

# Evoluindo Redes Neurais Convolucionais na Detecção de Emoções Usando Micro AGs

Diogenes Ademir Domingos  
Universidade Estadual do Maranhão  
(UEMA)  
Centro de Ciências Tecnológicas.  
São Luís, Maranhão, Brasil  
diogenes.20211001317@aluno.uema.br

Omar Andres Carmona Cortes  
Instituto Federal do Maranhão (IFMA)  
Departamento Acadêmico de  
Informática.  
São Luís, Maranhão, Brasil  
omar@ifma.edu.br

Fábio Manoel França Lobato  
Universidade Federal do Oeste do  
Pará (UFOPA)  
Instituto de Engenharia e Geociências.  
Santarém, Pará, Brasil  
fabio.lobato@ufopa.edu.br

## ABSTRACT

The Deep-emotive v.1 is a CNN that recognizes emotions by the human face's pictures. In this context, the CNN's structure creation depends on several hyperparameters, which impact the results positively or negatively. The Genetic Algorithm implementation allows us to explore the search space of these hyperparameters to find the best architecture for solving the problem. The defined search space is formed by the combination of both the number of convolutional layers and the fully connected ones, the number of filters for each layer, the size of filters, the subsampling type, and the number of nodes in the fully connected layer. This paper proposes to improve the Deep-Emotive network with the implementation of Convolutional Neural Networks (CNNs) architectures using Genetic Algorithms. The FER-2013 dataset was chosen to classify seven emotions by images of facial expressions, as it had the worst performance in the first version of the network, reaching an accuracy of 60.71%. This dataset has images with common problems for computer vision algorithms, such as occlusion, imbalance, perspective, noises, as well as images that do not exist in the context of emotions. The experiment's results indicate that the proposed approach can generate a CNN architecture with an accuracy of 63,84% in the train set and 62,39% in the validation set. Despite a low-performance rate, the experiments indicate that the algorithm can generate more adapted individuals who have already overcome the performance achieved by the first version of the network defined empirically. Thus, results show potential for exploitation in environments with more computational resources.

## KEYWORDS

Deep-emotive, Emotion recognition, Facial expressions, Convolutional neural network, Genetic Algorithms

## 1 INTRODUÇÃO

Redes Neurais Convolucionais, conhecidas pelo seu acrônimo em inglês - CNNs, são amplamente utilizadas em tarefas de visão computacional. Tais redes têm capacidade de extrair e aprender características por meio de transformações profundas e não lineares [1, 2]. Assim, é possível realizar o reconhecimento de bordas, objetos, faces, emoções, entre outros. Porém, a eficiência dessas redes depende da correta definição dos seus hiperparâmetros e de sua estrutura, os quais normalmente são definidos empiricamente por um especialista, demonstrando, assim, a necessidade de, além do pleno domínio de problema, grande experiência em criação de redes [1-3].

Como exemplo, observa-se a arquitetura Deep-Emotive v.1 [4], ilustrada na Fig. 1, criada empiricamente, a qual realiza a detecção de emoções por meio de expressões faciais obtida de imagens, independentemente de qualquer extração de características manual. A estrutura da rede é subdividida em dois blocos macro: bloco de extração de características, o qual contém três sub-blocos convolução, cada qual consiste em uma camada de convolução, uma de ativação e uma de subamostragem; e bloco de classificação, o qual possui duas camadas densas. Tal arquitetura foi treinada e validada no sob o conjunto de dados *Cohn-Kanade AU, versão 2 (CK+)* [5], apresentando precisão de 98,71%. Além disso, a rede foi validada por meio de transferência de aprendizagem, aplicando o aprendizado adquirido em diferentes conjuntos de dados, obtendo, assim, 97,67% de precisão no conjunto de dados *Japanese Female Facial Expression (JAFFE)* [6] e 60,71% no *Facial Expression Recognition 2013 (FER-2013)* [7].

No entanto, observa-se que há grande quantidade de estruturas possíveis para criação de uma arquitetura CNN e o espaço de busca aumenta exponencialmente com o número de camadas nela inclusa, o que torna inviável a aplicação de uma busca exaustiva. De fato, pesquisas atuais têm mostrado que o aumento na profundidade das redes, assim como a criação de novos módulos, melhora a precisão das redes CNN. Porém, em que pese a eficiência apresentada, a definição da arquitetura dessas redes ainda é realizada de forma empírica [8, 9].

Além disso, com base em *no free lunch theorem* [10], não há uma solução única para todos os problemas. Pois, a função de custo  $F(x)$  para qualquer algoritmo que apresente desempenho superior a outro na resolução de uma classe de problemas  $X_i$  dentro de um universo  $K$  terá o desempenho compensado na resolução de todos os outros problemas de otimização do espaço  $K - X_i$ . Assim, em média, o custo para encontrar a solução de todos os problemas será o mesmo para qualquer método de solução.

