KAM - a Kernel Attention Module for Emotion Classification with EEG Data

Dongyang Kuang[1][0000−0002−4862−7182] and
Craig Michoski[2][0000−0002−6356−233X]

[1] School of Mathematics (Zhuhai),
Sun Yat-sen University, Zhuhai, Guangdong, China
dykuang@outlook.com
https://dykuang.github.io/
[2] Oden Institute for Computational Engineering and Sciences,
University of Texas at Austin, Austin, USA
michoski@oden.utexas.edu

Abstract. In this work, a kernel attention module is presented for the task of EEG-based emotion classification with neural networks. The proposed module utilizes a self-attention mechanism by performing a kernel trick, demanding significantly fewer trainable parameters and computations than standard attention modules. The design also provides a scalar for quantitatively examining the amount of attention assigned during deep feature refinement, hence help better interpret a trained model. Using EEGNet as the backbone model, extensive experiments are conducted on the SEED dataset to assess the module's performance on within-subject classification tasks compared to other SOTA attention modules. Requiring only one extra parameter, the inserted module is shown to boost the base model's mean prediction accuracy up to more than 1% across 15 subjects. A key component of the method is the interpretability of solutions, which is addressed using several different techniques, and is included throughout as part of the dependency analysis.

Keywords: Kernel Attention · EEG · Emotion Classification.

1 Introduction

Correctly identifying human emotion using classification strategies has long been a topic of interest in brain computer interfaces (BCI) and their applications. According to the review [17] on classification algorithms utilized in EEG studies, there are five major categories of classifiers currently under investigation, which are: i) conventional classifiers [20, 13–15], ii) matrix and tensor based classifiers [5], iii) transfer learning based methods [2, 7], iv) deep learning algorithms and advanced statistical approaches [4, 19], and v) multi-label classifiers [18, 3, 21]. While many classification approaches have been explored in the context of EEG signal processing, the classification pipeline itself has still frequently involved extensive manual preprocessing and feature engineering, often requiring intervention from domain experts as well as the experimental operators involved in the acquisition of the EEG signals themselves.

The success of neural network tools has been shown to alleviate some of limitations in the classical pipeline by providing, for example, faster predictions and reducing the need to manually preprocess data. One problem that remains however, is that the scale of learnable parameters in a classification network can be too large relative to the input data size. That is, in contrast to areas such as image classification, the availability of clean and open-sourced EEG data sets is comparably quite small in size. As a consequence, basic research calls specifically for data efficient and parameter efficient, as well as human interpretable models in order to provide penetrating insight into human emotion classification given relatively sparse data sampling.

2 Related Work

Below we briefly review some salient results in the literature relevant to the present work.

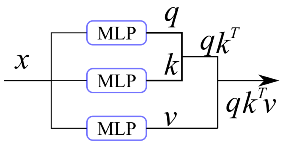

Fig. 1. The basic self-attention mechanism.

Self-Attention: Self-attention was originally introduced in the field of natural language processing (NLP) [1], where the self-attention mechanism operates as a key component in transformer modules. These standard self-attention designs rely on MLPs, instead of the more conventionally used convolutional layers, to generate an attention matrix. Adapting the concept of self-attention to computer vision has been remarkably successful, and has demonstrated impressive performance to date on a variety of different tasks, e.g. [6, 16]. As a consequence, through self-attention approaches, the field of image classification— long dominated by an assortment of solutions utilizing convolution neural networks—has discovered a promising new tool for (potentially) broad application.

Alternative Attention Approaches: In addition to self-attention, there exists many other types of attention modules in the field of computer vision. Two prominent examples of these are the Squeeze-and-Excitation (SE) network [10] and the Convolutional Block Attention Module (CBAM) [22], where the SE network won the ImageNet2017 championship, while CBAM sequentially infers attention maps along both channel and spatial dimensions for adaptive feature refinement. These methods, and others, are indicative of the increased contemporary importance attention-style approaches are having within computer vision.

EEGNet: EEGNet was proposed in [12] for a compact network design aimed at finding better generalizations across different BCI paradigms. Using depthwise/separable convolution layers, the EEGNet network contains considerably fewer trainable parameters than models constructed with regular convolutional layers, while still showing commensurate performance. EEGNet is also equipped with a convolutional layer of kernel size equal to the total number of EEG channels, enabling the ability to investigate spatial patterns learned during training in order to elucidate underlying electrophysical principles. Because of both the effectiveness and parsimonious nature of the EEGNet design, we adopt it as the backbone network for comparing and interpreting different attention modules.

