AVALIAÇÃO DE TÉCNICAS DE TRANSFERÊNCIA DE APRENDIZADO EM CNNS

Bernardo Augusto Godinho de Oliveira* Pedro Augusto Pinho Ferraz* Thiago Melo Machado-Coelho† Álvaro Henrique de Araújo Rungue* Willian Antônio dos Santos* Flávia Magalhães Freitas Ferreira* Luís Fabrício Wanderley Góes* Gustavo Luís Soares* Carlos Augusto Paiva da Silva Martins*

*Pontifícia Universidade Católica de Minas Gerais
Programa de Pós-Graduação em Engenharia Elétrica
Av. Itaú 525, 30535-012, Belo Horizonte, MG, Brasil

†Universidade Federal de Minas Gerais
Programa de Pós-Graduação em Engenharia Elétrica
Av. Antônio Carlos 6627, 31270-901, Belo Horizonte, MG, Brasil

Email: bernardo.godinho@sga.pucminas.br, pferraz@sga.pucminas.br, thmmcoelho@ufmg.br, alvaro.rungue@sga.pucminas.br, willian.santos.838460@sga.pucminas.br, flaviamagfreitas@pucminas.br, lfwgoes@pucminas.br, gsoares@pucminas.br, capsm@pucminas.br.

Abstract— Convolutional neural networks (CNNs) require a high amount of training data to achieve a good performance, and this data can often be difficult to obtain or costly. Transfer learning techniques are commonly used to reduce this requirement and often increase the final performance of the network due to increased generalization. The objective of this work is to compare different transfer learning techniques and to analyze the results obtained by each of them. The results show that retraining only the final layers increases the performance of the network by 2.5 times, in comparison to retraining the whole network and 6.54 times in conparison to training without the pretraining step.

Keywords— transfer learning, deep learning, object detection

Resumo— As redes neurais convolucionais (CNNs) demandam uma quantidade considerável dados para obter desempenho satisfatório, estes dados muitas vezes podem ser de difícil obtenção ou de alto custo. Nesse contexto, as técnicas de transferência de aprendizado têm sido utilizadas com sucesso para reduzir estes dados e, muitas vezes, aumentam o desempenho final da rede. O objetivo deste trabalho é comparar diferentes técnicas de transferência de aprendizado e analisar os resultados obtidos em cada uma delas. Os resultados mostram que o retreinamento apenas das camadas finais aumenta o desempenho da rede em 2.5 vezes em comparação com o retreino completo, e 6.54 vezes em comparação com o treinamento puro.

Palavras-chave— transferência de aprendizado, aprendizado profundo, detecção de objetos

1 Introdução

As redes neurais convolucionais (CNN - *Convolutional Neural Networks*) (Lecun et al., 2001) têm sido uma maneira eficiente de classificar imagens (Liu et al., 2017). Como as CNNs podem extrair recursos de diferentes locais da imagem, elas são mais tolerantes a mudanças de escala e distorções em comparação com outros tipos de redes neurais (Leng et al., 2016). Normalmente, um classificador avançado e complexo, como uma CNN profunda, é necessário para obter uma alta acurácia. No entanto, o uso desses modelos complexos causam um aumento no custo computacional para o treinamento (Felzenszwalb et al., 2010).

A detecção de objetos refere-se à tarefa de encontrar a posição dos objetos em uma imagem ou um vídeo. Ela pode ser acoplada a um modelo de reconhecimento para identificar os objetos detectados (Van de Sande et al., 2011). Tanto a detecção como o reconhecimento são modelados para que cada tipo de objeto seja uma classe. Assim, uma imagem está associada a uma ou várias instâncias desses objetos, e existe um conjunto finito de tipos de objetos. As detecções são exibidas dentro de caixas delimitadoras, que representam a posição e tamanho do objeto, como visto na Figura 1.

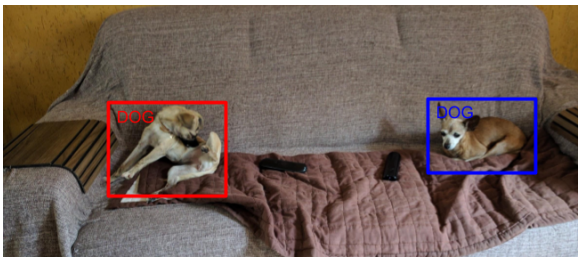

Figura 1: Caixas delimitadoras obtidas na detecção e reconhecimento de objetos.

