ISNN: 2358-6052

ESTUDO COMPARATIVO ENTRE DUAS MÉTRICAS DE AVALIAÇÃO PARA EXTRATORES DE CARACTERÍSTICAS DE UMA CNN: ASSINATURA DA IMAGEM X CLUSTERIZAÇÃO

Bruno Alberto Soares Oliveira [1]; Servílio Souza de Assis [2]; Luana da Costa Faria [3]; Alan Cândido de Souza [4]; Frederico Gadelha Guimarães [5];

1 Programa de Pós-Graduação em Engenharia Elétrica, UFMG, Belo Horizonte – MG; brunoalbertobambui@ufmg.br
2 Programa de Pós-Graduação em Engenharia Elétrica, UFMG, Belo Horizonte – MG
3 Engenharia de Controle e Automação, UFMG, Belo Horizonte – MG
4 Programa de Pós-Graduação em Engenharia Elétrica, UFMG, Belo Horizonte – MG
5 Machine Intelligence and Data Science (MINDS) Laboratory, UFMG, Belo Horizonte – MG; fredericoguimaraes@ufmg.br

RESUMO

A extração de características envolve a redução da quantidade de recursos necessários para descrever um grande conjunto de dados. Ao realizar a análise de dados complexos, um dos principais problemas decorre do número de variáveis envolvidas. O processamento com um grande número de variáveis geralmente requer uma grande quantidade de memória e poder computacional, além de poder fazer com que um algoritmo de classificação se sobreponha a amostras de treinamento e generalize mal a novas amostras. As clássicas Redes Neurais Convolucionais (CNN) necessitam em seu processo de treinamento que o erro seja calculado somente na camada de saída após ter passado por um classificador, para que posteriormente esse erro se retropropague e respectivamente atualize tais extratores de características das camadas convolucionais. O sistema proposto utiliza um modelo de CNN para gerar alguns extratores de características e emprega ou a métrica de similaridade Perceptual Hash ou a métrica de clusterização para ajustar os valores de tais extratores em tempo de treinamento da CNN, buscando, ao final do treinamento, concluir se esses extratores conseguem capturar as informações mais relevantes das imagens. A primeira técnica é uma impressão digital de um arquivo multimídia derivado de vários recursos de seu conteúdo. Ao contrário das funções hash criptográficas que dependem do efeito avalanche de pequenas mudanças na entrada que levam a mudanças drásticas na saída, os pHashs são “próximos” um do outro se os recursos forem semelhantes. Já o segundo método, este é uma técnica de aprendizado de máquina que envolve o agrupamento de pontos de dados. Para avaliar o desempenho do algoritmo proposto, são aplicados os métodos em dois conjuntos de dados: Mnist e Fashion Mnist, uma vez que ambos datasets são utilizados em toda literatura como forma de benchmark para aplicações com imagens. Com os resultados pôde-se concluir a eficiência do método proposto.

INTRODUÇÃO:

A Inteligência Computacional (IC) pode ser definida como sendo a teoria, design, aplicação e desenvolvimento de paradigmas computacionais baseados sob conceitos inspirados na natureza. Tradicionalmente, os três pilares principais da IC são: Redes Neurais, Sistemas Difusos e Computação Evolucionária. No entanto, com o tempo, muitos paradigmas da computação inspirada na natureza evoluíram (JANG et al., 1997). Assim, a IC é um campo em evolução e, atualmente, além das três principais constituintes, englobam paradigmas da computação como inteligência ambiental, vida artificial, aprendizado cultural, redes endócrinas artificiais, raciocínio social e redes hormonais artificiais.

A IC desempenha um papel importante no desenvolvimento de sistemas inteligentes de sucesso, incluindo jogos e sistemas de desenvolvimento cognitivo. Nos últimos anos tem havido uma explosão de pesquisas sobre Aprendizado Profundo, em particular, Redes Neurais Convolucionais profundas. Atualmente, o aprendizado profundo se tornou o método central da Inteligência Artificial (IA). De fato, alguns dos sistemas de IA mais bem sucedidos são baseados em IC.

