



MODELO DE CLASSIFICAÇÃO DE IMAGENS DE DOCUMENTOS COMERCIAIS BRASILEIROS EMBARCADOS EM DISPOSITIVOS MÓVEIS UTILIZANDO A TRANSFERÊNCIA DE APRENDIZAGEM

Trabalho de Conclusão de Curso
Engenharia da Computação

Rodrigo Felippo Cordeiro Pereira
Orientador: Prof. Dr. Byron Leite Dantas Bezerra



UNIVERSIDADE
DE PERNAMBUCO

**Universidade de Pernambuco
Escola Politécnica de Pernambuco
Graduação em Engenharia de Computação**

**RODRIGO FELIPPO CORDEIRO
PEREIRA**

**MODELO DE CLASSIFICAÇÃO DE
IMAGENS DE DOCUMENTOS
COMERCIAIS BRASILEIROS
EMBARCADOS EM DISPOSITIVOS
MÓVEIS UTILIZANDO A
TRANSFERÊNCIA DE
APRENDIZAGEM**

Monografia apresentada como requisito parcial para obtenção do diploma de Bacharel em Engenharia de Computação pela Escola Politécnica de Pernambuco – Universidade de Pernambuco.

Recife, Dezembro 2019.

Rodrigo Felippo Cordeiro Pereira

Modelo de classificação de imagens de documentos comerciais brasileiros embarcado em dispositivos móveis utilizando a transferência de aprendizagem/ Rodrigo Felippo Cordeiro Pereira – Recife – PE, Brasil, dezembro 2019 -

14 p.

Orientador: Byron Leite Dantas Bezerra

Trabalho de Conclusão de Curso – Engenharia da Computação

Escola Politécnica de Pernambuco

Universidade de Pernambuco, agosto de 2020

1. Classificação de imagem. 2. Documentos comerciais brasileiros. 3. Dispositivos móveis. 4. Transferência de aprendizagem I. Prof. Dr. Byron Leite Dantas Bezerra. II. Universidade de Pernambuco. III. Escola politécnica, IV. Título

MONOGRAFIA DE FINAL DE CURSO

Avaliação Final (para o presidente da banca)*

No dia 19/12/2019, às 09h, reuniu-se para deliberar sobre a defesa da monografia de conclusão de curso do(a) discente **RODRIGO FELIPPO CORDEIRO PEREIRA**, orientado(a) pelo(a) professor(a) **BYRON LEITE DANTAS BEZERRA**, sob título Avaliação de modelos de classificação de imagens de documentos comerciais brasileiros utilizando a transferência de aprendizagem de modelos modernos de reconhecimento de imagem., a banca composta pelos professores:

BRUNO JOSÉ TORRES FERNANDES (PRESIDENTE)

BYRON LEITE DANTAS BEZERRA (ORIENTADOR)

Após a apresentação da monografia e discussão entre os membros da Banca, a mesma foi considerada:

Aprovada Aprovada com Restrições* Reprovada

e foi-lhe atribuída nota: 7 (SETE)

*(Obrigatório o preenchimento do campo abaixo com comentários para o autor)

O(A) discente terá 15 dias para entrega da versão final da monografia a contar da data deste documento.

BRUNO JOSE TORRES

FERNANDES:05699196498

Assinado de forma digital por BRUNO
JOSE TORRES FERNANDES:05699196498
Dados: 2020.11.24 13:09:39 -03'00'

AVALIADOR 1: Prof (a) **BRUNO JOSÉ TORRES FERNANDES**

BYRON LEITE DANTAS

BEZERRA:96974990987

Assinado de forma digital por BYRON
LEITE DANTAS BEZERRA:96974990987
Dados: 2020.11.24 12:18:08 -03'00'

AVALIADOR 2: Prof (a) **BYRON LEITE DANTAS BEZERRA**

AVALIADOR 3: Prof (a)

* Este documento deverá ser encadernado juntamente com a monografia em versão final.

Autorização de publicação de PFC

Eu, Rodrigo Felipe Loureiro Pereira, autor do projeto
de final de curso intitulado:

Modelo de classificação de imagens de documentos
comerciais brasileiros embarcados em dispositivos
móveis utilizando a transferência de aprendizagem

autorizo a publicação de seu conteúdo na internet nos portais da Escola
Politécnica de Pernambuco e Universidade de Pernambuco.

O conteúdo do projeto de final de curso é de responsabilidade do autor.

