http://journal.unoeste.br/index.php/ce/index
DOI: 10.5747/ce.2021.v13.n2.e355
ISSN on-line 2178-8332

Colloquium Exactarum

Submetido: 15/03/2021 Revisado: 20/05/2021 Aceito:15/07/2021

APLICAÇÃO DE REDES NEURAIS CONVOLUCIONAIS NO RECONHECIMENTO DE CARACTERES EM PLACAS INFORMATIVAS JAPONESAS

APPLICATION OF CONVOLUTIONAL NEURAL NETWORKS IN THE CHARACTER RECOGNITION ON JAPANESE INFORMATIVE PLATES

Rafael Yuji Hirata Furusho[1], Francisco Assis da Silva[1], Leandro Luiz de Almeida[1], Danillo Roberto Pereira[1], Mário Augusto Pazoti[1], Almir Olivette Artero[2], Marco Antonio Piteri[2]

[1]Faculdade de Informática de Presidente Prudente, Unoeste - Universidade do Oeste Paulista, Presidente Prudente
rafayujisam@gmail.com, chico@unoeste.br, llalmeida@unoeste.br, danilopereira@unoeste.br, mario@unoeste.br
[2]Faculdade de Ciências e Tecnologia, UNESP - Universidade Estadual Paulista
Departamento de Matemática de Computação, Presidente Prudente
almir.artero@unesp.br, marco.piteri@unesp.br

RESUMO – Ao contrário da maioria dos países ocidentais, que possuem um alfabeto base derivado do Latim, o Japão possui dois alfabetos silabários denominados de Hiragana e Katakana, e um alfabeto derivado do chinês, denominado de Kanji. Por causa da grande diferença na forma de escrita desses alfabetos orientais em relação aos alfabetos ocidentais, os algoritmos de reconhecimento óptico de caracteres (OCR) baseados em alfabetos ocidentais tendem a não detectar eficientemente os caracteres japoneses. Este trabalho contribui com uma metodologia aplicando técnicas de processamento digital de imagens, como segmentação baseada em intervalos de cores, detecção de bordas e técnicas de morfologia matemática, para detectar placas informativas de trânsito japonesas, corrigir a perspectiva e segmentar os caracteres contidos nela. Foi utilizada uma rede neural convolucional para realizar a classificação de caracteres Hiragana contidos nas placas segmentadas, com uma taxa de acerto de 94,37%.
Palavras-chave: CNN, OCR, Hiragana, processamento digital de imagens, visão computacional.

ABSTRACT – Unlike most Western countries, which have a Latin-derived base alphabet, Japan has two syllabic alphabets called Hiragana and Katakana, and a Chinese alphabet, called Kanji. The vast differences in the writing of these Eastern alphabets to Western alphabets, Western alphabet-based OCR algorithms tend not to efficiently detect Japanese characters. This work contributes to a methodology applying digital image processing techniques, such as color range-based segmentation, edge detection and mathematical morphology techniques, to detect Japanese traffic informational plates correctly the perspective and segment the characters contained in it. A convolutional neural network was used to perform the classification of Hiragana characters contained in the segmented plates, with accuracy of 94.37%.
Keywords: CNN, OCR, Hiragana, digital image processing, computer vision.

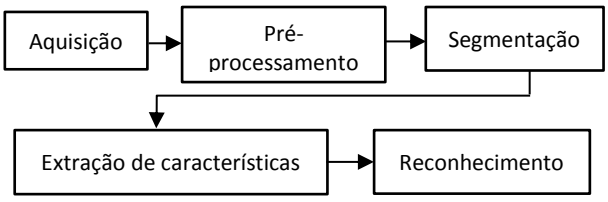
1. INTRODUÇÃO

O Japão é um país historicamente rico, com belas paisagens naturais e um grande polo tecnológico, tornando-o atrativo para turistas e estrangeiros buscando novas oportunidades de emprego. Durante o ano de 2017, de acordo com os dados emitidos pela *Japan National Tourism Organization* (JNTO, 2018), o Japão recebeu cerca de 28.691.100 turistas. Sabendo que o Japão utiliza de um conjunto de caracteres e ideogramas próprios, que se diferencia da maioria dos outros países, a utilização de ferramentas de tradução como dicionários, se torna inviável por necessitar que o turista reconheça os caracteres do alfabeto japonês.

O processo de detecção e reconhecimento de placas de sinalização de trânsito ainda é um processo desafiador, e possui uma variedade de aplicações para este tipo de tecnologia (GREENHALGH; MIRMEHDI, 2014). Sistemas de apoio ao motorista, levantamento de estruturas de rodovias e estradas e sistemas de carros autônomos são exemplos de tecnologia que fazem uso de processos de detecção e reconhecimento automático de placas de trânsito.

Com a necessidade de agilizar processos que envolvem digitação ou leitura de escritas contidas em imagens, como, por exemplo, documentos digitalizados (MANTAS, 1986) e placas de licenciamento veicular (ANAGNOSTOPOULOS et al., 2008), aplicações de OCR (*Optical Character Recognition*) surgiram como forma de solução para esses problemas. Os algoritmos de OCR buscam simular a capacidade humana de leitura, identificando símbolos gráficos que compõem um alfabeto. É possível esquematizar as etapas de um OCR por meio de um diagrama de blocos de um sistema de visão artificial, como está apresentado na Figura 1.

Figura 1. Esquematização em diagrama de blocos das etapas de um OCR baseado em um Sistema de Visão Artificial.



Fonte: (Autor, 2019).

A partir da aquisição da imagem, é realizado um pré-processamento para remoção de elementos na imagem que interferem no reconhecimento e segmentação do objeto de interesse em relação ao fundo. Após isso, realiza-se a extração de características, para então realizar a classificação do objeto.

Pelo fato de existirem um grande número de variações do alfabeto japonês, os algoritmos OCR ocidentais não possuem bom desempenho (DAS; BANERJEE, 2015). Parte dessa dificuldade está relacionada ao grande número de caracteres pertencente ao alfabeto japonês, o que dificulta a segmentação e o levantamento de características classificadoras. As redes neurais convolucionais (*Convolutional Neural Network* – CNN) surgiram como uma solução para reconhecer textos manuscritos, principalmente dígitos e caracteres chineses pela sua capacidade de aprendizado supervisionado utilizado camadas de convolução (TSAI, 2016). Este trabalho objetivou o desenvolvimento de uma metodologia de OCR para identificação de caracteres contidos em placas informativas japonesas e apresentar seu conteúdo no alfabeto romano. Para isso, a metodologia desenvolvida foi dividida em três etapas, segmentação da região da placa e correção da perspectiva, segmentação dos caracteres, e classificação de caracteres.