No aprendizado de máquina, reconhecimento de padrões e processamento digital de imagens, a extração de características inicia a partir de um conjunto inicial de dados medidos e constrói recursos destinados a serem representativos e não redundantes, facilitando as etapas subsequentes do aprendizado, da generalização e, em alguns casos, levando para melhores interpretações humanas. A extração de recursos é um processo de redução de dimensionalidade, em que um conjunto inicial de variáveis brutas é reduzido a grupos mais gerenciáveis para um futuro processamento, uma vez que, apesar da redução de dimensionalidade, ainda descreve de forma precisa e completa o conjunto de dados original (LADHA e DEEPA, 2011).

ISNN: 2358-6052

Quando os dados de entrada para um método computacional são muito grandes para serem processados e são supostamente redundantes, esses dados podem ser transformados em um conjunto reduzido de recursos. Determinar um subconjunto de recursos iniciais é chamado de seleção de recursos ou extração de características (ALPAYDIN, 2009). Espera-se que os recursos selecionados contenham as informações relevantes dos dados de entrada, para que a tarefa desejada possa ser executada usando essa representação reduzida em vez dos dados iniciais completos.

Modelos de aprendizagem de máquina profundos (ou também popularmente conhecidos como Deep Learning) vem alcançando resultados extremamente satisfatórios em aplicações de visão computacional (KRIZHEVSKY et al., 2012) e também no reconhecimento de fala (GRAVES et al., 2013).

As CNNs utilizam camadas com filtros de convolução que são aplicados a características locais (GRAVES et al., 2013). De acordo com KRIZHEVSKY et al. (2012), uma rede neural convolucional (CNN) é uma classe de redes neurais profundas, mais comumente aplicadas à análise de imagens digitais. As CNNs usam relativamente pouco pré-processamento em comparação com outros algoritmos de classificação de imagem. Isso significa que a rede aprende os filtros que, nos algoritmos tradicionais, foram manipulados à mão. Essa independência do conhecimento prévio e do esforço humano no design de recursos é uma grande vantagem. Suas principais aplicações são em imagem e vídeo de reconhecimento, sistemas de recomendação, classificação de imagens, análise de imagem médica e processamento de linguagem natural.

A eficiência das CNNs no processo de reconhecimento de imagens é uma das principais razões pelas quais o mundo acordou para a eficácia da aprendizagem profunda. Atualmente, existem inúmeros centros de pesquisas que estão impulsionando grandes avanços na visão computacional que tem aplicações bastante úteis para carros autônomos, robótica, drones, segurança, diagnósticos médicos e tratamentos para deficientes visuais.

Uma CNN consiste em uma camada de entrada e uma de saída, além de várias camadas ocultas. As camadas ocultas de uma CNN consistem em camadas convolucionais, a camada de ativação, que em grande parte das aplicações é utilizada a RELU, camadas de agrupamento, camadas totalmente conectadas e as camadas de normalização (KARPATHY et al., 2014). As camadas convolucionais realizam toda a parte da extração de características da imagem inicialmente para que, posteriormente, alimente uma Rede Neural Densa de forma que essa última possa realizar o processamento para classificação.

Devido à crescente digitalização, a autenticação de conteúdo multimídia está se tornando cada vez mais importante. Autenticação em geral significa decidir se um objeto é autêntico ou não, ou seja, se corresponder a um determinado objeto original. A autenticação depende muito do tipo do objeto. Ao autenticar um arquivo executável é importante que cada bit corresponda exatamente ao executável original. As funções hash criptográficas são adequadas para essas tarefas.

Um objeto multimídia, por exemplo, uma imagem, pode ter diferentes representações digitais que parecem todas iguais à percepção humana. Diferentes representações digitais podem emergir de uma imagem através de etapas de processamento de imagem, como corte, compressão ou equalização de histograma. Cada uma dessas etapas de processamento de imagem altera a representação binária da imagem.

As chamadas funções Perceptual Hash (pHash) foram propostas para estabelecer a “igualdade perceptual” do conteúdo multimídia. Nos últimos anos, um crescente interesse científico e industrial na tecnologia de pHash tem sido visto. Tais funções foram desenvolvidas para diferentes tipos de mídia digital, a exemplo áudio, imagem ou vídeo. As funções de hash perceptuais extraem determinados recursos do conteúdo multimídia e calculam um valor de hash com base nesses recursos. Ao autenticar um objeto multimídia, os valores de hash do objeto original e do objeto a ser autenticado são comparados usando funções específicas. Essas funções calculam uma pontuação de distância ou similaridade entre dois valores de pHashs (ZAUNER, 2010).