Rodrigo Estipoco
Assinatura do Discente

19/12/2019
Data

[Assinatura]
Assinatura do(a) Orientador(a)

19/12/2019
Data

[Assinatura]
Assinatura do(a) Professor de TCC

19/12/2019
Data

Modelo de classificação de imagens de documentos comerciais brasileiros embarcado em dispositivos móveis utilizando a transferência de aprendizagem

Classification model of Brazilian commercial documents images embedded in mobile devices using the transfer learning

Rodrigo Felippo Cordeiro Pereira¹  <https://orcid.org/0000-0002-8450-0939>

Byron Leite Dantas Bezerra¹  <https://orcid.org/0000-0002-8327-9734>

¹ Escola Politécnica de Pernambuco, Universidade de Pernambuco, Recife, Brasil.

E-mail do autor principal: Rodrigo Pereira rfcpc@ecomp.poli.br

Resumo

Atualmente, a extração de informações sobre o tipo de um documento é realizado por meio de uma identificação humana, isto é, o procedimento é feito de maneira manual. Com o advento da popularização de dispositivos móveis e a larga utilização em diversos setores empresariais e financeiros, tem causado um aumento no volume de dados que são aceitos na forma de imagens capturadas por esses dispositivos. Entre essas imagens encontram-se documentos de identificação e comprovantes residenciais. A análise manual dessas imagens torna o trabalho mais lento e sobrecarregado. Esse trabalho pretende avaliar se a classificação de imagem de documentos, normalmente feita por humanos, pode ser realizada por um sistema de classificação de documentos baseado em redes neurais. Outro objetivo deste trabalho é avaliar a viabilidade deste sistema de classificação ser embarcado e usado de modo eficaz e eficiente em dispositivos móveis. O sistema proposto neste trabalho foi construído a partir da transferência de aprendizagem do modelo de classificação de imagens MobileNet versão 2, que tem uma arquitetura simplificada com utilização de convoluções separáveis. Para treinamento e avaliação do sistema proposto, foi utilizada uma base de documentos comerciais brasileiros com 10988 imagens. Os resultados mostram que o modelo proposto é capaz de classificar as imagens dos documentos com uma acurácia de 98,34%. Finalmente, apresentamos o protótipo do sistema embarcado em um Smartphone Android, ao mesmo tempo em que verificamos a viabilidade de sua utilização neste dispositivo.

Palavras-Chave: Classificação de imagem; Documentos comerciais brasileiros; Dispositivos móveis; Transferência de aprendizagem;

Abstract

Currently, the extraction of information about the type of a document is carried out by means of human identification, that is, the procedure is done manually. With the advent of the popularization of mobile devices and widespread use in several business and financial sectors, it has caused an increase in the volume of data that is accepted in the form of images captured by these devices. Among these images are identification documents and residential vouchers. Manual analysis of these images makes work slower and overloaded. This work intends to evaluate if the classification of document images, normally made by humans, can be performed by a document classification system based on neural networks. Another objective of this work is to evaluate the feasibility of this classification system to be shipped and used effectively and efficiently on mobile devices. The system proposed in this work was built from the transfer of learning from the MobileNet image

classification model version 2, which has a simplified architecture with the use of separable convolutions. For training and evaluation of the proposed system, a base of Brazilian commercial documents with 10988 images was used. The results show that the proposed model is capable of classifying the images of the documents with an accuracy of 98.34%. Finally, we present the prototype of the system embedded in an Android Smartphone, while verifying the feasibility of using it on this device.

Key-words: *Image classification; Brazilian commercial documents; Mobile devices; Transfer learning;*

1. Introdução

Nos dias atuais algumas empresas estão utilizando o processamento de imagens para processar documentos comerciais brasileiros como pode ser observado em [1]. Algumas aplicações já são utilizadas como por exemplo, extração de informações de documentos utilizando *Optical Character Recognition* (OCR) e detecção de bordas do documento em uma imagem. No entanto, a análise para extrair as informações pertinentes ao tipo do documento é realizado por meio de identificação humana e passivo de erro. Um estudo feito por RUSSAKOVSKY, Olga et al. em 2015 [2] indica que o erro humano para classificar uma imagem é em torno de 4,7%. Portanto, seria desejável utilizar um software de classificação de imagens que consiga superar ou equiparar essa taxa de classificação humana.

Esse trabalho tem o objetivo de avaliar um modelo capaz de ser embarcado em dispositivos móveis para classificar tipos documentos comerciais brasileiros em uma imagem. Essa classificação de imagem através de um modelo computacional pode trazer melhorias importantes para empresas no que tange maior velocidade para o fluxo de processos dos documentos. Esse trabalho também pretende avaliar se a classificação de imagem normalmente feita por humanos, pode ser realizada por um sistema de classificação de documentos baseado em redes neurais.