O trabalho está organizado da seguinte maneira. Na Seção 2 são descritos os trabalhos relacionados com suas técnicas e resultados, que serviram de base para a elaboração deste trabalho. Na Seção 3 é apresentada uma metodologia para resolver o problema de detectar e classificar os caracteres contidos em placas informativas japonesas. A Seção 4 apresenta os resultados obtidos a partir da metodologia desenvolvida. Por fim, na Seção 5 são feitas as considerações finais e propostas de trabalhos futuros.

2. TRABALHOS RELACIONADOS

Segundo Anagnostopoulos et al. (2008), algoritmos de reconhecimento de placas de licenciamento possuem basicamente três etapas de processamento, a primeira consiste em localizar a região da placa, em seguida segmentar os caracteres contidos nessa região, para que na etapa final, realize o reconhecimento dos caracteres. Pelo fato de as imagens utilizadas, geralmente, apresentar grandes variações de iluminação, diferentes resoluções e formatos do

objeto desejado, estes algoritmos atuam apenas em situações específicas.

Oliveira (2010) apresentou um método para localizar e reconhecer caracteres em placas de licenciamento veicular. O processo foi dividido em três módulos principais, que trabalham com a entrada de uma imagem em escala de cinza e retorna sete caracteres extraídos dessa imagem. A primeira etapa consistiu em localizar a região da placa por meio da concentração de mudança brusca de intensidade de pixels. A segunda etapa visou segmentar os caracteres da placa, utilizando operações morfológicas e aplicação de filtros para eliminar ruídos e destacar os caracteres. Por fim, na última etapa, cada caractere segmentado é analisado comparando ele com uma série de máscaras para classificá-lo em um dos 36 caracteres alfanuméricos.

Gao et al. (2006) apresentaram uma proposta para segmentação de placas de trânsito usando cores e características de formas, extraídas baseando-se em um modelo de visão humana. No trabalho, foi utilizado o modelo de cor CIECAM97, para extração das cores utilizadas na segmentação e classificação das placas, e para a extração de características, foi utilizado o modelo FOSTS (*Foveal System for Traffic Signs*). O modelo de cor CIECAM97 apresentou ser eficaz para a segmentação das placas em diversas situações, enquanto o modelo FOSTS colaborou com a classificação de placas separadas por cores.

Greenhalgh e Mirmehdi (2012) propuseram um sistema de detecção e reconhecimento de placas de trânsito em tempo real, que detecta regiões candidatas das placas utilizando o algoritmo MSER (*Maximally Stable Extremal Regions*), o que oferece robustez às variações das condições de iluminação. No algoritmo de reconhecimento das placas de trânsito os autores utilizaram uma cascata de SVM (*Support Vector Machine*), treinadas utilizando o algoritmo HOG (*Histogram Of Gradients*). Para realizar o treinamento, foram utilizados modelos sintéticos disponibilizados *online*. Segundo os autores, o sistema proposto apresentou ser eficiente em situações com clima diversificado e em altas velocidades do veículo.

Rodrigues (2012) propôs um protótipo de arquitetura para detectar e reconhecer placas de sinalização utilizando redes neurais. A arquitetura proposta é dividida em duas partes, detecção e classificação, respectivamente. A detecção utiliza um módulo baseado em saliência construído por meio de uma pirâmide Gaussiana e operadores locais de orientação, atingindo uma taxa de acerto de 84%. A classificação, em contrapartida, não obteve resultados satisfatórios ao utilizar uma rede neural monolítica, por causa da variabilidade das imagens geradas pelo algoritmo de detecção.

Rosa (2017) apresentou um trabalho de um sistema de reconhecimento automático de placas de sinalização de trânsito, que possui cinco etapas: aquisição da amostra, seguida de um pré-processamento com o uso do espaço cor HSV (*Hue*, *Saturation*, *Value*) e técnicas como limiarização (*threshold*), morfologia matemática (CANDEIAS, 1997) e detectores de borda. A segmentação das placas foi realizada utilizando a transformada de Hough generalizada. A extração de características das placas segmentadas foi realizada por meio do método HOG (*Histogram Of Gradients*), para então classificá-la com o uso de SVM's (*Support Vector Machine*). O experimento utilizou duas formas de aquisição de imagens, a primeira constituída de apenas imagens estáticas e a segunda de imagens capturadas a partir de um carro em movimento. O processo de segmentação obteve uma taxa de sucesso de 70%, enquanto que a etapa de classificação obteve uma taxa de sucesso de 90% no reconhecimento das placas.

3. METODOLOGIA DESENVOLVIDA

3.1. Visão Geral

As placas informativas japonesas possuem uma grande variedade de formatos, o que dificulta o processo de detecção em imagens. Por este motivo, a segmentação necessita funcionar de forma genérica para tratar questões de posição e formato da placa na imagem. Durante o processo de captura, as imagens podem apresentar uma variação de iluminação, e as placas podem ser imageadas em diferentes pontos de vista. Para minimizar esses efeitos, um pré-processamento da imagem torna-se necessário, desta forma, busca-se remover imperfeições na imagem, tais como a presença de pixels ruidosos e brilho/contraste inadequados.

Foi utilizada uma suavização gaussiana na imagem de entrada para remover ruídos, e em seguida foi obtida uma imagem binária a partir dos canais de cor *H* (*Hue*) e *S* (*Saturation*) do espaço de cores HSV aplicando uma condição baseada em limites de intervalo da cor azul, definida empiricamente. Como as placas possuem uma cor predominante na tonalidade

azul, esta cor foi utilizada para separar a região da placa do fundo e para obter o contorno da região da placa (imagem binária).

Por causa possibilidade da existência de linhas divisórias nas placas, foi necessário utilizar a operação de morfologia matemática denominada de fechamento para eliminar essas linhas e unir as partes.

Para detectar o contorno da placa foi utilizado o algoritmo Canny (CANNY, 1986), seguido de um detector de contornos (SUZUKI; ABE, 1985). Os contornos encontrados são filtrados, e é selecionado apenas o de maior dimensão, que representa o contorno da placa.

