Predicting Auditory Spatial Attention from EEG using Single- and Multi-task Convolutional Neural Networks

Zhentao Liu[1], Jeffrey Mock[2], Yufei Huang[1], Edward Golob[2]
[1]Department of Electrical and Computer Engineering, [2]Department of Psychology,
the University of Texas at San Antonio, TX 78249

***Abstract*—Recent behavioral and electroencephalography (EEG) studies have defined ways that auditory spatial attention can be allocated over large regions of space. As with most experimental studies, behavior and EEG were averaged over 10s of minutes because identifying abstract feature spatial codes from raw EEG data is extremely challenging. The goal of this study is to design a deep learning model that can learn from raw EEG data and predict auditory spatial information on a trial-by-trial basis. We designed a convolutional neural network (CNN) model to predict the attended location or other stimulus locations relative to the attended location. A multi-task model was also used to predict the attended and stimulus locations at the same time. Based on the visualization of our models, we investigated features of individual classification tasks and joint feature of the multi-task model. Our model achieved an average 72.4% in relative location prediction and 90.0% in attended location prediction individually (AUROC's). The multi-task model improved the performance of attended location prediction by 3%. Our results show that deep learning methods are able to define abstract neural codes in EEG thought to neural mechanisms of human spatial cognition and attention.**

Keywords—spatial attention, auditory, convolutional neural network, multi-task learning

Supported by NIH grant DC014736.

I. INTRODUCTION

Intelligent behavior requires the ability to both focus spatial attention to help accomplish the current goal while also being responsive to unexpected events at other locations in space. Behavioral and neurophysiological studies suggest that one way spatial attention handles this "dual mandate" is by having a gradient where attentional processing benefits progressively decrease with distance from the attended location [1]. Neurophysiological studies using EEG show that spatial attention gradients likely generated by fronto-parietal brain regions transform absolute spatial locations into a coordinate system centered on the currently attended location [2].

In this paper, we propose a deep convolutional neural network (CNN) model that can extract EEG features that represent various spatial codes operative in an auditory spatial attention task. The CNN model can learn local, lower level features through spatial filters and temporal filters, and then represents higher-level features in the deeper layers [3]. Recent years, CNN models have proven to be successful in many fields such as computer vision, speech recognition, and natural language processing. One useful aspect of CNN models is their effectiveness in end-to-end learning; i.e., learning from raw data without extensive preprocessing and a priori feature selection. This is especially attractive in Brain-Computer Interface (BCI) because it is difficult for humans to select all relevant features from complex EEG signals. We also propose a multi-task learning method to define relations between different tasks and features. Multi-task learning is derived from inductive transfer which can improve network performance by introducing inductive bias [4]. In our case, the inductive bias is contributed by an auxiliary task which causes the model to learn extra features from both tasks to reduce overfitting.

In summary, we proposed individual CNN and multi-task models (MTM) to perform two different predictions in the auditory spatial attention experiment. Participants were told to attend to either a left or right side location while listening to sounds coming from one of five evenly spaced locations within a 180º frontal horizontal plane (45º apart). First, we predicted the sound location about the attended location (i.e. 45º, 90º, 135º, 180º away from the attended location, termed "Relative Location Prediction"). Second, we predicted where a subject was attending (i.e. left or right side, termed Attended Location Prediction). Each model was interpreted through visualization of learning related features.

II. EXPERIMENT AND DATA PREPROCESSING

A. Participants

Forty-four participants were included in this study.

B. Stimuli

Five virtual white noise burst sounds (0.1–10 kHz, 200 ms duration, 5 ms rise/fall times, ~60 dB nHL) were made to correspond to five locations, each 45º apart in the 180º frontal

azimuth plane. The spatialized sounds were created by applying appropriate interaural time and level differences, and head-related transfer functions for each spatial location. Stimuli were presented with insert earphones rather than free-field speakers in order to limit the influence of visual indicators of sound sources and avoid changes in the relationship between sound source location and the ears due to head movements.