Para realizar a classificação de imagens foi utilizada a técnica de transferência de aprendizagem. Essa técnica de transferência de aprendizagem permite, por exemplo, utilizar uma rede neural que já foi treinada com uma base de imagens, e retreinar uma ou mais camadas da rede, com a finalidade de classificar outros tipos de imagens, como pode ser observado em [3].

O treinamento do modelo abordado neste trabalho pode ser executado em menos de 30 minutos em um *laptop* sem precisar de um elevado processamento computacional como pode ser observado em [4]. Outro propósito deste trabalho é verificar se é possível usar um modelo de rede neural que possa oferecer um bom desempenho em um *smartphone*.

2. Referencial Teórico

A transferência de aprendizagem é capaz de reutilizar o processamento de um modelo de uma rede neural que já foi treinada em uma tarefa relacionada e reutilizar em esse processamento em um novo modelo como mostrado por HOWARD, Andrew G. et al. [3]. Em virtude desse potencial, essa técnica de transferência de aprendizagem foi incorporada no Tensorflow, uma plataforma de código aberto para aprendizado de máquina. O Tensorflow possui um ecossistema flexível e abrangente de ferramentas, bibliotecas e recursos da comunidade que permite que os pesquisadores desenvolvam técnicas de aprendizagem de máquina e os desenvolvedores construam e implantem facilmente aplicativos com essa mesma técnica [4]. Através do Tensorflow é possível utilizar a transferência de aprendizagem para classificar imagens, reconhecer textos, detectar objetos, reconhecer faces humanas, dentre outras possibilidades [4]. É válido ressaltar que a programação da rede de classificação pode ser feita através do módulos do Tensorflow utilizando a linguagem de programação Python[4].

A técnica de transferência de aprendizagem através do Tensorflow permite criar novos modelos que podem ser utilizados como ferramenta para empresas que dispõem de um baixo processamento de computacional, podendo também ser executada em dispositivos móveis. A transferência de aprendizagem através do Tensorflow já foi utilizada para identificação de fungos através de um modelo de classificação de imagem compatível com dispositivos móveis para reconhecer espécies de fungos escandinavos [5]. O Treinamento foi feito em um conjunto de dados fornecido pela *Danish Mycological Society*. Além dessas aplicações, essa técnica foi utilizada para classificar leucemia do tipo linfoblasto B a partir de várias imagens de leucemia [6].

Esses são apenas alguns exemplos de como essa técnica já é utilizada na área de classificação de imagens com diversos fins. Todos os exemplos citados a respeito de aplicações da transferência de aprendizagem exigem uma interação com alguma base de dados que vai ser usada para retrainar a rede, além do algoritmo que vai ser responsável para executar o retrainamento.

Existe um desafio anual que é conhecido como *ImageNet Large Scale Visual Recognition Challenge* (ILSVRC). O ILSVRC usa uma lista de 1000 categorias ou "classes" de imagens, incluindo 90 das 120 raças de cães classificadas pelo ImageNet completo. O ImageNet é um conjunto de dados de mais de 15 milhões de imagens de alta resolução rotuladas com cerca de 22.000 categorias. O ILSVRC usa um subconjunto do ImageNet de cerca de 1000 imagens em cada uma das 1000 categorias. Ao todo, existem cerca de 1,2 milhão de imagens de treinamento, 50.000 imagens de validação e 100.000 imagens de teste.

Em 2012, uma rede neural convolucional (CNN) profunda chamada AlexNet atingiu um erro de 16,4% no ILSVRC [7]. Nos próximos dois anos, as taxas de erro caíram para alguns pontos percentuais. O avanço de 2012 melhorou quantitativamente e marcou o início do avanço na inteligência artificial em todo o setor. Em 2015, pesquisadores da Microsoft relataram que suas CNNs excederam a capacidade humana nas tarefas restritas do ILSVRC utilizando uma rede neural convolucional profunda chamada ResNet que atingiu um erro de 3,57% [8]. Parte das imagens contidas na base de dados ImageNet podem ser observadas na Figura 1.

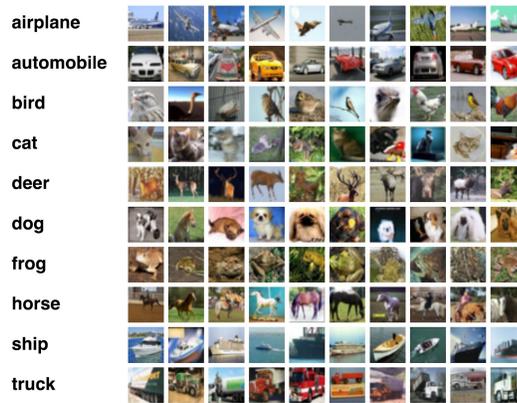


Figura 1: Parte das imagens contidas na base de dados ImageNet.

