INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO PIAUÍ
CAMPUS TERESINA CENTRAL
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

ERMESON DOS SANTOS SOUSA

SISTEMA DE AUXÍLIO À APRENDIZAGEM DE CARACTERES UTILIZADOS NA LÍNGUA JAPONESA (KANJI)

TERESINA, PIAUÍ

2019

ERMESON DOS SANTOS SOUSA

SISTEMA DE AUXÍLIO À APRENDIZAGEM DE CARACTERES UTILIZADOS NA LÍNGUA JAPONESA (KANJI)

Projeto apresentado à Banca Examinadora como requisito para aprovação na disciplina de Trabalho de Conclusão de Curso II do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Piauí.

Orientador: Prof. D^{r}. Thiago Alves Elias da Silva

TERESINA, PIAUÍ

2019

Dados Internacionais de Catalogação na Publicação (CIP) de acordo com ISBD

Sousa, Ermeson dos Santos

S725s Sistema de auxílio à aprendizagem de caracteres utilizados na língua japonesa (Kanji) / Ermeson dos Santos Sousa. - 2019.
32 f.: il. color.

Trabalho de Conclusão de Curso (Tecnologia em Análise e Desenvolvimento de Sistemas) - Instituto Federal do Piauí, Campus Teresina Central, 2019.
Orientador : Prof. Dr. Thiago Alves Elias da Silva.

1. Kanji. 2. Aprendizagem de Kanji. 3. Computação visual. 4. Redes neurais. 5. Reconhecimento Óptico de Caracteres (OCR). I.Título.

CDD - 004.21

Elaborado por Sindya Santos Melo CRB 3/1085

ERMESON DOS SANTOS SOUSA

SISTEMA DE AUXÍLIO À APRENDIZAGEM DE CARACTERES UTILIZADOS NA LÍNGUA JAPONESA (KANJI)

Projeto apresentado à Banca Examinadora como requisito para aprovação na disciplina de Trabalho de Conclusão de Curso II do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Piauí.

Aprovado pela banca examinadora em _____.

BANCA EXAMINADORA:

Prof. Dr. Thiago Alves Elias da Silva
Instituto Federal de Educação, Ciência e Tecnologia do Piauí - IFPI

MEMBRO 1
Instituto Federal de Educação, Ciência e Tecnologia do Piauí - IFPI

MEMBRO 2
Instituto Federal de Educação, Ciência e Tecnologia do Piauí - IFPI

MEMBRO 3
Instituto Federal de Educação, Ciência e Tecnologia do Piauí - IFPI

TERESINA, PIAUÍ

2019

AGRADECIMENTOS

Agradeço primeiramente a Deus, por ter me dado toda a força necessária pra concluir esse projeto.

Agradeço em seguida a meus pais, que assim como muitos, aguardaram pacientes pela conclusão do curso e me deram o apoio necessário para terminar a jornada.

Agradeço à minha irmã, por todo apoio dado à mim nos momentos difíceis dessa jornada.

Agradeço a todos os meus amigos, os quais não vou citar nomes pra não esquecer ninguém, por toda paciência que tiveram comigo nos meus dias de estresse.

Agradeço profundamente aos mestres e professores que partilharam de seus conhecimentos por todos esses anos.

Agradeço imensamente também a todos os colegas e amigos que aqui fiz, os quais também partilharam de seus conhecimentos.

Agradeço ao Instituto Federal do Piauí, pela infraestrutura necessária para a realização deste trabalho.

Agradeço especialmente aos meus orientadores, Thiago Alves Elias da Silva e Fernando Castelo Branco Gonçalves Santana, por sua amizade, conselhos e ensinamentos que levarei por toda a minha vida.

”No final, tudo vai dar certo.”
(Kinomoto Sakura, Cardcaptors Sakura)

RESUMO

O aprendizado de uma nova língua sempre é uma tarefa cheia de percalços, por fatores como falta de contato, gramática e pronúncia. Essa dificuldade é ainda maior com línguas que não utilizam o sistema de escrita romano, como línguas orientais, tendo como exemplo o Japonês e o Chinês. Uma das principais dificuldades relacionada ao aprendizado dessa língua é o aprendizado do *kanji*, sistema de escrita utilizado tanto no Japonês como no Chinês. Esse sistema de escrita consiste de diferentes ideogramas, onde cada um pode representar uma ideia ou conceito diferente. Tendo em vista essa dificuldade, esse trabalho apresenta um aplicativo para dispositivos *Android*, utilizando redes neurais, para auxiliar nesse aprendizado.

Palavras-chaves: *Kanji*, Aprendizado de *Kanji*, Computação visual, Rede Neurais, OCR

ABSTRACT

Learning a new language is always a task full of obstacles, such as lack of contact, grammar and pronunciation. This difficulty is even greater with languages that do not use the Roman writing system, such as Eastern languages, taking Japanese and Chinese as an example. One of the main difficulties related to learning this language is the learning of kanji, a writing system used in both Japanese and Chinese. This system consist of different ideograms, with differents meanings. Given this difficulty, this work aims to develop an application for Android devices, using neural networks, to aid in this learning.

Key-words: Kanji, Kanji learning, Visual computing, OCR, Neural Networks.

LISTA DE ILUSTRAÇÕES

SUMÁRIO

1 INTRODUÇÃO

A língua é um sistema simbólico formado por palavras e por leis combinatórias que permitem o exercício da linguagem verbal [CEREJA; MAGALHAES, 2004], ou seja, a fala humana.

O aprendizado de uma nova língua pode trazer inúmeros benefícios para a vida de uma pessoa. Entre os principais, estão a melhoria nas capacidades cognitivas [BIALYSTOK; MARTIN, 2004], na habilidade de tomar decisões [KEYSAR; HAYAKAWA; AN, 2012] e, em alguns casos, o adiamento, em até 5 anos, de um possível aparecimento da doença de Alzheimer [CRAIK; BIALYSTOK; FREEDMAN, 2010].

Alguns fatores podem influenciar esse aprendizado. Entre esses fatores, se encontram a idade [LIGHTBOWN et al., 1999], a estratégia de aprendizado utilizada [DÖRNYEI; SKEHAN, 2003] e fatores pessoais, como ansiedade, personalidade da pessoa e aptidão da mesma para esse aprendizado [ELLIS, 2004].

Além disso, podem ser encontradas certas dificuldades durante esse processo, como problemas com a fonologia, sintaxe e semântica [CHEN; CHANG, 2004], falta de exposição à língua [GOH, 2000] e a falta de motivação do estudante durante o processo de aprendizado [MASGORET; GARDNER, 2003].

A dificuldade encontrada durante esse processo pode ser ainda maior no aprendizado de línguas cujo sistema de escrita é não-alfabético [ROSE; HARBON, 2013]. Segundo classificação adotada pelo Instituto Linguístico da Defesa, algumas línguas com esse tipo de sistema são classificadas com "nível 4", atribuído a partir do tempo que se necessitaria para adquirir níveis de proeficiência [PAXTON; SVETENANT, 2014].

Um exemplo desses sistemas de escrita seria o *kanji*, utilizado principalmente na língua japonesa, juntamente com *hiragana* e *katakana* [DUNCAN et al., 2013], e chinesa, de forma isolada. O *kanji* teve sua origem milhares de anos atrás, a partir de pictografias utilizadas pelos chineses para representar o mundo à sua volta [TAMAOKA et al., 2002]. Foram adotados na língua japonesa como uma forma de representar o idioma falado [TAMAOKA, 2003].