Os diferentes pontos de vista de captura de imagens provocam a necessidade de executar uma correção de perspectiva da placa na imagem, em que se faz necessário detectar os cantos da placa para realizar essa correção. Os caracteres contidos na placa são então segmentados utilizando operações de morfologia matemática para unir partes dos caracteres, e então é aplicado um detector de bordas, seguido de um detector de contornos, para localizar os caracteres.

Para realizar a classificação dos caracteres foi utilizada uma rede neural convolucional com sete camadas de profundidade. O treinamento foi realizado utilizando um *dataset* contendo 75 tipos de caracteres do alfabeto Hiragana escrito à mão com 160 variações cada.

3.2. Recursos Utilizados

Para o desenvolvimento do trabalho foi utilizado um computador com processador Intel Core i7 5500U de 2.4GHz, 8GB de memória RAM, e uma placa de vídeo NVIDIA GeForce 920M com 4GB de memória de vídeo.

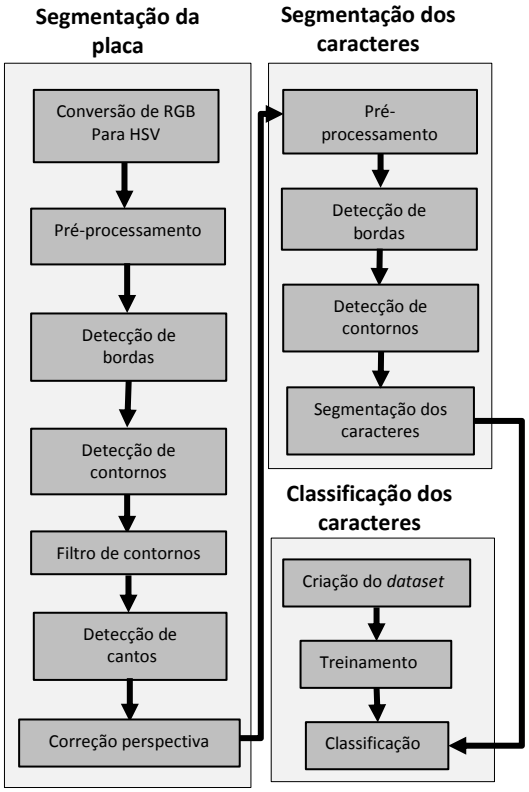
As imagens utilizadas para os experimentos foram adquiridas por meio de capturas de tela do Google Street View, de algumas ruas e rodovias do Japão. Foram obtidas 50 imagens contendo em cada imagem uma placa informativa japonesa.

Para a programação dos algoritmos da metodologia desenvolvida neste trabalho foi utilizada a linguagem de programação Python, bem com a biblioteca de visão computacional OpenCV, bibliotecas Keras e TensorFlow para a criação e treinamento da rede neural convolucional.

3.3. Métodos

O trabalho foi dividido em três etapas inspirado na metodologia proposta por Anagnostopoulos et al. (2008), que dividia o processo em localizar a região da placa, segmentar os caracteres contidas nela, e em seguida, realizar a classificação dos caracteres. A linha de raciocínio do autor, empregada para placas de licenciamento veicular, foi adaptada, neste trabalho, para a utilização em placas informativas de trânsito japonesas, seguindo o fluxograma representado na Figura 2.

Figura 2. Fluxograma de processos da metodologia desenvolvida neste trabalho.



Fonte: (Autor, 2019).

3.3.1. Segmentação das placas

O sistema de sinalização de autoestradas japonesas conta com placas de via expressa de duas tonalidades: verde e azul, sendo escolhidas as placas de coloração azul para este trabalho. Pela necessidade de se destacar no campo de visão do motorista, as placas informativas possuem alta saturação (canal *S* do espaço de cores HSV), fazendo com que a cor azul seja apresentada de forma intensa. Essa coloração azul das placas é definida pela JIS Z 9103 (JSAA, 2019), que corresponde aos valores R:25, G:113, B:255 no espaço de cor RGB (*Red*, *Green*, *Blue*).

Visto que regiões que apresentam uma concentração de pixels com alto nível de saturação podem pertencer à região da placa, elas se tornam candidatas a ser região de interesse.

Para realizar a conversão da imagem do espaço de cor RGB para o espaço de cor HSV, são utilizadas as fórmulas dadas pelas Equações 1, 2 e 3 (SHAIK et al., 2015).

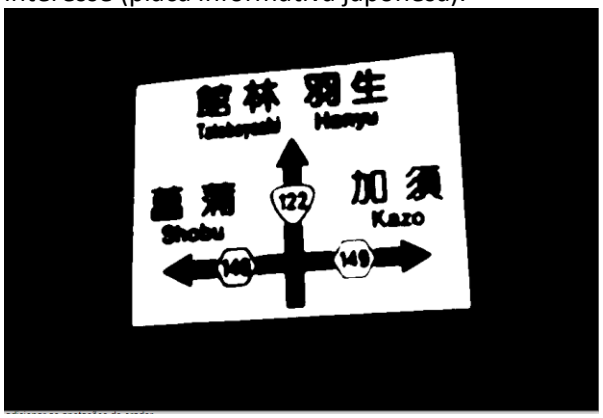
$$H = arccos \frac{\frac{1}{2}(2R - G - B)}{\sqrt{(R-G)^2 - (R-G)(G-B)}} \quad (1)$$

$$S = \frac{\max(R,G,B) - \min(R,G,B)}{\max(R,G,B)} \quad (2)$$

$$V = \max(R,G,B) \quad (3)$$

Um limite de intervalo da cor azul foi definido baseado nos canais *H* e *S*, para destacar a placa informativa japonesa contida na imagem (SOLOMON; BRECKON, 2011). O resultado da aplicação dessa técnica produz uma imagem binária contendo a placa (região de interesse) destacada do fundo com todos os pixels zerados (mudados para a cor preta), como está apresentado na Figura 3.

Figura 3. Exemplo da binarização da região de interesse (placa informativa japonesa).



Fonte: (Autor, 2019).

Na imagem binária (Figura 3) são aplicadas operações de morfologia matemática para diminuir a quantidade de ruídos, que podem interferir na análise da imagem, e unir as possíveis partes da placa separadas por divisões internas. Foram utilizadas operações de morfologia matemática erosão e dilatação, com um elemento estruturante 5x5 preenchido e origem no centro.

