Redes Neurais Profundas Triplet Aplicadas à Classificação de Sinais em Interfaces Cérebro-Computador

Pedro R. A. S. Bassi, Willian Rampazzo e Romis Attux

***Resumo*— Neste trabalho, propõe-se uma abordagem para classificação de sinais de EEG em interfaces cérebro-computador que tem por base uma rede neural profunda triplet. A rede é testada em uma base de dados com dez usuários, em uma configuração que exclui os dados do usuário avaliado do processo de treinamento da rede. Os resultados são promissores, embora seja necessário realizar novas investigações no sentido de analisar a perspectiva de aproveitamento adicional das relações não-lineares presentes nos dados.**

***Palavras-Chave*— Interfaces cérebro-computador, redes neurais profundas, SSVEP, redes triplet.**

***Abstract*— This work proposes an approach to EEG signal classification in brain-computer interfaces based on triplet deep neural networks. The network is tested over a database with ten users, in a setup that excludes the evaluated user's data from the network training process. The results are promising, although further investigation is necessary to verify if there are additional nonlinear relationships in the data to be explored.**

***Keywords*— Brain-computer interfaces, deep learning, SSVEP, triplet neural networks.**

I. Introdução

Neste artigo, é proposta uma nova abordagem para classificação de sinais cerebrais obtidos por eletroencefalografia (EEG) no âmbito de interfaces cérebro-computador (BCIs, do inglês *brain-computer interfaces*) baseadas em potenciais visualmente evocados em estado estacionário (SSVEP, do inglês *steady state visually evoked potentials*). O cerne da abordagem é o uso de redes neurais artificiais profundas, mais especificamente as do tipo triplet [1], junto a imagens compostas de espectrogramas dos sinais cerebrais. O projeto de BCIs é um tema de grande relevância em domínios que vão de tecnologias assistivas até o projeto de jogos de computador [2].

O modus operandi utilizado foi transformar o problema de classificação de sinais SSVEP em um problema de classificação de imagens, para então utilizar uma arquitetura profunda, estado-da-arte, para resolvê-lo. As imagens foram geradas usando a transformada de Fourier de tempo curto (STFT, do *inglês short-time Fourier transform*), criando uma base de dados formada por espectrogramas. Elas, então, foram submetidas a um pré-processamento, como a aplicação de um filtro excluindo regiões que não continham frequências de interesse, ou suas primeiras harmônicas superiores.

Redes neurais triplet foram propostas em [1], tendo por fundamento uma arquitetura, geralmente profunda e convolucional, concebida para realizar um mapeamento das imagens de entrada em um espaço euclidiano de pequena dimensão na saída, no qual imagens da mesma classe são mapeadas de forma a ficarem próximas e as de classes diferentes a ficarem distantes. Isto é realizado minimizando-se uma métrica denominada função de custo triplet.

As redes foram testadas sobre uma base de dados construída pelo grupo de pesquisa, numa configuração com duas frequências de interesse. Os resultados mostram que a rede tem um desempenho promissor, embora pareça haver espaço para aprimoramento da solução. Além disso, consideramos que o uso de redes profundas aplicadas à BCI pode ser um primeiro passo na obtenção de abordagens robustas para múltiplos conjuntos de dados e abordagens com transferência de conhecimento (*transfer learning*).

Este artigo está dividido da seguinte forma: na seção II, apresentaremos o paradigma SSVEP; na seção III, faremos uma breve explicação sobre redes neurais e mostraremos a arquitetura da rede neural utilizada; na seção IV, será discutida a questão de como aplicamos uma rede profunda triplet em SSVEP; na seção V, serão expostos e analisados os resultados obtidos. Por fim, será feita uma discussão sobre os estudos realizados, na seção VI.