Isso posto, o trabalho proposto visa identificar a possibilidade de criar a arquitetura de uma rede neural de convolução adequada de maneira sistemática, selecionando, randomicamente, os hiperparâmetros que a define, dentre espaço de busca determinado. Para tanto, optou-se por utilizar um Algoritmo Genético (AG), o qual será responsável por explorar com eficiência as soluções possíveis. Nessa abordagem, o AG não é utilizado para o treinamento das redes, apenas para a definição da estrutura da CNN, ou seja, é responsável pela construção da população inicial de indivíduos, nesse contexto representados pelas redes CNNs, bem como pelas operações genéticas, quais sejam: seleção, mutação e cruzamento, para

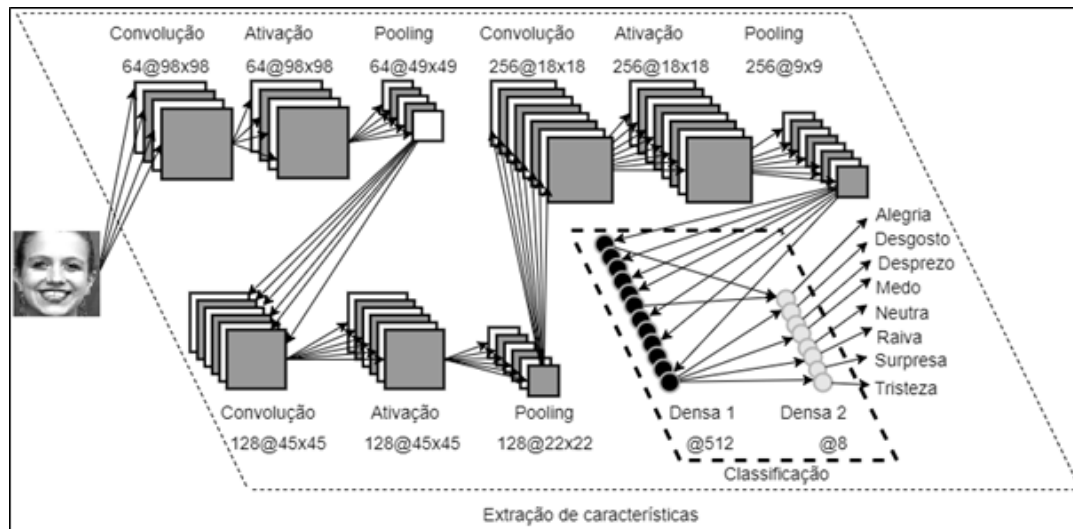


Figure 1: Arquitetura da Rede Deep-Emotive v.1.

que novos indivíduos competitivos sejam gerados a partir da geração anterior e os indivíduos fracos sejam eliminados. A função de aptidão de cada indivíduo é determinada pela métrica de acurácia sob o conjunto de dados de referência, obtida após o processo completo de treinamento de cada CNN. O processo genético termina após um número pré-definido de gerações. Destaca-se ainda que não foram utilizadas outras métricas mais robustas para a função de aptidão visto que o objetivo principal é verificar a viabilidade da criação da arquitetura CNN de forma evolutiva e não a otimização dos resultados. Porém, após encontrar a melhor arquitetura dentro do espaço de busca definido, recomenda-se treinar esse indivíduo e validar sua performance por meio outras métricas, além da definida nessa proposta.

Por fim, cumpre informar que, a definição escolhida é computacionalmente cara, pois é necessário realizar o treinamento de cada indivíduo (CNN) gerado para analisar sua aptidão. Assim, esse trabalho executou os procedimentos em um conjunto de dados relativamente pequeno denominado FER-2013, o mesmo utilizado na etapa da transferência de aprendizado da rede Deep-Emotive v.1, a fim de verificar a capacidade do algoritmo genético para encontrar estruturas de rede eficazes, e, em seguida, transferir as estruturas de melhor classificação, se existentes, para realizar o reconhecimento visual em conjuntos de dados de grande escala.

Nesse contexto, o trabalho está dividido da seguinte forma: a Seção 2 apresenta abordagens semelhantes ao trabalho proposto; a Seção 3 detalha o método proposto para geração das CNNs; a Seção 4 apresenta os resultados obtidos nos experimentos; finalmente, a Seção 5 traz as conclusões e as recomendações para trabalhos futuros.

## 2 TRABALHOS RELACIONADOS

Primeiramente, cabe ressaltar que, em que pese os correlatos apresentados a seguir abordarem estratégias semelhantes, eles não foram executados sob conjuntos de dados específicos para o problema proposto. Atualmente existem diferentes abordagens para a

exploração e ajuste de hiperparâmetros, dentre elas abordagens com Algoritmos genéticos, são métodos meta-heurísticos inspirados pelo processo biológico de evolução, utilizados em problemas de otimização para encontrar o máximo ou mínimo de uma função. AG está incluído no campo de estudo denominado computação evolutiva [11], o qual busca imitar os processos biológicos de reprodução e seleção natural. Possui operadores de seleção, que permite preservar indivíduos mais adaptados, cruzamento, que permite a criação de novos indivíduos a partir dos genes dos indivíduos selecionados e mutação, que permite a alteração autônoma de um indivíduo dentro do espaço de busca.

Em [12], os autores apresentam uma abordagem para criar a arquitetura da rede automaticamente utilizando AG, representando cada estrutura CNN como uma *String* binária de comprimento fixo. O AG é utilizado para inicializar a população inicial e para operações genéticas padrão (seleção, mutação e cruzamento), a fim de gerar indivíduos competitivos e eliminar os fracos. A função de aptidão foi definida pela precisão de reconhecimento de cada indivíduo sob o conjunto de dados e é utilizada para competição entre os indivíduos. Tal abordagem foi executada no conjunto de dados CIFAR10 [13] e obtiveram uma acurácia média de 76.58% e desvio padrão de 0.55.

Na mesma senda, [14] apresenta outra abordagem para criação da arquitetura CNN por meio de AG. Os hiperparâmetros que são definidos pelo AG são a quantidade mapas de recursos (filtros) das camadas de convolução e o tipo de subamostragem que será utilizada, além, claro, das operações genéticas, ignorando outros hiperparâmetros intrínsecos às CNN. O algoritmo proposto é validado no conjunto de dados CIFAR10 e CIFAR100 [13], os quais são amplamente utilizados em problemas de classificação de imagens e obteve um desempenho de 96.78% e 79.47%, respectivamente.