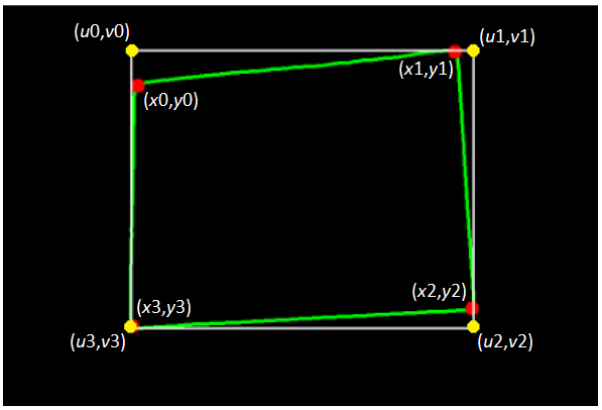
Utilizou-se o algoritmo de detecção de bordas Canny (CANNY, 1986) para detectar todas as bordas contidas na imagem. Em seguida, um algoritmo de detecção de contornos (SUZUKI; ABE, 1985) foi utilizado com o objetivo de encontrar todos os contornos fechados. Os contornos são filtrados, selecionando o contorno com maior área detectada.

No momento da aquisição da imagem, o ângulo de captura pode causar distorções de perspectivas. Essas distorções interferem nas etapas de detecção, segmentação e reconhecimento dos caracteres da placa.

Foi utilizada uma matriz de transformação (REYES et al., 2015), para corrigir essa distorção de perspectiva. Para isso, foi aplicado um algoritmo de detecção de cantos *Harris Corner Detector* (HARRIS; STEPHENS, 1988), obtendo os valores das coordenadas *x* e *y* dos quatro cantos da placa, que são utilizados na matriz de transformação.

Esta matriz de transformação utiliza as coordenadas (*x*0, *y*0), (*x*1, *y*1), (*x*2, *y*2) e (*x*3, *y*3) dos pontos adquiridos pelo detector de cantos e mais quatro pontos da região de interesse (*u*0, *v*0), (*u*1, *v*1), (*u*2, *v*2) e (*u*3, *v*3), gerados com base no retângulo mínimo do contorno da placa, como mostra a Figura 4.

Figura 4. Contorno da região da placa, com os cantos destacados em vermelho e o retângulo mínimo com os cantos em amarelo.



Fonte: (Autor, 2019).

Essa transformação é realizada utilizando a Equação 4, e as coordenadas da placa (x_n, y_n) (elemento *A* da equação), para se obter as coordenadas do retângulo mínimo (u_n, v_n) (elemento *B* da equação).

$$A.X = B \quad (4)$$

O elemento *X* representa um coeficiente desconhecido, utilizado para completar a matriz de transformação. A Equação 4 pode ser representada de forma expandida conforme a Equação 5.

$$\underbrace{\begin{bmatrix} x0 & y0 & 1 & 0 & 0 & 0 & -x0u0 & -y0u0 \\ x1 & y1 & 1 & 0 & 0 & 0 & -x1u1 & -y1u1 \\ x2 & y2 & 1 & 0 & 0 & 0 & -x2u2 & -y2u2 \\ x3 & y3 & 1 & 0 & 0 & 0 & -x3u3 & -y3u3 \\ 0 & 0 & 0 & x0 & y0 & 1 & -x0v0 & -y0v0 \\ 0 & 0 & 0 & x1 & y1 & 1 & -x1v1 & -y1v1 \\ 0 & 0 & 0 & x2 & y2 & 1 & -x2v2 & -y2v2 \\ 0 & 0 & 0 & x3 & y3 & 1 & -x3v3 & -y3v3 \end{bmatrix}}_{\mathbf{A}} * \underbrace{\begin{bmatrix} a0 \\ a1 \\ a2 \\ b0 \\ b1 \\ b2 \\ c0 \\ c1 \end{bmatrix}}_{\mathbf{X}} = \underbrace{\begin{bmatrix} u0 \\ u1 \\ u2 \\ u3 \\ v0 \\ v1 \\ v2 \\ v3 \end{bmatrix}}_{\mathbf{B}} \quad (5)$$

Para se obter o valor da matriz *X*, utiliza-se a decomposição LU (*Lower* e *Upper*) (BUNCH; HOPCROFT, 1974), o qual, resolve um sistema linear separando a matriz em duas partes, como consta na Equação 6.

$$A = L.U \quad (6)$$

O método de eliminação de Gauss (KAW; KALU; NGUYEN, 2018) zera todos os elementos abaixo da diagonal principal, obtendo então a matriz *U*. A matriz *L* por sua vez é formada com os valores dos multiplicadores utilizados para zerar os elementos da matriz *U* respectivamente, tendo os valores da diagonal principal acima, zerados. Após obter as matrizes *L* e *U*, substitui-se a matriz *A* pelas matrizes *L* e *U* formando $L.(U.X) = B$, resultado na Equação 7:

$$L.Y = B \quad (7)$$

Onde *Y* é:

$$U.X = Y \quad (8)$$

Resolvendo a Equação 7, os valores desconhecidos dos coeficientes são encontrados.

$$ui = \frac{(a0*xi+a1*yi+a2)}{c0*xi+c1*yi+1} \quad (9)$$

$$vi = \frac{(b0*xi+b1*yi+b2)}{c0*xi+c1*yi+1} \quad (10)$$

A partir das Equações 9 e 10 é possível obter o novo valor para as coordenadas (*x*, *y*) transformando-as e obtendo-se então (*u*, *v*). Ao aplicar para todos os pixels da imagem, a placa é distorcida, corrigindo a perspectiva, como ilustrado na Figura. 5.

Figura 5. Correção de perspectiva. a) Placa deformada. b) Placa com a perspectiva corrigida.



a) b)

Fonte: (Autor, 2019).

3.3.2. Segmentação dos caracteres

Após corrigir a região de interesse, são extraídos os caracteres presentes na placa. Para realizar a segmentação dos caracteres da placa foi utilizado o algoritmo de limiarização de Otsu (OTSU, 1979), o qual separa os caracteres do fundo. Para tratar as partes desconexas dos caracteres, foi utilizada uma operação de morfologia matemática, dilatação, que utiliza um elemento estruturante 5x5 (o mesmo utilizado na Seção 3.3.1). Após realizada a operação de dilatação, aplicou-se o mesmo algoritmo de detecção de bordas usado na segmentação da região de interesse, Canny (CANNY, 1986) e o algoritmo de detecção de contornos (SUZUKI; ABE, 1985), para localizar todos os contornos dentro da região de interesse.