C. *Experimental Paradigm*

A schematic of the paradigm with a sample stimulus sequence is shown in **Figure 1**. Participants had a response pad and listened to sequences of the sounds randomly presented from five possible locations in the frontal azimuth plane (p=.20/location; left to right: -90°, -45°, 0°, +45°, +90°). In each block the participant was given a target location (left or right, in separate blocks) and asked to make fast but accurate button responses to sounds at the target location. There were 150 stimuli per block, and each target location had two blocks (4 total blocks). Behavioral measures to targets included median reaction time, hit rate (percentage of responses to target) and false alarm rate (percentage of responses to non-targets).

D. *EEG Recording*

All EEG data were recorded in a sound-attenuated booth, and continuously digitized at 500 Hz with a 64-channel EEG system (Compumedics Neuroscan).

E. *Data preprocessing*

In this study, we minimally preprocessed the EEG datasets to maximize CNN end-to-end learning ability. We rejected bad channels by a standard deviation threshold [-2,2], and then used spherical interpolation. Data were filtered (1-45 Hz) and then epoched (-1.2 to 1.2 s, relative to sound onset). After preprocessing we obtained 25,975 samples, and each sample had 64 channels and 350 time points. All preprocessing work was done on EEGLAB, which is an open source software toolbox.

III. Method

A. *Convolutional neural networks*

In general, convolutional neural networks for EEG signals learn local features through convolution kernels, and transform high dimensional features into lower dimensions that contain global information about the original data [5].

In this study, we used a blocked design for our CNN model. In Block 1 we apply two convolutional kernels in sequence. First, we used kernels of size (channels, 1) to learn spatial correlation; with 64 weights for all 64 channels. We then used kernels of size (1, C) to learn temporal correlations. Hyperparameter C was determined during hyperparameter optimization. The two convolutions were kept linear because there is no significant performance improvement for nonlinear activation [6]. After convolution operations, we performed Batch Normalization and Exponential Linear Unit (ELU) activation [7]. Then a maxpooling layer was added to reduce the dimension of features. Regularization used both L1-L2 norm with dropout to limit overfitting.

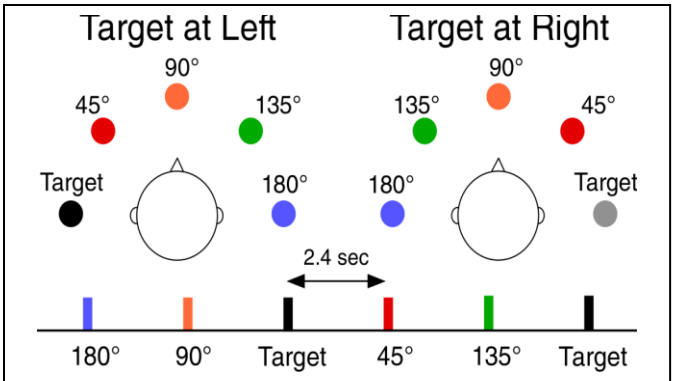


Figure 1. Schematic of sound location. Sound locations are shown relative to the listener's hear for each target location. Sounds at each location were equally probable and randomly presented with a constant onset asynchrony of 2.4s.

Block 2 has a similar structure to block1 except spatial filters are not included. Block 3 and block 4 have the Block 2 structure, and will be added to fit different tasks.

The classification block collects low dimension features directly after the last convolution block. Activation functions are set up as Sigmoid for binary classification, with SoftMax for multi-label classification.

B. *Relative location prediction*

In this task, we refer to "relative location" as the angular distance between the attended location and a given sound's location. When subjects attend to the left, speakers (left to right: -90°, -45°, 0°, 45°, 90°) will be labeled as 0, 1, 2, 3, 4, respectively. When the attended location is on the right, labels will be (left to right) 4, 3, 2, 1, 0. Thus, the model will classify EEG data into these 5 classes without knowing the absolute location of the attended location (code=0 for both left and right attended locations).

We use four convolution blocks in this task; each block has the same dropout rate of 0.6 and regularization rate of 0.001. We choose the first three blocks with a temporal filter size of (1,10), and the last block as (1,6) due to epoch length. In compiling we use optimizer Adam with an adaptive learning rate of 0.01 and decay=0.001 in every epoch during 400 training iterations. All parameters and hyperparameters are chosen by Hyperas grid searching algorithm. The architecture is shown in **Table 1**.

