Grocery Product Recognition to Aid Visually Impaired People

André de Lima Machado[1], Kelson Aires[1], Rodrigo Veras[1], Laurindo Britto Neto[1]
[1]*Department of Computing – Federal University of Piauí*, Teresina, Brazil
4ndre.machado@gmail.com, {kelson, rveras, laurindoneto}@ufpi.edu.br

***Abstract*—This paper proposes a new approach in object recognition to assist visually impaired people. This approach achieved accuracy rates higher than the approaches proposed by the authors of the selected datasets. We applied Data Augmentation with other techniques and adjustments to different Pre-trained CNNs (Convolutional Neural Networks). The ResNet-50 based approach achieved the best results in the most recent datasets. This work focused on products that are usually found on grocery store shelves, supermarkets, refrigerators or pantries.**

***Index Terms*—Grocery Product Recognition, Object Recognition, Computer Vision**

I. Introdução

Segundo a Organização Mundial de Saúde [1], no mundo existem cerca de 2,2 bilhões de pessoas cegas ou com baixa visão. O Instituto Brasileiro de Geografia e Estatística [2] estimou que existem cerca de 35,8 milhões de brasileiros com deficiência visual (18,8% da população do Brasil); 6,6 milhões de brasileiros cegos ou com baixa visão (3,4% da população), sendo 506,3 mil cegos (0,3% da população).

As tecnologias assistivas visam auxiliar pessoas com deficiência a alcançar maior independência e inclusão. Existem diversas abordagens em visão computacional e aprendizado de máquina, que são exploradas na criação de sistemas para o auxílio de pessoas com deficiência visual, por exemplo: navegação [3], leitura de textos [4], identificação de cédulas monetárias [5], reconhecimento de faces [6], [7], entre outros.

Das diversas possibilidades de auxílio que a visão computacional e o aprendizado de máquina podem apresentar para as tecnologias assistivas, este trabalho focou nas abordagens em reconhecimento de objetos que possam ajudar pessoas com deficiência visual a reconhecer produtos que, geralmente, são encontrados em prateleiras de mercearias, supermercados, geladeiras ou despensas. Um sistema para reconhecer tal categoria de objetos seria bastante útil para a pessoa com deficiência visual, visto que ela poderia verificar, sem ajuda de terceiros, se na geladeira de sua casa ou em sua despensa, por exemplo, tem uma caixa de suco ou uma caixa de leite.

O reconhecimento de objetos possui diversos desafios, tais como: condições descontroladas nos ambientes, grandes variações na iluminação, variações de fundo, variações da orientação e distância do objeto em relação à câmera, oclusões parciais etc. Há ainda o problema de existirem objetos diferentes visualmente, porém idênticos ao tato, como, por exemplo, uma caixa de leite versus uma caixa de suco, entre outros.

O objetivo deste trabalho é propor uma abordagem em reconhecimento de produtos de mercearia para auxiliar pessoas com deficiência visual, baseada em Rede Neural Convolucional (*Convolutional Neural Network* — CNN), que utiliza técnicas de Transferência de Aprendizado (*Transfer Learning*), Ajuste Fino (*Fine-tuning*), Aumento de Dados, *Batch Normalization* e *Drop Out*. Tal abordagem será incorporada a um sistema *wearable* [6], [7], usando uma arquitetura cliente-servidor. Assim, esse sistema ajudaria aos seus usuários na identificação de itens encontrados em diversos ambientes, como em sua própria casa, ou até mesmo o auxiliando a fazer compras de modo independente.

O restante deste trabalho está organizado da seguinte forma: a Seção II fornece uma visão geral dos artigos da literatura científica relacionados à este trabalho; a Seção III descreve brevemente os materiais, como as bases de dados, e a metodologia utilizados para validar a abordagem proposta; a Seção IV explica em detalhes a abordagem proposta; a Seção V apresenta os resultados obtidos pelos experimentos de validação da abordagem proposta; na Seção VI são discutidos os resultados apresentados; e, finalmente, as conclusões e perspectivas para trabalhos futuros são fornecidas pela Seção VII.

II. Trabalhos Relacionados

Machado et al. [8] realizaram uma revisão sistemática da literatura, que identificou o estado da arte em reconhecimento de produtos de mercearia. Além disso, tal trabalho descreveu e analisou cinco bases de dados públicas para a classificação de produtos de mercearia. Dentre os trabalhos, esta seção destaca os mais relevantes.