Outra abordagem usando AG é o *NeuroEvolution of Augmenting Topology* (NEAT) [15]. O NEAT aborda a evolução tanto da topologia quanto dos pesos de Redes Neurais Artificiais (RNAs). Ademais, pode-se citar que, a principal diferença consiste no fato de que o

DeepNEAT [16] representa cada gene como uma camada da rede, enquanto o NEAT representa cada gene como um neurônio. O trabalho Cooperative DeepNEAT (CoDeepNEAT) [17] é uma evolução do DeepNeat para evoluir duas subpopulações. Redes Neurais Evolutivas Multinódicas para o Aprendizado Profundo (MENNDL) [18] foram criadas para otimizar os hiperparâmetros de tamanho e quantidade de núcleos da CNN usando algoritmo genético.

### 3 EVOLUINDO REDES NEURAIS CONVOLUCIONAIS

Essa seção detalha o método proposto. Inicialmente, apresenta-se a codificação da arquitetura de rede CNN como um indivíduo. Em seguida, apresenta as operações e o fluxo de execução do algoritmo genético.

#### 3.1 Fluxo do Algoritmo Genético

A Fig. 2 ilustra o fluxo do Algoritmo Genético utilizado na abordagem proposta. Nota-se que foi aplicado uma implementação padrão de AG. Frente ao escasso recurso computacional disponível, optou-se por uma abordagem de micro algoritmo genético, o qual difere do AG tradicional pela quantidade baixa de indivíduos e gerações. Primeiramente, 10 indivíduos da população inicial são inicializados aleatoriamente e cada gene do indivíduo contém um dos valores possíveis apresentados na Tabela 1. Em seguida cada indivíduo tem seu fitness avaliado. A precisão de cada rede CNN treinada é atribuída como medida de adequação a cada indivíduo. A condição de parada é determinada pela quantidade máxima de gerações. Neste trabalho, utilizaram-se 10 gerações, uma vez que não há um ponto ótimo ou subótimo de resolução pré-definido.

Enquanto a condição de parada não for satisfeita, ocorrer a evolução da população atual por meio de operações genéticas, quais sejam: seleção pelo método da roleta, no qual o melhor indivíduo possui maior probabilidade de ser selecionado; cruzamento uniforme, no qual os genes dos indivíduos são cruzados individualmente entre os progenitores, com uma probabilidade de 80% de ocorrer; e mutação, no qual alterar os genes de cada indivíduo independentemente dos genitores, com uma baixa probabilidade (10%) de ocorrer para preservar as boas propriedades de um indivíduo sobrevivente. Os cinco (05) melhores indivíduos e sua correspondente arquitetura CNN são salvos no *Hall of Fame* (HoF). O elitismo ocorre com a troca do indivíduo com menor aptidão da geração anterior pelo melhor indivíduo avaliado na geração corrente. Esta abordagem não garante uma concreta convergência devido às restrições criadas para reduzir o custo computacional, mas trabalha com a aproximação.

#### 3.2 Representação da rede CNN - Indivíduo

Cada indivíduo foi modelado como um vetor de inteiros de cinco (05) posições. Cada posição representa um gene, conforme pode ser observado na Tabela 1. O primeiro gene define a quantidade de camadas de convolução da rede, limitado ao máximo de cinco (05) camadas devido ao custo de processamento. O segundo gene é responsável por criar a quantidade de filtros das camadas de convolução. Esse gene é intrinsecamente depende do gene anterior, pois, é determinado randomicamente tantas vezes a quantidade de

camadas de convolução. O terceiro gene estipula a dimensionalidade de cada filtro. O quarto gene determina qual das técnicas de subamostragem será utilizada. O último gene define a quantidade de neurônios utilizado na primeira camada totalmente conectada localizada na saída da rede.

**Table 1: Representação do fenótipo do indivíduo.**

Gene	Valores
Camadas de convolução	3
Filtros	256
Tamanho do filtro	5x5
Tipo de subamostragem	<i>AveragePooling</i>
Neurônios na camada densa	512

Observa-se que, o genótipo não abrange outros hiperparâmetros intrínsecos da arquitetura de redes CNNs, como por exemplo, função de ativação, responsável pela ativação do neurônio na rede, a qual definimos como PReLU; camadas de *dropout*, que são responsáveis por prevenir o sobre ajuste da rede, o qual optou-se por manter as mesmas definidas na arquitetura Deep-Emotive v.1.; e, por fim, a função de otimização do aprendizado, o *Adaptive Moment Estimation* (Adam) foi selecionada, a qual é computacionalmente eficiente, tem pouca necessidade de memória, invariante ao reescalonamento diagonal de gradientes e é adequado para problemas que são grandes em termos de dados/parâmetros [19].

A Tabela 2 lista os possíveis valores máximos que cada gene pode assumir, ilustrando, assim, o espaço de busca do problema. Verifica-se que foi delimitado a quantidade máxima de camadas de convolução em 5 camadas, devido ao custo computacional. Além disso, foi definido o máximo de 256 filtros para cada camada de convolução. O tamanho máximo dos *kernel* é de 10x10. ademais, definiu-se apenas dois tipos de subamostragem possíveis (*max-pooling* e *averagepooling*). Por fim, a quantidade máxima de 1.024 neurônios na primeira camada totalmente conectada. A segunda camada totalmente conectada possui o número fixo de 7 neurônios, representado cada emoção na saída da rede. Com isso, o espaço de busca definido ficou restrito à combinação de  $5 * 256 * 10 * 2 * 1024$ .