Um Perceptual Hash é uma impressão digital de um arquivo multimídia derivado de vários recursos de seu conteúdo. Ao contrário das funções hash criptográficas que dependem do efeito avalanche de pequenas mudanças na entrada que levam a mudanças drásticas na saída, os hashes perceptuais são “próximos” um do outro se os recursos forem semelhantes.

De acordo com KOZAT et al. (2004), os pHashs devem ser robustos o suficiente para levar em conta transformações ou "ataques" em uma determinada entrada e ainda serem flexíveis o suficiente para distinguir entre arquivos diferentes. Tais ataques podem incluir rotação, inclinação, ajuste de contraste e diferentes formatos/compactações. Todos esses desafios tornam o estudo perceptivo interessante em um campo de estudo e na vanguarda da pesquisa em ciência da computação.

As possíveis aplicações incluem proteção de direitos autorais, pesquisa de similaridade de arquivos de mídia ou até análise forense digital. NIU e JIAO (2008) exemplifica que o YouTube poderia manter um

ISNN: 2358-6052

banco de dados de hashes que foram enviados pelos principais produtores de filmes aos quais eles detêm os direitos autorais. Se um usuário enviar o mesmo vídeo para o YouTube, o hash será quase idêntico e poderá ser sinalizado como possível violação de direitos autorais. O hash de áudio pode ser usado para marcar automaticamente arquivos MP3 com informações ID3 adequadas, enquanto o hash de texto pode ser usado para detecção de plágio.

Segundo SEIF (2019), clustering é uma técnica de aprendizado de máquina que envolve o agrupamento de pontos de dados. Para um conjunto de pontos de dados é possível um algoritmo de agrupamento para classificar cada ponto de dados em um grupo específico. Em teoria, esse tipo de algoritmo separa os pontos de dados que estão no mesmo grupo, uma vez que possuem propriedades e/ou recursos semelhantes, enquanto existe outro grupo de pontos de dados que possivelmente possuem propriedades e/ou recursos altamente dissimilares. O Clustering é um método de aprendizado não supervisionado e é uma técnica comum para análise de dados estatísticos usada em muitos campos.

O algoritmo k-means é um método de quantização vetorial que é altamente popular para análise de cluster em mineração de dados. O k-means visa particionar n observações em k clusters nos quais cada observação pertence ao cluster com a média mais próxima, servindo como um centroide do cluster. Isso resulta em um particionamento do espaço de dados nas células de Voronoi (HARTIGAN e WONG, 1979).

O sistema proposto utiliza um modelo de CNN para gerar alguns extratores de características e emprega ou a métrica de similaridade Perceptual Hash ou a métrica de clusterização para ajustar os valores de tais extratores em tempo de treinamento da CNN, buscando, ao final do treinamento, concluir se esses extratores conseguem capturar as informações mais relevantes das imagens.

Assim, neste trabalho, propõe-se utilizar a métrica de similaridade pHash e uma métrica de clusterização para avaliar e atualizar os valores dos extratores de características gerados por uma CNN. Para avaliar o desempenho das métricas propostas são aplicados os métodos em dois conjuntos de dados: Mnist e Fashion Mnist.

METODOLOGIA:

Para o desenvolvimento deste trabalho foi utilizada a linguagem de programação R no IDE R Studio, em um Notebook da marca Gateway, com seis GB de memória RAM e processador Intel Core i3-2328M. Algumas bibliotecas importantes que foram utilizadas são: RSNNS, OpenImageR e caret.

Para um melhor entendimento da implementação proposta é apresentado um fluxograma na Figura 1 que ilustra as etapas da realização de cada experimento.

