IDENTIFICAÇÃO DE MASSAS EM ULTRASSONOGRAFIAS DE FÍGADO UTILIZANDO REDES NEURAIS CONVOLUCIONAIS

Liziane Zerbin Gallert [1]

Sandro da Silva Camargo [2]

Resumo:

Avanços na análise computadorizada de imagens médicas propiciaram o surgimento de métodos e técnicas que auxiliam no diagnóstico precoce de uma série de doenças. Atualmente esses diagnósticos se dão por meio de inspeção visual por parte de especialistas humanos. Este artigo descreve o desenvolvimento de um modelo computacional - baseado em redes neurais convolucionais (CNN) - que objetiva auxiliar o médico radiologista na confirmação de um diagnóstico, por meio da identificação de massas em achados de imagens ultrassonográficas de fígado. Em geral, processo de treinamento de uma rede neural exige um alto poder de processamento para dados densos. Quando se trata de processamento de imagens, esta questão é ainda mais realçada, visto que o número de entradas da rede é proporcional ao número de pixels da imagem de interesse, justificando o uso de processamento paralelo. É importante ressaltar que a classe de imagens médicas não pode permitir armazenamento com perdas de dados e exige precisão em sua identificação. O modelo de CNN proposto foi capaz de realizar a identificação dessas massas com uma acurácia de 96,67%, para o conjunto de imagens de teste analisadas.

Palavras-chave: Rede neural convolucional. GPU. Imagem de ultrassom.

Modalidade de Participação: Iniciação Científica

IDENTIFICAÇÃO DE MASSAS EM ULTRASSONOGRAFIAS DE FÍGADO UTILIZANDO REDES NEURAIS CONVOLUCIONAIS

[1] Aluno de graduação. lizianegallert@gmail.com. Autor principal

[2] Docente. sandrocamargo@unipampa.edu.br. Orientador

IDENTIFICAÇÃO DE MASSAS EM ULTRASSONOGRAFIAS DE FÍGADO UTILIZANDO REDES NEURAIS CONVOLUCIONAIS

1. INTRODUÇÃO

A ultrassonografia é um exame de diagnóstico muito utilizado na medicina moderna que tem como objetivo visualizar, em tempo real, qualquer órgão ou tecido do corpo. As imagens ultrassonográficas coletadas durante exames necessitam de tratamento, extração dos dados relevantes e classificação, a fim de gerar informações úteis para contribuir com o diagnóstico. Atualmente esses diagnósticos se dão por meio de inspeção visual por parte de especialistas humanos. Com o rápido crescimento do volume de imagens produzidas, sua análise minuciosa por esses especialistas torna-se difícil e cansativa, aumentando a probabilidade deste processo ser suscetível a erro.

Dessa forma, ocorre a demanda por métodos que possibilitem o processamento e análise da imagem e a extração de características, não somente de maneira mais eficiente, mas que também mantenha um alto grau de confiabilidade, simulando ou atuando como auxiliar no trabalho dos especialistas humanos. Técnicas ou ferramentas de inteligência artificial surgem como opções na automatização desses métodos, em especial as Redes Neurais Artificiais (RNA). Estudos demonstram o grande potencial das RNA em sistemas de apoio ao diagnóstico, principalmente nas aplicações de classificação de padrões utilizando imagens (MENESES et al., 2008; MARCOMINI, 2013; AMBRÓSIO, 2002).