II. Potenciais Visualmente Evocados em Regime Estacionário (SSVEP)

BCIs possibilitam aos seus usuários uma nova via de comunicação, diferente das vias biologicamente convenci-

Pedro R. A. S. Bassi, Faculdade de Engenharia Elétrica e de Computação, Universidade de Campinas (UNICAMP), e-mail: p157007@dac.unicamp.br; Willian Rampazzo, Faculdade de Engenharia Elétrica e de Computação, Universidade de Campinas (UNICAMP) / Brazilian Institute of Neuroscience and Neurotechnology (BRAINN), Campinas, São Paulo, Brasil, e-mail: willianr@dca.fee.unicamp.br; Romis Attux, Faculdade de Engenharia Elétrica e de Computação, Universidade de Campinas (UNICAMP) / Brazilian Institute of Neuroscience and Neurotechnology (BRAINN), Campinas, São Paulo, Brasil, e-mail: attux@dca.fee.unicamp.br. Este trabalho foi financiado pelo PIBIC/SAE/UNICAMP, pela Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) (proc. 2018/04100-3) e pelo CNPq (proc. 305621/2015-7). O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nivel Superior - Brasil (CAPES) - Código de Financiamento 001.

onais [2]. Para tal, podem ser utilizados diferentes paradigmas [3], como o baseado em imagética motora, o fundamentado no potencial P300 e o que utiliza potenciais visualmente evocados em regime estacionário (SSVEP). Neste trabalho, optamos pelo último, que apresenta um bom nível de robustez e uma linha de projeto simples e direta.

Um sistema BCI baseado em SSVEP requer um dispositivo de estimulação visual, como um monitor ou uma matriz de LEDs, por exemplo, para exibir um padrão oscilando nas frequências desejadas (geralmente não ultrapassando 30Hz)[3]. O uso de diferentes padrões oscilando no dispositivo, cada um em uma frequência, compõe o conjunto de comandos que um usuário pode transmitir ao sistema.

O ato de um usuário concentrar sua atenção em um dos estímulos visuais possibilita a detecção, através de análise dos sinais elétricos no escalpo, de oscilações na região do córtex visual com a mesma frequência do estímulo visual ou em suas harmônicas, tendo estas oscilações uma relação sinal-ruído que pode ser considerada boa [4].

Tipicamente, o processo de detecção/tomada de decisão é realizado a partir de uma representação no domínio da frequência, na qual atua um classificador. Um bom desempenho desse dispositivo é fundamental para o sucesso do projeto da BCI, ou seja, para o efetivo aproveitamento da informação subjacente aos sinais medidos.

Trabalhos recentes, que abordam a aplicação de redes neurais profundas ao problema de classificação de sinais SSVEP, utilizam arquiteturas convolucionais e entropia cruzada como função de custo. Em [5] e [6] os autores demonstram que é possível construir um teclado virtual treinando redes neurais convolucionais profundas e considerando um dos sujeitos como o de teste. Mas, até o momento, não há na literatura um estudo de classificação de sinais SSVEP utilizando redes neurais profundas triplet.

A. Dados de SSVEP utilizados

Em um experimento de SSVEP aprovado pelo Comitê de Ética da UNICAMP (CAAE 58592916.9.1001.5404), foram registrados estímulos visuais de 10 voluntários saudáveis. A interface de estimulação apresentava dois quadrados de 3,8cm em padrão xadrez oscilando nas frequências de 12Hz (esquerda) e 15Hz (direita) em um monitor com taxa de atualização de 60Hz. No protocolo definido, cada voluntário participou de 8 sessões de 12 segundos para cada uma das frequências.

Os sinais foram registrados por EEG e amostrados a 256Hz, utilizando um total de 16 eletrodos organizados no escalpo segundo o sistema internacional 10-10, privilegiando a região do córtex visual, conforme a Figura 1. Cada sessão de 12 segundos contém 3072 valores. Maiores detalhes sobre a aquisição dos dados estão disponíveis [7].

III. Redes neurais artificiais

Redes neurais artificiais (NNs, do inglês *neural networks*) e, mais recentemente, redes neurais artificiais profundas (DNNs, do inglês *deep neural networks*) têm obtido resultados expressivos em problemas de reconhecimento de padrões. O número de parâmetros das NNs tem crescido seguindo a capacidade computacional disponível e de modo proporcional à lei de Moore: a quantidade de neurônios tem dobrado a cada 2 anos [9]. Desta forma, cresce continuamente o escopo de problemas que podem ser abordados por DNNs.