As principais características do *kanji* são a ordem dos traços, suas leituras e os radicais que o formam [MORI, 2014]. Normalmente, possuem dois tipos de leituras: a On, baseada na pronúncia original chinesa, e a Kun, baseada na pronúncia japonesa [TANAKA, 2015] [VERDONSCHOT et al., 2013] [KUWANA, 2016].

Entre as principais dificuldades que o aprendizado de *kanji* pode trazer, estão a complexidade, as múltiplas leituras que um *kanji* pode ter, a grande quantidade de *kanji* existentes [GAMAGE, 2003] e grande possibilidade da ocorrência de homofonia, ou seja, *kanji*, e consequentemente palavras, com sons semelhantes, mas grafias completamente diferentes [TAMAOKA et al., 2002].

Com base no que foi apresentado, esse trabalho apresenta um sistema que auxilia o estudante de língua japonesa no aprendizado do *kanji*, através de redes neurais.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

O objetivo geral deste trabalho é o desenvolvimento de um sistema destinado ao auxílio do aprendizado de caracteres utilizados na língua japonesa (*kanji*), voltado para estudantes da língua japonesa, utlizando redes neurais profundas.

1.1.2 Objetivos Específicos

- Construir um modelo de identificação de *kanji*, utilizando aprendizagem de máquina;
- Desenvolver um aplicativo Android que utilize o modelo construído na identificação dos *kanji*;

2 FUNDAMENTAÇÃO TEÓRICA

Nessa seção serão apresentados, de forma detalhada, os conceitos necessários para entendimento do presente trabalho.

2.1 *KANJI*

Kanji são ideogramas originários da China, que representam a língua através de morfemas (baseadas no significado), ao invés de fonemas (baseadas no som) [ROSE, 2013]. Exemplos de *kanjis* podem ser vistos na figura 1. Foram levados ao Japão por volta do século 1 D.C. [LIBRENJAK; VUČKOVIĆ; DOVEDAN, 2012], como uma forma de representar o idioma falado [TAMAOKA, 2003].

Ilustração 1 – Exemplo de kanjis

São utilizados conjuntamente com dois alfabetos fonéticos: o *hiragana* (utilizado principalmente como partículas, partes flexionáveis de verbos e adjetivos) e o *katakana* (utilizados principalmente com *loan words*, termos científicos, algumas onomatopeias, etc) [HINO; MIYAMURA; LUPKER, 2011] [CRUZ; TELES, 2017].

Mais de 10000 *kanji* podem ser encontrados na literatura japonesa; ao mesmo tempo, o conhecimento dos 2000 *kanji* mais utilizados permitem aos falantes da língua a fluência na leitura da língua, já que constituem mais de 99% dos textos escritos no idioma. [ROSE, 2013]

Os kanji são caracterizados pelos radicais que o formam [HIGUCHI et al., 2015], sua ordem de traços [MORI, 2014], suas leituras [TANAKA, 2015] [VERDONSCHOT et al., 2013] e seu significado [OGIHARA et al., 2015]. Normalmente, possuem dois tipos de leituras: a On, baseada na pronúncia original chinesa, e a Kun, baseada na pronúncia japonesa [TANAKA, 2015] [VERDONSCHOT et al., 2013] [PERELLÓ, 2016].

Existem 214 radicais na língua japonesa [WINIWARTER, 2017], divididos em 7 categorias: *hen* (encontrados do lado esquerdo do *kanji*), *tsukuri* (encontrados do lado direito), *kanmuri* (encontrados na parte de cima), *ashi* (encontrados na parte de baixo), *kanmae* (radicais

que envolvem todos os outros elementos de um *kanji*), tare (radicais cujo formato extende-se pela parte de cima e pelo lado direito do *kanji*) e *nyou* (radicais cujo formato extende-se pela parte de baixo e pelo lado direito do *kanji*) [NISHIO, 2018] [KUWANA, 2016].

2.1.1 *Kyoiku Kanji*

Entre os *kanji* mais importantes, podemos destacar o *kyouiku kanji*, lista de 1006 *kanji* ensinados nos 6 anos iniciais da educação básica japonesa [KOINUMA; TEZUKA, 2016] e constituem cerca de 95% dos *kanji* utilizados [PAXTON; SVETENANT, 2014]. Ela é uma sub-seção do *Joyou Kanji*, lista de 2136 *kanji* mais utilizados, ambas definidas pelo Ministério de Educação, Cultura, Esportes, Ciência e Tecnologia do Japão [BOSSARD, 2015].

Eles são classificados entre *níveis 1-6*, dependendo do ano em que ele é ensinado [WINIWARTER, 2017]. Possuem a seguinte divisão: 80 no primeiro ano, 160 no segundo, 200 nos terceiro e quarto anos, 185 no quinto ano e 181 no sexto e último ano [TAEKO et al., 2013] [PAXTON; SVETENANT, 2014].

2.2 COMPUTAÇÃO VISUAL

Computação visual é uma área que envolve a aquisição, representação, processamento, análise, síntese e uso de informações visuais [GRÖLLER, 2013]. Além disso, ela estuda o desenvolvimento de sistemas artificiais capazes de desenvolver uma percepção de meio ambiente através de informações visuais [BASSO, 2018].

A computação visual envolve campos como os da computação gráfica, visão computacional, visualização, interação humano-computador [BADAM; FISHER; ELMQVIST, 2015], análise de dados, e-commerce e na área da aprendizagem [GUHA, 2013].

2.2.1 Visão Computacional

Visão computacional é um área da computação visual que tem como objetivo a análise, modificação e entendimento em auto-nível de imagens [PULLI et al., 2012], com o intuito de produzir dados numéricos ou simbólicos para realizar alguma função ou tomar decisões.

É uma área de pesquisa na automação industrial, diagnóstico médico e na navegação [FAHLGREN; GEHAN; BAXTER, 2015]. Essa área combina mecânica, instrumentação ótica, sensores magnéticos e tecnologia de processamento de imagens e vídeos digitais [ZHANG et al., 2014].

2.2.1.1 Binarização

Binarização é um processo de separação de valores de *pixels* de uma determinada imagem em dois valores distintos, como branco pra o pano de fundo e preto para o primeiro plano [FIRDOUSI; PARVEEN, 2014], sendo um passe chave do pré-processamento de qualquer

sistema de análise de imagens [MISHRA; ALAHARI; JAWAHAR, 2011]. Os métodos de binarização podem ser divididos em duas grande categorias: binarizações baseadas em *clustering* e binarizações baseadas em *threeshold* [MOGHADDAM; CHERIET, 2010].