Rivera-Rubio et al. [9] propuseram três abordagens: (1) *Scale-Invariant Feature Transform* (SIFT) + K-Means + *Support Vector Machine* (SVM) ; (2) SIFT + *Locality-constrained Linear Coding* (LLC) ; e (3) SIFT + *Principal Component Analysis* (PCA) + *Fisher Vector Encoding* + SVM. O artigo introduziu a base de dados *SHORT-100*. Foram realizados testes de acurácia em dois conjunto de imagens: (1) capturadas por *smartphones* e (2) extraídas de vídeos. No primeiro conjunto, a melhor acurácia média foi obtida pela abordagem SIFT + K-Means + SVM com 77,51%. No segundo conjunto, a melhor abordagem foi SIFT + LLC com acurácia média de 69,41%.

Esta pesquisa foi realizada com apoio da Fundação de Amparo à Pesquisa do Estado do Piauí (FAPEPI), por meio dos editais FAPEPI/CAPES Nº 005/2018 e FAPEPI/MCT/CNPq Nº 007/2018 (PPP).

Varol e Kuzu [10] consideraram o problema do reconhecimento de apenas uma mercadoria específica em imagens de prateleiras de diversos estabelecimentos. O produto considerado eram embalagens de cigarro. O artigo introduziu a base de dados *Grocery*. Os autores propuseram uma abordagem em que os objetos de interesse eram detectados pelo Viola-Jones, treinado com características de *Histogram of Oriented Gradients* (HOG). Da região detectada foi utilizada apenas 40% da imagem do topo da embalagem, local onde se encontra o logotipo, para classificação. Em seguida, a imagem do logotipo foi representada com informações de forma (descritas por SIFT) e cor (descritas pelo modelo HSV), por meio da abordagem BoW. Foram criados vocabulários visuais de forma e de cores, agrupando seus respectivos descritores por meio do K-means. Ao final, foi utilizado o classificador SVM para reconhecer a marca do produto. Foram realizados experimentos de acurácia que obtiveram os seguintes resultados: 85,9%, usando apenas descritores SIFT; 60,5%, usando apenas HSV; e 92,3%, usando ambos descritores.

Jund et al. [11] fizeram uso de *Transfer Learning*, por meio da arquitetura CaffeNet. Os autores propuseram uma *Fine-tuning* CNN (FT-CNN). O artigo introduziu a base *Freiburg Groceries*, na qual realizaram testes de acurácia. A abordagem proposta pelos autores obteve acurácia média de 78,9%.

Klasson et al. [12] utilizaram três diferentes CNNs pré-treinadas: AlexNet, VGG-16 e DenseNet-169. Elas serviram como descritores de características para um classificador SVM. As camadas foram extraídas a partir de CNNs com e sem a aplicação de *Fine-tuning* sobre elas. Além disso, foi proposta uma outra abordagem, utilizando apenas FT-CNNs como descritores e classificadores. O artigo introduziu a base de dados *Grocery Store*. Tal base permitiu testes de classificação de produtos específicos (*Fine-grained classification*) e teste de classificação de categorias de produto (*Coarse-grained classification*). A abordagem FT-CNN DenseNet-169 com o SVM obteve melhor taxa de acurácia, 85,0%, classificando produtos específicos. Já a abordagem CNN DenseNet-169, sem *Fine-tuning*, com o classificador SVM obteve melhor taxa de acurácia, 85,2%, classificando categorias de produtos.

Em contrapartida, as abordagens propostas neste trabalho, em especial a baseada na CNN ResNet-50, obtiveram resultados superiores a todos os trabalhos relacionados.

III. Materiais e Métodos

Os experimentos para validação da abordagem proposta neste trabalho foram realizados em uma máquina i7 com 3.00 GHz e memória RAM de 16 GB, equipada com a GPU NVIDIA GeForce GTX 1060 6 GB. A seguir são descritas as bases de dados e os métodos de validação utilizados.

A. Bases de Dados

Machado et al. [8] forneceram uma descrição e uma análise mais detalhada sobre as bases de dados utilizadas por este trabalho. Além disso, eles descreveram a base Grozi-120 [13]. Tal base de dados não foi utilizada neste trabalho. Apesar da base conter imagens de 120 produtos separados individualmente para testes de classificação, os autores da base realizaram testes de detecção em vez de testes de classificação. Eles utilizaram um outro grupo de imagens da base, que contém apenas fotos de prateleiras com vários produtos, para a detecção dos produtos. Isso inviabilizou a utilização da Grozi-120 para a comparação com as outras abordagens descritas neste trabalho, que são voltadas para testes de classificação.