Ao obter os contornos, um retângulo mínimo para cada contorno detectado é determinado, definindo os valores de x_{min}, y_{min}, x_{max} e y_{max}.

Pela presença de outros objetos dentro da região de interesse como setas, números e símbolos, foi utilizada uma filtragem determinada de forma empírica para encontrar os caracteres, como mostra o sistema de Inequações 11.

$$\begin{cases} 11 \geq P \leq 26 \\ R \geq 85 \end{cases} \quad (11)$$

onde *P* representa a proporção entre a altura do objeto em relação à altura da placa, e *R* a proporção entra a altura do objeto em relação a sua largura, representados respectivamente pelas Equações 12 e 13.

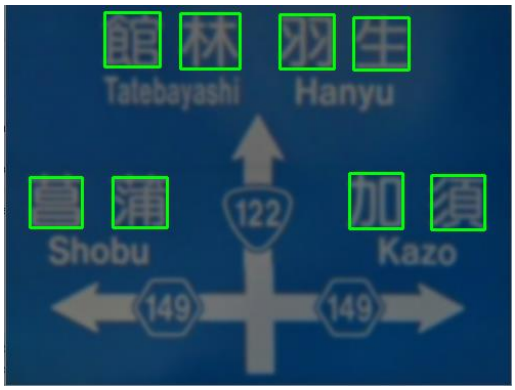
$$P = \frac{h * 100}{H} \quad (12)$$

$$R = \frac{h * 100}{w} \quad (13)$$

Onde *h* representa a altura do objeto, *H* a altura da placa e *w* a largura do objeto.

Ao aplicar o filtro dado pelo sistema de Inequações 11, os caracteres que estiverem fora deste intervalo são desconsiderados (Figura 6).

Figura 6. Contorno dos caracteres detectados.



Fonte: (Autor, 2019).

3.3.3. Classificação dos caracteres

Devido à grande variação de atributos dos caracteres japoneses, em especial o Kanji, é necessária uma solução que seja capaz de levantar e analisar esses atributos sem interferência humana, tornando as Redes Neurais Convolucionais (CNN) (ALBAWI; MOHAMMED; AL-ZAWI, 2017) uma solução eficiente para esse problema. Isso se dá porque dispensam a necessidade de realizar um levantamento manual de características classificadoras, e são capazes de aprender de forma supervisionada, os padrões para o reconhecimento dos caracteres japoneses.

Redes neurais convolucionais possuem camadas (*Layers*) que executam tarefas específicas. As camadas de convolução são responsáveis por definir os filtros pelos quais realizam o aprendizado da rede, usando pesos de ativação com base nos atributos da imagem.

As camadas de ativação aumentam o poder representacional do modelo através de operações elementares, como, por exemplo, a função de ativação ReLu (*Rectified Linear Unit*), que zera valores negativos e mantem constantes os valores positivos, evitando que pequenas alterações gerem grandes impactos na rede.

As camadas de *pooling* reduzem o tamanho de representação e a quantidade de parâmetros. Isto torna os mapas de ativação gerados pela camada de convolução invariantes à rotação, translação e escala.

As camadas de convolução, ativação e *pooling* podem se repetir de forma intercalada e compõem a primeira etapa da CNN, que realiza a extração de características.

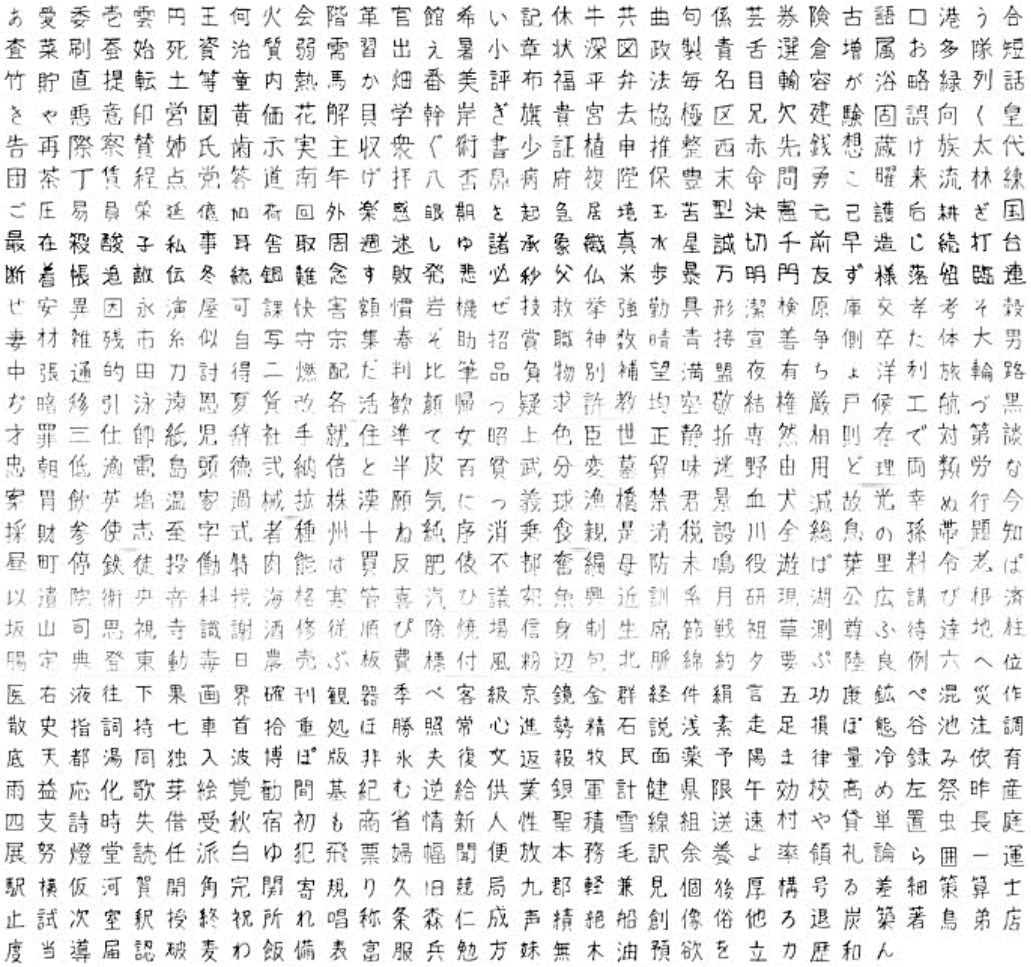
A segunda etapa da CNN possui, como camada de entrada, um vetor do mapa de ativação gerado pela primeira etapa, seguido de camadas densamente conectadas e funções de ativação. A camada de saída da CNN é responsável por classificar o valor de entrada em uma das classes da rede.

