Análise de Desempenho e Efetividade de Redes Neurais Convolucionais em Plataformas de GPU e CPU Aplicadas ao Reconhecimento de Emoções Através de Expressões Faciais em Seres Humanos *

Leandro P. Heck[1], **Cristiano A. Künas**[1], **Edson L. Padoin**[1]

[1]Universidade Regional do Noroeste do Estado do Rio Grande do Sul (UNIJUI)
Santa Rosa – RS – Brasil

{leandro.h, cristiano.kunas}@sou.unijui.edu.br, padoin@unijui.edu.br

Abstract. *Considering the growing interest in the field of human-computer interaction and that this iteration has become something more and more natural and social, together with the increase in the computational capacity provided by GPUs and CPUs, areas such as emotion recognition have been shown to be of great importance and relevance for the scientific community. However, even with several works done, detecting and recognizing emotions computationally and with the same ease that humans recognize is still a relevant problem to be explored. To this end, seeking to explore this theme, this work adopted the use of Convolutional Artificial Neural Networks (ANN) in the recognition of emotions in humans from facial expressions. The results showed that, with the training of an ANN in GPUs, it was possible to reduce the computational time by up to 89% and increase the accuracy to 65%.*

Resumo. *Considerando o crescente interesse no campo da interação humano-computador e que essa iteração vem se tornando algo cada vez mais natural e social, juntamente com o aumento da capacidade computacional proporcionada por GPUs e CPUs, áreas como reconhecimento de emoções tem se mostrado ser de grande interesse e relevância pela comunidade cientifica. Porém, mesmo com diversos trabalhos realizados, detectar e reconhecer emoções computacionalmente e com a mesma facilidade que humanos reconhecem ainda é um problema relevante a ser explorado. Para tal, buscando explorar esse tema, este trabalho adotou a utilização de Redes Neurais Artificiais (RNA) Convolucionais na realização do reconhecimento das emoções em humanos a partir de expressões faciais. Os resultados demonstraram que, com o treinamento de uma RNA em GPUs, foi possível reduzir o tempo computacional em até 89% e aumentar a acurácia para 65%.*

1. Introdução

Na atualidade há um crescente interesse na melhoria da interação entre humanos e computadores. Para que uma efetiva interface humano-computador inteligente seja alcançada é necessário que o computador relacione-se naturalmente com o usuário, semelhante à forma que os humanos interagem [Leão et al. 2012]. Algumas das maneiras de promover esse tipo de integração é com auxílio da Computação Afetiva, Inteligencia Artificial, *Machine Learning*, *Deep Learning*, entre outras.

*Trabalho desenvolvido com recursos do edital MCTIC/CNPq - Universal 28/2018 sob número 436339/2018-8 e do edital da VRPGPE bolsa PIBIC/UNIJUI.

Através destas abordagens é possível utilizar computadores para reconhecer, modelar e expressar as emoções e de que maneira podem responder às mesmas. Conforme Leão *et al.* (2012), fazer com que uma máquina reconheça, modele e expresse emoções não é uma tarefa simples. Quando seres humanos interagem entre si, boa parte dessa interação é baseada na linguagem verbal e na utilização da linguagem corporal por meio de gestos e expressões faciais que carregam e transmitem as emoções dos interlocutores.

Nas últimas décadas a comunidade científica vem tendo um crescente interesse no reconhecimento de emoções. Existem diversas maneiras de expressar as emoções humanas, e estas vêm sendo estudadas ao longo dos anos, e inúmeras fontes de dados têm sido exploradas, tais como textos, envio de *emoticons*, voz e as expressões faciais. Além das expressões faciais desempenharem um papel importante na relação entre as pessoas, ou seja, fornece informações que são relevantes sobre as emoções e intenções de um indivíduo durante o diálogo ou comunicação. O reconhecimento de tais emoções, nos possibilitam criar inúmeras aplicações de interação humano-computador e vem atraindo a atenção da comunidade científica.

Conforme a tecnologia de reconhecimento facial progride, também surgem novas maneiras de utilizá-la no nosso cotidiano, como para fazer pagamentos, ver quem está atento à aula, acordar um motorista sonolento, menus personalizados e marketing dirigido ao comércio. Imagine uma loja poder organizar todos os seus produtos, conforme a reação de seu clientes. Assim podendo deixar os produtos que mais chamam a atenção dos indivíduos com um maior destaque na loja ou vitrine, assim chamando a atenção e atraindo ainda mais clientes para a loja.