Para treinar completamente uma CNN, é necessário uma quantidade elevada de dados e alta variabilidade de características. Dependendo da aplicação, esse tipo de dado pode ser caro ou inviável de adquirir (Cao et al., 2018). Além disso, as redes neurais são sensíveis às diferenças de iluminação, contraste ou rotação. Se as imagens uti-

lizadas para treinamento não apresentarem uma variedade suficiente dessas condições, o classificador poderá não se adaptar a esses detalhes, reduzindo o desempenho da classificação. Dados indesejados, como partes do plano de fundo ou ruído, podem fazer com que o sistema se especialize apenas nos dados de treinamento, levando a uma generalização baixa (Tay e Cao, 2001).

A variabilidade na base de dados aumenta a probabilidade da rede aprender a detectar características de baixo nível corretamente, aumentando a acurácia para novos dados. As características de baixo nível representam os atributos mais básicos da imagem, como por exemplo: bordas, círculos, linhas e cores. Essas características são extraídas pela CNN, como visto na Figura 2, e utilizadas para o aprendizado e a classificação. Uma vez que as camadas intermediárias de uma CNN são geralmente extratores de características que podem ser generalizadas para conjuntos de dados diferentes, uma solução comum para o treinamento é usar um conjunto de dados maior para pré-treinar a rede e, em seguida, retreinar a rede para reconhecer as classes da base de dados alvo (Oquab et al., 2014). Esta técnica é amplamente utilizada para alcançar uma alta acurácia ao treinar redes utilizando um conjunto de dados insuficiente. Outra utilidade do retreino é adicionar novas imagens em uma rede treinada. Nesse caso, é possível retreinar apenas as camadas de reconhecimento com a nova base de dados (incluindo o novo objeto).

Figura 2: Representação criada utilizando as características extraídas pela CNN.

O objetivo deste trabalho é a avaliação de técnicas de transferência de aprendizado aplicado à *Fast and Lightweight Object Detection Network* (FLODNet), que exige menos recursos computacionais para treinamento e execução que outras CNNs estado-da-arte (de Oliveira et al., 2018), no contexto de detecção e reconhecimento de alimentos em imagens. A FLODNet é uma arquitetura de CNN com capacidade de realizar detecção e reconhecimento de objetos rapidamente e economizando recursos computacionais, tais como espaço de armazenamento, processamento e memória.

O artigo está organizado da seguinte forma. A Seção 2 apresenta os principais trabalhos relacionados. A Seção 3 fornece uma descrição da metodologia utilizada no trabalho, do processo de treinamento e das métricas usadas para avaliar os resultados. A Seção 4 apresenta os resultados obtidos. A conclusão e os trabalhos futuros são apresentados na Seção 5.

2 Trabalhos Relacionados

O uso de uma CNN como extrator de recursos treinada em um conjunto de dados diferente é explorado em (Bui et al., 2016). No projeto foi utilizada a AlexNet (Krizhevsky et al., 2012) com treinamento feito com a base do ImageNet (Deng et al., 2009) e uma rede neural recursiva com um classificador para extrair recursos e classificar imagens. No trabalho de Wollmann et al. (2018), o pré-treinamento é utilizado para aumentar o desempenho do treinamento final da rede.

O trabalho de Haarburger et al. (2018) utiliza a transferência de aprendizado para diminuir a quantidade de imagens necessárias para treinar uma rede neural profunda, uma vez que imagens médicas são difíceis de serem obtidas e difíceis de serem classificadas manualmente para o aprendizado supervisionado. Como mostrado por Karpathy et al. (2014), técnicas de transferência de aprendizagem foram comparadas com o treinamento completo. As técnicas testadas foram: retreino apenas da camada final, retreino das 3 camadas finais, retreino de todas as camadas e o treinamento de toda a rede. O resultado mostrou que, para o conjunto de dados utilizado, o melhor resultado foi alcançado pelo retreino das 3 camadas finais da rede.