After the model is trained, we extract all spatial filters, each of them is a (64,1) vector and the elements are the weights the model learned for each channel. We mapped those weights back on to the scalp sites to study define signal topography. We assume that the most significant filter provides the most discriminative features, so the filters are ranked based on feature classification performance to find the best spatial features.

We also performed slope analysis on this task. Each of the feature maps for spatial filters is extracted as an ERP map, and then the linear slope of the amplitude across the 4 non-target locations was calculated at each time point.

Table 1. Relative Location Model

Layer	Output Shape	Layer	Output Shape
Input Layer	64, 350, 1	Batch Norm.	1, 25, 100
Conv2d	1, 350, 10	ELU	1, 25, 100
Conv2d	1, 341, 25	Max Pooling	1, 8, 100
Batch Norm.	1, 341, 25	Dropout	1, 8, 100
ELU	1, 341, 25	Conv2d	1, 3, 200
Max Pooling	1, 113, 25	Batch Norm.	1, 3, 200
Dropout	1, 113, 25	ELU	1, 3, 200
Conv2d	1, 104, 50	Max Pooling	1, 1, 200
Batch Norm.	1, 104, 50	Dropout	1, 1, 200
ELU	1, 104, 50	Flatten	200
Max Pooling	1, 34, 50	Dense	5
Dropout	1, 34, 50	Softmax	5
Conv2d	1, 25, 100		
Total params: 187,046 Trainable params: 187,038			

Table 2. Attended Location Model

Layer	Output Shape	Layer	Output Shape
Input Layer_1	64, 350, 1	Conv2d	1, 25, 100
Conv2d	1, 350, 10	Batch Norm.	1, 25, 100
Conv2d	1, 341, 25	ELU	1, 25, 100
Batch Norm.	1, 341, 25	Max Pooling	1, 8, 100
ELU	1, 341, 25	Dropout	1, 8, 100
Max Pooling	1, 113, 25	Flatten	800
Dropout	1, 113, 25	Input Layer_2	1
Conv2d	1, 104, 50	Embedding	1, 800
Batch Norm.	1, 104, 50	Flatten	800
ELU	1, 104, 50	Multiply	800
Max Pooling	1, 34, 50	Dense	2
Dropout	1, 34, 50	Sigmoid	2
Total params: 72,239 Trainable params: 72,233			

C. Attended location prediction

In this task, we want to predict the subject's attended location based on EEG signals. From a previous study using averaged EEG signals the gradient away from left and right attended locations was very similar. This posed a challenge for the model to tell the difference between attending to the left and right side. In preliminary work accuracy never exceeded 60%. Our solution was to feed the sound location along with the EEG signal to the model using label embedding so that model can use one more factor to make a prediction. Absolute locations are labeled as 1, 2, 3, 4, 5 from left to right as another input for the model.

We used a similar architecture with the relative location prediction task, but had three blocks instead of four. Each block has a dropout rate of 0.5 and regularization rate of 0.001. All three blocks of temporal filter's size is (1,10). We added one embedded layer to model to expand sound location from a single number to a vector that is also learned by the model. We merge this vector with low dimension features from the top convolution layer, then send the merged feature to the classifier. For compiling we used the optimizer Adam with an adaptive learning rate of 0.01 and decay=0.001 for every epoch (600 training iterations). All parameters and hyperparameters were chosen by the Hyperas grid searching algorithm. The architecture is shown in **Table 2**.

After the model was trained, we extracted all spatial filters from the first block, and the weights were mapped onto the scalp to visualize their topography. As in this task, we do not have 5 classes to label, we use Elastic Net to do a "one stone two birds" classification task. In Elastic Net regression 350 time points were treated as features, and the algorithm will assign a coefficient for each of the points [8]. From the coefficients, we can construct a heatmap to show significant time points from all 350 features. At meantime, regression performance could give us the spatial filter ranking which is similar to what we did in the previous task.

D. Multi-task model (MTM)

An MTM was used to combine the attended location task and relative location task in one model. The multi-task model learned features from both the attended and relative locations, which were expected to improve model performance on each task. However, the attended location prediction needed to take sound location for input. Consequently, the shared feature learned by model contains absolute location information while relative location prediction represents the distance between sound and attended locations. Use of absolute and relative coding can pose a challenge. For example, in the relative location task, the model does not distinguish location 0°left attending) and location 0° (right attending) because the distances from those two locations to target are both zero, hence they should have the same label. However, in the attended location task, these two locations do have two different labels (left vs. right). Therefore, we mainly focus on improving attended location task performance rather than both the tasks.