1) SHORT-100: Rivera-Rubio et al. [9] desenvolveram a base de dados *SHORT-100*, com conjuntos de treino e teste definidos. Inicialmente, a base era chamada de *SHORT-30*, pois possuía apenas 30 classes de produtos. Entretanto, somente o seu conjunto de treino foi atualizado para 100 produtos. Cada uma das 30 classes do conjunto de teste foi dividida em dois grupos: *Blindfolded* (BF) – imagens capturadas por pessoas com os olhos vendados e *Sighted* (SG) – imagens tiradas por pessoas com visão. As imagens desses conjuntos de teste foram adquiridas por meio de 30 *smartphones* distintos. Durante a captura, enquanto uma das mãos segurava o *smartphone*, a outra segurava o produto. Isso fez com que as imagens pudessem conter outros objetos no fundo como a mão do fotógrafo. Cada conjunto de teste ainda se subdivide em imagens fixas (*Still Images* - ST) e imagens retiradas de *frames* de vídeo (*Video Frames* - VF), porém com boa qualidade. A quantidade de imagens em cada um desses subconjuntos é: ST-SG: 2.797; VF-SG: 92.293; ST-BF: 1.225; VF-BF: 39.121. No total, o conjunto de teste possui 135.436 imagens. Neste trabalho, os testes foram limitados aos subconjuntos de SG (ST-SG e VF-SG), pois os autores dessa base também restringiram seus testes a esses subconjuntos.

2) Grocery: Varol e Kuzu [10] desenvolveram a base *Grocery*. Essa base consiste em imagens de um único produto, apenas embalagens de cigarro. Essa base de dados possui produtos rotulados divididos em dez classe (marcas dos cigarros). Ela também possui 10.440 produtos não rotulados, que podem ser usados como uma classe negativa. Para os produtos rotulados, a base fornece dois conjuntos: um com as imagens do produto inteiro (imagens dos produtos) e o outro com as imagens recortadas na parte superior (40% do topo da imagem) com a marca do produto (imagens das marcas). Ambos já estão separados em conjuntos de treino e de teste. O conjunto de imagens das marcas pode ser considerado como um pré-processamento sobre o conjunto dos produtos, em que foram removidas a parte inferior da imagens. A parte inferior dessas imagens consistem em figuras de advertência que as embalagens de cigarros são obrigadas a ter, independente da marca, o que pode confundir métodos de reconhecimento.

3) Freiburg Groceries: Jund et al. [11] desenvolveram a base de dados *Freiburg Groceries*. Tal base contém 4.947 imagens de produtos capturadas por *smartphones* em supermercados. A *Freiburg Groceries* é dividida em 25 classes desbalanceadas, cada uma possuindo de 97 a 370 imagens, que não correspondem a produtos específicos, mas a categorias de produtos (*coarse classification*). Não foi fornecido uma divisão em conjuntos de treino e de teste.

4) Grocery Store: Klasson et al. [12] desenvolveram a base *Grocery Store*. Tal base contém 5.125 imagens naturais de 81 classes de frutas, legumes e produtos em caixas (iogurte, leite e suco), para classificação *fine-grained*. As imagens são também divididas em 43 categorias, em que, por exemplo, os produtos *Royal Gala* e *Granny Smith* pertencem à mesma categoria *Apple*. Isso viabiliza a classificação *coarse-grained*. A base possui os conjuntos de treino e de teste bem definidos.

B. Metodologia Experimental

Foram realizados testes de acurácia nas bases de dados, da mesma forma adotada por seus autores. As bases *Grocery* e *Grocery Store* foram divididas pelo método *holdout*, pois possuem conjuntos de treino e teste bem definidos. Na base *SHORT-100*, os autores forneceram um código-fonte, em que computam a acurácia média usando o *5-fold cross validation* (*5-fold*) apenas nos grupos ST-SG e VF-SG. A *Freiburg Groceries* também usou o *5-fold*, sendo necessário seguir as instruções dos autores para repetir os seus testes. Para a realização de comparações, foram usadas as métricas de validação a seguir: função *loss* e taxa de acurácia. Em alguns casos, foram aplicados os testes de hipótese *Wilcoxon Rank Sum* (WRS) e *Wilcoxon Signed Rank* (WSR) para verificar diferenças estatísticas.

IV. Abordagem Proposta

A abordagem proposta neste trabalho reusou diferentes CNNs pré-treinadas fornecidas pela biblioteca Keras[1]. Como neste trabalho é estabelecido um problema de classificação multi-classe, foi usada na última camada a função de ativação *softmax*. Foram realizados testes visando aumentar a acurácia e diminuir a função *loss*, por meio do uso dos métodos: transferência de aprendizado com ajuste fino, aumento de dados, camadas de *Batch Normalization* e de *Drop Out*. Foram feitos ajustes na resolução das imagens e no valor do tamanho do *mini-batch*. Utilizou-se funções de normalização fotométrica da biblioteca Keras de cada rede pré-treinada. No restante desta seção serão fornecidos mais detalhes sobre a abordagem proposta e sobre cada método utilizados.