Este trabalho visa a realização do reconhecimento das 6 emoções consideradas básicas por Ekman (1973), que são felicidade, tristeza, medo, surpresa, raiva, nojo, mais a emoção neutra. Para esta tarefa, serão utilizadas imagens de expressões faciais humanas, e para a extração das características, será utilizada uma Rede Neural Convolucional (CNN - *Convolutional Neural Network*). O principal objetivo é realizar a análise e comparação dos tempos da CNN nas diferentes arquiteturas de CPU e GPU. Além de observar o desempenho obtido pela CNN nas duas arquiteturas.

O restante do trabalho está organizado da seguinte forma. A Seção 2 discute os trabalhos relacionados. A Seção 3 apresenta a metodologia utilizada na implementação e o ambiente de execução utilizado na realização dos testes. Os resultados são discutidos na Seção 4, seguidos das conclusões e perspectivas de trabalhos futuros.

2. Trabalhos Relacionados

No trabalho de Bartlett *et al.* (2003) é apresentado um estudo com o objetivo de localizar automaticamente faces em um fluxo de vídeo e codificar a expressão visual de maneira dinâmica. Os autores discutem que a comunicação face a face é uma operação em tempo real e com uma escala de tempo em 40 milissegundos. O sistema é capaz de detectar sete expressões: neutra, raiva, desgosto, medo, alegria, tristeza e surpresa, com um diferencial de outros trabalhos pois opera em tempo real. Este sistema foi treinado e testado utilizando a base de dados *CohnKanade AU-Coded Expression Database*. Esta base de dados contém o registro facial de 210 adultos na faixa etária entre 18 e 50 anos de idade, sendo 69% do sexo feminino e 31% masculino, e, 81% Euro-Americanos, 13% Afro-Americanos e 6% de outros grupos étnicos. Os experimentos realizados compararam o desempenho do reconhecimento da abordagem de detecção automática com a abordagem de detecção manual, não encontrando nenhuma diferença significativa entre elas. O sistema apresentou um nível de precisão de 93% de reconhecimento, na seleção de uma

das 7 opções de expressões faciais.

O trabalho desenvolvido por Tang and Huang (2008), tem como objetivo reconhecer as seis emoções básicas e universais através de expressões faciais utilizando da geometria 3D. Esta abordagem extrai características que são invariantes sob efeito de iluminação ou postura, características que, os autores consideram como obstáculos para o reconhecimento facial em imagem 2D. Neste trabalho foi utilizado, como base de treinamento e teste, a base de dados *BU-3DFE*. Esta base de dados é composta por 100 indivíduos, sendo que 60% do sexo feminino e 40% do sexo masculino, com variedade étnicas, incluindo branco, preto, Leste Asiático, Médio Oriente Asiático, Latino-Americano entre outras. Para este trabalho foi constatado que a abordagem produziu um aumento absoluto de 3,5% na taxa média de reconhecimento. Esta abordagem obteve uma precisão média de 87,1%, sendo que a maior taxa foi de 99,2% para o reconhecimento da expressão facial da surpresa.

No trabalho desenvolvido por Amin *et al.* (2017), elaborou-se uma RNA que possui a finalidade de reconhecer emoções através de expressões faciais utilizando a técnica de aprendizado profundo. Segundo o autor, a utilização de redes neurais convolucionais, na abordagem de identificação de emoções, atinge uma precisão média de 60%. Para o desenvolvimento do trabalho foi utilizado a base de dados *Facial Expression Recognition 2013 (FER-2013)*. Esta base de dados foi elaborada e desenvolvida por Pierre Luc Carrier e Aaron Courville, e possui um total 35887 imagens, com dimensões de 48x48 *pixels* As amostras possuem as seis expressões faciais consideradas como básicas por Ekman (1973), adicionadas da expressão neutra. As amostras desta base de dados estão distribuídas em: 8989 imagens de alegria; 547 de desgosto (nojo); 5121 de medo; 6198 de neutra; 4953 de raiva; 4002 de surpresa; e 6077 de tristeza. O trabalho apresentou bons resultados, alcançando uma precisão média de 61,05% para a classificação das sete emoções. Analisando os resultados, os autores constataram que a emoção de alegria possui a maior taxa de reconhecimento. Porém, o modelo não consegue classificar perfeitamente as emoções de medo e tristeza.

Diferente dos trabalhos aqui apresentados, a nossa proposta busca desenvolver uma RNA que reconheça expressões faciais, e a partir destas identificar a emoção presente em uma determinada imagem. Propondo o desenvolvimento de uma RNA em arquitetura *GPU*, visando obter um melhor desempenho, tanto no seu treinamento, quanto na análise e processamento de suas entradas. Para que seja possível obter um resultado satisfatório no menor tempo possível.