Our multi-task model contains three blocks as a shared feature extractor while keeping two task-specific output layers. We added one dense layer before attended location task classifier for conveniently merging the embedded labels. The architecture is shown in **Table 3**. The loss function L_{total} of MTM is a linear combination of the above two tasks L_i as Eq (1) [9].

$$L_{\text{total}} = \sum \alpha_i L_i \quad (1)$$

In the model selection process, we put coefficients α_1 and α_2 in Hyperas searching space so that we can search for a

Table 3. Multi-task model

Layer	Output Shape	Layer	Output Shape
Input Layer_1	64, 350, 1	ELU	1, 25, 120
Conv2d	1, 350, 15	Max Pooling	1, 8, 120
Conv2d	1, 341, 30	Dropout	1, 8, 120
Batch Norm.	1, 341, 30	Flatten	960
ELU	1, 341, 30	Dense	5
Max Pooling	1, 113, 30	Softmax	5
Dropout	1, 113, 30	Input Layer_2	1
Conv2d	1, 104, 60	Embedding	1, 200
Batch Norm.	1, 104, 60	Flatten	200
ELU	1, 104, 60	Dense	200
Max_pooling2d	1, 34, 60	Multiply	200
Dropout	1, 34, 60	Dense	2
Conv2d	1, 25, 120	Sigmoid	2
Batch Norm.	1, 25, 120		
Total params: 294,304 Trainable params: 294,298			

relatively good linear loss combination to maximize the model performance.

IV. Result and Discussion

A. *Prediction of the relative location*

Model performance for relative location is shown in **Figure 2**. As our data are balanced in 5 labels, we use one vs all ROC_AUC to evaluate performance. Results show class 0 (attended target) has the best prediction, and likely reflects attention bias specific to targets. On the other hand, class 2 (90° direction) is quite difficult due to its neutral position.

We extracted features associated with the 10 spatial filters and performed classification (**Table 4**). Spatial filter 9 had the best performance, which produced the most significant features. The topography spatial filter 9 is shown in **Figure 3**.

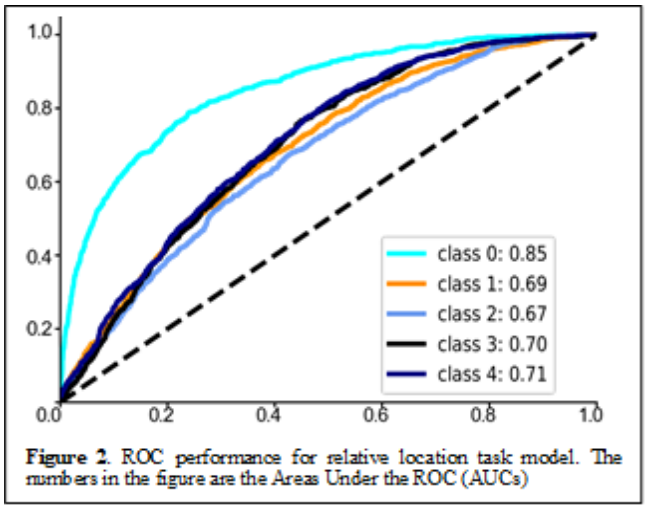


Figure 2. ROC performance for relative location task model. The numbers in the figure are the Areas Under the ROC (AUCs)

Table 4. Spatial filter performance (relative location)

Spatial1	Spatial2	Spatial3	Spatial4	Spatial5
60.40%	52.07%	60.10%	61.81%	57.59%
Spatial6	Spatial7	Spatial8	Spatial9	Spatial10
57.46%	53.45%	49.29%	67.50%	49.97%

Event-related potentials for each class were from generated from the spatial filters, and all five classes show a positive potential (P300) at a latency of ~500ms. Event-related potentials from Spatial filter 9 have three clear sensory peaks (~100-200 ms), as well as positivity at ~440-550 ms after sound onset that reflected attention gradients. We further performed attention gradient analysis by fitting the linear slope of potentials across the 4 non-target locations (350 time points and slope values) in **Figure 4**. The slope over time curve corresponds to significant EEG gradients discovered in previous work [2].