SEED: The SEED dataset [23] includes multiple physiological signals that evaluate self-reported emotional responses, classified into *Positive*, *Neutral*, and *Negative* reactions taken from 15 participants, and is one of the standard datasets for benchmarking EEG signal classification strategies. The data was collected with 62 EEG channels using the 10–20 international standard.

Our work: In this paper we consider a self-attention mechanism for boosting the backbone model's performance in a parameter efficient way and for providing better interpretation. However, at the outset, it is worth noting that several potential difficulties arise when conceptualizing the incorporation of a self-attention mechanism into the EEGNet framework. First, as discussed in [6], transformers tend to be quite data hungry models, failing to outperform regular convolution-based networks when the dataset's size is not large enough. In the area of emotion classification using EEG signals, this data-thirst requirement can become prohibitive. One of the reasons for this "data hungry" aspect of transformers is due to the MLP layers generally involved in the attention mechanism. These dense layers, not surprisingly, tend to contain many more parameters than convolutional layers that comprise popular emotion classification frameworks. Consequently, a difficulty arises when self-attention is applied directly to frameworks such as EEGNet, since the resulting hybrid frameworks tend to substantially undermine the primary advantage of the underlying base models; for example, EEGNet would no longer be a lightweight and compact model, but instead become a data hungry model focusing entirely on accuracy over pragmatic utility.

Thus, the primary goal of the present work is to find a way to incorporate the self-attention mechanism into EEGNet in a way that can still preserve EEGNet's pragmatic utility by maximizing the parameter efficiency in the design of the attention module. The solution presented in this paper is called a Kernel Attention network Module (KAM), and can be described as:

1. Utilizing a kernel function to produce the proper attention matrix instead of relying on MLP layers; thus reducing the number of both parameters and computations required. Moreover, benefiting from the one parameter design, specific techniques are then able to be employed for more effective interpretability techniques as well.
2. With the proposed module inserted along with only one additional parameter, the predictive performance of the baseline model—in our case, EEGNet—can be boosted up to more than 3% for some subjects on within-subject classifications and more than 1% overall on mean performance.

3 Kernel Attention Module

Figure 1 gives a simple illustration of how the basic self-attention mechanism works when applied on some feature x. First, the feature x is mapped to three different features of the same size via $q = \phi_q(x)$, $k = \phi_k(x)$, and $v = \phi_v(x)$. The mappings ϕ_j for $j \in \{q, k, v\}$ are achieved using MLP blocks. Next, an attention matrix is formed by computing qk^T which will then be used as a prefactor on v to produce the attention output. Computationally, this procedure can be summarized as: $x \leftarrow [\phi_q(x)\phi_k(x)^T]\phi_v(x)$.

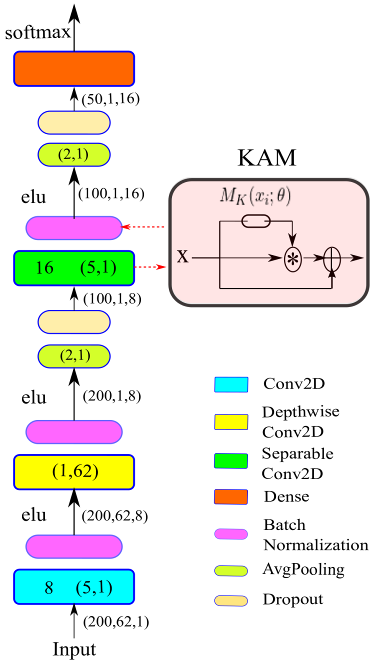

Fig. 2. EEGNet with KAM inserted. Some important hyperparameters, kernel shapes and tensor sizes are also shown.