3. Metodologia

Para a implementação da RNA proposta utilizou-se da linguagem de programação *Python*. As principais bibliotecas utilizadas são o *TensoFlow 2.0* e o *Keras*, que são voltadas ao aprendizado de máquina profundo. O *Keras* é o *framework* mais utilizado na área pela sua facilidade de uso e rápida prototipagem. No *Tensoflow 2.0*, o *Keras* foi "incorporado" ao *TensorFlow* através do módulo *tf.keras*. A RNA também utiliza as ferramentas: *OpenCV*, *Scikit-Learn*, *Numpy*, *Pandas*, *Matplotlib*.

A base de dados utilizada para este trabalho foi *FER-2013* (*Facial Expression Recognition 2013*). Conforme já mencionado está base de dados reúne um conjunto de dados *open-source* criado por Pierre- Luc Carrier e Aaron Couville, depois compartilhado publicamente para uma competição *Kaggle*[1] em 2013. Para iniciar o desenvolvimento da RNA é preciso aplicar algumas transformações nas informações do arquivo *FER2013.csv*. No momento de realizar

[1] *Kaggle* é uma plataforma web de competições de *Data Science*. Atualmente é propriedade da empresa *Google*

a leitura do arquivo *csv*, as informações estão no formato de *string*, é necessário converte-las para um *array*, para isso é utilizado a função *tolist()*, que converter os dados da base para uma lista. Na elaboração, optou-se por implementar uma rede neural convolucional, que vem sendo aplicada com sucesso no processamento e análise de imagens digitais [Vargas et al. 2016].

Para a criação do modelo da RNA, é definido um número total de filtros (*num_features*) como 32, a divisão do trabalho, no caso, para realizar os ajustes dos pesos da RNA (*batch_size*) como 16, e um total de 100 épocas, para realizar o treinamento. Também é definida uma métrica de parada (*EarlyStopping()*), no período de 15 épocas. O modelo de RNA possui uma sequência de quatro camadas de convolução, em cada uma é utilizando a função *Conv2D*, que recebe o total de filtros. Recebe um *kernel_size* de tamanho 3x3, que funciona como filtros que enxergam pequenos quadros e vai percorrendo toda a imagem captando os seus traços mais relevantes. Neste filtro é pego ponto a ponto e realizada a multiplicação por um detector de características, definido pela própria biblioteca utilizada, formando no final um mapa de características (*feature map*). É utilizado o hiperparâmetro *padding* = *same*, caso necessário será inserido uma coluna extra a matriz, totalmente zerada.

Em cada camada convolucional é utilizada a função de ativação ELU, também é usado a função de *MaxPolling2D*, que recebe um *kernel_size* de tamanho 2x2, que retorna o maior número da unidade, e passa este valor como saída. Essa sumarização de dados serve para diminuir a quantidade de pesos a serem aprendidos pela Rede Neural, além de também evitar *overfitting*. Cada camada convolucional possui um *Dropout* de 20%. Depois de realizado os processos anteriores é chamada a função de *Flatten()*, que recebe a matriz criada nas etapa de *Polling* como entrada, e a transforma em um *array*, ou seja, converte a matriz para um vetor de características, que é utilizado como entrada para o treinamento da RNA. A classe *Model* da API funcional da biblioteca *Keras* é utilizada no desenvolvimento do modelo de dados. A compilação do modelo configura o processo de aprendizado. Sendo definido o otimizador (*adam*), a função de perda (*binary_crossentropy*) e as métricas (*accuracy*).

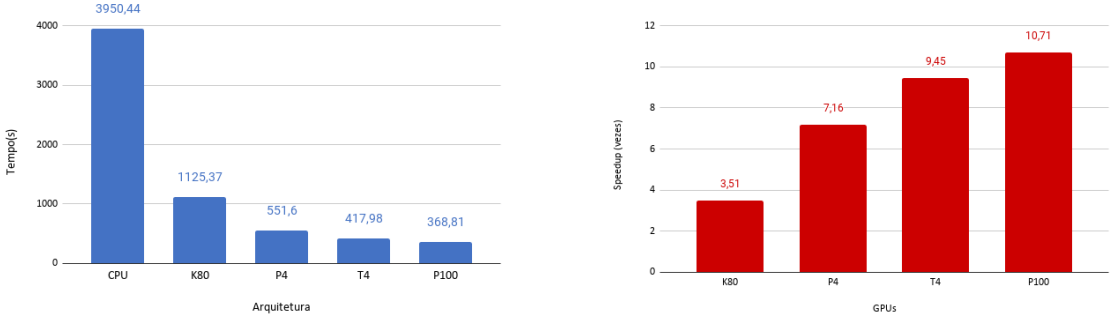
O ambiente de execução utilizado foi a plataforma do Google Colab, um serviço de nuvem gratuito hospedado pelo Google para Aprendizado de Máquina e Inteligência Artificial (IA), possuindo aceleradores de GPU grátis, bibliotecas já pré-instaladas, construído com base no *Jupyter Notebook*, suporta comandos *bash* além de armazenar os *notebooks* no próprio *Drive*. Possui uma Máquina Virtual Linux Ubuntu 18.04 LTS, e suporte ao CUDA, em sua versão 10.1. A plataforma possui 4 GPUs disponíveis, a NVIDIA Tesla K80, Tesla P4, Tesla T4 e a Tesla P100, cada GPU com suas respectivas características.