B. *Prediction of the attended location*

Model performance is shown in **Figure 5**. With the embedded label's help, attended location prediction achieved 90% ROC_AUC performance. Similar to the relative location task, we extracted features from 10 spatial filters and performed classification. Unlike the previous task, the attended location only had 2 labels (left, right). Thus we did not perform gradient analysis. Instead, we used Elastic Net Regression for feature selection. The regression performance ranking results are in

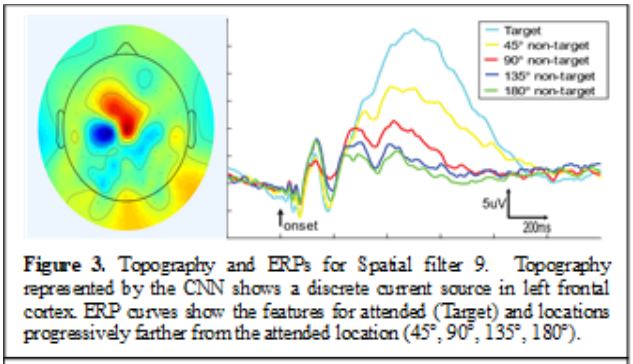


Figure 3. Topography and ERPs for Spatial filter 9. Topography represented by the CNN shows a discrete current source in left frontal cortex. ERP curves show the features for attended (Target) and locations progressively farther from the attended location (45°, 90°, 135°, 180°).

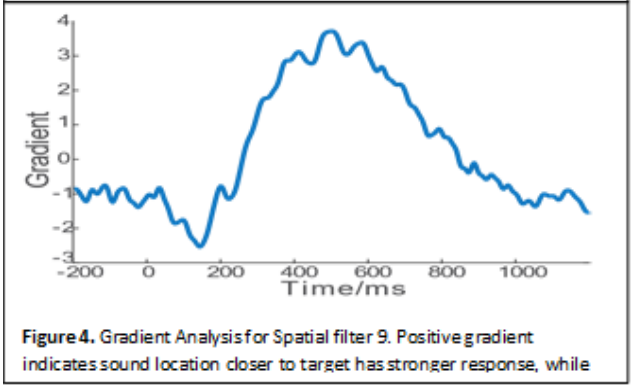


Figure 4. Gradient Analysis for Spatial filter 9. Positive gradient indicates sound location closer to target has stronger response, while

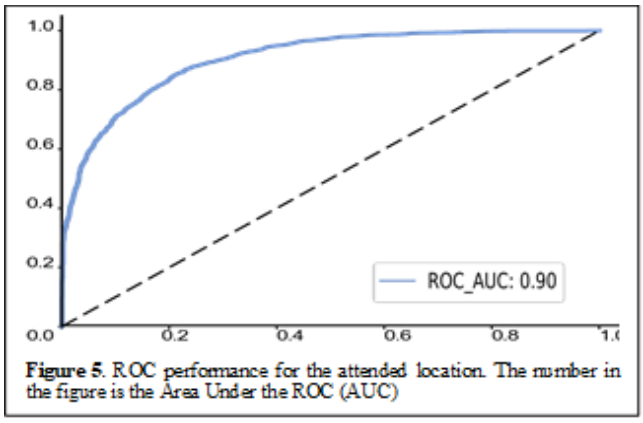


Figure 5. ROC performance for the attended location. The number in the figure is the Area Under the ROC (AUC)

Table 5. Spatial filter performance (attended location)

Spatial1	Spatial2	Spatial3	Spatial4	Spatial5
20.31%	21.66%	22.23%	25.60%	22.52%
Spatial6	Spatial7	Spatial8	Spatial9	Spatial10
22.14%	25.12%	23.29%	26.66%	22.91%

Table 5. Spatial filters 1, 3, 4, and 9 were selected due to their high ranking. In the Elastic Net, the regression function follows Eq (2) [8].

$$\beta = \text{argmin}\ (||y - X\beta||^2 + \lambda_2||\beta||^2 + \lambda_1||\beta||_1) \qquad (2)$$