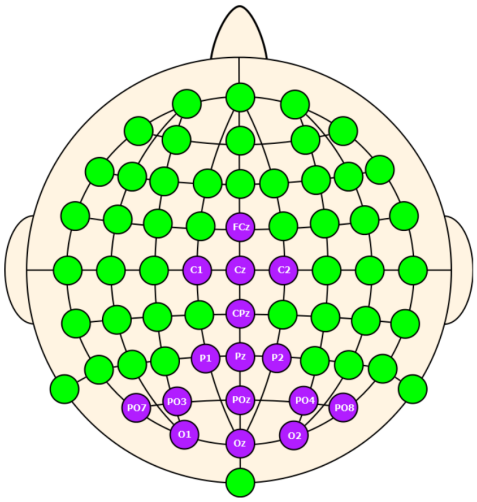

Fig. 1. Disposição dos eletrodos para aquisição dos sinais. Imagem extraída de [8].

A. Visão geral sobre redes neurais artificiais

Uma NN é uma estrutura altamente paralelizada, formada por unidades básicas de processamento simples interconectadas. Essas unidades, inspiradas no neurônio biológico, são chamadas neurônios artificiais [10]. Seu modelo mais comum opera realizando uma combinação linear das entradas, definida por um conjunto de pesos sinápticos. O resultado desta operação passa por uma função de ativação, monovariável e geralmente não-linear, como, por exemplo, a função unidade linear retificadora (ReLU, do inglês *rectified linear unit* - $y(x) = \max(0, x)$). A saída da função de ativação é a saída do neurônio e pode servir de entrada a outros neurônios da rede.

Uma NN do tipo *feedforward* tem sua estrutura definida por uma camada de entrada, uma ou mais camadas intermediárias e uma camada de saída, sem a presença de laços de realimentação. Uma NN é considerada profunda quando possui um número de camadas considerado "elevado", apesar de não haver um consenso sobre quantas camadas são necessárias para tal [9].

Dentre as redes utilizadas em problemas envolvendo imagens, as redes convolucionais vêm se destacando [9]. Estas redes possuem camadas que realizam operações de convolução discreta em um ou mais canais, ou seja, realizam convoluções em paralelo, com parâmetros (filtros) diferentes, sobre as entradas. As saídas das operações de convolução passam por uma função de ativação e podem sofrer *pooling*.

O processo de treinamento de uma NN consiste em otimizar os parâmetros livres da rede (pesos sinápticos,

filtros) de modo a diminuir o erro gerado pela comparação da saída esperada com a saída gerada pela rede. Os dados usados durante o treinamento de uma NN são divididos em três conjuntos: de treinamento, de validação e de teste. Para que se tenha melhor generalização, são tipicamente utilizadas técnicas de regularização. Exemplos de técnicas de regularização são o *dropout*, o *max pooling* e os métodos baseados na penalização da norma de um conjunto de pesos [9].

B. Redes Triplet

As redes triplet [1], têm um funcionamento semelhante ao de redes siamesas [11]. Seu principal uso tem sido reconhecimento facial e são o estado-da-arte na base de dados de rostos LFW (*Labeled Faces in the Wild*), com acurácia de 99.63% [1].

O funcionamento das redes triplet envolve mapear suas entradas em um espaço vetorial euclidiano de dimensão relativamente pequena, no qual entradas que sejam da mesma classe, como fotos de uma mesma pessoa em reconhecimento facial, sejam representadas em pontos próximos, enquanto entradas de classes diferentes estejam separadas por uma margem mínima. Desta forma, o problema de classificação de imagens se resume a calcular a distância euclidiana entre as saídas da rede para entradas diferentes e verificar, assim, se elas pertencem à mesma classe.

Para o mapeamento descrito acima estas redes minimizam a função custo de mesmo nome das redes (triplet) [1]. Para seu cálculo, uma tripla de entradas (ou triplet) é enviada à rede, contendo uma amostra âncora, x_a, uma amostra positiva (da mesma classe da âncora), x_p, e uma amostra negativa (de uma classe diferente da âncora), x_n. A rede gera o mapeamento para essas três entradas, criando $f(x_a)$, $f(x_p)$ e $f(x_n)$, respectivamente. O erro, L, para as três entradas é calculado segundo a Equação 1, que representa a função custo triplet.

$$L = ||f(x_a) - f(x_p)||_2^2 - ||f(x_a) - f(x_n)||_2^2 + \alpha \qquad (1)$$

Ao minimizar a função custo triplet, minimiza-se a distância euclidiana entre os mapeamentos $f(x_a)$ e $f(x_p)$, ao mesmo tempo que se reforça a margem α (um hiperparâmetro da rede) entre esses mapeamentos e $f(x_n)$.