Foi proposto por HOWARD, Andrew G. et al. [3] uma nova arquitetura de modelo chamada MobileNets com base em convoluções separáveis em profundidade. Em seu trabalho foram investigadas algumas das importantes decisões de arquitetura que levam para um modelo eficiente para classificação de imagens da ImageNet com o baixo processamento, para classificar 1000 classes de imagens diferentes. Em seguida, foi demonstrado como construir MobileNets menores e mais rápidos para reduzir o tamanho e a latência do processamento. Logo após, foram comparados diferentes modelos de MobileNets com o objetivo de avaliar tamanho e precisão da rede. Dessa forma, a MobileNet pode atuar em uma ampla variedade de tarefas. A Mobilenet V2 é a segunda versão de uma arquitetura de modelo com base em convoluções em profundidade baseada na ImageNet [2].

Modelo de classificação de imagens de documentos comerciais brasileiros embarcado em dispositivos móveis utilizando a transferência de aprendizagem

A MobileNet V2 exige um menor poder de processamento para ser executada sendo adequada para utilizar em dispositivos móveis. Como esse modelo é baseada na ImageNet, foi treinado usando um conjunto de dados de 1.000 classes do conjunto de dados original da ImageNet com mais de 1 milhão de imagens de treinamento. A arquitetura da MobileNet é representada na Figura 2. Arquitetura da MobileNet utiliza camadas de convolução de profundidade e convolução pontual que substituem as camadas de convolução padrão, exceto a primeira camada, essa utiliza uma camada totalmente convolucional. A arquitetura detalhada é apresentada na Tabela 1.

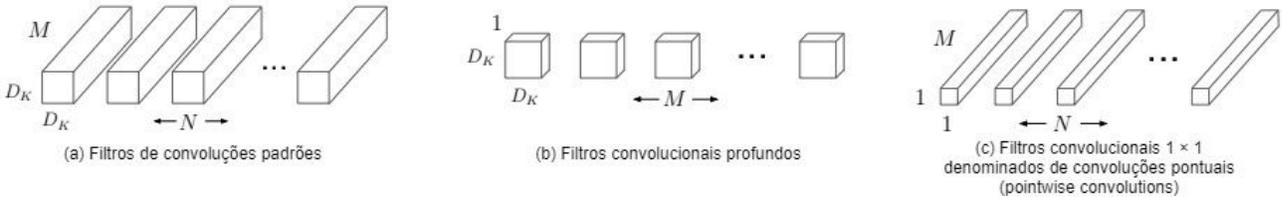


Figura 2: Os filtros convolucionais padrão em (a) são substituídos por duas camadas: convolução em profundidade em (b) e convolução pontual em (c) para criar um filtro separável em profundidade.

Fonte: HOWARD, Andrew G. et al. (2017, p.9).

Tabela 1: Arquitetura da MobileNet V2.

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32 \text{ dw}$	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64 \text{ dw}$	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128 \text{ dw}$	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128 \text{ dw}$	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256 \text{ dw}$	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256 \text{ dw}$	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
$5 \times$ Conv dw / s1	$3 \times 3 \times 512 \text{ dw}$	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512 \text{ dw}$	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024 \text{ dw}$	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

Fonte: HOWARD, Andrew G. et al. (2017, p.9).

3. Materiais e Métodos

Este trabalho propõe a transferência de aprendizagem de um modelo de uma rede neural pré-treinada e incluir essa aprendizagem em um novo modelo capaz de classificar imagens de documentos comerciais brasileiros. Para tal, utilizamos um algoritmo em python integrado com o Tensorflow. A máquina utilizada tem as seguintes configurações: Sistema operacional: Windows 10; Memória Ram: 8 GB e Processador: Intel Core I7-8700. A Figura 3 ilustra a sequência dos passos para avaliação, que serão descritos de forma mais específica nas subseções posteriores.



Figura 3: Fluxo para avaliação dos modelos de classificação de imagens de documentos.

Fonte: Próprio autor (2020)

3.1 Separação da base de imagens

Primeiramente as imagens dos documentos foram montadas com planos de fundos diferentes e também foram rotacionadas e inclinadas em posições diferentes. O objetivo de aplicar essas técnicas citadas anteriormente é fazer com que a rede neural possa aprender a classificar o documento independente do plano de fundo, rotação e inclinação aplicada a imagem.