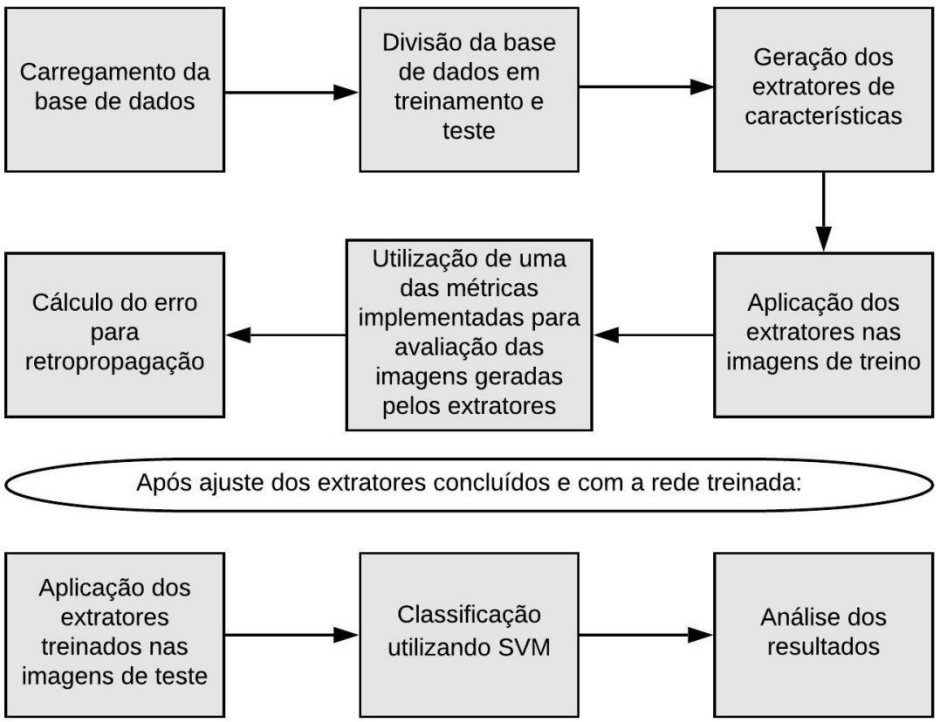

Figura 1: Fluxograma ilustrando as etapas do algoritmo implementado.

ISNN: 2358-6052

Inicialmente, define-se qual base de dados será utilizada e respectivamente o número de classes. Posteriormente, divide-se a base de dados escolhida em quatro subconjuntos: as imagens para treinamento da CCN, as imagens de teste da CNN, as classes das imagens de treinamento e as classes das imagens de teste. Com o conjunto de dados preparado, configuram-se os parâmetros do modelo sendo, uma camada de convolução constituída por seis extratores de característica de tamanho 3x3 cada, uma camada de pooling que possui um filtro de tamanho 2x2 que captura a média dos quatro píxeis nessa operação de convolução e a camada de flattern que é responsável por transformar a imagem de duas dimensões em apenas uma.

Com a arquitetura da rede definida, estabelece-se como 128 o tamanho de lotes e o número de épocas em cada execução sendo 10. O processo de treinamento é realizado através do algoritmo de backpropagation. Chamada a função de treinamento da CNN, as 128 imagens selecionadas aleatoriamente sofrem a operação de convolução na primeira camada e posteriormente este resultado entra na camada de pooling sofrendo uma nova redução de tamanho. Espera-se que essa nova imagem gerada represente bem a imagem original e, para verificar essa similaridade do quão bom foi feita a extração de características, foi utilizada uma das duas métricas implementadas neste trabalho.

O principal objetivo da função pHash é calcular a assinatura de uma imagem utilizando a Transformada Discreta de Cosseno (TDC). Ela recebe como entrada uma matriz bidimensional (a imagem), um inteiro especificando o tamanho dessa assinatura (representado por um vetor), um valor inteiro que especifica o fator de alta frequência a ser utilizado na TDC, um parâmetro que retorna o formato da assinatura (hexadecimal ou binário) e o método de redimensionamento.

A função pHash é utilizada dentro do método que implementa a métrica de avaliação dos extratores de características. Esse método recebe como parâmetro o lote das imagens originais, o lote das imagens após terem passado pela última camada de pooling e suas respectivas classes.