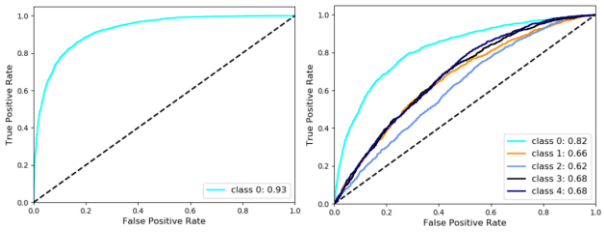
Hence each feature will be assigned one β and we can use β value to evaluate the feature's significance level. Heatmap of β which associates with 4 spatial filters is shown in **Figure 6**. It demonstrates that the most significant time window is between ~100-600 ms. Not every filter had a clear topographic pattern, which is likely due to noise from other sources affecting the minimum preprocessing method. Even with noise, the model still had 90% AUC performance.

C. Multi-task model

In our Multi-task model the attended location and relative location are predicted at the same time. One top consideration for multi-task models is the joint loss function construction. Due to different ranges of individual task loss functions, simply adding them together will cause performance collapse for a small loss range task [10]. In our case, the embedded label in the attended location prediction may be underdetermined. For example, an embedded label shows a significant difference between sounds from the left vs. right target, but when expressed as relative locations their labels are both class 0. The goal is simply to use relative location as a constraint to improve task performance. Therefore, we do not emphasize the relative location prediction. We used a linear combination of individual loss functions for the multi-task model joint loss function. Several pairs of weights that favor the attended location prediction were put into Hyperas searching grid for optimization, and (0.6, 0.4) was eventually selected. The two tasks' ROC_AUC performance are shown in **Figure 7**.

The attended location prediction had a notable 3% performance improvement, while relative location prediction dropped a little bit. However, for relative location prediction, class 0 still has the best performance among all locations while class 2 is the most challenging location for the model. The trend of performance maintained in the individual model, which implies that the relative location task was not collapsed. Thus, the feature selection mechanism is working properly.

To further explore what features are captured in MTM, we extracted the samples correctly classified in MTM but misclassified in the individual attended location task model. In total there are 457 samples (183 at -90° location, 274 at +90° location). Misclassified samples were passed to spatial filters, and then averaged with respect to the number of filters. Therefore we obtained weighted ERP feature maps. By comparing them with the samples that were correctly classified in the individual attended location task, we were able to identify



(a) (b)

Figure 7. ROC performance for the multi-task model. (a) Attended location (b) Relative locations. The numbers in the figure are the Areas Under the ROC (AUCs)

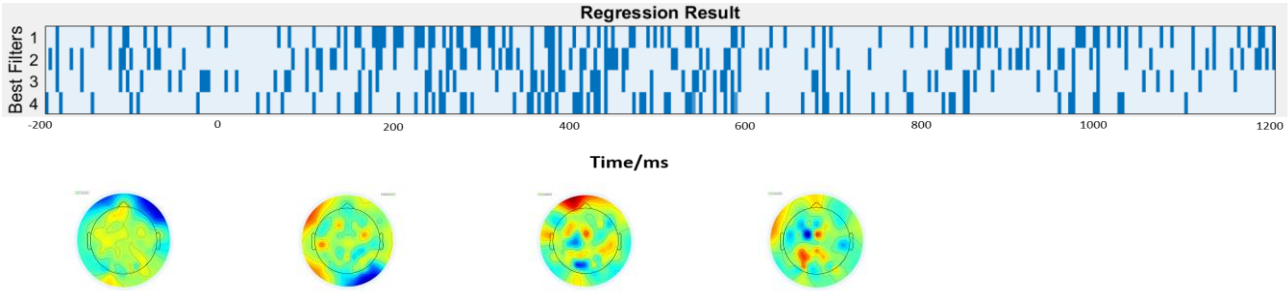


Figure 6. Regression Results and Topography. Top row shows the β values of the 4 best spatial filters' using Elastic Net regression. The blue vertical lines indicate significant time periods for regression. Bottom, row shows the topographic plots for the 4 spatial filters, indicating discrete cortical generators.

the extra features captured by MTM. The ERP heatmap is shown in **Figure 8**.

Note that differential samples had a lower amplitude in general, and the time window was much wider. From the relative location task, we learned that in the 440-550 ms time window sounds closer to attended location had larger responses (**Figure 9**). Hence, accurate prediction of the attended location depended on identifying a difference between the two curves.