A base de imagem foi dividida em sete classes de saídas: CNH aberta, CNH fechada parte superior, CNH fechada parte inferior, RG frente, RG verso, comprovante de residência e a **classe de outros documentos**. Essa última classe mencionada é composta por imagens aleatórias. Essa abordagem de utilizar a **classe outros** é adequada pois pode acontecer de a rede neural tentar classificar uma imagem que não seja de algum tipo de documento comercial brasileiro. Portanto, caso a imagem não for de nenhum dos outros 6 tipos de classificação a rede deve definir essa imagem como a **classe outros**.

No algoritmo de treinamento as imagens foram separadas em 80% para treino e 20% para validação. Na Tabela 2 podem ser observados os quantitativos de exemplos de imagens adicionados nos conjuntos de treinamento e validação, para cada uma das 7 classes de interesse. Na Figura 4 podem ser observadas algumas imagens de documentos comerciais brasileiros da base de treinamento.



Figura 4: Imagens de documentos comerciais brasileiros da base de treinamento.

Fonte: Próprio autor (2019)

Tabela 2: Quantidade de imagens para cada classe da rede neural de documentos comerciais brasileiros.

Classe	Treino (80%)	Validação (20%)	Total (100%)
CNH aberta	1441	360	1801
CNH fechada parte inferior	1415	353	1768
CNH fechada parte superior	1255	313	1568
Outros	1160	289	1449
Comprovante de residência	1159	289	1448
RG verso	1151	287	1438
RG frente	1213	303	1516
Total	8794	2194	10988

Fonte: Próprio autor (2020)

3.2 Treinamento

Após a separação da base de imagens aplicamos a técnica de transferência de aprendizagem a partir da arquitetura da MobileNet V2 treinada [3]. Foi aplicada uma camada *dropout* com *rate* igual a 0.2 para evitar o *overfitting* e uma camada final densa (totalmente conectada) seguida da função de ativação *softmax*. Foram utilizadas 7 classes como descrito na Tabela 2. A arquitetura de rede neural aqui proposta para classificação de documentos comerciais brasileiros é ilustrada na Tabela 3. As camadas adicionadas podem ser observadas em destaque na Tabela 3.

Tabela 3: Arquitetura da rede de classificação de documentos comerciais brasileiros.

Type / Stride	Filter Shape	Input Size
Conv / s2	3 x 3 x 3 x 32	224 x 224 x 3
Conv dw / s1	3 x 3 x 32 dw	112 x 112 x 32
Conv / s1	1 x 1 x 32 x 64	112 x 112 x 32
Conv dw / s2	3 x 3 x 64 dw	112 x 112 x 64
Conv / s1	1 x 1 x 64 x 128	56 x 56 x 64
Conv dw / s1	3 x 3 x 128 dw	56 x 56 x 128
Conv / s1	1 x 1 x 128 x 128	56 x 56 x 128
Conv dw / s2	3 x 3 x 128 dw	56 x 56 x 128
Conv / s1	1 x 1 x 128 x 256	28 x 28 x 128
Conv dw / s1	3 x 3 x 256 dw	28 x 28 x 256
Conv / s1	1 x 1 x 256 x 256	28 x 28 x 256
Conv dw / s2	3 x 3 x 256 dw	28 x 28 x 256
Conv / s1	1 x 1 x 256 x 512	14 x 14 x 256
5 x Conv dw / s1	3 x 3 x 512 dw	14 x 14 x 512
5 x Conv / s1	1 x 1 x 512 x 512	14 x 14 x 512
Conv dw / s2	3 x 3 x 512 dw	14 x 14 x 512
Conv / s1	1 x 1 x 512 x 1024	7 x 7 x 512
Conv dw / s2	3 x 3 x 1024 dw	7 x 7 x 1024
Conv / s1	1 x 1 x 1024 x 1024	7 x 7 x 1024
Avg Pool / s1	Pool 7 x 7	7 x 7 x 1024
FC / s1	1024 x 1000	1 x 1 x 1024
Dropout	1024 x 1000	1 x 1 x 1024
FC / s1	1024 x 1000	1 x 1 x 1024
Softmax / s1	Classifier	1 x 1 x 7

Fonte: Próprio autor (2020)

No processo de treinamento da arquitetura proposta, os pesos originais da MobileNet V2 são congelados, sendo atualizados apenas os pesos das camadas dropout e camada densa. Para treinamento do modelo foi utilizado o algoritmo de gradiente descendente estocástico com taxa de aprendizagem igual a 0,005 e um momento com valor de 0,9. A função para calcular a perda foi a entropia cruzada categórica com o objetivo de classificar apenas um resultado para os 7 tipos de classes. O modelo foi executado em 5 épocas com *batch size* igual a 32.