A. CNNs Pré-treinadas

Empregou-se diferentes CNNs, previamente treinadas na base de dados ImageNet, todas fornecidas pela biblioteca Keras: VGG-16, VGG-19, Inception-V3, Xception, ResNet-50 e DenseNet-201. A rede VGG-16 teve camadas de *Batch Normalization* adicionadas internamente, antes das camadas de *Pooling*. Ela também foi adicionada antes da camada densa de saída. Esse novo modelo será chamado de VGG-16-BN.

A rede VGG-19 também teve a adição dessas camadas internamente, ao que será referido como VGG-19-BN. Foi observado que usar qualquer CNN com os pesos zerados tinha resultados muito inferiores. Possivelmente, as redes VGG-16-BN e VGG-19-BN poderiam ter resultados ainda melhores, caso já tivessem as camadas internas de *Batch Normalization* quando foram treinadas na base ImageNet.

[1]https://keras.io/. Acesso em: 03 ago. 2021

Por conta de algumas CNNs exigirem maior poder computacional, algumas vezes foi necessário diminuir o tamanho do *mini-batch* (bs – *mini-batch size*) ou das dimensões da imagem para viabilizar a execução. Salvo exceções relatadas nos resultados, as configurações utilizadas nos testes foram as seguintes: execuções em 50 épocas, sendo interrompido antes se não houver melhora na *loss* da validação durante 30 épocas; redimensionamento da imagem para 200×200, exceto aos modelos computacionalmente maiores (ResNet-50: 150×150; Xception e DenseNet-201: 100×100); bs de valor 32; três camadas de *Drop Out* intercaladas com duas camadas densas de tamanho 1.024; uma camada de *Batch Normalization* mais uma camada densa com a função *softmax*, em que seu tamanho deve ser o número de classes da base.

B. Ajuste Fino Raso, Médio e Completo

Foram realizados testes com Ajuste Fino raso (ou *shallow Fine-tuning*), isto é, manteve-se como treináveis apenas as camadas densas adicionadas à rede, que diminuiu a complexidade da rede, exigindo menor poder computacional e, por isso, demonstrou maior rapidez na execução. Experimentos com Ajuste Fino médio também foram feitos. Contudo, em ambos, a acurácia foi muito inferior à obtida ao realizar o Ajuste Fino completo. Essa necessidade de melhor ajustar os descritores pode ser explicada devido aos tipos de classes em que eles foram pré-treinados, que, apesar de incluir alguns produtos, englobavam um vasto número de animais, plantas, paisagens e objetos distintos.

C. Operações de Aumento de Dados

A técnica de Aumento de Dados contribuiu bastante ao melhor desempenho do processo de reconhecimento. Foram exploradas diferentes faixas de valores para diversas das operações de aumento de dados fornecidas pelo Keras. Dentro dos valores definidos, a intensidade de cada operação é escolhida aleatoriamente em tempo de execução.

A quantidade de imagens geradas no processo de Aumento de Dados por época, para cada imagem de treino, foi igual à divisão inteira da quantidade de imagens de treino pelo bs. Após vários testes, algumas das operações de Aumento de Dados do Keras foram escolhidas, visto que favoreceram o aprendizado das CNNs, diminuindo o *overfitting*. A Tabela I mostra o nome dessas operações e os valores atribuídos a elas.

TABELA I
Operações de Aumento de Dados utilizadas.

Operação	Valor	Descrição
rotation_range	70	Rotaciona a imagem
width_shift_range	0.3	Move a imagem horizontalmente
height_shift_range	0.3	Move a imagem verticalmente
shear_range	10	Cisalhamento
zoom_range	[0.3,1.4]	Amplia ou diminui a escala da imagem
horizontal_flip	True	Gira a imagem horizontalmente
fill_mode	”constant”	Escolhe o preenchimento do fundo

D. Camadas de Batch Normalization Adicionadas

Ao substituir-se o classificador por camadas densas nas redes VGG-16 e VGG-19, sucedia uma enorme dificuldade no aprendizado. A adição de camadas de *Batch Normalization* elevou a curva de aprendizado inicial e possibilitou alcançar maiores taxas de acerto. Foram feitos testes com a adição antes e depois das camadas de *Pooling*, tendo melhores resultados precedendo a camada.

E. Camadas de Drop Out Adicionadas

Apesar de camadas de *Drop Out* serem um mecanismo para refrear o *overfitting*, dependendo da quantidade dessas camadas e dos valores selecionados para desconectar aleatoriamente neurônios da rede, a aprendizagem pode ser comprometida. Isso pode gerar resultados piores, principalmente em um reconhecimento multi-classe. Deve-se ter ainda mais cautela quando elas são utilizadas juntamente com camadas de *Batch Normalization*. Após diversos testes, utilizou-se como configuração três camadas de *Drop Out* intercaladas pelas camadas densas do classificador, possuindo os seguintes valores, da primeira para a última: 10%, 40% e 5%.