In the case where x has large feature dimension n, the attention is applied on segments of x in parallel with the resulting feature pieces subsequently concatenated afterwards—a mechanism referred to as *multi-head self-attention*. For more details on the underlying algorithms we refer the reader to [6]. Using the above basic self-attention format, KAM is constructed by replacing the inner product form $\phi_q(x)\phi_k(x)^T$ with a kernel matrix $M_K(x;\theta)$ subject to some parameter θ. For example, a Gaussian type kernel function can be used to generate $M_K^{ij} = \exp(-\alpha d(x_i, x_j)^2)$, where $d(\cdot,\cdot)$ denotes some distance metric, x_i is the ith row or column of feature block x depending on whether M_K is multiplied to x by left or right, $\theta = \alpha$ is the learnable parameter during training. In the KAM design, ϕ_v can be simply dropped to reduce the number of total parameters. Finally, a skip connection is included in the the KAM design that offers several potential benefits. On one hand, an additional skip can help better backpropagation of gradients to the blocks in front of the KAM. On the other, it provides an easy interface to quantitatively measure how much attention is actually being applied, requiring only an examination on the values of θ. For example, when $\alpha \to +\infty$ then $M_K(x;\theta) \to I$, meaning no cross attention among features is applied. However, if $\alpha \to 0$, then $M_K(x;\theta) \to J$ which is an attention matrix whose off-diagonal entries $J_{ij} = 1, i \neq j$, meaning deep features now equally contribute to others for refinement during training. The above procedure leads to our Kernel Attention Module (KAM) design as shown in Figure 2, where its symbolic form can be summarized as:

$$x \leftarrow x + M_K(x;\theta)x = (I + M_K(x;\theta))\,x. \tag{1}$$

The proposed KAM mechanism can also be easily applied with multiple heads. We further note that in the implementation in Section 4, an extended form is used for $M_K^{ij} = \exp(-\alpha d(x_i, x_j)^2)$, where $\alpha \in (a, \infty)$. If the lower bound is set to $a = 0$, the case can be readily interpreted as $\alpha = 1/\sigma^2$ where the parameter can be understood as a kind of "standard deviation". Setting a less than zero allows off-diagonal entries in M_K to have values greater than one, in which case a should be close to zero, i.e. $a \approx -\epsilon$ for ϵ small, to prevent numerical blow-up during training from poor matrix conditioning.

4 Experiments

In this research, we focused on model's performance on subject dependent classification tasks. In the spirit of [23] the data set was divided into non-overlapping epochs, each lasting one second, yielding ∼3300 epochs per trial, per subject, and ∼1060 epochs per labelled emotion. However, in contrast to [23], and most other studies where the training and testing data are manually split and models evaluated in a single pass, we adopt a cross-validation (CV) approach to improve evaluation robustness.

In the following benchmark using EEGNet with KAM, the data from each subject is split, taking 1/6 for validation during training. Five-fold cross validation (5-CV) is then performed on the remaining 5/6 of the data. Theses ratios are chosen to make the validation and test set roughly the same size during cross validation. For any model test, initial weights are set to the same values across each of the five folds. The network is trained for each fold over a maximum of 80 epochs, and the best model is selected at the epoch with the best validation accuracy. All experiments are trained with the same Adam optimizer configuration of an initial learning rate of 10^{-2} and a decay rate of 0.75, which only activates when no accuracy improvement is seen on the validation set in the past 10 epochs. A total of $5 \times 15 = 45$ training runs are conducted for each model compared in our benchmark. The code used for models' training and evaluation will be made available at https://github.com/dykuang/BCI-Attention

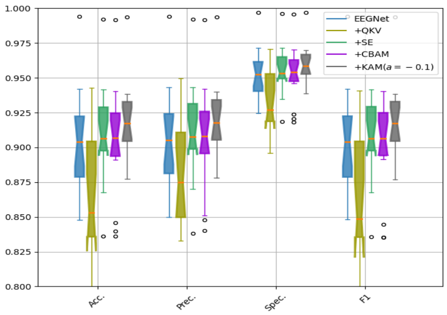

Fig. 3. Overall mean prediction performance across 15 subjects.

Benchmark: For benchmarking we compared five models: a) EEGNet, b) the basic QKV type attention from Figure 1, c) SE attention, d) CBAM attention, and e) KAM($a = -0.1$). All implementations herein are inserted at the same location shown in Figure 2. Note that the basic QKV attention module does not perform well here, which is likely due to it, as mentioned in [6], being data thirsty and SEED not being a large enough dataset to quench. It is also worth mentioning here that the version of KAM with α's lower bound $a = 0$ gives mean accuracy of 91.74% ± 3.02% which is slightly worse than the case of $a = -0.1$. This suggests that the extension of the lower bound to a negative value can potentially help during deep feature refinement. We also observe in our experiments that the training procedure for some subjects will push α slightly below zero for minimizing the loss function (see Figure 5).