Para o treinamento em lotes, enviam-se diversas triplas para a rede contendo pelo menos uma âncora de cada classe do problema em cada lote. O custo é calculado para cada tripla e somado para formar o erro do lote.

O erro calculado é retropropagado pelo algoritmo *back-propagation* e os parâmetros da rede são ajustados segundo algoritmo de otimização, como, por exemplo, o Adam [12], de forma a minimizar o erro.

Para gerar o mapeamento é comum o uso de DNNs, convolucionais e regularizadas, seguidas por uma normalização L2 de suas saídas, para garantir que $||f(x)||_2 = 1$.

C. Arquitetura Utilizada

Neste trabalho, adotamos a arquitetura de DNN triplet ilustrada na Figura 2 para resolver o problema de classificação dos sinais SSVEP. A arquitetura é composta por 4 camadas convolucionais 2D e 3 camadas densas, ou totalmente conectadas. Após a função de ativação da última camada densa é aplicada uma normalização L2. Desta forma, a DNN projeta suas imagens de entrada em um espaço euclidiano de dimensão 64. Sendo a DNN uma rede triplet, seu treinamento se dá com triplas de entradas (âncora, positiva e negativa) e com a minimização da função de custo triplet.

Um dos fatores que afetam o desempenho da abordagem triplet é o forte uso de regularização na rede. Na arquitetura avaliada temos *dropout* (50%), normalização do lote e *max pooling*. Isto se mostrou essencial para treinar a rede a partir dos dados de alguns indivíduos e obter um bom desempenho ao analisar informações de indivíduos que não foram usadas no treinamento. O uso de interpolações do tipo vizinho mais próximo, responsáveis por dobrar o tamanho das suas entradas, é necessário para contrabalancear o efeito da redução dos tamanhos das imagens nas operações de convolução e *pooling*, já que o modelo é uma rede profunda e trabalha com imagens pequenas.

Após o treinamento, para classificar uma nova imagem obtemos a saída (mapeamento) da rede para esta nova imagem e para 10 amostras aleatórias de cada classe, retiradas do conjunto de treinamento. Medimos então a distância euclidiana quadrática média entre todos os mapeamentos gerados, repetindo a operação para cada classe. O menor valor de distância quadrática média obtido entre o mapeamento da imagem de entrada e o mapeamento das imagens de outras classes corresponde à classe a qual a imagem de entrada pertence.

IV. Classificação de SSVEP com Redes Neurais

Com o intuito de aproveitar o bom desempenho de redes convolucionais no problema de classificação de imagens, os sinais de SSVEP foram transformados em imagens.

No processo de geração das imagens, os sinais SSVEP passaram por um janelamento, sendo os 12 segundos de uma sessão divididos em janelas de 3 segundos com deslocamento de 1 segundo, gerando 10 janelas por sessão. As janelas foram então transformadas em espectrogramas usando a STFT com janela de 256 amostras por segmento e deslocamento de 128 amostras. Por fim, o módulo da saída da STFT foi convertido para decibéis e transformados em imagens, estando seus valores entre 0 e 255, representando pixeis.

Nesta abordagem são consideradas também as frequências que estejam entre a faixa de 1Hz acima e 1Hz abaixo das frequências dos estímulos visuais (12Hz e 15Hz). Foram avaliados o uso das frequências dos estímulos visuais assim como a inclusão de suas primeiras harmônicas superiores. As imagens geradas sem as primeiras harmônicas possuem as frequências de 11, 12 e 13Hz, bem como 14, 15 e 16Hz. Adicionalmente, as imagens com a primeira harmônica superior possuem também as frequências de 23, 24 e 25Hz, bem como 29, 30 e 31Hz. As imagens finais geradas estão em escala de cinza e têm tamanho de 12 por