F. Pré-processamento

Usou-se como pré-processamento as funções de normalização fotométrica fornecidas pelo Keras com as CNNs pré-treinadas. Além disso, todos os pixels das imagens foram normalizados entre 0 e 1. Na base *Grocery Products* foi feito um pré-processamento de corte sobre as imagens de teste para remover as bordas, restando 70% da imagem original, o que melhorou os resultados, devido a vários casos com objetos de outras classes nas bordas.

G. Escolha do Otimizador e Taxa de Aprendizado

Foram comparados os resultados de diferentes otimizadores, que são responsáveis por minimizar o custo da função de perda. Os otimizadores utilizados foram: *Adaptive Moment Estimation* (Adam), *Root Mean Square Propagation* (RMSProp) e *Stochastic Gradient Descent* (SGD). Os otimizadores SGD e Adam obtiveram bons resultados. Porém, a taxa de aprendizado utilizada para o otimizador SGD foi de 0,01, enquanto para o Adam se fixou em 0,0001. Na maioria dos testes foi usado Adam.

H. Redução Automática da Taxa de Aprendizado

Um rápido aprendizado ocorreu nas primeiras épocas. Em seguida, como os otimizadores tinham dificuldade nos cálculos para encontrar a solução ótima, a redução automática da taxa de aprendizado melhorou o treinamento. Foi estabelecido que a taxa de aprendizado fosse reduzida pela metade cada vez que a *loss* do treinamento não obtinha melhora, com um mínimo de 3 épocas para poder ser reduzida novamente. Definiu-se que ela fosse reduzida só até 40 vezes o seu valor inicial.

I. Ajustes de Resolução, Mini-batch e Camadas Densas

O Manual do Keras[2] recomenda usar um tamanho de bs grande, pois aumentar o subconjunto de imagens processadas em paralelo pode melhorar o aprendizado, desde que não ocorra perda de memória no processamento. Preferencialmente, o número deve ser na forma de potência de dois. Aumentar a resolução das imagens e o valor do bs frequentemente melhora os resultados, mas ambos exigem maior capacidade computacional. Contudo, um deles pode ter maior influência sobre o aprendizado. Nesse caso, é possível aumentar um deles e diminuir o valor do outro moderadamente. Deve ser considerado também que valores altos para o bs aceleram a velocidade do treinamento, enquanto valores pequenos o atrasam. A quantidade e o tamanho das camadas densas também influem nos resultados. Porém, camadas grandes demais e em grande número podem prejudicar os resultados, além de exigir mais do hardware.

V. Resultados

Esta seção expõe os resultados dos testes realizados nas quatro bases de dados utilizadas, recorrendo-se à mesma forma de validação cruzada e de separação dos conjuntos de treino e teste que o fizeram seus autores. Apenas na base SHORT-100 não foram encontradas instruções dessa divisão.

A. Resultados com a SHORT-100

Rivera-Rubio et al. [9] alcançaram 77,51% de acurácia média no conjunto ST-SG e 69,41% no conjunto VF-SG. Os autores não relataram os desvios padrões das acurácias médias. A base SHORT-100 é extensa, com 92,3 GB de espaço em disco (após a remoção de diretórios repetidos nas pastas que possuem *frames* de vídeo). Consequentemente, os testes nessa base foram custosos. Por isso, nela foram utilizadas apenas as redes ResNet-50 e VGG-16-BN. Em testes preliminares, tais abordagens já demonstravam a obtenção dos melhores resultados.

Neste trabalho, os testes com *5-fold* dividiram em cinco partes os grupos ST-SG e VF-SG, calculando a média e o desvio padrão da acurácia e da *loss*. Para os testes com o grupo VF-SG serem mais rápidos, eles foram executados por apenas cinco épocas. Na rede VGG-16-BN e ResNet-50, o valor do bs foi aumentado, respectivamente, para 128 e 64, e as imagens foram reduzidas para 100×100. No conjunto ST-SG, as redes ResNet-50 e VGG-16-BN obtiveram acurácia média igual a, respectivamente, $94,75 \pm 1,79\%$ e $94,75 \pm 1,24\%$. A rede VGG-16-BN alcançou *loss* média de $0,1780 \pm 0,0438$. No conjunto VF-SG, as redes ResNet-50 e VGG-16-BN chegaram à acurácia média, respectivamente, de $83,79 \pm 2,92\%$ e $82,39 \pm 3,35\%$, sendo que tais redes obtiveram *loss* média de, respectivamente, $0,9710 \pm 0,2620$ e $0,8154 \pm 0,0927$.