Ilustração 2 – Exemplo de binarização

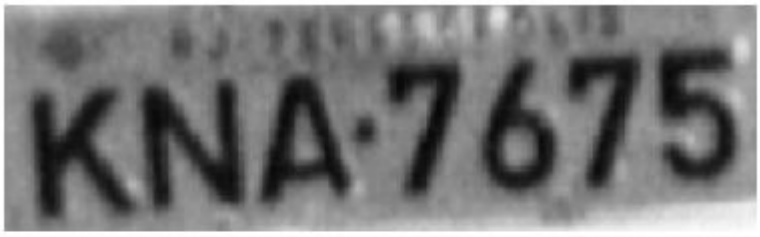


Figura 1.4 – Placa antes da binarização

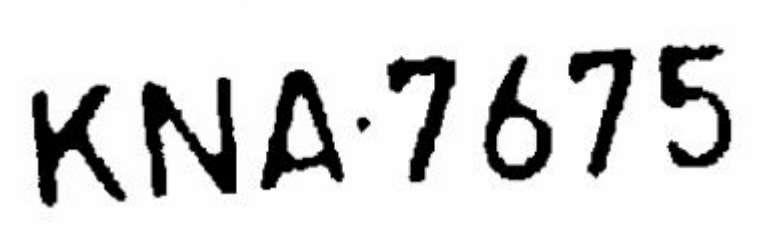


Figura 1.5 – Placa depois da binarização

As binarizações baseadas em *clustering* geralmente utilizam *features* baseados em modelos para diferenciar textos do *background* e classificá-los [MOGHADDAM; CHERIET, 2010]. Exemplos desse tipo de classificação seriam a classificação *fuzzy* [PAPAMARKOS, 2001] e segmentação recursiva no espaço PCA [DRIRA, 2006].

Já as binarizações baseadas em threshold consistem na tentativa de localizar um limite para realizar a separação entre background e foreground [SEHAD et al., 2013]. Exemplos desse tipo de binarização seriam o método de OTSU, métodos baseados na entropia [SILVA et al., 2008] e métodos baseados nos atributos dos ruídos [MOGHADDAM; CHERIET, 2010].

2.2.1.2 OCR (Optical Character Recognition)

OCR, acrônimo para Optical Character Recognition [ANDRADE, 2016], é uma tecnologia de visão computacional, capaz de reconhecer caracteres a partir de uma imagem, sejam eles impressos, datilografados ou manuscritos [ROWINSKI; KEUTZER, 2016].

A ideia de reconhecimento de caracteres foi introduzida primeiramente pela invenção do scanner de retina, um sistema de transmissão de imagem que fazia o uso de mosaico de fotocélulas [HAMAD; KAYA, 2016]. Um grande avanço nessa área ocorreu em 1890, com a invenção do scanner sequencial por Nipkow [BERCHMANS; KUMAR, 2014].

Gustav Tauschek, de Vienna, na Áustria, patenteou um máquina básica de leitura OCR em 1929. Paul Handel, da General Electric, com um pedido de patente de um sistema similar, em

abril de 1931, nos Estados Unidos [GARG et al., 2017], onde Tauschek conseguiu sua patente em 1935 [BERCHMANS; KUMAR, 2014]. Ambas as máquinas utilizavam discos circulares com os *templates* cravados neles. A imagem a ser reconhecida era, então, segurada na frente do disco e iluminada. A luz era refletida de um parte dela, sendo focada através do buraco de *template* e detectada por um foto-sensor [BERCHMANS; KUMAR, 2014].

Mori *et al.* (1992) dividiu os sistema de OCR comerciais em três gerações: a primeira é caracterizada pelo reconhecimento de textos impressos [KUMAR; JINDAL; SHARMA, 2011]. O sistema típico dessa geração era o NCR 420, que utilizava uma fonte especial para os numerais e 5 símbolos, chamadas de NOF ou bicodes [MORI; SUEN; YAMAMOTO, 1992].

A segunda geração é caracterizada pelo reconhecimento de textos manuscritos [KUMAR; JINDAL; SHARMA, 2011]. Inicialmente, era restrita à números, somente. Tais máquinas surgiram entre metade da década de 1960 e começo da década de 1970. O sistema mais famoso dessa época foi o IBM 1287, exibido na World Fair em Nova York, no ano de 1965. Esse sistema era híbrido em relação ao seu hardware, ou seja, combinava tecnologia analógica e digital [MORI; SUEN; YAMAMOTO, 1992].

A terceira geração, começada na década de 1970 [MORI; SUEN; YAMAMOTO, 1992], caracteriza-se pelo reconhecimento de textos impressos que contenham degradação, e de textos que possuam textos, gráficos e símbolos matemáticos [KUMAR; JINDAL; SHARMA, 2011].

2.3 REDES NEURAIS

Redes neurais é um grande conjunto de sistemas, que engloba sistemas dinâmicos não-lineares, modelos de redução de dados e modelos de regressão não-linear flexível. [SARLE, 1994]. Elas são frequentemente utilizadas para classificar padrões, baseado em padrões aprendidos a partir de exemplos.

Redes neurais buscam representar o cérebro humano em um computador. Elas funcionam assim por dois motivos [BUDIWATI; HARYATNO; DHARMA, 2011]:

- O conhecimento é obtido através de um processo de aprendizagem;
- Neurônios, também conhecidos como integridades sinápticas, são utilizados para guardar o conhecimento;

Diferentes paradigmas empregam diferentes regras de aprendizagem, mas todos, de algum maneira, utilizam padrões estatísticos de um conjunto de exemplos de treinamento para classificarem novos padrões [SPECHT, 1990].

Redes neurais possuem três usos principais: análise de dados, modelagem de sistemas nervosos biológicos e de inteligência, e como processadores e controladores de sinais adaptativos implementados em hardware para aplicações em robôs [SARLE, 1994].

2.3.1 Redes Neurais Convolucionais

Rede neural convolucional é um conjunto de camadas que transforma uma camada de entrada, que contém dados pra alimentação, em uma camada de saída, que contém o resultado desejado [CHABANNE et al., 2017].

Elas podem ser aplicadas em vários tipos de sistemas, como, por exemplo, reconhecimento de imagens [SIMONYAN; ZISSERMAN, 2014] e vídeos [DING; TAO, 2017], OCR[LV, 2011], classificação de imagens [HOWARD, 2013], sistemas de recomendações [OORD; DIELEMAN; SCHRAUWEN, 2013], análises de imagens médicas [LI et al., 2014] e processamento de linguagem natural [COLLOBERT; WESTON, 2008].

2.3.1.1 Processo de propagação reversa

O processo de propagação reversa é um paradigma para aprendizagem e treinamento de redes neurais, onde o valor de cada camada é ajustado continuamente [JIN et al., 2000]. É uma dos paradigmas de redes neurais mais estudados e, entre suas qualidades, estão a auto-aprendizagem, a auto-adaptação, a robustez e a habilidade de generalização [DING; SU; YU, 2011].

Esse paradigma é composto de três camadas: a de entrada, a escondida e a de saída. O seu funcionamento ocorre em três passos: o de aprendizagem, ou treinamento, o de testes e o de validação [PANDA; CHAKRABORTY; PAL, 2008]. Ele é geralmente utilizado para o treinamento de redes neurais convolucionais [NG; TAY; GOI, 2013].

Esse tipo de rede usa os *outputs* de erro para mudar os valores de forma inversa. Para obter esse valor, há primeiro a propagação de forma inversa ou direta. Quando esse movimento acontece, os neurônios serão ativados pela função sigmóide (Ilustração 3) [BUDIWATI; HARYATNO; DHARMA, 2011].