Alguns trabalhos utilizam técnicas de criação de dados sintetizados para melhorar o treinamento e aumentar a generalização. Neste processo, várias técnicas são aplicadas para gerar imagens usadas posteriormente para o treinamento. No trabalho de Masi et al. (2016), um conjunto de dados de rostos foi usado para sintetizar novas imagens com mudanças na pose, aparência e expressões faciais. Diversas combinações de técnicas foram exploradas e apresentaram ganhos significativos em comparação com o conjunto de dados original. No trabalho (Sapp et al., 2008), o fundo da imagem foi alterado. O conjunto de dados original foi obtido usando um fundo verde para que o algoritmo pudesse reconhecê-lo e fazer alterações para criar novas imagens.

Similar ao trabalho de Haarburger et al. (2018), este trabalho utiliza a transferência de aprendizado como forma de suprir a deficiência de imagens para treino. Além disso, as diversas formas de transferência são comparadas, como feito por Karpathy et al. (2014). A criação de dados sintetizados foi necessária para aumentar a vari-

abilidade de características, como no trabalho de Masi et al. (2016) e Sapp et al. (2008).

3 Metodologia

Diferentes técnicas de transferência de aprendizado foram analisadas utilizando a FLODNet em comparação com o treino completo da rede partindo de valores aleatórios. A arquitetura da FLODNet, explicada com mais detalhes em (de Oliveira et al., 2018), pode ser vista na Figura 3 e é brevemente explicada nas seções seguintes.

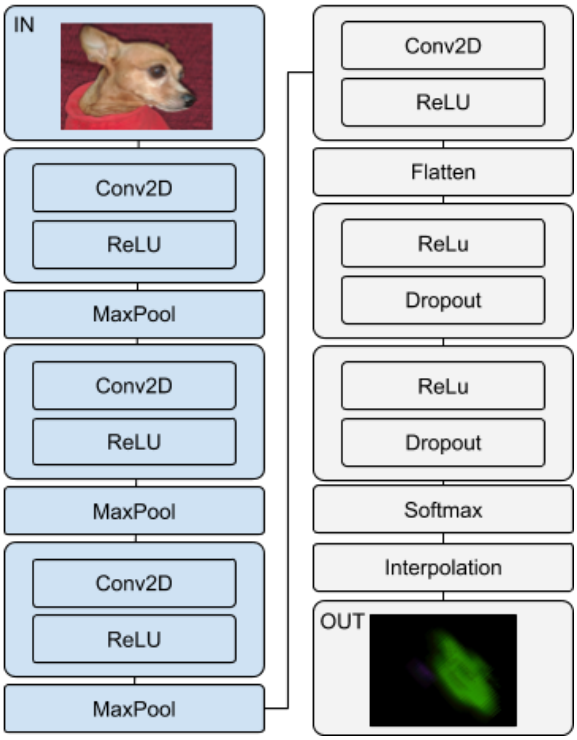

Figura 3: Arquitetura da FLODNet. A saída representa as regiões detectadas pela rede. O retreino parcial afeta apenas as camadas em cinza enquanto que o retreino completo afeta todas as camadas da rede.

3.1 Obtenção e Preparação da Base de Dados

A base de dados contém 200 imagens de pratos com uma ou mais classes de alimentos categorizados em 12 classes. Todas as imagens foram obtidas em 1936x1296 *pixels*. Posteriormente, elas tiveram seu conteúdo segmentado e classificado manualmente com o auxílio do software LabelMe (Russell et al., 2008). O conjunto de imagens foi então dividido em dois grupos, treino e teste, de modo que as imagens de teste não sejam utilizadas durante o treino.

As redes neurais são muito sensíveis às transformações na imagem, o que pode dificultar a detecção e reconhecimento de objetos em condições diferentes daquelas para as quais foi treinada. As técnicas utilizadas para criar dados sintéticos neste trabalho foram: alterar o fundo das imagens, adicionar ruído aleatório, rotacionar objetos, redimensionar a imagem, alterando a proporção da imagem, recortar partes aleatórias, transformar a textura interna, alterar o brilho e o contraste. Em geral, treinar a rede com um conjunto de dados que já contém essas transformações permite à rede generalizar melhor os recursos de cada classe e reconhecê-los nessas condições. Com essas técnicas foi possível obter 20.000 novas imagens para treino e teste. As imagens foram pré-processadas por um *script* que prepara o conjunto de dados para o treino. Um arquivo *Comma-Separated Values* (CSV) é criado mapeando os objetos que podem ser encontrados em cada região das imagens, para que a rede possa ser treinada para classificar essas regiões.