Table 1. Mean accuracy reported from different models.

Models	EEGNet	+QKV	+SE	+CBAM	+KAM($a = -0.1$)
Parameters	3851	4940	3933	4033	3852
Acc(%)	90.34 ± 3.69	86.81 ±4.19	91.20 ±3.42	90.40 ±3.97	91.89 ±2.76

Channel Attention: The inserted KAM module can potentially change the kernel weight originally designed in EEGNet during the model's decision process. Particularly, kernel weights in the first depthwise convolution layer (see Figure 2) were treated as a representation for relative attention across different channels[3] in [12]. They can be affected when different attention modules are inserted. As the depthwise convolution applies one kernel to each of the eight channels, there are eight kernels associated to the architecture in Fig. 2. For clarity here, to illustrate the effect of these kernels on the network, we only examine the kernel applied on the first feature channel.

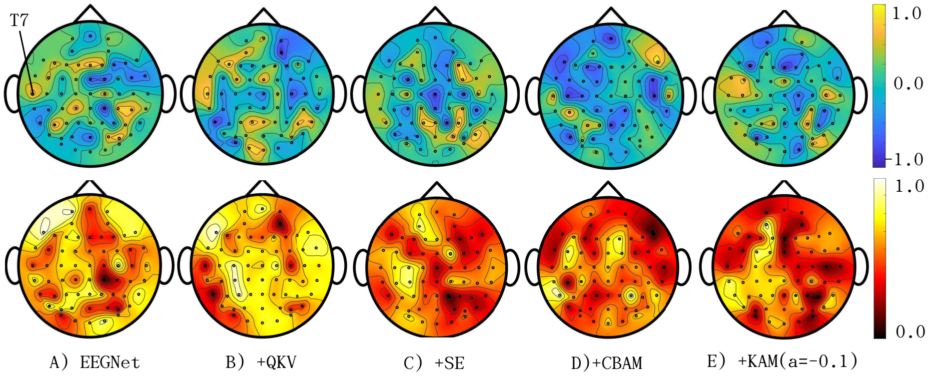

Fig. 4. Kernel weights mapped onto scalp maps. The first row shows the normalized mean. The second row shows the normalized standard deviation from the 5-CV.

To visualize the different spatial attention patterns discovered represented by the selected kernel weights of different models, we present scalp maps in Fig. 4. These maps are generated by training the models under 5-CV for subject S01, where mean and standard deviations are shown. It can be clearly seen overall that different modules can result in different channel attention patterns. While it may be difficult to immediately associate the mean value mappings to informative clinical interpretations, the spatial magnitude of the standard deviation does provide a way to measure different models' confidence in assigning kernel weights across different regions. For example, one thing to observe here is that

[3] These are kernel weights in the first depthwise convolutional layer. The shape is of(1, 62) and can be directly associated with the 62 EEG sensor locations on scalp.

all models visualized high mean attention values around the T7 region with relatively low uncertainty (represented by the std value). This observation seems to support some studies reporting correlations between emotional deficiency and memory development with specific temporal lobe function, such as in the diathesis of schizophrenia, e.g. [8], and in types of memory enhancement in forms of dementia, e.g. [11]. As an alternative research direction, how to enforce one's prior knowledge on task related scalp patterns so that the posterior learned patterns are robust to inserted attention modules is also important for building a better human interpretable model.

Dependency on $\boldsymbol{\alpha}$: For better interpret the effect of module parameter α from KAM in trained models, we organised this section. Figure 5(A) shows the distribution over learned α during 5-CV for each subject. Among these, only experiments with data from subject S02 and S13 yield instances where $\alpha < 0$, while all other trainings find $\alpha \geq 0$. However, per subject speaking, the change in the lower bound on α does have a noticeable impact. For example, in the data from subject S05, S06, and S13 the learned α values cluster at locations close to zero, but we observed from our experiments that this small deviation from zero results in noticeable accuracy differences. This may because of the fact that small α will correspond to the case with large σ, i.e. further away from zero attention as explained above in Section 3.