4. Resultados

Para criar a matriz de confusão foram utilizadas as imagens do subconjunto de validação. É possível observar que os documentos obtiveram sucesso na maioria da classificação das imagens. No caso da CNH aberta, CNH fechada inferior e superior a taxa de acerto não foi tão assertiva em comparação às outras classes de saída. Isso sugere que a rede tem o desempenho melhor em classificar imagens com características mais visualmente perceptíveis.

A rede pode reduzir a taxa de acerto da classificação ou até mesmo classificar a imagem de forma equivocada, caso algumas das classes de saídas tenham características semelhantes. Já a classe de outros documentos foi misturada de imagens com características diferentes das outras classes de saída e obteve a maior taxa de acerto.

Em resumo, podemos observar que a utilização da transferência de aprendizagem, associada ao ajuste de arquitetura da MobileNet V2 aqui proposto, permitiu classificar imagens de documentos comerciais com características visualmente diferentes, independente do conteúdo textual, posição, inclinação e outras distorções. Os resultados podem ser observados na matriz de confusão apresentada na Tabela 4.

Tabela 4: Matriz de confusão da rede neural de documentos comerciais brasileiros

		Mobilenet V2						
		Classe prevista						
Classe real		CNH aberta	CNH fechada parte inferior	CNH fechada parte superior	Outros	Comprovante de residência	RG verso	RG frente
	CNH aberta	352	6	2	0	0	0	0
	CNH fechada parte inferior	2	348	0	0	3	0	0
	CNH fechada parte superior	4	1	308	0	0	0	0
	Outros	0	1	0	287	1	0	0
	Comprovante de residência	6	6	2	0	273	2	0
	RG verso	0	5	0	0	0	282	0
	RG frente	1	1	1	0	0	0	300

Fonte: Próprio autor (2019)

Na avaliação da acurácia para a 5 épocas foi obtida uma taxa média de 95,04%, um desvio padrão de aproximadamente 0,043 e a maior taxa foi de 97,90% para a base de treino. Para a base de validação foi obtida uma acurácia média de 96,99%, um desvio padrão de aproximadamente 0,012 e a maior taxa foi de 98,34%. Na Figura 5 podem ser observados os resultados das acurácias de treinamento e validação.

Também é válido ressaltar que o dropout atua apenas no processo de treinamento o que gera uma regularização para a saída dos dados. Já no processo de validação não ocorre a regularização, o que explica uma taxa de erro menor na validação.

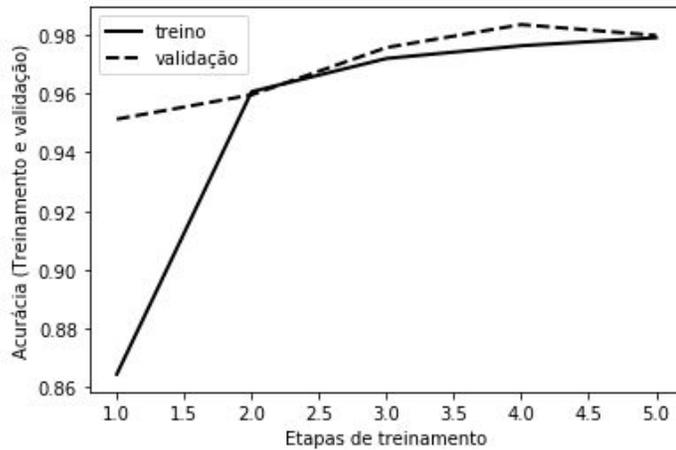


Figura 5: Gráfico representando a acurácia do modelo de classificação de imagens de documentos comerciais brasileiros em relação às etapas de treinamento.

Fonte: Próprio autor (2020)

Na avaliação da função de perda para as 5 épocas executadas foi obtida uma função de perda média de 0,6435, um desvio padrão de aproximadamente 0,080 e a menor taxa foi de 0,5855 para a base de treino. Para a base de validação foi obtida uma função de perda média de 0,5987 um desvio padrão de aproximadamente 0,025 e a menor taxa foi de 0,5762. Na figura 6 podem ser observados os resultados das funções de perdas de treinamento e validação.