Ilustração 3 – Função Sigmóide

$$f(x) = \frac{1}{1 + e^{-x}}$$

2.4 TENSORFLOW

O Tensorflow é um sistema de aprendizagem de máquina que opera em larga escala, em ambientes heterogêneos [ABADI, 2016]. É uma bilioteca open-source, desenvolvida pela Google para computação numérica, utilizada hoje em dia em grandes empresas [ERTAM; AYDIN, 2017]. O Tensorflow suporta uma grande variedade de aplicações, mas é especialmente utilizado para treinamento e inferência com redes neurais profundas [ABADI, 2016].

O Tensorflow foi criado pelo time de pesquisa da Google Brain para substituir o Theano [MUSHTAQ, 2018], uma biblioteca Python que permitia definir, otimizar, e avaliar expressões matemáticas envolvendo arrays multi-dimensionais eficientemente [TEAM et al., 2016].

3 TRABALHOS RELACIONADOS

Este capítulo apresenta trabalhos relacionados ao processamento e análise de imagens contendo kanji e similares.

3.1 KANJI SNAP - APLICATIVO PARA SMARTPHONE BASEADO EM OCR PARA O APRENDIZADO DE CARACTERES JAPONESES - KANJI

Aplicativo feito com o objetivo de auxiliar estudantes estrangeiros durante o aprendizado dos caracteres utilizados na língua japonesa (kanji). Para esse fim, foi utilizada a API de reconhecimento de kanji oferecida pela NTT Docomo.

O aplicativo oferece para o usuário, além de informações básicas como leituras e significados do kanji, palavras que o contenham e exemplos de frases com o mesmo. Além disso, oferece um sistema de gamificaçao, onde os estudantes usuários da aplicação podem ter controle do nível de seu próprio nível de aprendizado.

O aplicativo funciona da seguinte maneira: O usuário, primeiramente, tira uma foto do kanji do qual ele queira saber informações sobre. Após a captura da imagem, o usuário pode escolher entre duas maneiras de reconhecimento: reconhecimento de toda a imagem ou de uma área especificada pelo próprio usuário.

Feita a escolha, a imagem é, então, enviada, através de um Requisição HTTP, para a API da Docomo, onde ocorre seu pré-processamento, o reconhecimento da área onde está inserido o kanji e o OCR propriamente dito.

Após o processamento, o resultado é enviado de volta para o aplicativo por meio de uma HTTPResponse, no formato JSON, contendo o caracteres extraídos, junto com uma lista de palavras identificadas e sua localizações nas imagens.

Para a obtenção das informações úteis para o usuário, são utilizados dois dicionários, disponibilizados por Jim Breen. O primeiro é um dicionário de kanji, convertido para um banco de dados SQLite, e com informações detalhadas de kanjis. O segundo é um dicionário Japonês-Inglês online, que pode ser acessado por meio de um API, onde é possível fazer simples buscas através da própria URL, por meio de string queries.

O principal ponto negativo desse trabalho é a necessidade da existência de uma conexão de rede para o funcionamento total do mesmo, não podendo ser utilizado como aplicativo *stand-alone*. O KanjiAPP faz o reconhecimento dos kanji sem a necessidade de conexão com a Internet.

3.2 RECONHECENDO CARACTERES JAPONESES MANUSCRITOS UTILIZANDO REDES NEURAIS CONVOLUCIONAIS PROFUNDAS

Esse trabalho buscou fazer o reconhecimento de caracteres japoneses (*kanji*, *hiragana* e *katakana*). Para esse propósito, foram utilizadas redes neurais convolucionais profundas e o dataset fornecido pelo Instituto Nacional de Ciência e Tecnologia Industrial Avançada (*AIST*, da sigla em inglês), um dos datasets também utilizados nesse trabalho.

O modelo foi implementado no Tensorflow, na interface Keras. Os recursos computacionais usados foram fornecidos pelo grupo de pesquisa em computação da Universidade de Stanford.

Foram testadas nesse trabalho onze diferentes tipos de redes neurais convolucionais profundas. A arquitetura geral consiste de uma camada convolucional relativamente pequena, seguida de uma camada de ativação e uma de centralização máxima. Isto é, então, repetido, onde, durante cada repetição, a camada convolucional aumenta seu tamanho em cada profundidade.

Para esse trabalho, buscou-se realizar quatro diferentes tarefas de classificação:

- Diferenciação dos *scripts* entre *hiragana*, *katakana* e *kanji*;
- Classificação dos caracteres *katakana*;
- Classificação dos caracteres *hiragana*;
- Classificação dos caracteres *kanji*.

Os caracteres *katakana* utilizados foram retirados do *dataset ETL-1* e os caracteres *hiragana* e *kanji* foram retirados do *dasaset ETL-8*. Todos os caracteres utilizados foram binarizados em preto e branco utilizando o método de Otsu.

O treinamento foi realizado com a função de ativação de *softmax*, com a utilização do otimizador de Adam. Foram utilizados no treinamento, 80% de todos os exemplos disponíveis do *dataset* correspondente, com 20% destes exemplos sendo utilizados para validação cruzada. O treinamento foi terminado quando a perda ocorrida na validação e na taxa de acurácia começaram a estabilizar.

A acurácia de classificação foi utilizada como métrica principal para avaliar o desempenho dos modelos em todas as tarefas. Na primeira tarefa, que consistia na diferenciação entre *hiragana*, *katakana* e *kanji*, o melhor desempenho foi realizado pelo modelo M11 com 11 camadas de peso, com acurácia resultante de 99.30%.

Já na segunda tarefa, a de classificação de *hiragana*, o melhor modelo foi o modelo M7-2, de 7 camadas de peso, com a acurácia de 96.50%. A terceira tarefa, a de classificação de *katakana*, teve seu melhor resultado com o modelo de 12 camadas de peso (M12), com a acurácia de 98.19%.

A quarta e última tarefa, a de classificação de *kanji*, foi realizada mais eficientemente pelo

modelo M7-2, com acurácia de 99.64% no processo de classificação. Quando foi realizada uma classificação combinada, combinando todas as tarefas anteriores, a melhor taxa de classificação foi obtida pelo modelo M7-1, de 7 camadas de peso, que obteve acurácia de 99.53%.

O trabalho descrito foi utilizado como base para uma melhor interpretação dos conceitos, pela utilização de redes neurais e da mesma base de dados de *kanji* para seu desenvolvimento.

3.3 APLICAÇÃO DE RECONHECIMENTO DE PADRÕES DE CARACTERES JAPONESES (KANA) UTILIZANDO USANDO REDES NEURAIS

Aplicação que busca reconhecer e identificar caracteres japoneses (*hiragana* e *katakana*) através de padrões encontrados nos mesmos, utilizando redes neurais para esse fim. Para o treinamento e testificação da rede, foi utilizado o método de propagação reversa.

Inicialmente, os caracteres utilizados no treinamento e na testificação (as imagens utilizadas em cada um desses passos tiveram diferentes fontes) foram escaneados de maneira *off-line*, em diferentes fases. Os caracteres consistem de imagens, no formato bitmap, de tamanho 16 x 16 *pixels*, consistindo tanto de caracteres manuscritos quanto caracteres impressos.

Após a captura dos caracteres, foi realizado o pré-processamento das imagens. Esse processo consiste na limpeza do ruído e na redução da espessura do caractere. Para a realização desse processo, foi utilizado o algoritmo de Diluição Básica (*Basic Thinning*, em inglês).