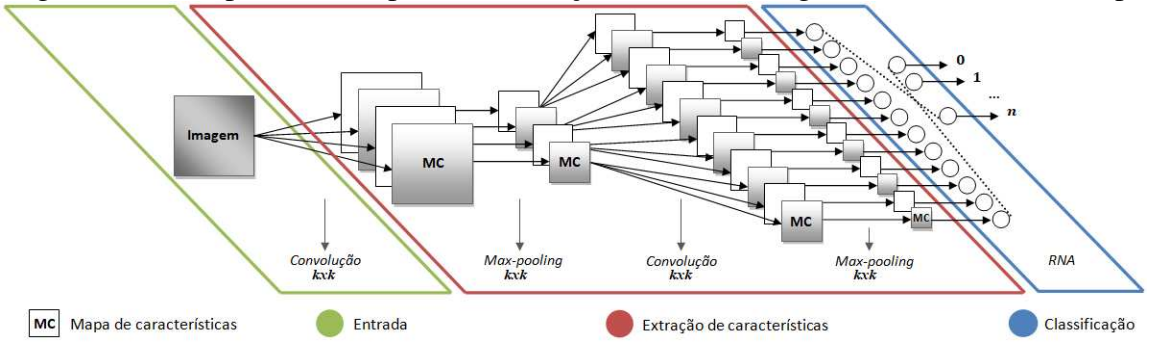
Dentre os diferentes tipos de RNA, as CNN (*Convolutional Neural Network*), base deste estudo, vêm sendo aplicadas com sucesso no processamento e análise de imagens digitais. Ela aprimora a generalização das RNA, possuindo a habilidade de lidar com deformações e variações nos dados de entrada devido à combinação da sua arquitetura das idéias dos campos receptivos, compartilhamento de pesos e subamostragem temporal ou espacial (LECUN; BENGIO, 2003).

Neste contexto, objetiva-se o desenvolvimento de um modelo computacional - baseado em CNN – para auxiliar o médico radiologista na confirmação de um diagnóstico, por meio da identificação de massas em achados de imagens ultrassonográficas de fígado.

2. METODOLOGIA

É usual que as imagens de entrada utilizadas pela CNN passem por um processo de extração de características, constituídas por pelo menos uma camada de convolução seguida ou não por uma camada de *pooling*, também conhecida como redução ou subamostragem. A Figura 1 explicita as camadas básicas de uma CNN para classificação típica.

Figura 1 – Exemplo de CNN para classificação com dois estágios e suas diferentes etapas.



Fonte: Adaptado de LECUN; BENGIO, 2003.

O propósito principal da etapa convolucional em uma CNN é detectar características ou recursos visuais nas imagens (TORRES, 2016). Essa extração de características se dá na camada oculta do processo de convolução. Primeiramente, cada *pixel* de uma imagem de interesse é atribuído a um neurônio na entrada da camada de extração de características. Na CNN os neurônios de entrada da camada de extração de características não são completamente conectados aos neurônios da camada oculta; a conexão se dá de maneira local e com dependência espacial na distribuição dos *pixels* (TORRES, 2016). O tamanho das janelas convolutivas podem variar de camada para camada e possuem formato quadrado de $altura\ x\ largura$, ou seja, kxk.

A camada de *pooling* condensa a saída da camada convolucional - através de conexões locais - e cria uma versão compactada da informação, que servirá de entrada para a próxima camada de convolução (TORRES, 2016). Esse processo representa o campo receptivo. Conforme Torres (2016) existem vários métodos para comprimir informação, porém o mais habitual é conhecido como *max-pooling*, que consiste em reduzir as informações apenas mantendo o valor máximo na região kxk considerada.

A etapa de classificação diz respeito ao uso de uma rede neural no reconhecimento e interpretação dos dados extraídos da imagem de entrada, cuja saída corresponde à quantidade de respostas possíveis após o processo de análise. Neste contexto, as CNN são capazes de prever ou extrapolar dos padrões identificados, para dados desconhecidos, viabilizando a análise de grandes quantidades de imagens e a extração de conhecimento das mesmas, produzindo saídas consistentes para entradas não apresentadas anteriormente. Como ponto negativo, o processo de treinamento demanda um tempo longo por ser computacionalmente custoso.

Sendo assim, optou-se por comparar o desempenho, em termos de tempo de execução, de uma CPU (*Central Processing Unit*) e uma GPU (*Graphics Processing Unit*) do modelo de CNN proposto. Para tal levou-se em conta o tempo computacional para executar o treinamento da rede e a análise das imagens durante o processo de teste.

Por fim foram realizados testes simuladores, com consequente verificação da acurácia dos diagnósticos em imagens ultrassonográficas de fígado, visando identificar a presença de massas nas mesmas. Foram então avaliados os tempos de processamento total para a realização das análises propostas e o percentual de acertos nos diagnósticos.