3.2 Pré-treino

O conjunto de dados *Dogs vs. Cats*, que contém 25.000 imagens rotuladas como cão ou gato, do Kaggle (*Dogs vs. Cats | Kaggle*, n.d.), foi usado para pré-treinar a FLODNet. A quantidade de dados, a variabilidade de cores e formas nesse conjunto de dados ajuda a generalizar características de baixo nível nos filtros convolucionais presentes nas camadas iniciais da rede.

3.3 Técnicas de Treino

Levando em consideração a hierarquia das camadas da FLODNet, as seguintes formas de treino foram escolhidas para comparação:

- Treino Completo (Sem Retreino) – O treinamento é feito de forma padrão, ou seja, todas as camadas da FLODNet são treinadas na base de dados final (alimentos), contendo 20.000 imagens, partindo de pesos iniciais aleatórios e o resultado será avaliado;
- Retreino Completo – Após o treino na base de dados de pré-treino (Dogs vs. Cats) a FLODNet será completamente retreinada na base de dados final (alimentos), e o resultado será avaliado;
- Retreino Parcial I – Após o treino na base de dados de pré-treino (Dogs vs. Cats), apenas as camadas destacadas em cinza na Figura 3 são retreinadas para a base de dados final (alimentos), limitado ao tempo de execução do treino completo, seguindo-se a avaliação dos resultados;
- Retreino Parcial II – Similar ao item anterior, contudo, o tempo limite de execução será o mesmo do retreino completo, seguindo-se a avaliação dos resultados;

- Retreino Parcial III – Similar aos dois itens anteriores, sem interrupção forçada por tempo, seguindo-se a avaliação dos resultados;

Tabela 1: Técnicas de treino.

	Pré-treino	**Camadas Retreino**	**Tempo Limite**
Treino Completo	Não	-	Não
Retreino Completo	Sim	Todas	Não
Retreino Parcial I	Sim	Em cinza na Figura 3	Tempo do Treino Completo
Retreino Parcial II	Sim	Em cinza na Figura 3	Tempo do Retreino Completo
Retreino Parcial III	Sim	Em cinza na Figura 3	Não

Os retreinos parciais I e II são importantes para comparação com os resultados obtidos no treino completo e no retreino completo, como visto na Tabela 1, considerando que o retreino parcial III, o treino completo e o retreino completo finalizaram o treino com tempos de execução diferentes.

3.4 Métricas

Para medir a exatidão das detecções é utilizada a razão entre a interseção e a união das regiões delimitadas pelas caixas (IoU), como descrito na Equação 2.

$$\text{IoU(A,B)} = \frac{|A \cap B|}{|A \cup B|}, \quad \text{sendo } 0 \leq \text{IoU(A,B)} \leq 1 \tag{1}$$

Como mostrado em (Krähenbühl e Koltun, 2014), o IoU representa com exatidão a similaridade entre as regiões de *pixels*. Os objetos detectados são considerados corretos se coincidirem com a classe esperada e se suas caixas de delimitação se sobrepuserem em pelo menos 50% em relação à detecção original, chamada de *ground truth*.

Para calcular o IoU geral, denominado de IoU_g, a média dos IoUs obtidos em cada detecção de um mesmo objeto, em toda a base de dados, é calculada como apresentado na equação 2.

$$\text{IoU}_\text{g} = \frac{\sum_{b=0}^{\text{b-1}} \text{IoU(bbox[b],ground truth[b])}}{\text{b}} \tag{2}$$

Nesse caso, a variável *b* representa uma única detecção, que é comparada ao *ground truth*. Essa métrica é útil para medir a exatidão de detecção das caixas de delimitação propostas.

3.5 Equipamento

Todas as fases de treinamento foram realizadas em um laptop com CPU Intel Core i7-6700HQ, GeForce GTX 970m 3 GB, SSD de 500GB e Arch Linux. Os testes foram executados em um laptop com CPU Intel Core i7-3610QM, disco rígido de 500 GB 5400rpm e Arch Linux.