O passo seguinte é o de extração das características (*features*), utilizando processamento de histogramas. Os padrões utilizados são acumulados dos *pixels* pretos de cada linha, coluna, e diagonais direita e esquerda de cada imagem, o que resulta em 16 linhas, 16 colunas e 36 diagonais direita e esquerda para cada caractere. Após a extração, cada matriz formada é salva no banco de dados para a posterior alimentação do banco.

Os processos de classificação e treinamento utilizaram redes neurais, junto ao processo de propagação reversa. Nessa etapa, o modelo foi alimentado utilizando os histogramas obtidos na fase anterior.

A testificação foi realizada em duas diferentes fases. A primeira, com imagens utilizadas durante o treinamento, obteve taxas de acerto entre 91 e 93%. Já, a segunda fase, que utilizou as imagens de testes não utilizadas durante o treinamento, e obtidas separadamente em relação à elas, obteve resultado que variaram entre 26 e 40%.

Esse trabalho é relacionado pela utilização de redes neurais para identificação de caracteres utilizados na língua japonesa. Porém, o trabalho descrito consiste na identificação de um tipo de caractere (*hiragana* e *katakana*), enquanto o presente trabalho identifica outro tipo, o *kanji*.

4 KANJIAPP

Esse capítulo explica o desenvolvimento de uma aplicação *Android*, que utiliza aprendizagem de máquina e redes neurais, implementados por meio do *Tensorflow*, da *Google*, para reconhecer e fornecer informações de caracteres japoneses (*kanji*), com a finalidade de auxiliar estudantes da língua.

4.1 MATERIAIS UTILIZADOS

Essa seção dedica-se a explicar os materiais utilizados para a confecção do modelo e o desenvolvimento da aplicação.

4.1.1 Datasets

Foram escolhidos os seguintes datasets para a realização desse trabalho:

- Kanji300: dataset contendo quantidade variada de imagens de 300 *kanji*. É disponibilizado pelo NDL Lab, serviço experimental da Biblioteca Nacional da Dieta (*Kokuritsu Kokkai Toshokan*). Pode ser encontrado no link: https://lab.ndl.go.jp/cms/kanji300;
- ETL Character Database: dataset produzido pelo Laboratório de Eletrotécnica (atual Instituto Nacional de Ciência e Tecnologia Industrial Avançada (AIST)), com cooperação da Associação de Desenvolvimento da Indústria Eletrônica do Japão (atual Associação de Indústria de Tecnologia da Informação e Eletrônica do Japão), universidades e outras organizações de pesquisas. Contém cerca de 1.200.000 caracteres escritos a mão e impressos, vindos de cerca de 800 diferentes kanji, obtidos entre os anos de 1973 e 1984. Pode ser encontrado no seguinte link: http://etlcdb.db.aist.go.jp/

4.2 DESENVOLVIMENTO

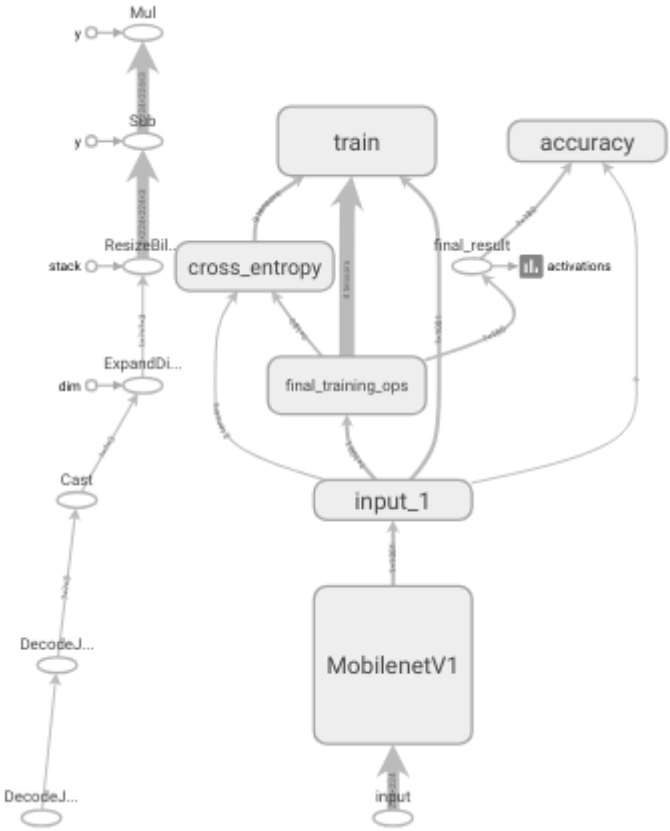
Essa seção explicará como foi feito o desenvolvimento do aplicativo, desde a fase do treinamento do modelo, até a fase de confecção da aplicação propriamente dita.

4.2.1 Treinamento

O treinamento foi realizado a partir de uma versão modificada de um código de retreino, disponibilizado pela própria Google. O modelo utilizado foi treinado na arquitetura MobileNet V1, demonstrada na ilustração 4. Essa arquitetura foi escolhida por ser a mais indicada para uso em dispositivos móveis [HOWARD et al., 2017].

A arquitetura MobileNet utiliza convoluções separáveis profundas, uma forma de convoluções fatorizadas que fatorizam uma convolução comum em uma convolução profunda e uma convolução 1 x 1, chamada de convolução pontual [HOWARD et al., 2017].

Ilustração 4 – MobileNet V1



Nessa arquitetura, a convolução profunda aplica um filtro pra cada um canal inserido. Após esse passo, a convolução pontual aplica uma convolução 1 x 1 para combinar as saídas da primeira convolução. Esta separação possui o efeito de diminuir drasticamente o tamanho do modelo e o custo computacional [HOWARD et al., 2017].

Nesse treinamento, foi utilizada a técnica de aprendizagem por transferência (do inglês *transfer learning*). Essa técnica consiste na utilização de um conhecimento adquirido em uma tarefa similar para o aperfeiçoamento da aprendizagem na tarefa desejada [TORREY; SHAVLIK, 2010].

A técnica descrita no trecho anterior foi usada para treinar o código já existente com imagens disponíveis dos kanji, e utilizar o mesmo código para o reconhecimento dentro do aparelho.

Para a realização do treinamento, os dois *datasets* foram divididos em suas respectivas categorias, onde cada categoria representava um *kanji* diferente, e foram binarizadas, para melhorar o seu contraste.

Após esse passo, o *script* foi executado, gerando os *bottlenecks*. Os *bottlenecks* são, de acordo com definição fornecida pelo Google, a camada anterior à camada responsável pelo treinamento propriamente dito. Eles contém um conjunto de valores utilizados pelo classificador

para distinguir entre as classes treinadas, ou seja, valores simples o suficiente para guardar informações sobre as imagens utilizadas no treinamento.

Após a geração dos *bottlenecks*, o treinamento em si é iniciado. Durante cada passo do treinamento, são escolhidas, aleatoriamente, 10 imagens. O *bottleneck* de cada uma delas é, então, recuperado, e alimentado à camada final do modelo, onde são feitas predições. Com as predições feitas, elas são comparadas com o seus respectivos labels, e, a partir disso, os pesos são atualizados, treinados por meio do processo de propagação reversa, através de camadas de filtros convolucionais.