[2]https://keras.io/getting_started/faq/. Acesso em: 03 ago. 2021

B. Resultados com a Grocery

Varol e Kuzu [10] avaliaram a acurácia da abordagem proposta por eles, apenas usando o grupo de imagens das marcas dos cigarros (40% do topo da imagem). A abordagem proposta por eles alcançou 92,3% de acurácia.

Nos testes deste trabalho, como a base possui apenas 10 classes, as CNNs testadas tiveram altas taxas de acurácia já nas primeiras épocas. Por isso, os testes foram executados em apenas 30 épocas. No conjunto com imagens das marcas, as CNNs VGG-16, VGG-16-BN, VGG-19-BN, ResNet-50, Inception-V3, Xception e DenseNet-201 tiveram resultados de acurácia bastante próximos, sendo a média deles de $96,06 \pm 0,51\%$. Apesar da acurácia alta, ela melhorou com ajustes pela diminuição da resolução para aumentar o valor do bs, chegando ao melhor resultado com 98,18% de acurácia e 0,0991 de *loss*, por meio da abordagem ResNet-50 com bs igual a 64 e resoluções de 100×100. A rede VGG-16-BN, com bs configurado em 128 e resolução de 100×100, alcançou 97,21% de acurácia e 0,1014 de *loss*. As acurácias obtidas para as CNNs pré-treinadas VGG-16, VGG-19-BN, Inception-V3, Xception e DenseNet-201 foram, respectivamente: 95,85%, 95,36%, 95,59%, 96,81% e 96,24%. Observa-se que todas as outras redes obtiveram taxas de acurácia menores que as abordagens ResNet-50 e VGG-16-BN.

C. Resultados com a Freiburg Groceries

Na base Freiburg Groceries, Jund et al. [11] obtiveram $78,9 \pm 0,50\%$. Neste trabalho, a VGG-16-BN obteve a acurácia média de $86,10 \pm 2,20\%$ e a *loss* média de $0,5294 \pm 0,0856$. A ResNet-50 obteve a acurácia média de $86,65 \pm 2,92\%$ e a *loss* média de $0,5971 \pm 0,1360$. Como os testes nessa base demoraram bastante, devido a realização do *5-fold*, foram utilizados apenas as redes ResNet-50 e VGG-16-BN. Em testes preliminares, tais abordagens já demonstravam a obtenção dos melhores resultados.

D. Resultados com a Grocery Store

Klasson et al. [12] realizaram vários testes utilizando algumas camadas pré-treinadas combinadas com diferentes classificadores, alcançando, no melhor resultado, 85,0% de acurácia na classificação *fine-grained*, e 85,2% na classificação *coarse-grained*. Eles também utilizaram uma abordagem com *Fine-tuning*, mas apenas para a classificação *fine-grained*, alcançando 84,0% de acurácia por meio da rede DenseNet-169. Utilizando VGG-16, eles atingiram somente 73,8%.

Neste trabalho, na classificação *fine-grained*, o resultado mais alto de acurácia foi obtido em um experimento com a rede DenseNet-201, com três camadas densas, dimensões 200×200 e bs de valor 8. Esse teste alcançou 90,84% de acurácia e 0,3991 de *loss*. Devido ao baixo valor do bs, foram necessárias aproximadamente 40 horas para finalizar o treinamento, tornando essa abordagem ineficiente. Portanto, esse não foi considerado o melhor resultado. O treinamento da DenseNet-201 nas configurações normais durou cerca de 30 minutos, porém alcançou 84,16% de acurácia.

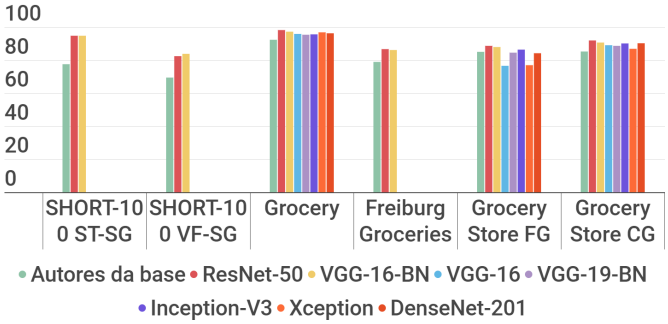

Fig. 1. Resultados de acurácia

Na classificação *fine-grained*, os melhores resultados foram alcançados pela rede ResNet-50. Ela conquistou 88,63% de acurácia e 0,4232 de *loss* na classificação *fine-grained*. Também foram computadas as acurácias para as CNNs VGG-16-BN, VGG-16, VGG-19-BN, Inception-V3, Xception e DenseNet-201, que obtiveram os seguintes valores respectivos: 87,91%, 76,54%, 84,50%, 86,34%, 76,92% e 84,16%.