**Table 2: Tabela de valores possíveis para cada gene do indivíduo**

Gene	Valores possíveis
Camadas de convolução	$[1 + 2 + \dots + 5]$
Filtros	$[1 + 2 + \dots + 256]$
Tamanho do filtro	$[(1, 1) + (2, 2) + \dots + (10, 10)]$
Tipo subamostragem	$[MaxPooling, AveragePooling]$
Neurônios na camada densa	$[1 + 2 + \dots + 1.024]$

## 4 EXPERIMENTOS

Nesta seção serão apresentados os resultados dos experimentos realizados. Conforme exposto anteriormente, essa abordagem requer poder computacional relativamente alto, então a estratégia abordada explora estruturas de rede promissoras em um conjunto de dados relativamente pequeno, para, em seguida, se relevante, transferir estas estruturas para um ambiente em grande escala.



Figure 2: Fluxo do Algoritmo Genético da Rede Deep-Emotive v.2.

Por questões de legibilidade e a fim de possibilitar a comparação fidedigna em caso de replicação dos experimentos, todas as imagens do artigo encontram-se disponíveis em <https://github.com/diogenesis/deepemotive>.

#### 4.1 Infraestrutura computacional e estratégia de execução

A abordagem deste trabalho foi implementada em um ambiente com Python 3.9 e foi executado em um sistema operacional Unix (*Ubuntu18.04.5LTS*). Cada arquitetura CNN desenvolvida e avaliada foi implementada usando a estrutura de aprendizado profundo do Framework Tensorflow. Os experimentos foram executados na *GPU TeslaV100 – SXM2*, com 54.8 GB RAM, 8 cores de processamento e *Cuda.v.11.1.TC45506.29190527.0*.

Devido ao custo computacional para calcular a função de aptidão de cada indivíduo, foi necessário estipular um *trade-off* que garantisse a convergência aceitável, considerando o tamanho da população, número de gerações, e o tempo necessário para a evolução. Assim, optou-se por utilizar uma variação da técnica denominada Micro Algoritmo Genético ( $\mu GA$ ), introduzida por [20]. Tal técnica consiste em um AG que evolui uma pequena população de indivíduos, além de que a população é reinicializada segundo um critério de convergência, bem como, não aplicada a etapa de mutação, visto que diversidade dos indivíduos é provida por meio das reinicializações executadas. Para o trabalho proposta, optou-se por utilizar uma população pequena de indivíduos, além de manter a etapa de mutação. A Tabela 3 apresenta o setup do experimento realizado.

Table 3: Tabela de parâmetros da execução do AG

Parâmetros	Valor
Número de gerações	10
Tamanho da população	10
Probabilidade de cruzamento	0.80
Probabilidade de mutação	0.10
Elitismo	1
Épocas de treinamento fitness	50
Hall of Fame	5

#### 4.2 Deep-emotive v.2 e o conjunto de dados FER-2013

Optou-se por utilizar o conjunto de dados FER-2013 pelo baixo desempenho atingido pela arquitetura definida empiricamente em Deep-Emotive v.1. (60.71%). Tal conjunto de dados possui 35.887 imagens, possuem tamanho de 48 x 48 pixels em escala de cinza, porém foram redimensionadas para 32 x 32, a fim de otimizar o processamento. Além disso, as amostras estão distribuídas em: 7.215 imagens de alegria; 436 de desgosto; 4.097 de medo; 4.965 de neutra; 3.995 de raiva; 3.171 de surpresa; e 4.830 de tristeza, demonstrando o desbalanceamento do conjunto de dados. Então, foi necessário a aplicação da técnica de *data-augmentation*, a fim de reduzir os problemas de balanceamento entre as classes. Tal estratégia visa aumentar a diversidade do conjunto de treinamento aplicando transformações aleatórias, como rotação, translação, reflexo, entre outras. Ademais, esse conjunto de dados apresenta imagens com problemas consolidados na área de visão computacional, como oclusão, perspectiva, ruídos, além de imagens que não fazem parte do contexto do tema abordado no trabalho, o que impossibilita obter o



conhecimento necessário para reconhecer a emoção por meio das expressões faciais, conforme pode ser observado na Fig. 3(a), bem como, possui baixa similaridade em relação ao conjunto de dados CK+, Fig. 3(b).

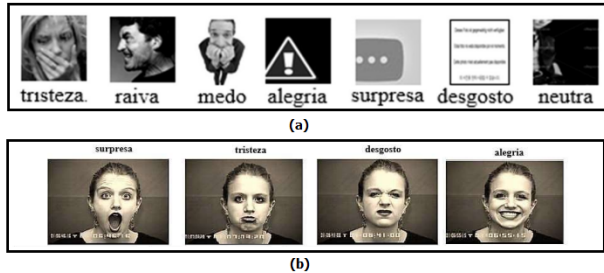


Figure 3: Dissimilaridade e problemas do conjunto de dados FER-2013.

A Fig. 4 apresenta o desempenho dos cinco melhores indivíduos para cada geração sob o conjunto de dados de treinamento. Nota-se que o melhor desempenho alcançado foi na geração 65,90% na quarta geração. No entanto, observa-se uma linearidade entre os melhores. Em sequência, na Fig. 5, observa-se o desempenho dos cinco melhores indivíduos no conjunto de dados de validação. O melhor indivíduo obteve 58,50% de acurácia na sexta geração. Nesse gráfico, nota-se pela linha de tendência um crescimento linear durante as gerações, indicando a evolução do desempenho e início de uma convergência.