No final, o algoritmo forneceu dois arquivos: um arquivo com a extensão .pb, correspondente ao modelo treinado, e um arquivo com a extensão .txt, correspondente a lista de *labels* das imagens utilizadas para o treinamento do modelo.

4.3 KANJIAPP

O aplicativo foi desenvolvido para a plataforma Android, por ser, atualmente, o sistema operacional mobile mais utilizado do mundo, segundo relatório da StatsCounter [STATSCOUNTER, 2019 (acessado em 29 de maio de 2019)].

O aplicativo em si possui três telas principais: a tela de início, que apresenta informações básicas, e o acesso para outras partes do aplicativo; a câmera, onde é feita a captura da imagem para o reconhecimento; e a parte das informações sobre o *kanji*, que pode ser acessada a partir da câmera.

4.3.1 Tela de início

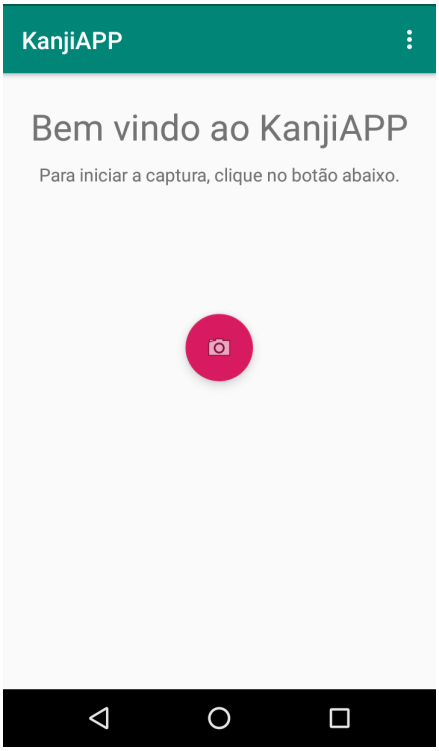
Como demonstrado na ilustração 3, a tela inicial possui um botão principal, utilizado para levar o usuário até a câmera, que fará a captura da foto para identificação do caractere desejado.

4.3.2 Câmera

A câmera, cujo acesso ocorre a partir da tela de início, é utilizada para a captura da imagem a ser fornecida ao modelo, tendo como objetivo do reconhecimento do *kanji* capturado. Nessa tela, pode-se observar o *kanji* reconhecido, escrito em letras brancas, na parte superior da câmera, além de um botão na parte de baixo, que nos permite ver informações sobre o mesmo.

A construção da câmera é feita em duas partes: a principal e o fragmento (onde está contida a câmera). A imagem é, então, analisada pelo classificador (modelo treinado anteriormente), com o resultado, no fim, sendo mostrado na parte superior da tela, como demonstrado na ilustração 4.

Ilustração 5 – Tela inicial



Para poder obter, então, informações sobre o *kanji* identificado nessa tela, o usuário pode clicar no botão localizado na parte de baixo da mesma, onde está escrito "Informações sobre o *Kanji*", o que o levará para a tela explicada no tópico seguinte.

4.3.3 Informações sobre o *Kanji*

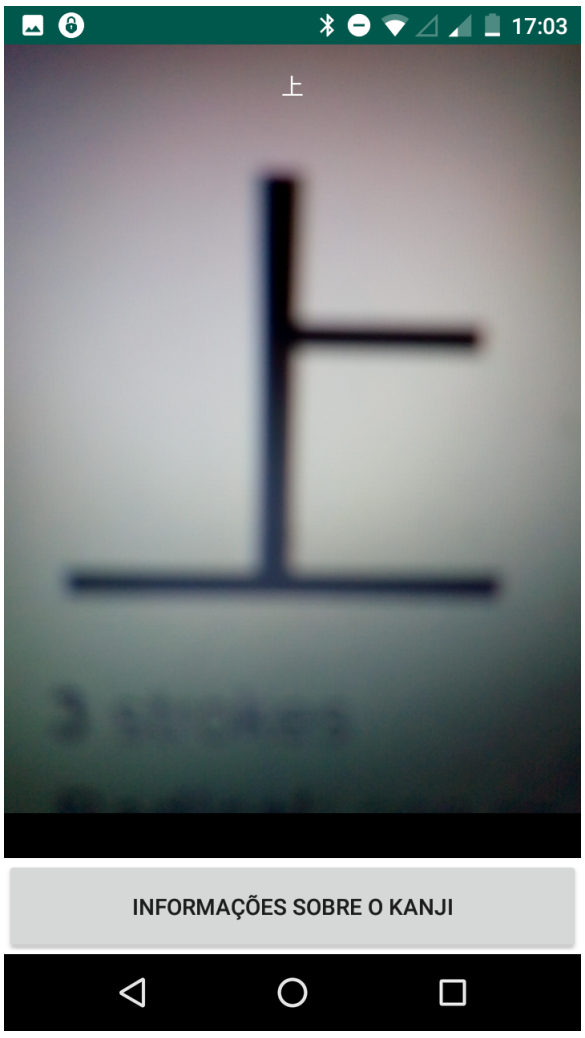
Nessa tela, demonstrada na ilustração 5, é possível ver algumas informações sobre o *kanji* identificado na tela anterior. As informações foram fornecidas ao aplicativo através de arquivos de texto (formato .txt) e retiradas do site *jisho.org*.

FIGURE[HTB] EXEMPLO DE INFORMAÇÕES DE UM KANJI SENDO FORNECIDAS [SCALE=0.60]IMAGENS/SCREENSHOT$_2$0190617 − 010503.$PNGFIG$: $BINAREXEMP1FIGURE$

As informações mostradas nessa tela consistem em:

- Imagem do *kanji* procurado, com a sua respectiva ordem de traços;
- Leituras (on e kun) do *kanji*;
- Significado(s) do *kanji*;
- Informações básicas sobre o *kanji*, tais como o ano da escola básica onde ele é ensinado, sua frequência em jornais, etc;
- Exemplos de palavras escritas utilizando esse *kanji*.

Ilustração 6 – Câmera



5 CONCLUSÃO

Esse trabalho apresentou o processo de desenvolvimento de um aplicação que utiliza redes neurais convolucionais implementadas no TensorFlow para identificar e fornecer informações sobre caracteres japoneses (*kanji*), com o objetivo de auxiliar no aprendizado de estudantes da língua.

Essas dificuldades incluem a complexidade, as múltiplas leituras que um *kanji* pode ter, a grande quantidade de *kanji* existentes [GAMAGE, 2003] e grande possibilidade da ocorrência de homofonia, ou seja, *kanji*, e consequentemente palavras, com sons semelhantes, mas grafias completamente diferentes [TAMAOKA et al., 2002].

Pelas dificuldades apresentadas, esse trabalho consistiu na apresentação do desenvolvimento de um aplicativo móvel, por conta de sua natureza portável, o que facilitaria o seu uso pelos alunos da língua.

A aplicação foi desenvolvida para *Android* (sistema operacional móvel mais utilizado) e utilizou redes neurais convolucionais, presentes no modelo *MobileNet*, do *Tensorflow*, e a técnica de aprendizagem por transferência, para fazer a identificação dos caracteres pretendidos.