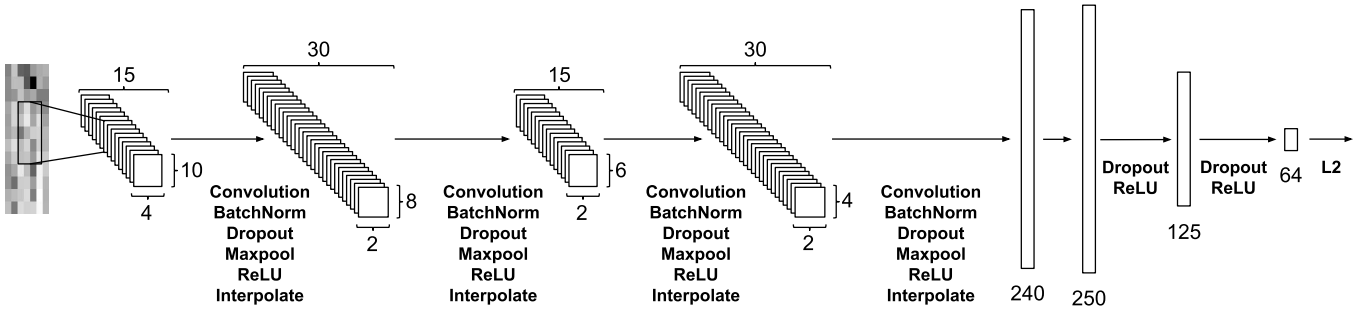

Fig. 2. Proposta de arquitetura DNN para mapeamento no espaço euclidiano de amostras no formato de imagem de espectrograma nas frequências fundamentais mais uma harmônica superior.

7 caso incluam as harmônicas e 6 por 7 caso contrário. Um exemplo de uma amostra com a primeira harmônica superior está ilustrado na Figura 3.

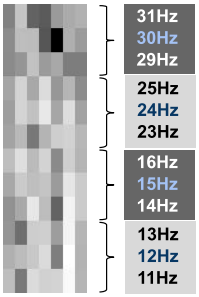

Fig. 3. Exemplo de imagem gerada, referente a um estímulo de 12Hz com sua harmônica em 24Hz, no sujeito 8, eletrodo Oz, sessão 1 e janela 2.

A base de dados de espectrogramas foi dividida em três conjuntos: treinamento, validação e teste. O conjunto de teste contém todos os dados de um único sujeito, dentre os 10. Os dados dos outros 9 sujeitos foram então embaralhados e separados aleatoriamente entre os conjuntos de treinamento e validação, com o conjunto de treinamento contendo 70% destes dados e o conjunto de validação, 30%. A decisão de separar os conjuntos dessa forma tem por objetivo fazer com que a rede generalize melhor ao avaliar dados vindos de novos sujeitos, que não participaram do treinamento.

Para o treinamento da rede triplet são necessárias triplas de dados, formadas por uma âncora, um positivo (da mesma classe da âncora) e um negativo (de classe diferente da âncora). Eles são obtidos do conjunto de treinamento e, para gerar cada tripla, primeiro decide-se, aleatoriamente, por uma âncora e depois são escolhidos, também aleatoriamente, o elemento positivo e o negativo. Em cada época, todas as imagens do conjunto de treinamento são usadas como âncora uma única vez. Os testes efetuados com outras formas de gerar as triplas, como as descritas em [1], não resultaram na melhora do desempenho ou na redução do tempo de treinamento para o problema tratado.

Todas redes foram treinadas por 3000 épocas, com *mini-batches* de 1024 triplas, função custo triplet, descrita na seção III-B, com uma margem, α, de 0,8. Utilizamos o otimizador Adam, com taxa de aprendizado $0,001$. Todo treinamento foi feito em duas GPUs NVidia GTX 1080 e foi utilizada a biblioteca PyTorch para sua realização. O treinamento de uma rede nessas condições levou em média 1 hora e 40 minutos. Para cada época, foi considerado o erro no conjunto de validação, calculado também a partir da função custo triplet com margem de 0,8, de modo a escolher a rede que minimizasse esse erro. Por fim, para testar as redes, foi utilizado o método de classificação descrito na seção III-C, baseado nas distâncias euclidianas entre a saída da rede para uma imagem do conjunto de teste e saídas para amostras de imagens de cada uma das classes, retiradas do conjunto de treinamento.