4 Resultados

A Tabela 2 mostra os valores mínimo, máximo e mediana do IoU_g, dos tempos de treino e retreino, bem como do tempo efetivo, em três execuções de cada uma das diferentes técnicas de retreino propostas. Uma vez que um dos objetivos seja conseguir que a rede trabalhe com uma nova base de dados sem que seja necessário refazer o retreino, o tempo importante a ser considerado é o tempo efetivo, que é o tempo necessário para a rede obter conhecimento necessário para trabalhar com uma nova base. Esse tempo considera apenas o tempo necessário para que a rede seja executada em uma nova base de dados. A mediana será utilizada em todas as análises e comparações a seguir.

Como mostrado na Tabela 2, o melhor resultado foi obtido pelo Retreino Parcial III, similar ao obtido no trabalho de Karpathy et al. (2014). A mediana do IoU_g no Treino Completo foi igual a $0,081$, enquanto que no Retreino Completo e no Retreino Parcial III foi igual a $0,217$ e $0,530$, respectivamente.

O Treino Completo (sem retreino) faz com que a rede se torne altamente especialista na base de dados utilizada para o treino em poucas iterações. Nesse caso, como a rede alcança acurácia total na base de treino, o processo é finalizado por não alteração nos valores. Contudo, o aprendizado altamente especialista na base de treino faz com que a rede não generalize o conhecimento aprendido para novas imagens, como as imagens utilizadas no teste. Isso caracteriza um treinamento ineficiente da rede. O valor de IoU_g obtido reflete a incapacidade da rede de reconhecer os alimentos quando treinada dessa forma.

Embora o Retreino Completo permita o aprendizado de características não existentes na base de dados através do pré-treino, essas características são sobrescritas durante o retreino. Esse fato, conhecido como esquecimento catastrófico (French, 1999), diminui a capacidade de generalização da rede, causando um efeito parecido com o descrito no parágrafo anterior.

O Retreino Parcial permite que a CNN mantenha as características aprendidas e retreine somente as camadas utilizadas como classificadores. Desta forma, o esquecimento catastrófico é redu-

Tabela 2: Comparação entre diversas técnicas de treino.

		IoU_g	Tempo Tre.	Tempo Ret.	Tempo Efetivo
Treino Completo	Min	0,035	21min	-	21min
	Med	0,081	40min	-	40min
	Max	0,090	46min	-	46min
Retreino Completo	Min	0,211	618min	92min	92min
	Med	0,217	632min	94min	94min
	Max	0,221	641min	94min	94min
Retreino Parcial I	Min	0,145	618min	40min*	40min*
	Med	0,192	632min	40min*	40min*
	Max	0,194	641min	40min*	40min*
Retreino Parcial II	Min	0,281	618min	94min**	94min**
	Med	0,465	632min	94min**	94min**
	Max	0,487	641min	94min**	94min**
Retreino Parcial III	Min	0,524	618min	218min	218min
	Med	0,530	632min	219min	219min
	Max	0,561	641min	235min	235min

* Retreino terminado prematuramente para comparação com Treino Completo (sem retreino).
** Retreino terminado prematuramente para comparação com Retreino Completo.

zido, o que aumenta a capacidade de generalização. O IoU_g reflete a qualidade do treino obtido utilizando essa técnica. A interrupção prematura do Retreino Parcial, como ocorrido no Retreino Parcial I e II, causou uma perda de desempenho, refletida no IoU_g, menor em comparação com o Retreino Parcial III.

A mediana do IoU_g no Retreino Parcial I aponta um desempenho $2,3$ vezes superior ao Treino Completo, utilizando o mesmo tempo efetivo. O menor valor obtido para o IoU_g ainda é $1,55$ vezes maior que o maior valor obtido pelo Treino Completo. Mesmo interrompendo o retreino nos ciclos iniciais do treinamento, ele foi superior ao Treino Completo.

O Retreino Parcial II obteve um desempenho $2,1$ vezes superior ao Retreino Completo. Similar ao analisado no parágrafo anterior, o Retreino Parcial II também obteve como mínimo um valor superior ao máximo do Retreino Completo.