5.1 TRABALHOS FUTUROS

Ente os possíveis aspectos que podem ser melhorados futuramente, com relação a esse trabalho estão:

- Segmentação: Atualmente, o aplicativo funciona somente com *kanjis* isolados, fora de textos. Para trabalhos futuros, espera-se a elaboração de um algoritmo de segmentação, para permitir o funcionamento dos mesmos com textos completos escritos em língua japonesa;
- Quantidade de *kanjis*: Atualmente, o aplicativo interpreta apenas os 1006 *kanjis* que fazem parte do *Kyouiku Kanji*. Para trabalhos futuros, pretende-se aumentar a quantidade de *Kanji* utilizados na elaboração do sistema;
- Melhorias gerais: Utilização de novas arquiteturas de rede para testes de performance, além de melhorias no design e na disponibilização de informações para o usuário.

REFERÊNCIAS

ABADI, M. Tensorflow: learning functions at scale. In: ACM. *Acm Sigplan Notices*. [S.l.], 2016. v. 51, n. 9, p. 1–1.

ANDRADE, L. B. d. República e federalismo no estado do rio grande do sul: história jurídica e social na primeira república. 2016.

BADAM, S. K.; FISHER, E.; ELMQVIST, N. Munin: A peer-to-peer middleware for ubiquitous analytics and visualization spaces. *IEEE Transactions on Visualization and Computer Graphics*, IEEE, v. 21, n. 2, p. 215–228, 2015.

BASSO, M. A framework for autonomous mission and guidance control of unmanned aerial vehicles based on computer vision techniques. 2018.

BERCHMANS, D.; KUMAR, S. Optical character recognition: an overview and an insight. In: IEEE. *Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2014 International Conference on*. [S.l.], 2014. p. 1361–1365.

BIALYSTOK, E.; MARTIN, M. M. Attention and inhibition in bilingual children: Evidence from the dimensional change card sort task. *Developmental science*, Wiley Online Library, v. 7, n. 3, p. 325–339, 2004.

BOSSARD, A. An implementation of japanese characters cartography as a learning tool. *Information Engineering Express*, v. 1, n. 1, p. 10–19, 2015.

BUDIWATI, S. D.; HARYATNO, J.; DHARMA, E. M. Japanese character (kana) pattern recognition application using neural network. In: IEEE. *Proceedings of the 2011 International Conference on Electrical Engineering and Informatics*. [S.l.], 2011. p. 1–6.

CEREJA, W. R.; MAGALHAES, T. C. *Português: linguagens/literatura, gramática e redação*. [S.l.]: Atual, 2004.

CHABANNE, H. et al. Privacy-preserving classification on deep neural network. *IACR Cryptology ePrint Archive*, v. 2017, p. 35, 2017.

CHEN, T.-Y.; CHANG, G. B. The relationship between foreign language anxiety and learning difficulties. *Foreign Language Annals*, Wiley Online Library, v. 37, n. 2, p. 279–289, 2004.

COLLOBERT, R.; WESTON, J. A unified architecture for natural language processing: Deep neural networks with multitask learning. In: ACM. *Proceedings of the 25th international conference on Machine learning*. [S.l.], 2008. p. 160–167.

CRAIK, F. I.; BIALYSTOK, E.; FREEDMAN, M. Delaying the onset of alzheimer disease: bilingualism as a form of cognitive reserve. *Neurology*, AAN Enterprises, v. 75, n. 19, p. 1726–1729, 2010.

CRUZ, B. Y.; TELES, S. Desenvolvimento de interface para jogo educativo voltado para estudantes de língua japonesa. 2017.

DING, C.; TAO, D. Trunk-branch ensemble convolutional neural networks for video-based face recognition. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 40, n. 4, p. 1002–1014, 2017.

DING, S.; SU, C.; YU, J. An optimizing bp neural network algorithm based on genetic algorithm. *Artificial intelligence review*, Springer, v. 36, n. 2, p. 153–162, 2011.

DÖRNYEI, Z.; SKEHAN, P. 18 individual differences in second language learning. *The handbook of second language acquisition*, p. 589, 2003.

DRIRA, F. Towards restoring historic documents degraded over time. In: IEEE. *Second International Conference on Document Image Analysis for Libraries (DIAL'06)*. [S.l.], 2006. p. 8–pp.

DUNCAN, K. J. K. et al. Inter-and intrahemispheric connectivity differences when reading japanese kanji and hiragana. *Cerebral Cortex*, Oxford University Press, v. 24, n. 6, p. 1601–1608, 2013.

ELLIS, R. 21 individual differences in second language learning. *The handbook of applied linguistics*, Citeseer, p. 525, 2004.

ERTAM, F.; AYDIN, G. Data classification with deep learning using tensorflow. In: IEEE. *2017 International Conference on Computer Science and Engineering (UBMK)*. [S.l.], 2017. p. 755–758.

FAHLGREN, N.; GEHAN, M. A.; BAXTER, I. Lights, camera, action: high-throughput plant phenotyping is ready for a close-up. *Current opinion in plant biology*, Elsevier, v. 24, p. 93–99, 2015.

FIRDOUSI, R.; PARVEEN, S. Local thresholding techniques in image binarization. *International Journal Of Engineering And Computer Science*, IJECS India, v. 3, n. 3, p. 4062–4065, 2014.

GAMAGE, G. H. Perceptions of kanji learning strategies. *Australian Review of Applied Linguistics*, John Benjamins Publishing Company, v. 26, n. 2, p. 17–30, 2003.

GARG, A. et al. A brief review on text traces using ocr. *Imperial Journal of Interdisciplinary Research*, v. 3, n. 4, 2017.

GOH, C. C. A cognitive perspective on language learners' listening comprehension problems. *System*, Elsevier, v. 28, n. 1, p. 55–75, 2000.

GRÖLLER, E. Trends in visual computing. 2013.

GUHA, A. Bits. 2013.

HAMAD, K. A.; KAYA, M. A detailed analysis of optical character recognition technology. *International Journal of Applied Mathematics, Electronics and Computers*, v. 4, n. Special Issue-1, p. 244–249, 2016.

HIGUCHI, H. et al. Neural basis of hierarchical visual form processing of japanese kanji characters. *Brain and behavior*, Wiley Online Library, v. 5, n. 12, p. e00413, 2015.

HINO, Y.; MIYAMURA, S.; LUPKER, S. J. The nature of orthographic–phonological and orthographic–semantic relationships for japanese kana and kanji words. *Behavior research methods*, Springer, v. 43, n. 4, p. 1110–1151, 2011.

HOWARD, A. G. Some improvements on deep convolutional neural network based image classification. *arXiv preprint arXiv:1312.5402*, 2013.

HOWARD, A. G. et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

JIN, W. et al. The improvements of bp neural network learning algorithm. In: IEEE. *WCC 2000-ICSP 2000. 2000 5th international conference on signal processing proceedings. 16th world computer congress 2000*. [S.l.], 2000. v. 3, p. 1647–1649.

KEYSAR, B.; HAYAKAWA, S. L.; AN, S. G. The foreign-language effect: Thinking in a foreign tongue reduces decision biases. *Psychological Science*, v. 23, n. 6, p. 661–668, 2012. PMID: 22517192. Disponível em: <https://doi.org/10.1177/0956797611432178>.

KOINUMA, A.; TEZUKA, T. *Text processing apparatus and text display system*. [S.l.]: Google Patents, 2016. US Patent 9,519,637.