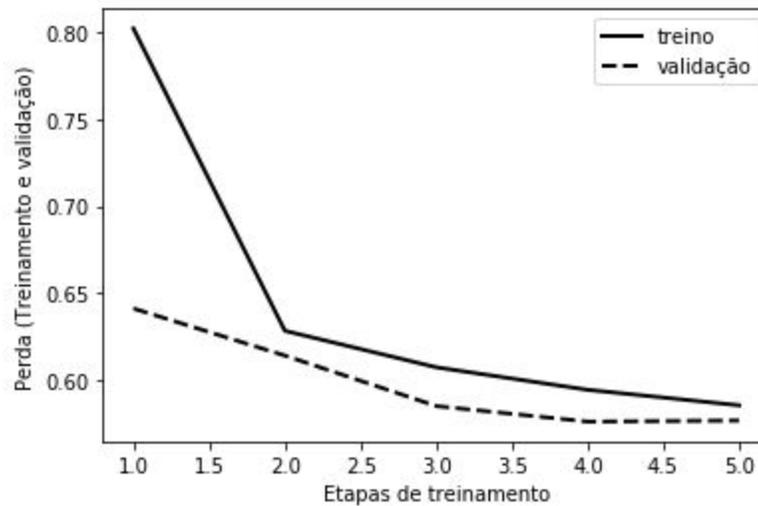


Figura 6: Gráfico representando a função de perda do modelo de classificação de imagens de documentos comerciais brasileiros em relação às etapas de treinamento.

Fonte: Próprio autor (2020)

5. Implementação e Avaliação do Sistema de Classificação em Smartphones

Nesta seção, apresentamos as decisões de projeto para implementação do Sistema descrito anteriormente para Smartphones na plataforma Android, bem como algumas considerações a respeito do protótipo desenvolvido.

Após a finalização do treinamento da rede, foi gerado um arquivo (*saved_model.pb*) com o tamanho de 2,32 MB. Esse arquivo representa o modelo salvo da rede de classificação. Em seguida, esse modelo foi convertido para outro tipo de arquivo (*converted_model.tflite*) com o tamanho de 8,7 MB. A finalidade dessa conversão foi utilizar o modelo embarcado em dispositivos móveis. Também foi gerado um arquivo (*labels.txt*) contendo os nomes das classes do modelo de classificação.

O teste foi realizado no dispositivo Samsung Galaxy J7 Neo com o sistema operacional Android versão 7.0, uma memória RAM de 2 GB, processador 1.6 GHz 8 Core, câmera com 13 Megapixels e resolução de 4128 x 3096 pixel.

Para utilizar o arquivo (*converted_model.tflite*) integrado com o aplicativo Android foi preciso utilizar as recomendações do Google como pode ser observado em [9]. Para embarcar o modelo no aplicativo foi necessário colocar os arquivos *converted_model.tflite* e *labels.txt* na pasta *Assets* do Android. Abaixo seguem os principais códigos de integração:

O primeiro bloco demonstra o construtor para o classificador de imagens. Como pode ser observado na Figura 7.

```
Classifier(Activity activity) throws IOException {
    tfliteModel = loadModelFile(activity);
    tflite = new Interpreter(tfliteModel, tfliteOptions);
    labels = loadLabelList(activity);
    imgData =
        ByteBuffer.allocateDirect(
            DIM_BATCH_SIZE *
            getImageSizeX() *
            getImageSizeY() *
            DIM_PIXEL_SIZE *
            getNumBytesPerChannel());
    imgData.order(ByteOrder.nativeOrder());
    Log.d(TAG, "Created a Tensorflow Lite Image Classifier.");
}
```

Figura 7: Construtor para o classificador de imagens.

Fonte: [9] (2020)

A próxima linha cria a instância do interpretador de classificação baseado no modelo de classificação construído. Como pode ser observado na Figura 8.

```
tflite = new Interpreter(tfliteModel, tfliteOptions);
```

Figura 8: Instância do interpretador de classificação.

Fonte: [9] (2020)

A linha seguinte identifica os dados (*ByteBuffer*) das imagens baseados em suas dimensões para serem classificadas. Como pode ser observado na Figura 9.

```
imgData = ByteBuffer.allocateDirect(  
    DIM_BATCH_SIZE *  
    getImageSizeX() *  
    getImageSizeY() *  
    DIM_PIXEL_SIZE *  
    getNumBytesPerChannel()  
);
```

Figura 9: Dados (*ByteBuffer*) das imagens para serem classificadas.

Fonte: [9] (2020)

A próxima linha identifica as classes de saída do modelo criado. Como pode ser observado na Figura 10.

```
labels = loadLabelList(activity);
```

Figura 10: Classes de saída do modelo de criado.