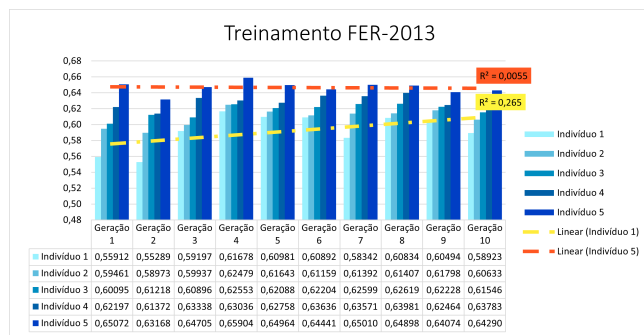


Figure 4: Gráfico de desempenho dos HoF5 no conjunto de dados de treinamento FER-2013

Semelhante ao observado na Fig. 4 e Fig. 5, pode-se verificar um aumento no desempenho de acurácia e redução do desvio padrão no desempenho dos indivíduos por meio do gráfico de médias, ilustrado na Fig. 6, bem como, pelo gráfico de desvio padrão apresentado na Fig. 7, nos quais o melhor indivíduo obteve 64,65% com desvio padrão de 0,0068 no conjunto de treinamento e 57,57% com desvio padrão de 0,0056 no conjunto de validação. Ademais, o fenótipo do melhor indivíduo é apresentado na Tabela 4. Esse foi definido pelo algoritmo genético como uma rede CNN de 3 camadas de convolução, com 22 filtros de tamanho 3x3, 243 filtros de tamanho 7x7 e 216 filtros de tamanho 2x2, respectivamente. O

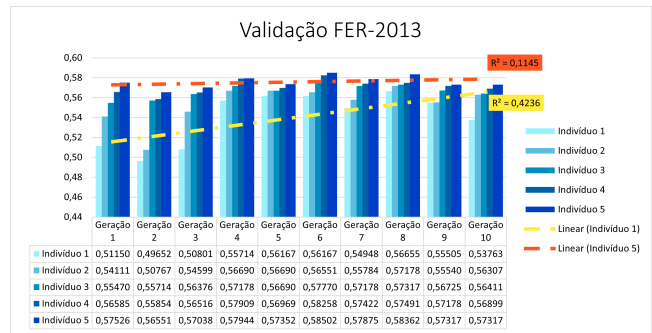


Figure 5: Gráfico de desempenho dos HoF5 no conjunto de dados de validação FER-2013

tipo de subamostragem, identificado pelo algoritmo genético como melhor para o problema, foi o *AveragePooling*. Ainda, o AG definiu a quantidade de 129 neurônios na camada totalmente conectada no final da rede.

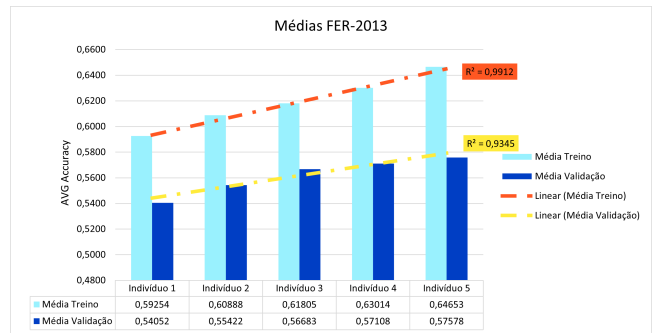


Figure 6: Gráfico de médias do desempenho do HoF5 em cada geração no conjunto de dados FER-2013.

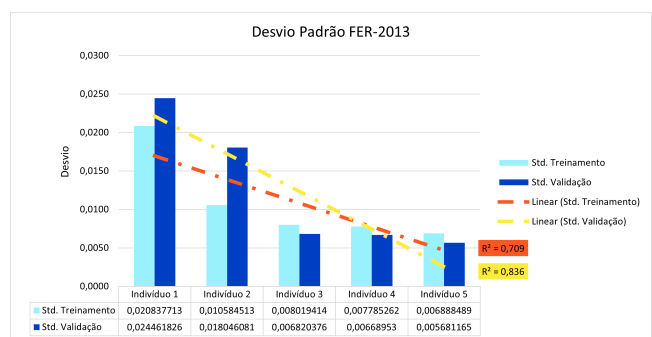


Figure 7: Gráfico do desvio padrão do HoF5 em cada geração no conjunto de dados FER-2013.

Após a aplicação da estratégia proposta, procedeu-se os ajustes finos e treinamento apartado do melhor indivíduo identificado pelo algoritmo genético. Com isso, a estrutura apresentada na Tabela 3, obteve um desempenho de 63,84% no conjunto de dados de treinamento e 62,39% no conjunto de dados de teste. A Fig. 8 apresenta o

Table 4: Fenótipo do melhor indivíduo para o conjunto de dados FER-2013

Camadas de Convolução			Tamanho dos Filtros		
Filtros na camada 1	Filtros na camada 2	Filtros na camada 3	Camada 1	Camada 2	Camada 3
22	243	216	3x3	7x7	2x2
Tipo de Subamostragem			Neurônios		
Pooling 1	Pooling 2	Pooling 3	Camada densa 1		
Average	Average	Average	129		

gráfico do treinamento e validação do melhor indivíduo, treinado separadamente. Esse experimento mostrou a capacidade da estratégia proposta, visto que o resultado encontrado, para o conjunto de dados de referência, supera o da primeira versão da rede Deep-Emotive (60,71%).

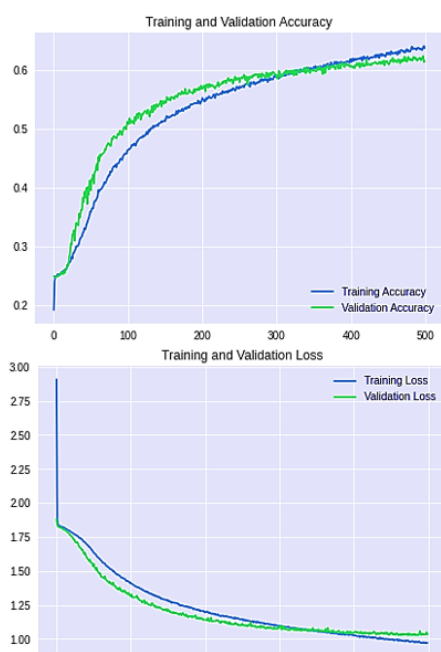


Figure 8: Gráfico do desempenho do melhor indivíduo identificado pelo AG. no conjunto de dados FER-2013.