No que diz respeito ao ambiente de desenvolvimento, optou-se pela utilização da interface *Anaconda Navigator*, uma GUI (*Graphical User Interface*) que permite o desenvolvimento de aplicações e o gerenciamento de pacotes (ANACONDA, 2017). Integrado a GUI, tem-se o *framework* CNTK (*Microsoft Cognitive Toolkit*), conhecido por implementar o treinamento de CNN com grande eficiência para dados de fala, imagem e texto segundo dados da NVIDIA (2017).

Para o desenvolvimento do código fonte, utilizou-se a ferramenta *Spyder* (*Scientific Python Development Environment*), disponível na interface *Anaconda Navigator*. O *Spyder* é um IDE (*Integrated Development Environment*) especializado na linguagem de programação *Python*, que permite edição, teste e depuração de código fonte (SPYDER, 2017). A Tabela 1 explicita o restante das decisões de projeto que envolvem *software* e *hardware* utilizadas no decorrer da execução da metodologia.

Tabela 1 - Especificações técnicas do *software* e *hardware* utilizados.

Tipo	***Material***	**Especificação técnica**
Software	Sistema operacional	*Windows* 10 PRO (64 *bit*)
	Linguagem de programação	*Python* (versão 3.5.2)
	Framework	CNTK 2.5.1 para GPU
	GUI	Anaconda3 (versão 4.1.1)

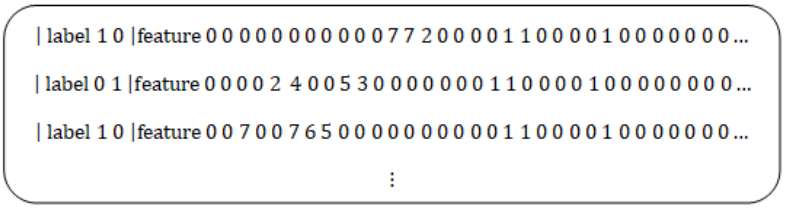
	IDE	*Spyder* (integrado ao Anaconda)
Hardware (CPU)	Processador	Intel (R) Core (TM) i5 – 7400 CPU @3.00 GHz
	Memória RAM	16 GB
Hardware (GPU)	Modelo da GPU	GeForce GTX 1050 Ti
	Cuda cores	768
	Arquitetura da GPU	Pascal
	Memória dedicada	4 G

Fonte: do autor, 2018.

A aquisição das imagens ultrassonográficas de fígados utilizadas deu-se através da plataforma Flickr, (FLICKR, 2017), onde encontram-se disponíveis para *download* nos álbuns de uma clínica especializada em ultrassonografia computadorizada, constituindo o banco de dados com 120 imagens. Na etapa de pré-processamento, com o auxílio do editor gráfico *open source* GIMP (GNU *Image Manipulation Program*), as imagens obtidas tiveram seu canal de cores alterado de RGB (*Red Green Blue*) para monocromático e removeu-se logos e rótulos que pudessem afetar os resultados. Ainda com auxílio da ferramenta GIMP, deu-se o processo de segmentação, onde o tamanho em *pixels* das imagens foi redefinido de forma que todas as imagens possuíssem o mesmo tamanho ($450x450$).

Para a representação e descrição - também conhecida como extração de características - desenvolveu-se um código em *Python*. O código visa extrair os valores de *pixel* das imagens resultantes do processo de segmentação, caracterizando com precisão as formas presentes nelas. Como essas imagens possuem um tamanho correspondente a $450x450$ *pixels*, um vetor com 202500 posições é extraído para cada imagem do banco de dados e armazenado em um arquivo de texto. A saída desta etapa é apresentada na Figura 2, onde *label* e *features* correspondem à saída esperada pela CNN (0 1 possui massa e 1 0 não possui massa) e o vetor de *pixels* extraído, respectivamente. Assim são gerados os arquivos de treino e de teste da rede neural.

Figura 2 – Imagem representada por seus valores de *pixel* e seu respectivo *label*.



Fonte: do autor, 2018.