A primeira parte do método consiste em calcular, através da função pHash, as assinaturas de cada imagem original e as assinaturas de cada imagem que representa as características extraídas das imagens originais. Extraído a assinatura de cada imagem é comparada a assinatura da imagem original com a assinatura da imagem das características extraídas, uma a uma, tendo como retorno da função um valor entre 0 e 1 que indica a similaridade entre essas duas imagens.

A principal utilização da função de ativação sigmóide em classificadores é retornar a probabilidade de uma entrada X pertencer a classe Y. Análogo a esse recurso, o método de simetria proposto retorna o quão similar são as imagens, fazendo com que seja possível utilizar esse valor de similaridade subtraído por 1, como o erro a ser retropropagado pelo algoritmo de backpropagation. Para as demais classes, é utilizado o valor de não similaridade entre as imagens e dividido igualmente entre o número de classes do problema.

A função cluster tem como objetivo encontrar o centroide de cada grupo de imagens pertencentes a mesma classe. Ela recebe como parâmetros as imagens que estão alocadas atualmente no batch daquela determinada iteração do processo de treinamento e suas respectivas classes.

Primeiramente, as imagens do batch são separadas por classe e, posteriormente, é medida a distância euclidiana de todas entre todas as imagens pertencentes a aquela determinada classe. Tendo obtido essas distâncias, é encontrado o centroide de cada grupo que contém a mesma dimensão das imagens.

Com todos os centroides encontrados, é calculada a distância de cada imagem do batch para cada centroide calculado, tendo assim, uma matriz de valores que correspondem a uma medida que posteriormente é utilizada como erro no algoritmo de backpropagation.

Implementado a métrica que afere o quão bem as imagens com as características extraídas se aproximam das imagens originais e tendo realizado o processo de treinamento, é aplicado o classificador SVM para verificar a acurácia em cima do conjunto de teste. Primeiramente, é utilizado o conjunto de treinamento para treinar o modelo da SVM. Posteriormente, são aplicadas as imagens contidas no conjunto de teste no modelo gerado pela CNN, obtendo-se como retorno uma matriz de características, que em seguida é aplicado no modelo gerado pelo SMV.

Os parâmetros utilizados para a função SVM foram: Validação cruzada repetida, fazendo-se o k=10 e repetindo o processo por 10 execuções, retornando a precisão final média dada as 10 iterações e um kernel de base radial, uma vez que o problema não é linearmente separável. Feito o treinamento, é realizada as classificações das imagens de teste e analisado os resultados através de uma matriz de confusão.

Com a implementação desenvolvida foram realizados alguns experimentos. Primeiramente, foi analisado o conjunto de dados Mnist utilizando a métrica pHash. Posteriormente, foi feito a execução do algoritmo para o conjunto de dados Fashion Mnist, primeiro com a métrica pHash e depois com a métrica de clusterização.

ISNN: 2358-6052

RESULTADOS E DISCUSSÕES:

A Figura 2 ilustra um exemplo de três imagens originais e três imagens de características geradas no processo de treinamento após terem passado na última camada de pooling.

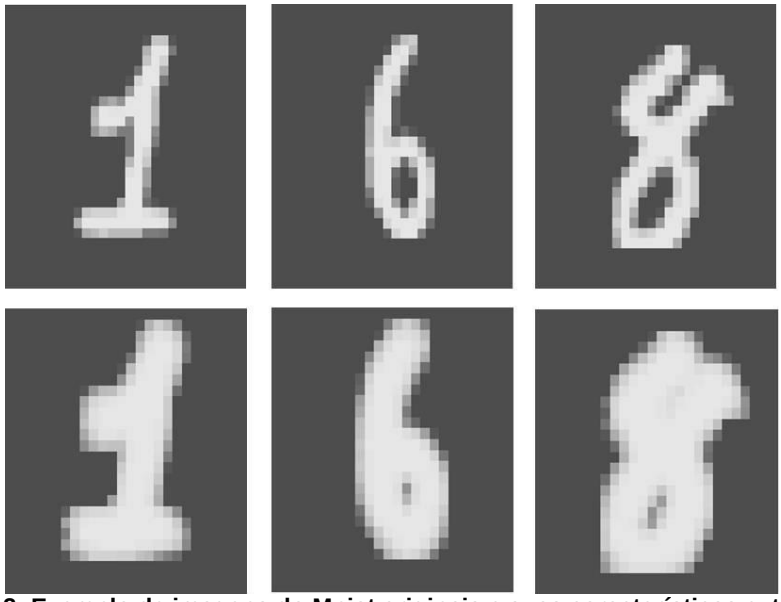

Figura 2: Exemplo de imagens do Mnist originais e suas características extraídas.

