysis and comparison. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 27, n. 1, p. 148–154, Jan 2005.

PINTO JÚNIOR, R. F.; DE PAULA JÚNIOR, I. C. **Static Hand Gesture ASL Dataset**. [S.l.]: IEEE Dataport, 2019.

PISHARADY, P. K.; SAERBECK, M. Recent methods and databases in vision-based hand gesture recognition. **Comput. Vis. Image Underst.**, Elsevier Science Inc., New York, NY, USA, v. 141, n. C, p. 152–165, dez. 2015. ISSN 1077-3142.

RAMER, U. An iterative procedure for the polygonal approximation of plane curves. **Computer graphics and image processing**, Elsevier, v. 1, n. 3, p. 244–256, 1972.

RANGA, V.; YADAV, N.; GARG, P. American sign language fingerspelling using hybrid discrete wavelet transform-gabor filter and convolutional neural network. **Journal of Engineering Science and Technology**, v. 13, p. 2655–2669, 09 2018.

RIOFRÍO, S.; POZO, D.; ROSERO, J.; VáSQUEZ, J. Gesture recognition using dynamic time warping and kinect: A practical approach. In: **2017 International Conference on Information Systems and Computer Science (INCISCOS)**. [S.l.: s.n.], 2017. p. 302–308.

SHERMAN, W. R.; CRAIG, A. B. **Understanding virtual reality: Interface, application, and design**. [S.l.]: Morgan Kaufmann, 2018.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. **arXiv preprint arXiv:1409.1556**, 2014.

STUTZ, D. **Recognizing Handwritten Digits using a Two-Layer Perceptron and the MNIST Dataset**. 2014. <https://davidstutz.de/recognizing-handwritten-digits-mnist-dataset-twolayer-perceptron/>. Acessado: 18/06/2018.

SUZUKI, S. *et al.* Topological structural analysis of digitized binary images by border following. **Computer Vision, Graphics, and Image Processing**, Elsevier, v. 30, n. 1, p. 32–46, 1985.

SZEGEDY, C.; IOFFE, S.; VANHOUCKE, V.; ALEMI, A. A. Inception-v4, inception-resnet and the impact of residual connections on learning. In: **Thirty-First AAAI Conference on Artificial Intelligence**. [S.l.: s.n.], 2017.

SZEGEDY, C.; LIU, W.; JIA, Y.; SERMANET, P.; REED, S.; ANGUELOV, D.; ERHAN, D.; VANHOUCKE, V.; RABINOVICH, A. Going deeper with convolutions. **CVPR 2015**, 2015.

SZEGEDY, C.; VANHOUCKE, V.; IOFFE, S.; SHLENS, J.; WOJNA, Z. Rethinking the inception architecture for computer vision. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2016. p. 2818–2826.

TECHNAVIO. **Global Robotics Market 2015-2019**. [S.l.], 2015.

TRIESCH, J.; MALSBURG, C. von der. A system for person-independent hand posture recognition against complex backgrounds. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 23, n. 12, p. 1449–1453, Dec 2001. ISSN 0162-8828.

VARGAS, A. C. G.; PAES, A.; VASCONCELOS, C. N. Um estudo sobre redes neurais convolucionais e sua aplicação em detecção de pedestres. In: CONFERENCE ON GRAPHICS, PATTERNS AND IMAGES (SIBGRAPI). São José dos Campos, 2016.

XU, G.; XIAO, Y.; XIE, S.; ZHU, S. Face detection based on skin color segmentation and adaboost algorithm. In: IEEE. **2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)**. [S.l.], 2017. p. 1756–1760.

YANG, N. A. M.-H.; TABB, M. Extraction of 2d motion trajectories and its application to hand gesture recognition. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE, v. 24, n. 8, p. 1061–1074, 2002.

ZUO, H.; FAN, H.; BLASCH, E.; LING, H. Combining convolutional and recurrent neural networks for human skin detection. **IEEE Signal Processing Letters**, IEEE, v. 24, n. 3, p. 289–293, 2017.