Na etapa de reconhecimento e interpretação, utilizou-se uma CNN, implementada em linguagem de programação *Python* com auxílio do *framework* CNTK. O modelo desenvolvido pode ser executado tanto em um CPU quanto em uma GPU sem prejuízo a identificação das massas nas imagens ultrassonográficas de fígado.

Os exemplos utilizados para treinar a rede neural consistiram tanto de exemplos positivos quanto de exemplos negativos, buscando atenuar o problema de reconhecimentos falsos ocasionais. Exemplos positivos são relativos aos dados de entrada que contém o alvo de interesse (massas) e negativos os que não contém.

As 120 imagens disponíveis no banco de dados foram separadas em dois grupos iguais (60 imagens cada), um utilizado para a criação do arquivo de teste e o outro para a criação do arquivo de treino. Das 60 imagens disponíveis no arquivo de treino, 30 são de exemplos positivos e 30 de exemplos negativos, gravados de forma intercalada. De forma

análoga, o arquivo de teste possui 60 imagens, sendo 30 de exemplos positivos e 30 de exemplos negativos.

3. RESULTADOS E DISCUSSÃO

Para o processo de treino e teste foram definidas uma janela convolutiva de $5x5$ e uma janela de *max-pooling* de $2x2$. Testou-se então o efeito da aplicação de uma camada de convolução seguida de *max-pooling*, na capacidade de predição da rede neural, como pode ser observado na Tabela 2.

A quantidade de camadas aplicadas ao vetor de *features* criado no processamento das imagens foi sendo aumentada até que a rede atingiu um erro de 3,33%, ou seja, o modelo de CNN proposto foi capaz de realizar a identificação de massas em imagens ultrassonográficas de fígado com uma acurácia e 96,67% para o conjunto de imagens de teste.

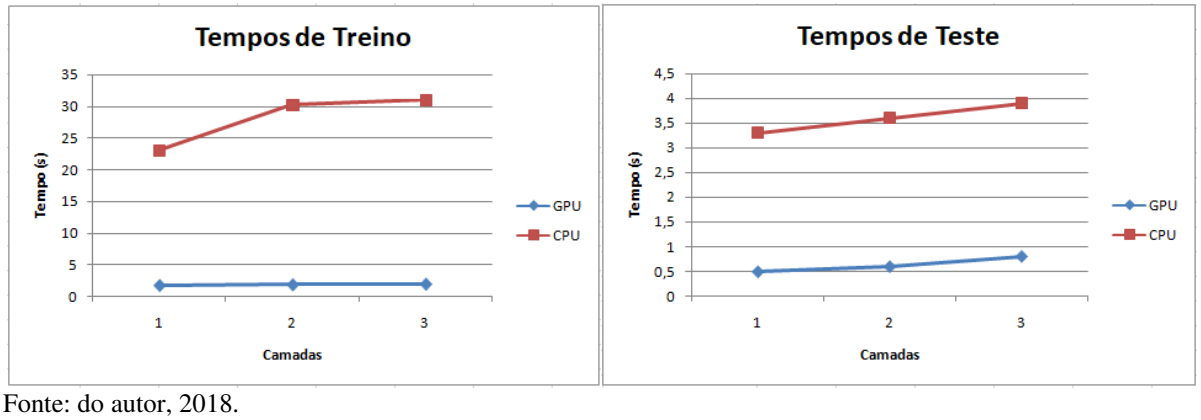
Tabela 2 – Tempos de treinamento e teste obtidos na CPU e na GPU.

Hardware	Camada de Convolução + *Max-pooling*	Tempo de treinamento (s)	Tempo de teste (s)	Erro (%)
CPU	1	23,1	3,3	50
	2	30,3	3,6	23,33
	3	31	3,9	3,33
GPU	1	1,7	0,5	50
	2	1,8	0,6	23,33
	3	1,9	0,8	3,33

Fonte: do autor, 2018.

Também é possível observar na Tabela 2 que o processo de treino e teste, para cada quantidade de camadas, foi realizado tanto utilizando a GPU quanto a CPU. É possível estabelecer então um paralelo entre os tempos de execução de cada uma como ilustra a Figura 3.