Fonte: [9] (2020)

A comando seguinte executa a classificação das imagens baseado nas classes do modelo. Como pode ser observado na Figura 11.

```
tflite.run(imgData, labelProbArray);
```

Figura 11: Comando para executar a classificação das imagens.

Fonte: [9] (2020)

A tela de câmera pode ser observada juntamente com a porcentagem de classificação das classes do modelo na parte inferior da tela do aplicativo na Figura 12.



Figura 12: Aplicativo com o modelo de classificação de imagens.

Fonte: [9] (2020)

As classificações das imagens são executadas a cada novo *frame* da câmera com um tempo de resposta de aproximadamente 0,5 segundos.

6. Conclusão

Neste trabalho foi proposta a avaliação de um modelo de classificação de imagens de documentos comerciais brasileiros utilizando a transferência de aprendizagem do modelo MobileNet V2. Essa avaliação resultante do modelo proposto pode facilitar a classificação de documentos em ambientes de produção empresarial. Além disso, possibilitar uma nova perspectiva para processamento na classificação de imagens em um contexto com baixos recursos computacionais disponíveis, como por exemplo em dispositivos móveis.

A principal contribuição desse trabalho está na possibilidade de aplicar o modelo proposto como sistema embarcado em dispositivos móveis e também ser utilizado como um sistema de classificação de documentos baseado em redes neurais em ambientes empresariais.

Foi realizado um experimento para verificar a assertividade do mecanismo de integração proposto, onde conseguimos uma acurácia de 98,34% para a base de validação em condições semelhantes ao conjunto de dados utilizado para treinamento. O fato de algumas classes apresentarem características semelhantes como por exemplo "CNH aberta" e "CNH fechada parte superior" e "CNH fechada parte inferior" pode ter influenciado nos resultados obtidos, apesar dos modelos demonstrarem sucesso na classificação das imagens.

Uma das vantagens percebidas foi o rápido processamento para construção dos modelos, pois como foi feita a transferência de aprendizagem não foi preciso utilizar um elevado processamento computacional. Para trabalhos futuros pode-se abordar uma forma de aumentar a base de dados para mais imagens diversificadas e selecionar classes de saídas com características mais distintas, com o intuito de deixá-lo mais completo e assertivo.

Também podem ser investigadas outros modelos de rede neural com objetivo de aumentar a acurácia do sistema de classificação com foco em imagens de documentos.

7. Referências

- [1] BANCO INTER. Baixe o App e venha para a nossa revolução. **Banco Inter**, 2019. Disponível em: <<https://www.bancointer.com.br/>>. Acesso em: 16 de set. de 2019.
- [2] RUSSAKOVSKY, Olga et al. **Imagenet large scale visual recognition challenge**. International Journal of Computer Vision, v. 115, n. 3, p. 211-252, 2015.
- [3] HOWARD, Andrew G. et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications. **Arxiv Preprint Arxiv:1704.04861**, 2017. Acesso em: 16 de set. de 2019.
- [4] GOOGLE BRAIN TEAM. How to Retrain an Image Classifier for New Categories. **TensorFlow**, 2019. Disponível em: <https://www.tensorflow.org/hub/tutorials/image_retraining>. Acesso em: 16 de set. de 2019.
- [5] GOOGLE BRAIN TEAM. fungi_mobile_V1. **TensorFlow**, 2019. Disponível em: <https://tfhub.dev/svampeatlas/vision/classifier/fungi_mobile_V1/1>. Acesso em: 20 de nov. de 2019.
- [6] KASSANI, Sara Hosseinzadeh et al. A Hybrid Deep Learning Architecture for Leukemic B-lymphoblast Classification. **arXiv preprint arXiv:1909.11866**, 2019. Acesso em: 16 de set. de 2019.
- [7] KRIZHEVSKY, Alex; SUTSKEVER, Ilya; HINTON, Geoffrey E. Imagenet classification with deep convolutional neural networks. In: **Advances in neural information processing systems**. 2012. p. 1097-1105. Acesso em: 7 de fev. de 2020.
- [8] HE, Kaiming et al. Deep residual learning for image recognition. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. 2016. p. 770-778. Acesso em: 7 de fev. de 2020.
- [9] GOOGLE BRAIN TEAM. Recognize Flowers with TensorFlow Lite on Android. **TensorFlow**, 2019. Disponível em: <<https://codelabs.developers.google.com/codelabs/recognize-flowers-with-tensorflow-on-android/#0>>. Acesso em: 23 de jan. de 2020.