Cumprir informar que foi realizado um experimento alterando a criação da população inicial, ou seja, ao invés de criar os indivíduos de forma totalmente aleatória, seguindo os parâmetros já apresentados, foi incluído na população inicial um indivíduo representando a arquitetura Deep-Emotive v.1, conforme Fig. 1, com o objetivo de os genes dessa arquitetura gerassem novos indivíduos mais adaptados, no entanto não se obteve desempenho superior ao apresentado.

Em seguida, frente aos resultados encontrados, decidiu-se validar o conjunto de dados FER-2013 em uma arquitetura já consolidada, como por exemplo, a VGG16. Tal arquitetura foi vencedora da competição *ImageNet Large Scale Visual Recognition Challenge* (ILSVR) em 2014 [8], além de ser considerada uma das arquiteturas de modelo de visão computacional no estado-da-arte. A rede tem 16 camadas, gerando cerca de, aproximadamente, 138 milhões de parâmetros treináveis.

Destarte, a fim de utilizar a VGG16 nesse trabalho, alguns ajustes foram necessários nas camadas inicial e na camada final da rede. As imagens foram redimensionadas para o tamanho 224x224 pixels, visto que é o tamanho aceito pela referida arquitetura, além disso, incluiu-se uma camada de saída personalizada para o domínio do problema, ou seja, uma camada totalmente conectada com 7 neurônios representando as emoções. Em seguida, procedeu-se o treinamento da rede VGG no referido conjunto de dados e o desempenho alcançado foi de 53,61% de acurácia. Vale ressaltar que, não foi realizado ajustes finos na rede VGG, além das mencionadas, utilizando-se, assim, a configuração padrão.

### 4.3 Deep-Emotive v.2 e o conjunto de dados CK+

Em sequência, frente aos resultados encontrados, decidiu-se validar a estratégia proposta nesse trabalho no conjunto de dados CK+, utilizado para o treinamento da referida rede na versão 1.0. A Fig. 9 apresenta o gráfico de desempenho no conjunto de treinamento CK+. Observa-se que foi possível atingir uma acurácia de 93,74% na quarta geração. Além disso, o algoritmo apresenta conversão na sexta geração de treinamento. Ainda, as linhas de tendência auxiliam a demonstrar a evolução dos indivíduos durante as gerações. A Fig. 10 apresenta o desempenho no conjunto de dados de validação CK+. Observa-se que foi possível atingir 97,61% de acurácia na quarta geração, bem como, o algoritmo converge na sexta geração.

Na mesma senda, pode-se verificar um aumento na acurácia e redução do desvio padrão do desempenho dos indivíduos por meio do gráfico de médias, ilustrado na Fig. 11, e gráfico de desvio padrão apresentado na Fig. 12, nos quais o melhor indivíduo obteve 92,60% com desvio padrão de 0,0094 no conjunto de treinamento e 96,71% com desvio padrão de 0,0087 no conjunto de validação.

Ademais, o fenótipo do melhor indivíduo para o conjunto de dados CK+ é apresentado na Tabela 5. Esse foi definido pelo algoritmo genético como uma rede CNN de 3 camadas de convolução, com 172 filtros de tamanho 11x11, 48 filtros de tamanho 5x5 e 252 filtros de tamanho 5x5, respectivamente. Nota-se valores próximos aos definidos empiricamente na rede Deep-Emotive v.1. O tipo de subamostragem, identificado pelo algoritmo genético como melhor para o problema, foi o *Average Pooling*, o mesmo da versão 1.0. Ainda, o AG definiu a quantidade de 512 neurônios na camada totalmente conectada no final da rede, semelhante a versão 1.0. A Fig. 13 apresenta o gráfico do treinamento e validação do melhor indivíduo para o conjunto de dados CK+, treinado separadamente. O experimento, novamente, mostrou a capacidade da estratégia proposta, visto que o resultado encontrado, para o conjunto de dados de referência, supera o da primeira versão da rede Deep-Emotive,

atingindo o valor de 99,32% contra 98,71% da primeira versão no conjunto de validação.

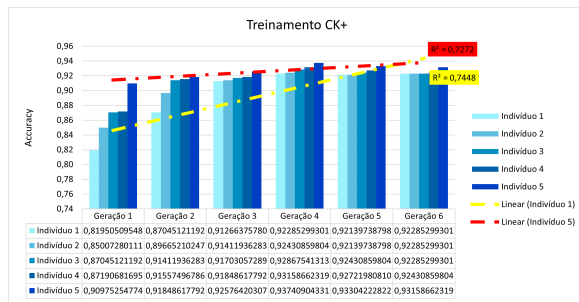


Figure 9: Gráfico de desempenho dos HoF5 no conjunto de dados de treinamento CK+

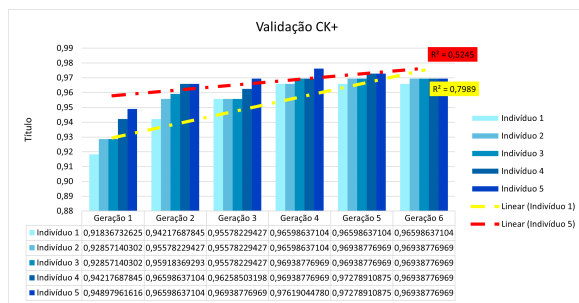


Figure 10: Gráfico de desempenho dos HoF5 no conjunto de validação CK+

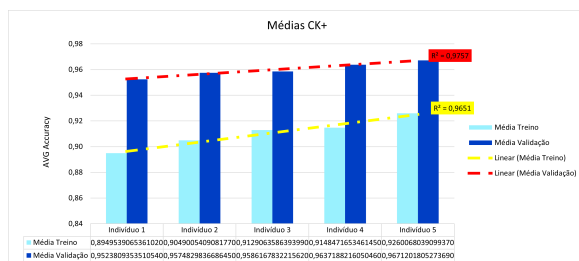


Figure 11: Gráfico de médias do desempenho do HoF5 no conjunto de dados CK+.