A NVIDIA Tesla K80 possui 24 GB de memória com largura de banda de memória incrivelmente rápida e desempenho de computação líder para cargas de trabalho de precisão única e dupla. Equipado com a tecnologia NVIDIA GPU *Boost*. Possui 4.992 CUDA cores, 8.73 TFLOPS de precisão simples, 2.91 TFLOPS de precisão dupla. A NVIDIA Tesla P4 possui arquitetura NVIDIA Turing, é alimentada por 320 núcleos de tensores NVIDIA Turing. Essa GPU é empacotada em um fator de forma *PCIe (Peripheral Component Interconnect Express)* pequeno de 70 *watts* e com baixo consumo de energia. Além de ser equipada com 2.560 CUDA Cores, com 8.1 TFLOPS de precisão simples e 65 TFLOPS de precisão mista. A NVIDIA Tesla T4 possui a arquitetura NVIDIA Pascal, desenvolvida especificamente para aumentar a eficiência de servidores em expansão que executam cargas de trabalho de aprendizado profundo. Possui 2.560 CUDA Cores, com 5.5 TFLOPS de Desempenho de Precisão Única. A NVIDIA Tesla P100 é desenvolvida com a arquitetura NVIDIA Pascal e projetado para aumentar a taxa

de transferência e para data centers HPC (*High Performance Computing*) e hiperescala. Possui 3.584 CUDA Cores, 4.7 TFLOPSde Desempenho de Dupla Precisão, 9.3 TFLOPS de Desempenho de Precisão Única e 18.7 TFLOPS de Desempenho de Meia-Precisão.

4. Resultados

Os resultados dos tempos de execução, são apresentados na 1(a). O *speedup* alcançado pela RNA comparado a execução em CPU, é apresentado na Figura 1(b), em ambas a Tesla P100 obteve o melhor resultado. O *speedup* do algoritmo executado em GPU apresentou um ganho de $10,71$ vezes sobre a CPU. Reduzindo o tempo de execução de $3950,44$ segundos para $368,81$ segundos, apresentando um ganho de $90,66\%$. O resultado do *speedup* das demais GPUs também foram satisfatórios. O *speedup* do algoritmo executado na GPU Tesla K80 apresentou um ganho de $3,51$ vezes sobre a CPU. Diminuindo o tempo de execução de $3950,44$ segundos para $1125,37$ segundos, obtendo um ganho de $71,21\%$. Já a Tesla P4 exibiu um ganho de $7,16$ vezes sobre a CPU. Reduzindo o tempo de execução de $3950,44$ segundos para $551,60$ segundos, alcançando um ganho de $86,04\%$. E por fim a Tesla T4 apontou um ganho de $9,45$ vezes comparada a CPU. Diminuindo o tempo de execução de $3950,44$ segundos pra $417,98$, tendo um ganho de $89,42\%$.



(a) tempo alcançado em cada arquitetura

(b) *Speedup* alcançado (vezes)

Figura 1. Comparativo dos tempos e *Speedup* alcançados.

A GPU que obteve a melhor acurácia foi a Tesla K80. O gráfico da acurácia e perda do treinamento da Tesla K80 é apresentado na Figura 2. Em relação ao gráfico de perda, é possível observar que a taxa começa alta e vai reduzindo, logo após a época 9, o percentual de perda começa a aumentar a cada nova época, obteve-se apenas uma perda de apenas $1,40$. Examinando os gráficos da Figura 2, nota-se que o aprendizado da rede manteve-se constante, desde seu início, com poucas variações. Isto ocorre, pela utilização de um percentual menor na técnica de *Dropout*. O fato da RNA ter apresentado pouca variação no momento do aprendizado, é a baixa taxa de *Dropout*. Pois durante o treinamento a rede neural já pode conter informações corretas sobre os dados de saída, e apenas uma pequena taxa da rede é zerada, ou seja, ela começa a se especializar mais rapidamente com os dados de saída, assim não apresentando muitas oscilações em seu treinamento. Quando a taxa de *Dropout* for maior as chances de eliminar informações que a rede neural já considera como corretas, são maiores. Dessa forma, a rede neural acaba precisando treinar mais, apresentando maiores oscilações em seu treinamento.