Therefore, it makes sense that the low amplitude samples had worse performance in the individual model. When the sample has a lower amplitude, the gradient of different location responses also becomes lower. As a result, the curves shown in Fig 9 would be flatter in general, which makes them more difficult to be classified correctly. We also observed that the EEG gradient slopes (Figure 3, 4) increase with each non-target trial after a target ("sequence effect", unpublished observations). The sequence effect likely played a significant role in this task, and the number of misclassified samples approximately matched the expected influence of sequence effects (around 10% out of 5195). These samples are classified correctly in MTM, because the feature selected by the relative location task is supposed to have a stronger ability to distinguish amplitudes for steeper slope gradients, which would enhance the difference of the attended location curve.

V. CONCLUSION AND FUTURE WORK

This study applies CNN models to predicting stimulus and attended locations from EEG recorded during in an auditory spatial attention experiment. Our classification results based on single trial event-related potential showed good performance on both individual task model and multi-task mode. We interpreted our results with CNN visualization method, which not only confirmed auditory spatial attention features form previous work but also successfully found new joint features.

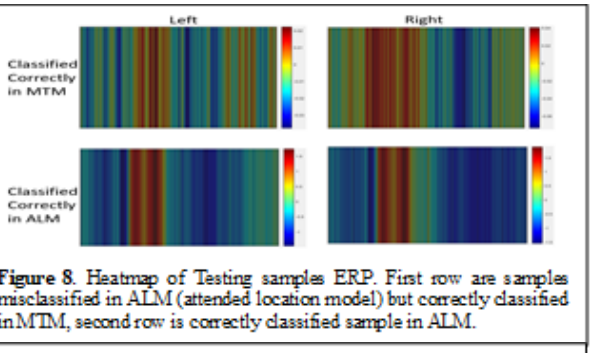


Figure 8. Heatmap of Testing samples ERP. First row are samples misclassified in ALM (attended location model) but correctly classified in MTM, second row is correctly classified sample in ALM.

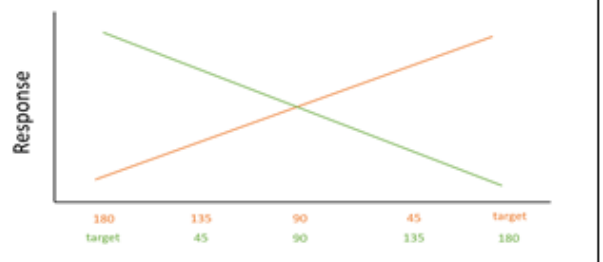


Figure 9. Schematic diagram of location response. Orange line is attending right and green line is attending left. Each relative location has a similar response, as explained in the Methods section.

References

[1] Mondor, T. A., & Zatorre, R. J. (1995). Shifting and focusing auditory spatial attention. *J Exp Psychol Hum Percept Perform, 21*(2), 387–409.

[2] Mock, J.R., Seay, M.J., Charney, D.R., Holmes, J.L., Golob, E.J. (2015). Rapid cortical dynamics associated with auditory spatial attention gradients. *Front. Neurosci*. 2015 Jun 2; 9:179

[3] Schirrmeister, R. T., et al. (2017). "Deep learning with convolutional neural networks for EEG decoding and visualization." Hum Brain Mapp 38(11): 5391-5420.

[4] Ruder, S. (2017). "An overview of multi-task learning in deep neural networks." arXiv preprint arXiv:1706.05098.

[5] Hertel, L., et al. (2015). Deep convolutional neural networks as generic feature extractors. 2015 International Joint Conference on Neural Networks (IJCNN), IEEE.

[6] Lawhern, V. J., et al. (2018). "EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces." Journal of neural engineering 15(5): 056013.

[7] Clevert, D.-A., et al. (2015). "Fast and accurate deep network learning by exponential linear units (elus)." arXiv preprint arXiv:1511.07289.

[8] Zou, H. and T. Hastie (2005). "Regularization and variable selection via the elastic net." Journal of the royal statistical society: series B (statistical methodology) 67(2):

[9] Kendall, A., et al. (2018). Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

[10] Phan, H., et al. (2017). "DNN and CNN with weighted and multi-task loss functions for audio event detection." arXiv preprint arXiv:1708.03211