O Retreino Parcial III proporcionou um ganho de $6,54$ vezes em relação ao Treino Completo e 2.52 vezes o Retreino Completo, repetindo o padrão a valores mínimos e máximos em comparação com o Retreino Parcial II. Contudo, essa técnica obteve um desempenho apenas $1,15$ superior levando $2,32$ vezes o tempo do Retreino Parcial II. Em comparação, o Retreino Parcial I, que levou $2,35$ vezes menos tempo que o Retreino Parcial II obteve um ganho de $2,42$. Isso mostra que, como esperado, o retreino perde eficiência a medida que cresce o número de iterações e pode ser interrompido prematuramente para reduzir custos causando baixas perdas em desempenho.

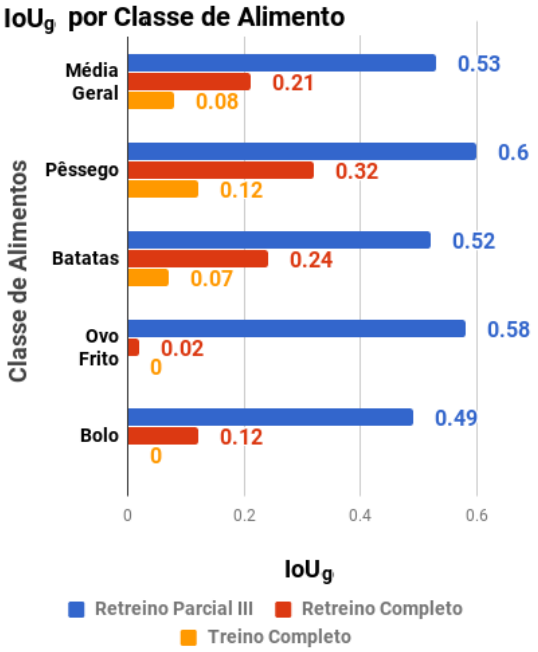

Figura 4: Valores de IoU para alimentos utilizando o Retreino Parcial III e Treino Completo.

O gráfico da Figura 4 mostra como o Retreino Parcial III afeta o reconhecimento para diferentes alimentos em relação ao Retreino Completo e ao Treino Completo. Alimentos com menor variação de formas e texturas, como o pêssego e as batatas, sofreram menos com a falta de generalização do Treino Completo e do Retreino Completo que alimentos como ovo frito e bolo. Por terem pouca variação, o pêssego e as batatas têm maior chance de serem detectados, uma vez que a CNN aprendeu características muito similares durante o treino. Esses alimentos mostram que, de fato, o valor baixo de IoU_g é devido à falta de generalização.

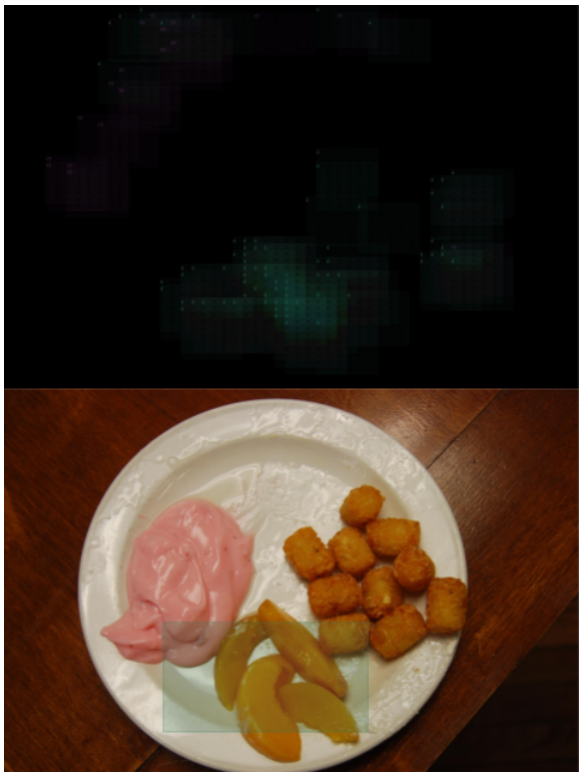

Figura 5: Mapa de detecções com os valores de confiança representados pela opacidade (acima) e imagem de saída (abaixo) da FLODNet completamente treinada, sem retreino.