## 5 CONCLUSÃO

Este artigo apresentou uma abordagem meta-heurística para a rede Deep-Emotive v.1. com a implementação de arquiteturas de redes neurais convolucionais sistematizadas utilizando micro algoritmos genéticos. A criação de CNNs depende de vários hiperparâmetros, os quais impactam significativamente os resultados. O AG permite explorar o espaço de busca desses hiperparâmetros visando de encontrar a melhor arquitetura para solução do problema. Cabe

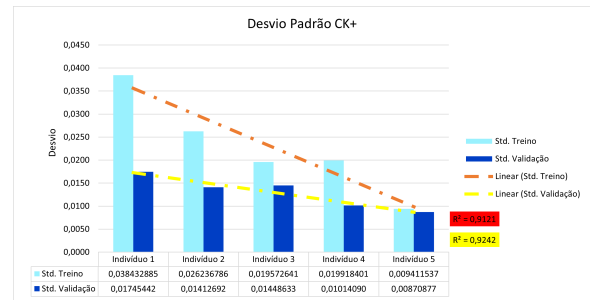


Figure 12: Gráfico de desvio padrão do HoF5 no conjunto de dados CK+.

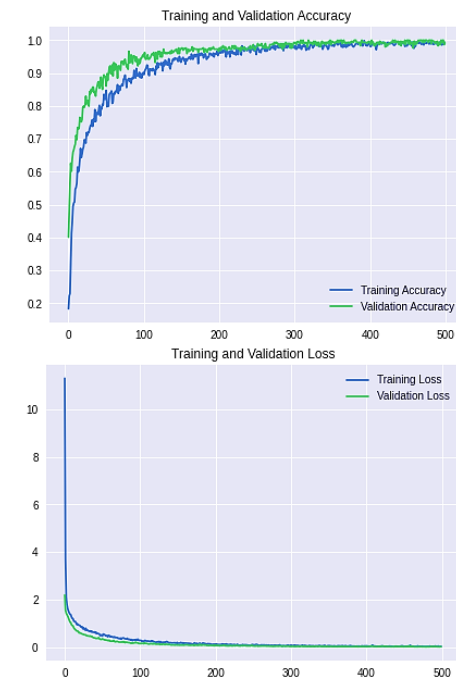


Figure 13: Gráfico do desempenho do melhor indivíduo identificado pelo AG no conjunto de dados CK+.

ressaltar que, em que pese os trabalhos correlatos abordarem estratégias semelhantes, eles não foram executados sob conjuntos de dados específicos para o problema proposto. Nesse trabalho, o espaço de busca é formado pela quantidade de camadas de convolução e camadas totalmente conectadas, assim como o número de filtros para cada camada, dimensão dos filtros, tipo de subamostragem e quantidade de neurônio na camada totalmente conectada no final da rede.

Os experimentos foram realizados sob dois conjuntos de dados distintos, a fim de classificar emoções por meio de imagens de expressões faciais. O conjunto de dados foi o FER-2013 [7], pois esse teve o pior desempenho (60,71%) na primeira versão da rede Deep-Emotive, a qual teve sua arquitetura definida empiricamente. Os experimentos apresentaram o desempenho superior ao empirismo,

Table 5: Fenótipo do melhor indivíduo para o conjunto de dados CK+

Camadas de Convolução			Tamanho dos Filtros		
Filtros na camada 1	Filtros na camada 2	Filtros na Camada 3	Camada 1	Camada 2	Camada 3
172	48	252	11x11	5x5	5x5
Tipo de subamostragem			Neurônios		
Pooling 1	Pooling 2	Pooling 3	Camada densa 1		
Average	Average	Average	512		

mostrando um aumento de 1,68%, ou seja, atingiu 63,84% no conjunto de dados de treino e 62,39% no conjunto de dados de validação, após os ajustes finos e treinamento o melhor indivíduo, superando o desempenho da primeira versão da rede. Cabe ressaltar que, conforme apresentado na Fig. 3(a), esse conjunto dados apresenta imagens com problemas consolidados na área de visão computacional, como oclusão, perspectiva, ruídos, além de imagens possuírem baixa similaridade ao contexto de reconhecer a emoção por meio das expressões faciais, o que traz complexidade ao problema. Além disso, utilizou-se uma arquitetura já consolidada (VGG16), em sua configuração padrão, para validação do conjunto de dados FER-2013 e o desempenho alcançado foi de 53.61% de acurácia para o referido conjunto de dados. Ademais, o limitado recurso computacional para aplicação da abordagem proposta, em que se fez necessário utilizar uma abordagem de micro algoritmo genético, bem como uma limitação do espaço de busca do AG, pode ter influenciado para o baixo desempenho nesse conjunto de dados. Tais fatos apontam que, o motivo do baixo resultado não está na estratégia abordada, mas sim, aparentemente, na natureza dos dados, pois, a abordagem proporcionou um aumento de 1,68% em relação à estrutura definida empiricamente, mostrando que o potencial da estratégia.