É possível identificar a olho humano através da Figura 2 que a extração foi bem sucedida, considerando que é possível visualizar qual é o dígito da imagem, mesmo depois de ter sofrido as operações de convolução.

A Tabela 1 demonstra o valor real das imagens originais ilustradas na Figura 2 e o valor previsto obtido pelas métricas utilizadas no desenvolvimento deste trabalho.

Tabela 1: Valores de saída das imagens ilustrados na Figura 2.

Esperado			Saída prevista – pHash			Saída prevista - Clusterização		
1	6	8	1	6	8	1	6	8
0	0	0	0.00000000	0.02469136	0.04938272	1.02E+05	9.89E+04	1.07E+05
1	0	0	1.00000000	0.02469136	0.04938272	1.14E+05	8.88E+04	4.06E+04
0	0	0	0.00000000	0.02469136	0.04938272	1.06E+05	1.12E+05	1.15E+05
0	0	0	0.00000000	0.02469136	0.04938272	1.02E+05	1.09E+05	1.04E+05
0	0	0	0.00000000	0.02469136	0.04938272	9.42E+04	8.24E+04	6.72E+04
0	0	0	0.00000000	0.02469136	0.04938272	8.64E+04	1.13E+05	1.06E+05
0	1	0	0.00000000	0.77777778	0.04938272	1.14E+05	1.25E+05	1.20E+05
0	0	0	0.00000000	0.02469136	0.04938272	1.01E+05	8.94E+04	1.15E+05
0	0	1	0.00000000	0.02469136	0.55555556	9.50E+04	9.30E+04	1.28E+05
0	0	0	0.00000000	0.02469136	0.04938272	8.57E+04	8.83E+04	9.64E+04

Como podem ser observados na Tabela 1, os valores da saída prevista se aproximam com os valores de saída real. Durante o treinamento, a tendência é que se aproximem cada vez mais como no exemplo da primeira imagem, que encontrou uma forte similaridade entre a imagem original e a imagem de características, significando que o extrator se comportou bem e realmente conseguiu capturar as características mais significativas da imagem.

É possível identificar na Tabela 1, para o dígito oito, utilizando a métrica pHash, uma não tão boa similaridade entre a imagem original com suas características extraídas. Por este motivo, o erro a ser retropropagado será maior no ajuste de peso dos extratores.

ISNN: 2358-6052

A Tabela 2 apresenta os resultados obtidos do conjunto de dados Mnist e Fashion Mnist, ambos com 10 classes.

Tabela 2: Acurácia obtida para o conjunto de teste.

Método	Mnist	Fashion
pHash	97.45%	87.44%
Cluster	97.64%	87.70%

Observando os valores ilustrados na Tabela 2 é possível concluir que existe um índice próximo de acerto ao se comparar as duas métricas implementadas neste trabalho. Durante o treinamento, notou-se que o erro para a métrica pHash foi aproximadamente 10 vezes menor do que utilizando a métrica de clusterização. Verificou-se também que no primeiro lote de imagens o erro estava muito alto e para o segundo lote de imagens houve uma brusca queda desse valor. Após essas primeiras atualizações, o erro de treinamento ficou oscilando entre valores próximos e assim permaneceu até o algoritmo terminar o número de épocas previamente estabelecido.

Ao se analisar a matriz de confusão do conjunto de dados Mnist e Fashion Mnist, ambos com 10 classes, tendo utilizado a métrica de avaliação pHash, foi possível concluir algumas suposições que possuem bastante relevância neste estudo. Uma observação importante a ser ressaltada é: um dos maiores erros de classificação observando-se a matriz de confusão obtida é o cruzamento da classe cinco com a classe três, ao se considerar o conjunto de dados Mnist. Ao se observar atentamente para esses dois dígitos, é possível notar a olho humano que a diferença entre eles é bem menor em relação com o cruzamento do dígito um e o dígito zero, por exemplo, o que justifica um maior erro e uma menor acurácia de classificação ao se comparar essas duas classes.