Na classificação *coarse-grained*, a ResNet-50 alcançou 91,89% de acurácia e 0,3154 de *loss*. As CNNs VGG-16-BN, VGG-16, VGG-19-BN, Inception-V3, Xception e DenseNet-201 obtiveram as seguintes acurácias respectivas: 90,62%, 89,02%, 88,63%, 90,08%, 86,84% e 90,20%. Observa-se que todas as outras redes obtiveram taxas de acurácia menores que as abordagens ResNet-50 e VGG-16-BN.

Por meio da Fig. 1 é possível visualizar graficamente, para cada base de dados apresentada e suas subdivisões, o resultado obtidos pelos autores, seguido dos resultados alcançados neste trabalho. A Tabela II reúne, para todas as bases de dados descritas neste trabalho (coluna Base de Dados), os melhores resultados de acurácia dos autores encontrados na literatura científica (coluna Literatura) e dos resultados das melhores abordagens obtidas por este trabalho (ResNet-50 (coluna ResNet-50) e a VGG-16-BN (coluna VGG-16-BN)), que superaram os resultados obtidos pelos autores da literatura

TABELA II
RESULTADOS DE ACURÁCIA DA LITERATURA VERSUS AS MELHORES ABORDAGENS DESTE TRABALHO (AS MELHORES TAXAS DE ACURÁCIA ESTÃO EM NEGRITO. PARA OS TESTES COM *k-fold*, OS DESVIOS PADRÕES TAMBÉM SÃO INFORMADOS).

Base de Dados		**Acurácia (em porcentagem)**		
		Literatura	**ResNet-50**	**VGG-16-BN**
SHORT-100 [9]	ST-SG	77,51	**94,75**$\pm 1,79\%$	**94,75**$\pm 1,24\%$
	VF-SG	69,41	$82,39 \pm 3,35\%$	**83,79**$\pm 2,92\%$
Grocery [10]		92,3	**98,18**	97,21
Freiburg Groceries [11]		$78,9 \pm 0,5\%$	**86,65**$\pm 2,92\%$	$86,10 \pm 2,20\%$
Grocery Store [12]	Classificação *fine-grained*	85,0	**88,63**	87,91
	Classificação *coarse-grained*	85,2	**91,89**	90,62

analisada. Tal comparação pode ser feita, visto que todos os testes foram realizados com a mesma metodologia experimental realizada pelos autores dos trabalhos relacionados.

VI. Discussão

Os experimentos realizados neste trabalho focaram na etapa de reconhecimento, não realizando testes em bases cujas imagens de teste se limitavam a prateleiras com diversas classes de produtos, visto que são usadas por trabalhos de detecção e não somente de reconhecimento. Portanto, todas as bases testadas possuem imagens de teste em que cada imagem corresponde a apenas uma classe. Com isso, este trabalho considerou o cenário de reconhecer um produto próximo ou na mão do usuário, ou ainda o cenário de reconhecer um produto que já foi detectado, independente de como foi realizada a detecção.

A abordagem proposta para o reconhecimento de produtos de mercearia consistiu em empregar CNNs para reuso, usando a função própria de normalização fotométrica da respectiva CNN pré-treinada, fornecida pelo Keras, aplicando *Fine-tuning* completo, dimensões de imagem reajustadas para 200×200 ou o mais próximo possível, bs igual a 32 ou mais, algumas operações de Aumento de Dados com valores específicos, normalização dos valores dos pixels, redução automática da taxa de aprendizado e, como classificador, três camadas de *Drop Out* intercaladas por duas camadas densas de tamanho 1.024, seguidas por uma camada de *Batch Normalization* e a camada densa de saída. O otimizador Adam (com taxa de aprendizado 0,0001) teve resultados levemente melhores que o SGD (com taxa de aprendizado 0,01). Algumas CNNs exigiram maior capacidade de hardware. Nesse caso, o bs foi mantido e a resolução padrão das imagens foi diminuída para a ResNet-50, a Xception e a DenseNet-201, o que agilizou um pouco o processo de treinamento nessas redes.

Como pode ser visto na Tabela II, na base de dados SHORT-100, a rede VGG-16-BN obteve alguns resultados melhores que a ResNet-50. Contudo, aplicando tanto o teste de hipótese WRS quanto o WSR nos testes com *5-fold* sobre o conjunto VF-SG, em que a rede VGG-16-BN havia sido superior à ResNet-50, obteve-se o *p-valor* $= 0,2222$ e *p-valor* $= 0,0625$, respectivamente. Isso mostra que, para esse caso, as abordagens não possuem diferenças estatísticas significantes com 95% de confiança.