V. Resultados

Foi treinada uma rede para cada um dos sujeitos, considerado como sujeito de teste, enquanto os dados dos outros 9 sujeitos foram usados como conjunto de treinamento e validação. Foram feitos testes utilizando apenas dados do eletrodo Oz e dos eletrodos O1, O2, Oz e POz, os quais privilegiam a região do córtex visual. Redes geradas por testes com outros conjuntos de eletrodos obtiveram um desempenho inferior aos testados neste trabalho. Foram, também, testadas imagens geradas com e sem a primeira harmônica superior das frequências dos estímulos visuais.

As redes foram treinadas para classificar os sinais SSVEP gerados com estímulos visuais nas frequências de 12Hz e 15Hz e os resultados de classificação foram comparados aos resultados de uma Máquina de Vetor-Suporte (SVM, *Suport-Vector Machine*) linear para a mesma entrada das DNNs.

Os resultados, em termos de taxa de acerto da classificação (acurácia) no conjunto de testes, assim como a média e o desvio padrão, usando 4 eletrodos, estão consolidados na Figura 4 e, usando 1 eletrodo, na Figura 5.

Observamos que os melhores resultados obtidos estão concentrados nos dados fornecidos apenas pelo eletrodo Oz, com uma média entre todos sujeitos de 78,5% para espectrogramas que incluem as harmônicas e 77,25% para espectrogramas que não contém as harmônicas, contra uma média entre todos sujeitos de 68,84% e 70,07%, respectivamente, utilizando os eletrodos O1, O2, Oz e POz. É também visível que, dependendo do sujeito de teste, o desempenho da rede difere consideravelmente.

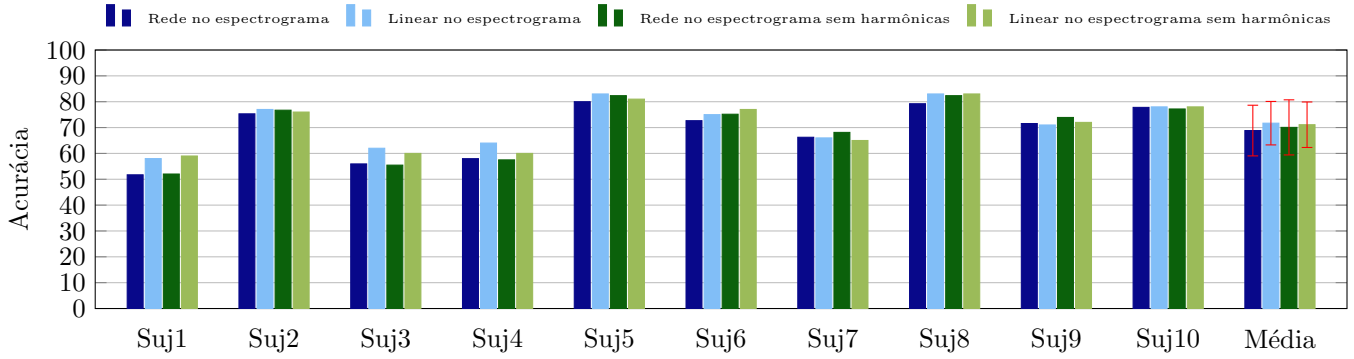

Fig. 4. Acurácia da DCNN Triplet e da SVM linear no espectrograma com e sem uma harmônica, dados dos eletrodos O1, O2, Oz, POz.