A arquitetura proposta para este trabalho possui sete níveis de profundidade, sendo três de convolução, intercaladas com camadas de *pooling*. O primeiro nível de convolução gera 64 mapas de ativação, e utiliza um *kernel* de dimensão 3x3 e uma função de ativação ReLu. A segunda, terceira e quarta camadas geram 128, 192 e 256 mapas de ativação, respectivamente, e possuem um *kernel* de dimensão 3x3, função de ativação ReLu e um *dropout* de 0,25. *Dropout* serve para eliminar pesos aleatórios para que o treinamento evite um treinamento excessivo da rede (*overfitting*).

A camada densamente conectada possui duas camadas de 1024 neurônios com função de ativação ReLu e *dropout* de 0,5. A última camada da rede densamente conectada possui 75 neurônios e uma função de ativação *softmax*.

Para o treinamento da rede, foram utilizadas imagens retiradas do *dataset* fornecido pelo Eletrotechnical Laboratory (ETL, 2019), que contém uma grande coleção de imagens com caracteres japoneses escritos à mão. O *dataset* escolhido para este projeto foi o ETL-8G (Figura 7), que possui 12000 imagens de caracteres Hiragana escritos à mão, divididos em 75 classes diferentes com 160 variações. Estas imagens passaram por um processo de binarização antes da realização do treinamento.

Figura 7. Imagem com os primeiros 956 registros do dataset ETL-8G.



Fonte: (ETLCDB, 2019).

A Figura 8 apresenta a arquitetura da rede utilizada neste trabalho.

Figura 8. Arquitetura da rede.

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 64, 64, 64)	640
activation_1 (Activation)	(None, 64, 64, 64)	0
max_pooling2d_1 (MaxPooling2	(None, 32, 32, 64)	0
conv2d_2 (Conv2D)	(None, 32, 32, 128)	73856
activation_2 (Activation)	(None, 32, 32, 128)	0
max_pooling2d_2 (MaxPooling2	(None, 16, 16, 128)	0
dropout_1 (Dropout)	(None, 16, 16, 128)	0
conv2d_3 (Conv2D)	(None, 16, 16, 192)	221376
activation_3 (Activation)	(None, 16, 16, 192)	0
max_pooling2d_3 (MaxPooling2	(None, 8, 8, 192)	0
dropout_2 (Dropout)	(None, 8, 8, 192)	0
conv2d_4 (Conv2D)	(None, 8, 8, 256)	442624
activation_4 (Activation)	(None, 8, 8, 256)	0
max_pooling2d_4 (MaxPooling2	(None, 4, 4, 256)	0
dropout_3 (Dropout)	(None, 4, 4, 256)	0
flatten_1 (Flatten)	(None, 4096)	0
dense_1 (Dense)	(None, 1024)	4195328
activation_5 (Activation)	(None, 1024)	0
dropout_4 (Dropout)	(None, 1024)	0
dense_2 (Dense)	(None, 1024)	1049600
activation_6 (Activation)	(None, 1024)	0
dropout_5 (Dropout)	(None, 1024)	0
dense_3 (Dense)	(None, 75)	76875
activation_7 (Activation)	(None, 75)	0

Fonte: (Autor, 2019).

4. EXPERIMENTOS

Para a realização dos experimentos, imagens de placas informativas japonesas foram obtidas por meio de captura de tela utilizando o Google Street View, em regiões do Japão (Figura 9). Para cada imagem, foi aplicado um pré-processamento, a placa foi segmentada (Figura 10) e os caracteres segmentados (Figura 11), conforme descrito na Seção 3. Ao todo foram obtidas 50 capturas de tela em diferentes condições de iluminação e pontos de vista. Além das placas, caracteres do alfabeto Hiragana também foram segmentados, para testar a eficiência da rede neural.

Os resultados obtidos na fase de segmentação das placas e dos caracteres foram satisfatórios, como mostrado na Tabela 1.

Tabela 1. Resultados obtidos na etapa de segmentação das placas e dos caracteres.

Frames	Acertos	Erros	Precisão
Segmentação das placas	44	6	88%
Segmentação dos caracteres	302	22	93,21%

Fonte: (Autor, 2019).

Figura 9. Captura da imagem de uma placa obtida pelo Google Street View.



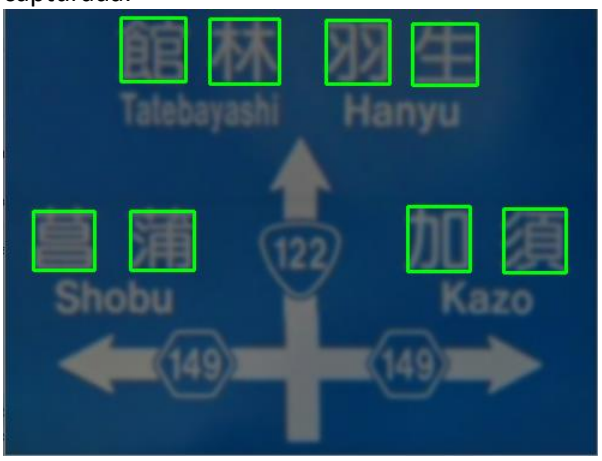
Fonte: (Autor, 2019).

Figura 10. Resultado do pré-processamento e segmentação da placa realizado sobre a imagem capturada.



Fonte: (Autor, 2019).

Figura 11. Caracteres segmentados da placa capturada.



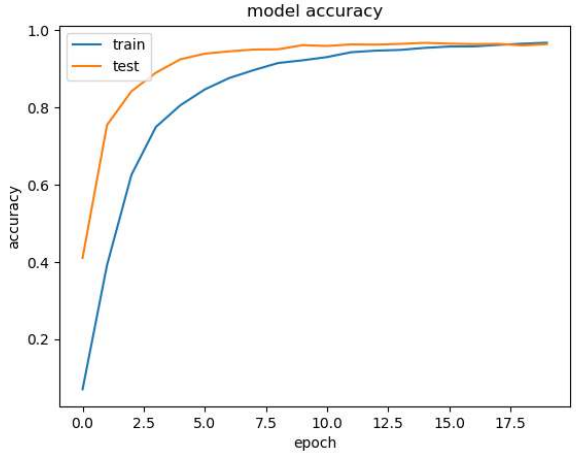
Fonte: (Autor, 2019).