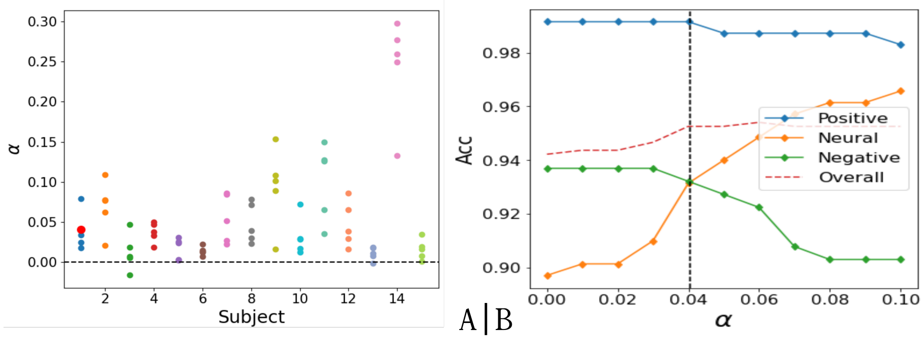

Fig. 5. *A*: Distribution of learned α value during the 5CV with EEGNet+KAM across the 15 subjects. *B*: Change of accuracy with varying value of α while freezing other parameters in the selected model (marked as red in first column i.e. subject S01 of *A*).

The one-parameter KAM design also makes it easy to analyze prediction accuracy as a function of the module parameter α. As an example, we choose the model trained from the first fold (marked with a red dot in Fig. 5(A)) for subject S01 and gather data from three trials each with a different emotion label for the dependency analysis. By varying α values in KAM while keeping other model parameters frozen, we can examine how α conditionally effects the prediction. In 5(B), it is interesting to see that the learned value (black vertical line) in KAM happens at a location where the overall accuracy line first rises to stabilize in this case. It is also interesting to observe that crossover between accuracy lines for “neutral" and “negative" happens at the same location of learned α. [4].

This dependency can also be examined via the distribution of $\frac{\partial f_i(x)}{\partial \alpha}|_x$ for varying α, where x is the input and $f_i(x)$ is the output of the corresponding neuron (before activation) from the last dense layer for label $i \in \{1, 2, 3\}$, i.e. positive, neutral, and negative emotion labels respectively. The result is gathered in Fig. 6(A-C) computed with the same data mentioned in the previous paragraph. Of note, the partial dependencies appear to show very similar patterns. That is, the variance in each is a decreasing function for $\alpha \in [0, 0.1]$. This can be explained by the fact that α is packaged inside an exponential form that maintains its character through differentiation. Finally, Fig. 6(D) shows a close look at the histogram of how these distributions differ at the learned $\alpha = 0.0406$ from the model's selection during training.

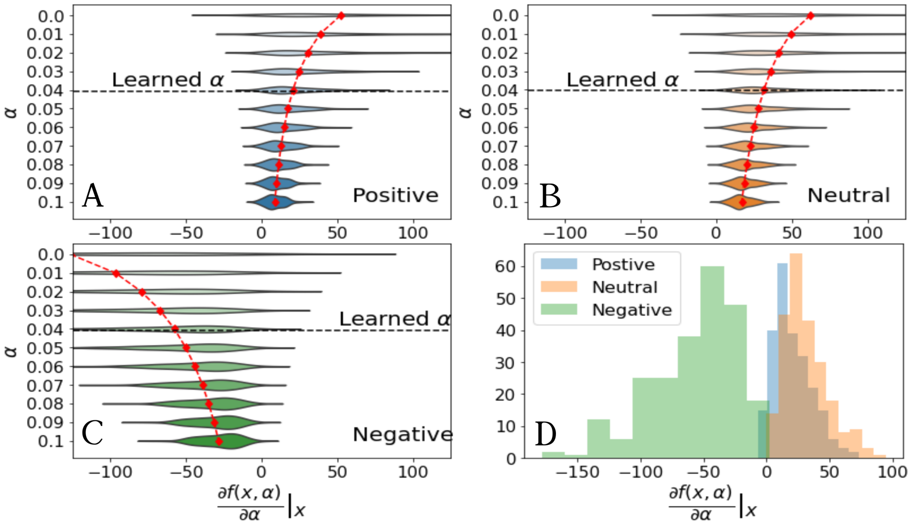

Fig. 6. A-C: Distribution at different α values corresponding to label “Positive", “Neutral" and “Negative". The mean is linked by dashed line. **D**: Distribution at learned $\alpha = 0.0406$ for different emotion labels.