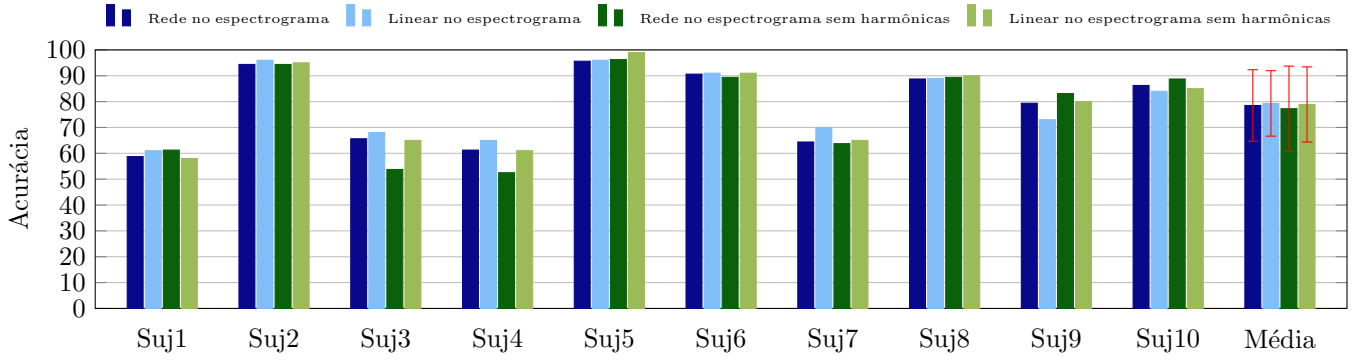

Fig. 5. Acurácia da DCNN Triplet e da SVM linear no espectrograma com e sem uma harmônica usando informação do eletrodo Oz.

Os resultados podem ser interpretados como sendo bastante promissores, uma vez que indicam potencial de aplicação com um número reduzido de eletrodos e em uma condição em que os dados do próprio usuário não constam no treinamento. Como ressalva, deve-se apontar que a diferença de desempenho não é expressiva em relação a um classificador linear, o que pode significar que há espaço para melhorias na metodologia.

VI. Conclusões

Este trabalho apresentou uma metodologia para classificação de sinais em BCIs baseadas em SSVEP, utilizando redes neurais profundas triplet. A proposta busca explorar o potencial dessa rede que, até onde pudemos verificar, não havia sido empregada neste contexto, para obter um projeto de interface que opere sem a necessidade de treinamento "indivíduo a indivíduo". Pelos resultados obtidos, foi possível constatar que a rede, na implementação atual, é capaz de obter um desempenho que pode ser considerado bom (mais de 70% de acurácia para a maior parte dos indivíduos). Por outro lado, o desempenho próximo ao de uma estrutura linear requer novas investigações, no sentido de avaliar se mudanças na arquitetura e/ou no treinamento podem levar a uma exploração mais ampla da informação subjacente aos dados.

Referências

[1] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.

[2] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain–computer interfaces for communication and control," *Clinical neurophysiology*, vol. 113, no. 6, pp. 767–791, 2002.

[3] F. Beverina, G. Palmas, S. Silvoni, F. Piccione, S. Giove, *et al.*, "User adaptive bcis: Ssvep and p300 based interfaces.," *PsychNology Journal*, vol. 1, no. 4, pp. 331–354, 2003.

[4] N. Galloway, "Human brain electrophysiology: Evoked potentials and evoked magnetic fields in science and medicine," *The British journal of ophthalmology*, vol. 74, no. 4, p. 255, 1990.

[5] T.-H. Nguyen and W.-Y. Chung, "A single-channel ssvep-based bci speller using deep learning," *IEEE Access*, vol. 7, pp. 1752–1763, 2019.

[6] J. J. Podmore, T. P. Breckon, N. K. N. Aznan, and J. D. Connolly, "On the relative contribution of deep convolutional neural networks for ssvep-based bio-signal decoding in bci speller applications," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, pp. 611–618, April 2019.

[7] S. N. Carvalho, T. B. Costa, L. F. Uribe, D. C. Soriano, S. R. Almeida, L. L. Min, G. Castellano, and R. Attux, "Effect of the combination of different numbers of flickering frequencies in an ssvep-bci for healthy volunteers and stroke patients," in *Neural Engineering (NER), 2015 7th International IEEE/EMBS Conference on*, pp. 78–81, IEEE, 2015.

[8] S. N. Carvalho, T. B. Costa, L. F. Uribe, D. C. Soriano, G. F. Yared, L. C. Coradine, and R. Attux, "Comparative analysis of strategies for feature extraction and classification in ssvep bcis," *Biomedical Signal Processing and Control*, vol. 21, pp. 34–42, 2015.

[9] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*, vol. 1. MIT press Cambridge, 2016.

[10] S. Haykin, *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1994.

[11] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a 'siamese' time delay neural network," in *Advances in neural information processing systems*, pp. 737–744, 1994.

[12] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *arXiv e-prints*, p. arXiv:1412.6980, Dec 2014.