Para o processo de classificação, quatro tipos diferentes de arquiteturas foram testados, baseando-se nos modelos propostos por Charles Tsai (TSAI, 2016), de classificação de caracteres japoneses manuscritos.

A primeira arquitetura testada possui 16 camadas de profundidade, e demonstrou não ser prática por conta do elevado custo de treinamento e resultados não satisfatórios. A segunda rede possui 11 camadas de profundidade, e apresentou ser razoavelmente rápida, mas com problemas de *overfitting*. A terceira e quarta redes possuem sete camadas de convolução, apresentando desempenhos similares, sendo a última (quarta rede), a escolhida para este trabalho, pelo fato da velocidade de treinamento ser superior a anterior (terceira rede).

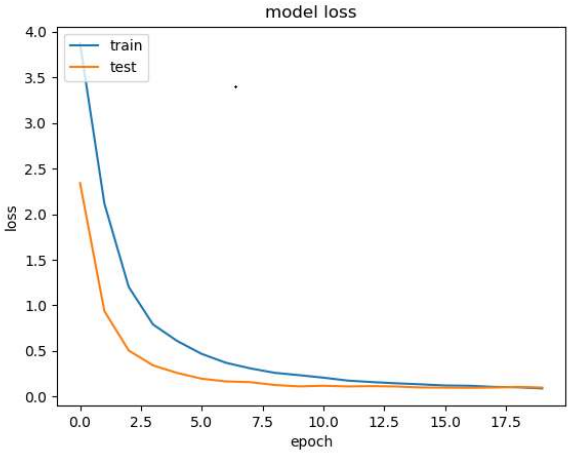
Foram realizadas 20 épocas de treinamento, com um lote (*batch*) de tamanho 16. Uma função de otimização Adam (KINGMA; BA, 2014) foi utilizada no processo de treinamento, o que demonstrou ser eficaz para problemas de convergência da rede, apresentando o desempenho mostrado nos gráficos das Figuras 12 e 13.

Figura 12. Gráfico de desempenho de acurácia da rede durante o treinamento e validação.



Fonte: (Autor, 2019).

Figura 13. Gráfico de desempenho de perda da rede durante o treinamento e validação.



Fonte: (Autor, 2019).

O processo de classificação teve resultados satisfatórios. Durante o treinamento da rede, a taxa de perda foi de 9,2% e a taxa de acurácia atingiu o valor 97,17%. Para os testes com imagens dos caracteres retirados da placa, a rede apresentou um resultado satisfatório, atingindo uma taxa de acerto de 94,37%.

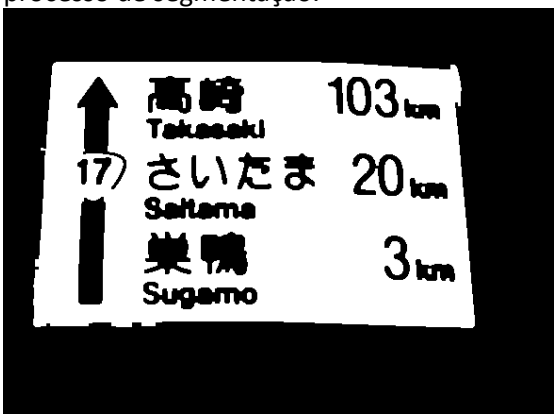
5. CONSIDERAÇÕES FINAIS

Este trabalho objetivou o desenvolvimento de uma metodologia para o reconhecimento ótico de caracteres em placas informativas japonesas. Apesar das limitações de hardware, o processo de classificação obteve uma boa taxa de acerto, comprovando ser uma alternativa promissora para projetos futuros com este tema.

O processo de segmentação apresentou resultados satisfatórios, mas que podem ser melhorados. Como apresentado anteriormente, algoritmos OCR para placas informativas de trânsito possuem limitações por causa das variações do ambiente. Os erros ocorridos durante as etapas de segmentação da placa e dos caracteres foram causados principalmente por níveis de iluminação difíceis de serem tratados de forma única. O processo de segmentação em si funciona de maneira eficaz, sendo capaz de binarizar a região da placa em relação ao fundo. Entretanto, algumas regiões podem ficar falhas, ocasionando erros no processo de detecção de cantos, como apresentado na Figura 14.

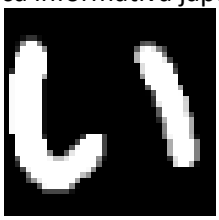
Os problemas detectados durante o processo de segmentação dos caracteres foram ocasionados pela presença de letras com partes desconexas entre si, segmentando o elemento em duas partes menores. Na Figura 15 é apresentado um exemplo de uma letra “i” do alfabeto Hiragana que ocasionou erro na segmentação por ter duas partes desconexas.

Figura 14. Placa com bordas defeituosas no processo de segmentação.



Fonte: (Autor, 2019).

Figura 15. Caractere “i” do alfabeto Hiragana retirado de uma placa informativa japonesa.



Fonte: (Autor, 2019).

O processo de classificação utilizando redes neurais convolucionais demonstrou ser eficiente, atingindo uma taxa de acerto no reconhecimento de 94,37% para caracteres retirados das imagens das placas. Os erros ocorridos neste processo são ocasionados

principalmente pela similaridade entre algumas letras.

Para evitar erros como este, poderia ser utilizado um *dataset* composto por imagens de letras extraídas diretamente de placas informativas japonesas ao invés de caracteres manuscritos, como foi o caso deste trabalho.

Outro trabalho de grande valia seria a implementação de janelas deslizantes no processo de segmentação dos caracteres. Dessa forma, cada parte da placa seria submetida a uma rede neural, que a classificaria em seu tipo (letra, símbolo ou número), para em seguida, ser classificado em qual classe de caractere pertence.

REFERÊNCIAS

ALBAWI, S.; MOHAMMED, T. A.; AL-ZAWI, S. Understanding of a convolutional neural network. In: INTERNATIONAL CONFERENCE ON ENGINEERING AND TECHNOLOGY (ICET). 2017. https://doi.org/10.1109/ICEngTechnol.2017.8308186

ANAGNOSTOPOULOS, C. N. E.; ANAGNOSTOPOULOS, I. E.; PSOROULAS, I. D; LOUMOS, V.; KAYAFAS, E. License Plate Recognition from Still Images and Video Sequences: A Survey. **IEEE Transactions on Intelligent Transportation Systems**, v. 9, n. 3, set. 2008. https://doi.org/10.1109/TITS.2008.922938

BUNCH, J. R.; HOPCROFT, J. E. Triangular factorization and inversion by fast matrix multiplication. **Mathematic of Computation,** v. 28, n. 125, p 231-236, 1974. https://doi.org/10.1090/S0025-5718-1974-0331751-8

CANDEIAS, A. L. B. **Aplicação da Morfologia Matemática à análise de imagens de sensoriamento remoto**. 1997. 187 f. Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Aplicadas, São José dos Campos, SP, 1997.