Prediction Transition Curve: In this section, we would like to explore how different models react when the input sample to be predicted is gradually transformed to another sample via some morphing operation g. In other words, we try to examine and interpret how different models react under a particularly selected “attack" g. Let $\{x_0, x_1, x_2\}$ denote 3 samples with labels 0, 1 and 2,

[4] This might be an interesting coincidence since we also had other cases in our experiments where they do not meet exactly.

and $\vec{p_i} = \{p_i^j\} = F(x_i)$ with $j = 0, 1, 2$ denoting model F's confidence (here using a softmax score of the last dense layer) assigning input x_i for label j. Then clearly by construction $\sum_j p_i^j = 1$ for any i. Further let $g_i^j(u), u \in [0, 1]$ denote an morphing operation between samples parameterized by u such that $x_i = g_i^j(0) * x_i$ and $x_j = g_i^j(1) * x_i$. The symbol $*$ here stands for certain abstract action operation. In trinary classification tasks, as u increases, $F[g_i^j(u)]$ will draw a curve in the hyperplane $x + y + z = 1$ inside the triangle formed by $[1, 0, 0], [0, 1, 0], [0, 0, 1]$. By checking these curves (we call them *prediction transition curves, PTC*), one can have an idea that how the trained model F reacts with respect to the morphing operation g on selected samples. The idea can be generated to higher dimensional cases with class categories $n > 3$, but it then becomes harder to visualize these cases as simple curves being embedded in higher dimensional simplices. This prediction transition curve provides a way for visualizing and interpreting model's predicting behavior under "attack" g for given inputs.

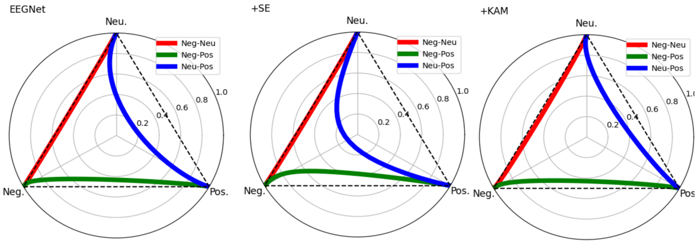

Fig. 7. The prediction transition curves from the three models on the same selected samples. From left to right: EEGNet, EEGNet+SE, EEGNet+KAM.

As a demonstration, we select three samples each with a different label, and EEGNet, EEGNet+SE, EEGNet+KAM all predict correctly on them. For simplicity, we choose the straightforward linear interpolation between samples for g, i.e. $g_i^j(u) * x_i = (1 - u)x_i + ux_j$. Notice that this definition of g is symmetric in terms of $g_i^j(u) * x_i = g_j^i(1 - u) * x_j$. So morphing from x_i to x_j and x_j from x_i end up with the sample path (regardless of the direction). Other types of morphing operations can also be used depending on one's prior on the (known or inferred) underlying relationships between samples. The prediction transition curves obtained from morphing with g are summarized in Figure 7. It can be seen that all three compared models have almost straight PTCs for the connection of "Negative- Neutral" and "Negative-Positive", meaning that when one input sample is slowly morphing to the other, the model transits its confidence between the two labels almost linearly while leaving the third label almost untouched. Curves linking "Neutral-Positive" are all curved to the center at different extents, suggesting that models are in some sense "hesitating" to assign "Negative" for intermediate samples generated by morphing between

"Neutral" and "Positive". This example is consistent with the observations in Figure 6 D) and provides a different angle suggesting that the trained models find it more difficult to separate "Positive" from "Neutral" emotions than separating "Negative" from "Positive" or "Neutral" from "Negative" emotions for the subject data under examination. An interesting followup question is whether this observation bears clinical significance as well, something which undoubtedly deserves consideration.

5 Conclusion

In this work, we present a kernel attention module that can be inserted into a network for deep feature refinement. Using EEGNet as the backbone model, the performance of KAM are benchmarked against several SOTA attention modules under cross validation with SEED dataset. With only one additional parameter, the idea behind KAM has demonstrated good potential for developing parameter efficient models that can simultaneously help human interpretation on trained models. Many follow-up studies are possible in this context, including investigating the effects of different kernels (other than Gaussian) alongside more exhaustive dependency analyses. Additionally, examining different training strategies, such as the masked-autoencoder discussed in [9], might also be beneficial.