KUMAR, M.; JINDAL, M.; SHARMA, R. Review on ocr for handwritten indian scripts character recognition. In: *Advances in Digital Image Processing and Information Technology*. [S.l.]: Springer, 2011. p. 268–276.

KUWANA, T. Reading strategies used by high school japanese language learners. 2016.

LI, Q. et al. Medical image classification with convolutional neural network. In: IEEE. *2014 13th International Conference on Control Automation Robotics & Vision (ICARCV)*. [S.l.], 2014. p. 844–848.

LIBRENJAK, S.; VUČKOVIĆ, K.; DOVEDAN, Z. Multimedia assisted learning of japanese kanji characters. In: IEEE. *MIPRO, 2012 Proceedings of the 35th International Convention*. [S.l.], 2012. p. 1284–1289.

LIGHTBOWN, P. M. et al. *How languages are learned*. [S.l.]: Oxford university press Oxford, 1999. v. 2.

LV, G. Recognition of multi-fontstyle characters based on convolutional neural network. In: IEEE. *2011 Fourth International Symposium on Computational Intelligence and Design*. [S.l.], 2011. v. 2, p. 223–225.

MASGORET, A.-M.; GARDNER, R. C. Attitudes, motivation, and second language learning: a meta–analysis of studies conducted by gardner and associates. *Language learning*, Wiley Online Library, v. 53, n. 1, p. 123–163, 2003.

MISHRA, A.; ALAHARI, K.; JAWAHAR, C. An mrf model for binarization of natural scene text. In: IEEE. *2011 International Conference on Document Analysis and Recognition*. [S.l.], 2011. p. 11–16.

MOGHADDAM, R. F.; CHERIET, M. A multi-scale framework for adaptive binarization of degraded document images. *Pattern Recognition*, Elsevier, v. 43, n. 6, p. 2186–2198, 2010.

MORI, S.; SUEN, C. Y.; YAMAMOTO, K. Historical review of ocr research and development. *Proceedings of the IEEE*, IEEE, v. 80, n. 7, p. 1029–1058, 1992.

MORI, Y. Review of recent research on kanji processing, learning, and instruction. *Japanese Language and Literature*, JSTOR, p. 403–430, 2014.

MUSHTAQ, M. A. Multilingual sentiment analysis as product reputation insight. 2018.

NG, C.-B.; TAY, Y.-H.; GOI, B.-M. Comparing image representations for training a convolutional neural network to classify gender. In: IEEE. *2013 1st International Conference on Artificial Intelligence, Modelling and Simulation*. [S.l.], 2013. p. 29–33.

NISHIO, K. *Image processing apparatus and image processing method*. [S.l.]: Google Patents, 2018. US Patent App. 15/661,394.

OGIHARA, Y. et al. Are common names becoming less common? the rise in uniqueness and individualism in japan. *Frontiers in psychology*, Frontiers, v. 6, p. 1490, 2015.

OORD, A. Van den; DIELEMAN, S.; SCHRAUWEN, B. Deep content-based music recommendation. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2013. p. 2643–2651.

PANDA, S. S.; CHAKRABORTY, D.; PAL, S. K. Flank wear prediction in drilling using back propagation neural network and radial basis function network. *Applied soft computing*, Elsevier, v. 8, n. 2, p. 858–871, 2008.

PAPAMARKOS, N. A technique for fuzzy document binarization. In: ACM. *Proceedings of the 2001 ACM Symposium on Document engineering*. [S.l.], 2001. p. 152–156.

PAXTON, S.; SVETENANT, C. Tackling the kanji hurdle: investigation of kanji learning in non-kanji background learners. *International Journal of Research Studies in Language Learning*, v. 3, n. 3, p. 89–104, 2014.

PERELLÓ, M. C. Aproximación a los estudios de la lengua japonesa: métodos de aprendizaje de kanji japoneses de nivel intermedio. 2016.

PULLI, K. et al. Real-time computer vision with opencv. *Communications of the ACM*, ACM, v. 55, n. 6, p. 61–69, 2012.

ROSE, H. L2 learners' attitudes toward, and use of, mnemonic strategies when learning japanese kanji. *The Modern Language Journal*, Wiley Online Library, v. 97, n. 4, p. 981–992, 2013.

ROSE, H.; HARBON, L. Self-regulation in second language learning: An investigation of the kanji-learning task. *Foreign Language Annals*, Wiley Online Library, v. 46, n. 1, p. 96–107, 2013.

ROWINSKI, Z.; KEUTZER, K. Namsel: An optical character recognition system for tibetan text. *Himalayan Linguistics*, v. 15, n. 1, 2016.

SARLE, W. S. Neural networks and statistical models. Citeseer, 1994.

SEHAD, A. et al. Ancient degraded document image binarization based on texture features. In: IEEE. *2013 8th International Symposium on Image and Signal Processing and Analysis (ISPA)*. [S.l.], 2013. p. 189–193.

SILVA, J. M. M. da et al. A new and efficient algorithm to binarize document images removing back-to-front interference. *J. UCS*, v. 14, n. 2, p. 299–313, 2008.

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

SPECHT, D. F. Probabilistic neural networks. *Neural networks*, Elsevier, v. 3, n. 1, p. 109–118, 1990.

STATSCOUNTER. *Operating System Market Share Worldwide*. 2019 (acessado em 29 de maio de 2019). Disponível em: <http://gs.statcounter.com/os-market-share>.

TAEKO, O. et al. Kyouiku kanji wo taishou to shita buhin (bushuu) wo kyouyuu suru kanjigun no imiteki rujisei ni kan suru kentou. *Toukai Gakuin Daigaku Kiyou*, v. 6, p. 217–223, 2013.

TAMAOKA, K. Where do statistically-derived indicators and human strategies meet when identifying on-and kun-readings of japanese kanji? *Cognitive Studies*, Japanese Cognitive Science Society, v. 10, n. 4, p. 441–468, 2003.

TAMAOKA, K. et al. A web-accessible database of characteristics of the 1,945 basic japanese kanji. *Behavior Research Methods, Instruments, & Computers*, Springer, v. 34, n. 2, p. 260–275, 2002.

TANAKA, M. Japanese kanji word processing for chinese learners of japanese: a study of homophonic and semantic primed lexical decision tasks. *Theory and Practice in Language Studies*, v. 5, n. 5, p. 900–905, 2015.

TEAM, T. T. D. et al. Theano: A python framework for fast computation of mathematical expressions. *arXiv preprint arXiv:1605.02688*, 2016.

TORREY, L.; SHAVLIK, J. Transfer learning. In: *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*. [S.l.]: IGI Global, 2010. p. 242–264.

VERDONSCHOT, R. G. et al. The multiple pronunciations of japanese kanji: A masked priming investigation. *The Quarterly Journal of Experimental Psychology*, Taylor & Francis, v. 66, n. 10, p. 2023–2038, 2013.

WINIWARTER, W. Kangaroo: the kanji game room. In: ACM. *Proceedings of the 19th International Conference on Information Integration and Web-based Applications & Services*. [S.l.], 2017. p. 535–544.

ZHANG, B. et al. Principles, developments and applications of computer vision for external quality inspection of fruits and vegetables: A review. *Food Research International*, Elsevier, v. 62, p. 326–343, 2014.