Ao se analisar a matriz de confusão do conjunto de dados Mnist e Fashion Mnist, ambos com 10 classes, tendo utilizado a métrica de avaliação de clusterização, foi possível seguir o mesmo raciocínio da matriz de confusão obtida nos resultados ao se utilizar a métrica pHash, uma vez que considerando a métrica de clusterização, para o conjunto de dados Mnist, foi possível perceber que uma das maiores quantidades de imagens classificadas erroneamente foram 13 imagens que pertencem a classe cinco mas foram classificadas pelo modelo sendo pertencentes a casse três.

CONCLUSÕES:

O presente trabalho propôs implementar duas métricas para avaliar o quão bom são os extratores de características que são atualizados durante o processo de treinamento de uma Rede Neural Convolucional. Uma das métricas implementada usa a função pHash que utiliza a Transformada Discreta de Cosseno para obter a assinatura entre duas imagens e posteriormente verificar o nível de similaridade entre elas.

Os resultados obtidos indicam um bom desempenho da utilização destas métricas para avaliar os extratores de características do modelo. Para a base de dados Mnist, obteve-se um valor de acurácia médio de 97.45% e 97.64%. Para a base de dados Fashion Mnist um valor de acurácia médio de 87.44% e 87.70%, utilizando a métrica pHash e a métrica de cluster, respectivamente.

REFERÊNCIAS BIBLIOGRÁFICAS:

ALPAYDIN, Ethem. **Introduction to machine learning**. MIT press, 2009.

GRAVES, Alex; MOHAMED, Abdel-rahman; HINTON, Geoffrey. Speech recognition with deep recurrent neural networks. In: **2013 IEEE international conference on acoustics, speech and signal processing**. IEEE, 2013. p. 6645-6649.

HARTIGAN, John A.; WONG, Manchek A. Algorithm AS 136: A k-means clustering algorithm. **Journal of the Royal Statistical Society. Series C (Applied Statistics)**, v. 28, n. 1, p. 100-108, 1979.

ISNN: 2358-6052

JANG, Jyh-Shing Roger; SUN, Chuen-Tsai; MIZUTANI, Eiji. Neuro-fuzzy and soft computing-a computational approach to learning and machine intelligence [Book Review]. **IEEE Transactions on automatic control**, v. 42, n. 10, p. 1482-1484, 1997.

KARPATHY, Andrej et al. Large-scale video classification with convolutional neural networks. In: **Proceedings of the IEEE conference on Computer Vision and Pattern Recognition**. 2014. p. 1725-1732.

KOZAT, Suleyman Serdar; VENKATESAN, Ramarathnam; MIHÇAK, Mehmet Kivanç. Robust perceptual image hashing via matrix invariants. In: **2004 International Conference on Image Processing, 2004. ICIP'04.** IEEE, 2004. p. 3443-3446.

KRIZHEVSKY, Alex; SUTSKEVER, Ilya; HINTON, Geoffrey E. Imagenet classification with deep convolutional neural networks. In: **Advances in neural information processing systems**. 2012. p. 1097-1105.

LADHA, L.; DEEPA, T. Feature selection methods and algorithms. **International journal on computer science and engineering**, v. 3, n. 5, p. 1787-1797, 2011.

LECUN, Yann et al. Gradient-based learning applied to document recognition. **Proceedings of the IEEE**, v. 86, n. 11, p. 2278-2324, 1998.

NIU, Xia-mu; JIAO, Yu-hua. An overview of perceptual hashing. **Acta Electronica Sinica**, v. 36, n. 7, p. 1405-1411, 2008.

SEIF, G. The 5 clustering algorithms data scientists need to know. **Towards Data Science. URL: https://towardsdatascience. com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68. Accessed**, v. 13, p. 2018, 2018.

ZAUNER, Christoph. Implementation and benchmarking of perceptual image hash functions. 2010.