Figura 3 – Comparação entre CPU e GPU para o processo de teste e de treino da rede neural.



Fonte: do autor, 2018.

Para ambos os casos de treinamento e teste (Figura 3) a GPU obteve um desempenho, em termos de tempo de execução, superior a CPU, mesmo com um banco de dados relativamente pequeno (*60* imagens com um formato de *450x450 pixels*).

4. CONSIDERAÇÕES FINAIS

Ressalta-se que o modelo de CNN ainda está em processo de desenvolvimento como parte de um trabalho de conclusão de curso, porém foi capaz de realizar a identificação das massas em imagens ultrassonográficas de fígado com uma precisão de 96,67%, para o conjunto de imagens analisados.

Kohavi (1995) define validação cruzada como uma técnica para avaliar a capacidade de generalização de um modelo, a partir de um conjunto de dados. Pretende-se dar confiabilidade ao resultado obtido, avaliando a capacidade de generalização do protótipo através da alteração do método de treino e teste para a aplicação da técnica estatística de avaliação cruzada *k-fold*.

É possível então expandir esse modelo de CNN para que, além de identificar as massas, consiga classificá-las. Para que a classificação ocorra, deve-se expandir o número de imagens no banco de dados e atribuir um *label* único a cada tipo de massa. Deve-se manter também um conjunto de imagens de fígados saudáveis com *label* próprio, objetivando evitar falsos reconhecimentos. Esse aumento no banco de dados levará a um aumento no tempo de treinamento e de teste da CNN, tornando a utilização de um ambiente de alta capacidade de processamento indispensável. Torna-se viável então o desenvolvimento de uma interface gráfica que permita ao usuário submeter ultrassonografias de fígado para análise, obtendo-se uma ferramenta de auxílio diagnóstico, que pode ser utilizada para fins educacionais, colaborando com o treinamento e qualificação de estudantes da área médica.

REFERÊNCIAS

AMBRÓSIO, P. E. Redes neurais artificiais no apoio ao diagnóstico diferencial de lesões intersticiais pulmonares. Universidade de São Paulo, São Paulo, BR, junho 2002.

ANACONDA. **Documentation: What is Anaconda Navigator?**. 2017. [Online; acessado em 20-Maio-2017]. Disponível em: <https://docs.continuum.io/anaconda/navigator/>.

FLICKR. **Sobre o Flickr**. 2017. [Online; acessado em 05-Junho-2017]. Disponível em: <https://www.flickr.com/about>.

GONZALEZ, R. C.; WOODS, R. E. **Processamento Digital de Imagens**. São Paulo, BR: Pearson, 2010. 624 p.

KOHAVI, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: ARBIB, M. A. (Ed.). **International joint Conference on artificial intelligence**. [S.l.: s.n.], 1995. v. 14, p. 1137–1145.

LECUN, Y.; BENGIO, Y. Convolutional networks for images, speech and neural networks. In: ARBIB, M. A. (Ed.). **The Handbook of Brain Theory and Neural Networks**. Cambridge, USA: M.I.T. Press, 2003. v. 2, p. 276–279.

MARCOMINI, K. D. Aplicação de modelos de redes neurais artificiais na segmentação de nódulos em imagens de ultrassonografia de mama. Universidade de São Paulo, São Paulo, BR, março 2013.

MENESES, A. A. de M. et al. Análise de imagens médicas através de sistemas computacionais inteligentes para apoio ao diagnóstico clínico. Outubro-dezembro 2008.

NVIDIA. **Deep Learning Frameworks**. 2017. [Online; acessado em 11-Maio-2017]. Disponível em: <https://developer.nvidia.com/deep-learning-frameworks>.

SPYDER. **Spyder - Documentation**. 2017. [Online; acessado em 05-Junho-2017]. Disponível em: <https://pythonhosted.org/spyder/>.

TORRES, J. **Hello World en Tensor Flow - para iniciarse en la programación Del Deep Learning**. Barcelona, ES: Undertile - BSC and UPC Barcelona Tech, 2016. 150 p.