VII. Conclusão e Trabalhos Futuros

A abordagem proposta baseada na ResNet-50 alcançou resultados de acurácia superiores, tanto sobre as abordagens baseada em outras CNNs quanto sobre as abordagens encontradas na literatura (veja a Tabela II). No único teste em que a ResNet-50 não superou a VGG-16-BN (SHORT-100 – VF-SG), o teste de hipótese indicou que as abordagens não possuem diferenças estatísticas significantes. Portanto, a abordagem baseada na ResNet-50 é a recomendada para a tarefa de reconhecer produtos de mercearia, usando a sua função própria de pré-processamento e todos os ajustes e métodos discutidos neste trabalho.

Na revisão sistemática de literatura realizada por Machado et al. [8], foi observado que a partir de 2016 os estudos se concentraram mais em CNNs. Nessa e em diversas áreas de aprendizado de máquina e visão computacional, as abordagens com CNNs lideram o estado da arte. Por isso, este trabalho focou em abordagens baseadas em CNNs, apesar da intenção inicial deste trabalho de usar a abordagem proposta em dispositivos *wearable* para auxiliar pessoas com deficiência visual. Em trabalhos futuros, pretende-se implementar a abordagem proposta usando uma arquitetura cliente-servidor. A ideia básica é transferir do cliente (a partir de um dispositivo *wearable*) para o servidor a tarefa de realizar o processamento mais custoso no reconhecimento. Com isso, será possível verificar se a abordagem funciona em tempo real em um sistema *wearable*. Pretende-se também projetar o sistema *wearable* de forma que economize recursos de hardware, como memória, bateria e processamento. Além disso, pretende-se realizar estudos com arquiteturas de CNNs menores, que apresentam boas taxas de acurácia e são voltadas a dispositivos *wearable*.

Referências

[1] WHO, "Blindness and vision impairment," 2021, [online] Disponível em: https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment [Acesso em: Ago. 2021].

[2] IBGE, "Releitura dos dados de pessoas com deficiência no censo demográfico 2010 à luz das recomendações do grupo de washington," IBGE, RJ, Nota tecnica 01/2018, 2018.

[3] D. Bal, M. M. Islam Tusher, M. Rahman, and M. S. Rahman Saymon, "Navix: A wearable navigation system for visually impaired persons," in *2nd STI*, 2020, pp. 1–4.

[4] G. Vaidya, K. Vaidya, and K. Bhosale, "Text recognition system for visually impaired using portable camera," in *2020 International Conference on Convergence to Digital World - Quo Vadis (ICCDW)*, 2020, pp. 1–4.

[5] L. P. Sousa, R. M. S. Veras, L. H. S. Vogado, L. S. Britto Neto, R. R. V. Silva, F. H. D. Araujo, and F. N. S. Medeiros, "Banknote identification methodology for visually impaired people," in *2020 IWSSIP*, 2020, pp. 261–266.

[6] L. Britto Neto, V. R. M. L. Maike, F. L. Koch, M. C. C. Baranauskas, A. Rocha, and S. Goldenstein, "A wearable face recognition system built into a smartwatch and the visually impaired user," in *ICEIS*, INSTICC. SciTePress, 2015, pp. 5–12.

[7] L. Britto Neto, F. Grijalva, V. R. M. L. Maike, L. C. Martini, D. Florencio, M. C. C. Baranauskas, A. Rocha, and S. Goldenstein, "A kinect-based wearable face recognition system to aid visually impaired users," *IEEE THMS*, vol. 47, no. 1, pp. 52–64, 2017.

[8] A. Machado, R. Veras, K. Aires, and L. Britto Neto, "A systematic review on product recognition for aiding visually impaired people," *IEEE LATAMT*, vol. 19, no. 4, pp. 592–603, 2021.

[9] J. Rivera-Rubio, S. Idrees, I. Alexiou, L. Hadjilucas, and A. A. Bharath, "Small hand-held object recognition test (short)," in *IEEE WACV*, 2014, pp. 524–531.

[10] G. Varol and R. S. Kuzu, "Toward retail product recognition on grocery shelves," in *ICGIP 2014*, Y. Wang, X. Jiang, and D. Zhang, Eds., vol. 9443, International Society for Optics and Photonics. SPIE, 2015, pp. 46 – 52.

[11] P. Jund, N. Abdo, A. Eitel, and W. Burgard, "The freiburg groceries dataset," 2016.

[12] M. Klasson, C. Zhang, and H. Kjellström, "A hierarchical grocery store image dataset with visual and semantic labels," in *IEEE WACV*, 2019, pp. 491–500.

[13] M. Merler, C. Galleguillos, and S. Belongie, "Recognizing groceries in situ using in vitro training data," in *IEEE CVPR*, 2007, pp. 1–8.