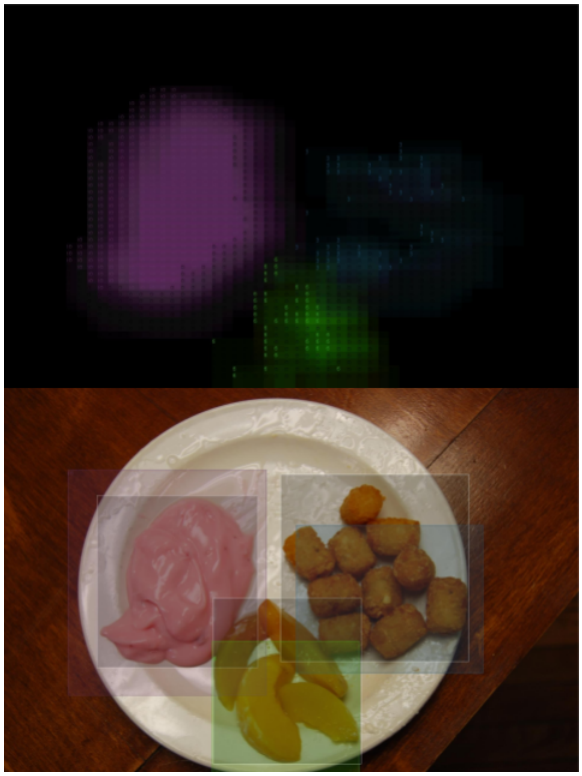

Figura 6: Mapa de detecções com os valores de confiança representados pela opacidade (acima) e imagem de saída (abaixo) da FLODNet parcialmente retreinada utilizando o Retreino Parcial III.

No pêssego, foi observada uma perda de desempenho de $1,87$ vezes no Retreino Completo em comparação com o Retreino Parcial III, enquanto que as batatas perderam em $2,16$ vezes. Essa diferença pode ser associada aos efeitos do esquecimento catastrófico causado pelo retreino de todas as camadas que fez com que a rede generalizasse menos as características da superfície frita.

As Figuras 5 e 6 mostram o mapa de detecções e a imagem de saída produzida pela FLODNet. Comparando os mapas de detecção é possível perceber que sem o Retreino Parcial, as detecções foram mais fracas ou, em alguns casos, inexistente. O mapa produzido pela rede parcialmente retreinada é mais bem definido e com áreas mais opacas, indicando maior confiança na detecção. Além disso, as detecções mostradas na figura de saída são mais exatas com o tamanho e posicionamento real do alimento.

5 Conclusão e Trabalhos Futuros

Como mostrado nos resultados, o Retreino Parcial da rede permite uma melhor generalização do conhecimento obtido durante o treino, aumentando o desempenho da rede. Treinar todas as camadas da rede diretamente na base de dados alvo, sem o passo de pré-treino, produziu resultados inconsistentes e baixo desempenho final. A quantidade e a variabilidade de amostras não foram suficientes para treinar a rede neural, o que demonstra que as técnicas de transferência de aprendizado causam um ganho significativo. O Retreino Parcial sem interrupção prematura obteve um desempenho 6.62 vezes maior nas imagens de teste, em comparação com o treino sem o passo de pré-treino.

O Retreino Completo permitiu à rede aprender características específicas da base de dados de retreino, fazendo com que ocorresse o esquecimento catastrófico das características mais generalistas aprendidas durante o pré-treino. Dessa forma, ocorreu um problema similar ao visto no Treino Completo sem o passo de pré-treino, a rede perdeu a capacidade de generalização. Contudo, o resultado obtido foi 2.81 vezes superior ao treino sem o passo de pré-treino.

No trabalho foram analisadas diferentes formas de treino utilizando a FLODNet para detectar e reconhecer alimentos em imagens, mostrando as vantagens e desvantagens de cada método. Os resultados obtidos permitiram uma análise quantitativa dos efeitos de uma má generalização dos dados e do esquecimento catastrófico no treino de uma CNN. Por fim, foi analisado o efeito em diferentes classes de objetos e o impacto no mapa de detecções.

Como trabalho futuro é importante analisar como as características da arquitetura da CNN podem melhorar ou prejudicar as técnicas de retreino. Outro ponto importante é investigar o

melhor momento para a parada prematura do retreino e formas de evitar o esquecimento catastrófico ao ser treinada para uma nova base de dados.