6 Acknowledgement

This work was partially supported by the Fundamental Research Funds for the Central Universities, Sun Yat-sen University (22qntd2901)

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate (2016)
2. Blankertz, B., Kawanabe, M., Tomioka, R., Hohlefeld, F., Müller, K.r., Nikulin, V.V.: Invariant common spatial patterns: Alleviating nonstationarities in brain-computer interfacing. In: Advances in neural information processing systems. pp. 113–120 (2008)
3. Blankertz, B., Lemm, S., Treder, M., Haufe, S., Müller, K.R.: Single-trial analysis and classification of ERP components—a tutorial. NeuroImage **56**(2), 814–825 (2011)
4. Cecotti, H., Graser, A.: Convolutional neural networks for p300 detection with application to brain-computer interfaces. IEEE transactions on pattern analysis and machine intelligence **33**(3), 433–445 (2010)
5. Congedo, M., Barachant, A., Bhatia, R.: Riemannian geometry for EEG-based brain-computer interfaces; a primer and a review. Brain-Computer Interfaces **4**(3), 155–174 (2017)
6. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)

7. Fazli, S., Popescu, F., Danóczy, M., Blankertz, B., Müller, K.R., Grozea, C.: Subject-independent mental state classification in single trials. Neural networks **22**(9), 1305–1312 (2009)
8. Goghari, V.M., MacDonald III, A.W., Sponheim, S.R.: Temporal lobe structures and facial emotion recognition in schizophrenia patients and nonpsychotic relatives. Schizophrenia bulletin **37**(6), 1281–1294 (2011)
9. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. arXiv preprint arXiv:2111.06377 (2021)
10. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7132–7141 (2018)
11. Kumfor, F., Irish, M., Hodges, J.R., Piguet, O.: Frontal and temporal lobe contributions to emotional enhancement of memory in behavioral-variant frontotemporal dementia and alzheimer's disease. Frontiers in behavioral neuroscience **8**, 225 (2014)
12. Lawhern, V.J., Solon, A.J., Waytowich, N.R., Gordon, S.M., Hung, C.P., Lance, B.J.: EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces. Journal of neural engineering **15**(5), 056013 (2018)
13. Li, J., Zhang, L.: Bilateral adaptation and neurofeedback for brain computer interface system. Journal of neuroscience methods **193**(2), 373–379 (2010)
14. Liu, G., Huang, G., Meng, J., Zhang, D., Zhu, X.: Improved GMM with parameter initialization for unsupervised adaptation of brain–computer interface. International Journal for Numerical Methods in Biomedical Engineering **26**(6), 681–691 (2010)
15. Liu, G., Zhang, D., Meng, J., Huang, G., Zhu, X.: Unsupervised adaptation of electroencephalogram signal processing based on fuzzy c-means algorithm. International Journal of Adaptive Control and Signal Processing **26**(6), 482–495 (2012)
16. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10012–10022 (2021)
17. Lotte, F., Bougrain, L., Cichocki, A., Clerc, M., Congedo, M., Rakotomamonjy, A., Yger, F.: A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update. Journal of neural engineering **15**(3), 031005 (2018)
18. Lotte, F., Congedo, M., Lécuyer, A., Lamarche, F., Arnaldi, B.: A review of classification algorithms for EEG-based brain–computer interfaces. Journal of neural engineering **4**(2), R1 (2007)
19. Lu, N., Li, T., Ren, X., Miao, H.: A deep learning scheme for motor imagery classification based on restricted boltzmann machines. IEEE transactions on neural systems and rehabilitation engineering **25**(6), 566–576 (2016)
20. Schlögl, A., Vidaurre, C., Müller, K.R.: Adaptive methods in bci research-an introductory tutorial. In: Brain-Computer Interfaces, pp. 331–355. Springer (2009)
21. Steyrl, D., Scherer, R., Faller, J., Müller-Putz, G.R.: Random forests in non-invasive sensorimotor rhythm brain-computer interfaces: a practical and convenient non-linear classifier. Biomedical Engineering/Biomedizinische Technik **61**(1), 77–86 (2016)
22. Woo, S., Park, J., Lee, J.Y., Kweon, I.S.: CBAM: Convolutional block attention module. In: Proceedings of the European conference on computer vision (ECCV). pp. 3–19 (2018)
23. Zheng, W.L., Lu, B.L.: Investigating critical frequency bands and channels for EEG-based emotion recognition with deep neural networks. IEEE Transactions on Autonomous Mental Development **7**(3), 162–175 (2015)