5. Conclusões e trabalhos futuros

Este trabalho abordou o desenvolvimento de uma rede neural convolucional para o reconhecimento de emoções através de expressões faciais, realizando inúmeros testes para analisar

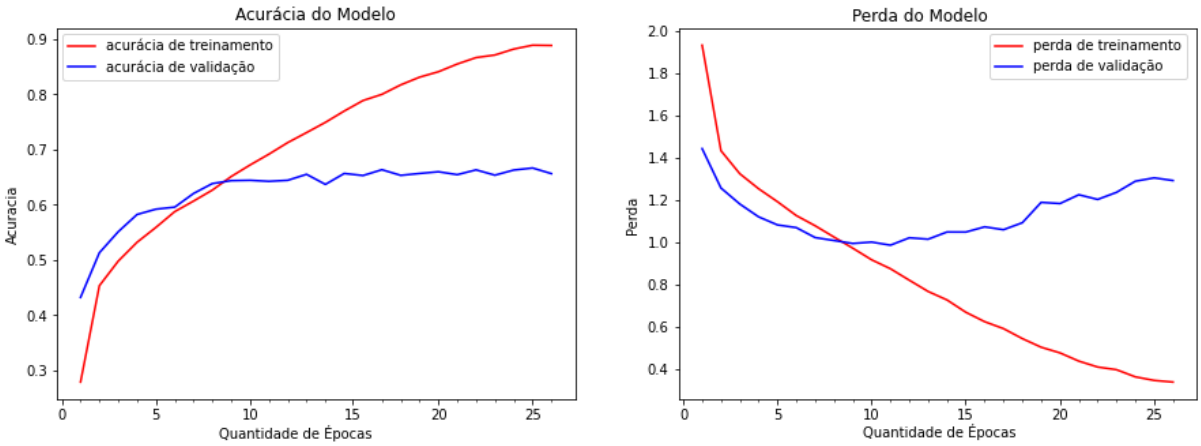


Figura 2. Gráfico da taxa de acurácia e perda do treinamento.

a avaliar o desempenho da aplicação executada em arquiteturas CPU e GPU. Com a nossa implementação foi possível aumentar a acurácia da RNA alcançando uma precisão de até $65,97\%$. Com relação ao tempo computacional a versão desenvolvida para GPU Tesla P100, obteve-se excelentes resultados, reduzindo o tempo de execução em até $10,71$ vezes se comparado com a execução na CPU.

Como trabalhos futuros pretende-se aplicar a solução desenvolvida em outras bases de dados para validar seu comportamento. Uma segunda iniciativa seria desenvolver a Rede Neural Artificial para fazer o reconhecimento das emoções em tempo real de pessoas em vídeos. Também poderia ser modificado o algoritmo da Rede Neural Artificial para que se possa utilizar o ambiente de execução em *TPU* do *Google Colab*, além de analisar a influência de outros hiperparâmetros, como taxa de aprendizagem, de *Dropout* e Função de Ativação.

Referências

Amin, D., Chase, P., and Sinha, K. (2017). Touchy feely: An emotion recognition challenge. *Palo alto: Stanford*.

Bartlett, M. S., Littlewort, G., Fasel, I., and Movellan, J. R. (2003). Real time face detection and facial expression recognition: development and applications to human computer interaction. In *2003 Conference on computer vision and pattern recognition workshop*, volume 5, pages 53–53. IEEE.

Ekman, P. (1973). Cross-cultural studies of facial expression. *Darwin and facial expression: A century of research in review*, 169222(1).

Leão, L. P., Bezerra, J. S., Matos, L. N., and Nunes, M. A. S. N. (2012). Detecção de expressões faciais: uma abordagem baseada em análise do fluxo óptico. *Revista GEINTEC-Gestão, Inovação e Tecnologias*, 2(5):472–489.

Tang, H. and Huang, T. S. (2008). 3d facial expression recognition based on properties of line segments connecting facial feature points. In *2008 8th IEEE International Conference on Automatic Face & Gesture Recognition*, pages 1–6. IEEE.

Vargas, A. C. G., Paes, A., and Vasconcelos, C. N. (2016). Um estudo sobre redes neurais convolucionais e sua aplicação em detecção de pedestres. In *Proceedings of the xxix conference on graphics, patterns and images*, volume 1.