Agradecimentos

Os autores deste trabalho agradecem à CAPES, CNPq e FAPEMIG pela contribuição financeira com relação a esta pesquisa.

Referências

Bui, H. M., Lech, M., Cheng, E., Neville, K. e Burnett, I. S. (2016). Object recognition using deep convolutional features transformed by a recursive network structure, *IEEE Access* **4**: 10059–10066.

Cao, H., Bernard, S., Heutte, L. e Sabourin, R. (2018). Improve the performance of transfer learning without fine-tuning using dissimilarity-based multi-view learning for breast cancer histology images, *International Conference Image Analysis and Recognition*, Springer, pp. 779–787.

de Oliveira, B. A. G., Ferreira, F. M. F. e d. S. Martins, C. A. P. (2018). Fast and lightweight object detection network: Detection and recognition on resource constrained devices, *IEEE Access* **6**: 8714–8724 – DOI: 10.1109/ACCESS.2018.2801813.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. e Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database, *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, IEEE, pp. 248–255.

Dogs vs. Cats | Kaggle (n.d.). `https://www.kaggle.com/c/dogs-vs-cats/data`. (Accessed on 06/02/2017).

Felzenszwalb, P. F., Girshick, R. B. e McAllester, D. (2010). Cascade object detection with deformable part models, *Computer vision and pattern recognition (CVPR), 2010 IEEE conference on*, IEEE, pp. 2241–2248.

French, R. M. (1999). Catastrophic forgetting in connectionist networks, *Trends in Cognitive Sciences* **3**(4): 128–135.

Haarburger, C., Langenberg, P., Truhn, D., Schneider, H., Thüring, J., Schrading, S., Kuhl, C. K. e Merhof, D. (2018). Transfer learning for breast cancer malignancy classification based on dynamic contrast-enhanced mr images, *in* A. Maier, T. M. Deserno, H. Handels, K. H. Maier-Hein, C. Palm e T. Tolxdorff (eds), *Bildverarbeitung für die Medizin 2018*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 216–221.

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R. e Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks, *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 1725–1732.

Krähenbühl, P. e Koltun, V. (2014). Geodesic object proposals, *European Conference on Computer Vision*, Springer, pp. 725–739.

Krizhevsky, A., Sutskever, I. e Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks, *Advances in neural information processing systems*, pp. 1097–1105.

Lecun, Y., Bottou, L., Bengio, Y. e Haffner, P. (2001). *Gradient-based learning applied to document recognition*, IEEE Press, pp. 306–351.

Leng, J., Li, T., Bai, G., Dong, Q. e Dong, H. (2016). Cube-cnn-svm: A novel hyperspectral image classification method, *2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 1027–1034.

Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y. e Alsaadi, F. E. (2017). A survey of deep neural network architectures and their applications, *Neurocomputing* **234**: 11–26.

Masi, I., Tran, A. T., Leksut, J. T., Hassner, T. e Medioni, G. (2016). Do we really need to collect millions of faces for effective face recognition?, *arXiv preprint arXiv:1603.07057* .

Oquab, M., Bottou, L., Laptev, I. e Sivic, J. (2014). Learning and transferring mid-level image representations using convolutional neural networks, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1717–1724.

Russell, B. C., Torralba, A., Murphy, K. P. e Freeman, W. T. (2008). Labelme: a database and web-based tool for image annotation, *International journal of computer vision* **77**(1-3): 157–173.

Sapp, B., Saxena, A. e Ng, A. Y. (2008). A fast data collection and augmentation procedure for object recognition., *AAAI*, Chicago, IL, pp. 1402–1408.

Tay, F. E. e Cao, L. (2001). Application of support vector machines in financial time series forecasting, *Omega* **29**(4): 309–317.

Van de Sande, K. E., Uijlings, J. R., Gevers, T. e Smeulders, A. W. (2011). Segmentation as

selective search for object recognition, *Computer Vision (ICCV), 2011 IEEE International Conference on*, IEEE, pp. 1879–1886.

Wollmann, T., Ivanova, J., Gunkel, M., Chung, I., Erfle, H., Rippe, K. e Rohr, K. (2018). Multichannel Deep Transfer Learning for Nuclei Segmentation in Glioblastoma Cell Tissue Images, Springer Vieweg, Berlin, Heidelberg, pp. 316–321.