O segundo conjunto de dados utilizado foi o Cohn-Kanade AU, versão 2 (CK+) [5]. Os experimentos apresentaram o desempenho superior ao empirismo, mostrando um aumento de 0.61%, ou seja, atingiu 99,32% no conjunto de dados de validação, após os ajustes finos e treinamento o melhor indivíduo, superando o desempenho da primeira versão da rede (98,71%) em que a arquitetura foi definida empiricamente. Além disso, esse conjunto de dados possui imagens criadas especificamente para o problema proposto, conforme Fig. 3(b), não apresentando os mesmos problemas da área de visão computacional do outro conjunto de dados.

Por fim, como trabalhos futuros, recomenda-se a extrapolação para AGs com maior espaço de busca, o que requer maior poder computacional para condução dos experimentos; inclusão de mais parâmetros da rede como funções de ativação de cada camada etc. Vislumbra-se também testar outras metaheurísticas como Particle Swarm Optimization (PSO) e Ant Colony Optimization (ACO) para realizar a hiperparametrização. Além disso, aplicar testes estatísticos para verificar o desempenho das abordagens implementadas..

## AGRADECIMENTOS

Este trabalho foi parcialmente financiado pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) - DT-308334/2020; PROGRAMA DE APOIO A NÚCLEOS EMERGENTES - PRONEM - FAPESPA/CNPq (Termo de Outorga n° 045/2021); e Acordo de Cooperação Técnica N° 02/2021 (Processo N° 38328/2020 - TJ/MA).

## REFERÊNCIAS

- [1] Yann LeCun, Yoshua Bengio, Geoffrey Hinton, et al. Deep learning. *nature* 521 (7553), 436-444. *Google Scholar Google Scholar Cross Ref Cross Ref*, 2015.
- [2] Wanderson Gonçalves e Gonçalves, Marcelo Henrique de Paula dos Santos, Fábio Manoel França Lobato, Ândrea Ribeiro-dos Santos, and Gilderlanio Santana de Araújo. Deep learning in gastric tissue diseases: a systematic review. *BMJ Open Gastroenterology*, 7(1), 2020. doi: 10.1136/bmjgast-2019-000371. URL <https://bmjopen gastro.bmj.com/content/7/1/e000371>.
- [3] Gustavo Almeida, Virgílio Soares, Marcio Ponte, and Fabio Lobato. Reconhecimento de plantas comestíveis não convencionais: uma análise de métodos de classificação aplicados à visão computacional. In *Anais do X Workshop de Computação Aplicada a Gestão do Meio Ambiente e Recursos Naturais*, pages 67-76. SBC, 2019.
- [4] Diógenes Ademir Domingos. Deep-emotive: protótipo de sistema para reconhecimento de expressões faciais utilizando aprendizado profundo. 2017.
- [5] Patrick Lucey, Jeffrey F Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2010 *IEEE Computer Society Conference on*, pages 94-101. IEEE, 2010.
- [6] Michael Lyons, Shigeru Akamatsu, Miyuki Kamachi, and Jiro Gyoba. Coding facial expressions with gabor wavelets. In *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, pages 200-205. IEEE, 1998.
- [7] Ian J Goodfellow, Dumitru Erhan, Pierre Luc Carrier, Aaron Courville, Mehdi Mirza, Ben Hamner, Will Cukierski, Yichuan Tang, David Thaler, Dong-Hyun Lee, et al. Challenges in representation learning: A report on three machine learning contests. In *International Conference on Neural Information Processing*, pages 117-124. Springer, 2013.
- [8] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [9] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1-9, 2015.
- [10] David H Wolpert and William G Macready. No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1):67-82, 1997.
- [11] Kenneth E Kinnear, William B Langdon, Lee Spector, Peter J Angeline, and Una-May O'Reilly. *Advances in genetic programming*, volume 3. MIT press, 1994.
- [12] Lingxi Xie and Alan Yuille. Genetic cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1379-1388, 2017.
- [13] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.
- [14] Yanan Sun, Bing Xue, Mengjie Zhang, Gary G Yen, and Jiancheng Lv. Automatically designing cnn architectures using the genetic algorithm for image classification. *IEEE transactions on cybernetics*, 50(9):3840-3854, 2020.
- [15] Kenneth O Stanley and Risto Miikkulainen. Evolving neural networks through augmenting topologies. *Evolutionary computation*, 10(2):99-127, 2002.
- [16] Risto Miikkulainen, Jason Liang, Elliot Meyerson, Aditya Rawal, Daniel Fink, Olivier Francon, Bala Raju, Hormoz Shahrzad, Arshak Navruzyan, Nigel Duffy, et al. Evolving deep neural networks. In *Artificial Intelligence in the Age of Neural Networks and Brain Computing*, pages 293-312. Elsevier, 2019.
- [17] Jonas da Silveira Bohrer, Bruno Iochins Grisci, and Marcio Dorn. Neuroevolution of neural network architectures using codeepneat and keras. *arXiv preprint arXiv:2002.04634*, 2020.
- [18] Steven R Young, Derek C Rose, Travis Johnston, William T Heller, Thomas P Karnowski, Thomas E Potok, Robert M Patton, Gabriel Perdue, and Jonathan Miller. Evolving deep networks using hpc. In *Proceedings of the Machine Learning on HPC Environments*, pages 1-7. 2017.
- [19] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [20] Kalmanje Krishnakumar. Micro-genetic algorithms for stationary and non-stationary function optimization. In *Intelligent control and adaptive systems*, volume 1196, pages 289-296. International Society for Optics and Photonics, 1990.