CANNY, J. F. A Computational Approach to Edge Detection, **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. PAMI-8, n. 6, 1986, p. 679-698. https://doi.org/10.1109/TPAMI.1986.4767851

CHEN, L.; WANG, S.; FAN, W.; SUN, J.; NAOI, S. Beyond human recognition: A CNN-based framework for handwritten character recognition. In: 3ª IAPR ASIAN CONFERENCE ON PATTERN RECOGNITION, p. 695-699, 2015. https://doi.org/10.1109/ACPR.2015.7486592

DAS, S.; BANERJEE, S. An Algorithm for Japanese Character Recognition. I.J. **Image, Graphics and Signal Processing,** v. 1, p. 9-15. 2015. https://doi.org/10.5815/ijigsp.2015.01.02

ETL. Electrotechnical laboratory character database. Disponível em: http://etlcdb.db.aist.go.jp/. Acessado em: 12 jun. 2019.

GAO, X. W; PODLADCHIKOVA, L.; SHAPOSHNIKOV, D.; HONG, K.; SHEVTSOVA, N. Recognition of traffic signs based on their colour and shape features extracted using human vision models. **Journal of Visual Communication and Image Representation**. v. 17, n. 4, p. 675-685, 2006. https://doi.org/10.1016/j.jvcir.2005.10.003

GREENHALGH, J.; MIRMEHDI, M. Real-Time Detection and Recognition of Road Traffic Signs. **IEEE Transactions on Intelligent Transportation Systems**, v. 13, n. 4, 2012. https://doi.org/10.1109/TITS.2012.2208909

GREENHALGH, J.; MIRMEHDI, M. Recognizing Text-Based Traffic Signs. **IEEE Transactions on Intelligent Transportation Systems**, v. 16, n. 3, 2014. https://doi.org/10.1109/TITS.2014.2363167

HARRIS, P.; STEPHENS, M. **A combined corner and edge detector.** Plessey Research Roke Manor, Reino Unido. 1988. https://doi.org/10.5244/C.2.23

JSAA. Japan Safety Appliances Association, JIS Z 9101 revisado – Esboço e comentário do JIS Z 9130. Disponível em: http://jsaa.or.jp/wp/wp-content/uploads/2018/07/5adbd5ec14cec6c33b3c5534e9aa52c8.pdf. Acessado em: 10 março 2019.

JNTO. Japan National Tourism Organization. Número de clientes estrangeiros que visitam o Japão em 2017. Disponível em: www.jnto.go.jp/jpn/statistics/since2003_tourists.pdf. Acessado em: 22 fev. 2018.

KAW, A.K.; KALU, E.E.; NGUYEN, D. Numerical methods with applications: chapter 04.06

Gaussian Elimination. University of South Florida. 2018. Disponível em: http://mathforcollege.com/nm/mws/gen/04sle/mws_gen_sle_txt_gaussian.pdf. Acessado em: 13 jan 2019.

KINGMA, D. P; BA, J. L; Adam: A Method for Stochastic Optimization. In: INTERNATIONAL CONFERENCE ON LEARNING REPRESENTATIONS, 2014.

MANTAS, J. An Overview of Character Recognition Methodologies. **Pattern Recognition**, v. 19, n. 6, p. 425-430. 1986. https://doi.org/10.1016/0031-3203(86)90040-3

OLIVEIRA, L. A. **Localização e reconhecimento de caracteres em placa de automóveis.** Trabalho de conclusão de curso – Escola de engenharia de São Carlos, da Universidade de São Carlos. São Carlos, 2010.

OTSU, N. A Threshold Selection Method from Gray-Level Histograms. **IEEE Transactions on Systems, Man and Cybernetics**, v. 9, n. 1, 1979. https://doi.org/10.1109/TSMC.1979.4310076

REYES, S. V.; CISNEROS, J. L. V.; VILLAGRANA, J. E. A.; AMBRIZ, F. J. B.; SIGG, M. L.; RAMÍREZ, C. H. C. Mapeos Proyectivos: la base para el Funcionamiento del Pizarrón Interactivo Wiimote Whiteboard. **Revista Ibérica de Sistemas e Tecnologias de Informação.** 2015.

RODRIGUES, F. A. **Localização e Reconhecimento de Placas de Sinalização Utilizando um Mecanismo de Atenção Visual e Redes Neurais Artificiais.** 2012. 124 f. Dissertação de mestrado - Universidade Federal de Campina Grande. Campina Grande, 2012.

ROSA, M. S. **Reconhecimento de placas de sinalização de trânsito via processamento digital de imagem e aprendizado de máquina.** Trabalho de conclusão de curso – Instituto Federal de Santa Catarina. São José, 2017.

SHAIK, B. K.; GENESAN, P.; KALIST, V; SATHISH, B. S.; JENITHA, J. M. M. Comparative study of skin color detection and segmentation in HSV and YCbCr color space. **Procedia Computer Science,** v. 57, p. 41-48, 2015. https://doi.org/10.1016/j.procs.2015.07.362

SOLOMON, C.; BRECKON, T. **Fundamentals of Digital Image Processing: A practical approach with examples in Matlab.** John Wiley & Sons, 2011. https://doi.org/10.1002/9780470689776

SUZUKI, S.; ABE, K. Topological Structural Analysis of Digitized Binary Images by Border Following. COMPUTER VISION, GRAPHICS, AND IMAGE, **Processing**, v. 30, No. 1, 1985. https://doi.org/10.1016/0734-189X(85)90016-7

TSAI, C. Recognizing Handwritten Japanese Characters Using Deep ConvolutionalNeural Networks. University of Stanford, California, 2016. Disponível em: https://www.semanticscholar.org/paper/Recognizing-Handwritten-Japanese-Characters-Using-Tsai/f3ee6bfaec669a2c8d087e2f11fa48aa7b45d6ea. Acessado em: 15 maio 2018.