Supplementary Materials

1 Optimal Gradient Blending

The theoretical rationale provided in this paper for the optimal gradient blending discussed in Section III.D closely aligns with the explanation given in [1,2]. Let $\mathcal{L}_{train}$ be the model's average training loss over the fixed training set, and $\mathcal{L}_{\diamond}$ be the true loss concerning the hypothetical target distribution. The overfitting measure at the training process at epoch n is defined as the gap between $\mathcal{L}_{train}(n)$ and $\mathcal{L}_{\diamond}(n)$, i.e. $O(n) = \mathcal{L}_{train}(n) - \mathcal{L}_{\diamond}(n)$. The quality of training between two model checkpoints at the training process at epoch n_0 and n can be measured by the changes in the overfitting measure $\Delta O(n_0, n)$ and the generalization measure $\Delta G(n_0, n)$. We can define the overfitting-to-generalization ratio (OGR) as:

$$ORG = \left|\frac{\Delta O(n_0, n)}{\Delta G(n_0, n)}\right| = \left|\frac{O(n) - O(n_0)}{\mathcal{L}_{\diamond}(n) - \mathcal{L}_{\diamond}(n_0)}\right| \tag{1}$$

While reducing the overall OGR throughout the training process is feasible, relying on this metric is not advisable. This is because very underfit models may still achieve a high OGR, which is misleading. In an alternative perspective, the objective is to address an infinitesimal problem: by combining several gradient estimates, the aim is to blend them to minimize an infinitesimal OGR^2. This ensures that each gradient step yields a gain at least as good as the single best-task network flow.

Given a single parameter update step with an estimated gradient $\hat{g}$. The distance between two checkpoints allows us to make an approximation: $\Delta O \approx \langle \Delta\mathcal{L}_{train} - \Delta\mathcal{L}_{\diamond}, \hat{g}\rangle$ and $\Delta G \approx \langle \Delta\mathcal{L}_{\diamond}, \hat{g}\rangle$. Therefore, OGR^2 for the single vector $\hat{g}$ is given as follows:

$$ORG^2 = \left(\frac{\langle \Delta\mathcal{L}_{train} - \Delta\mathcal{L}_{\diamond}, \hat{g}\rangle}{\langle \Delta\mathcal{L}_{\diamond}, \hat{g}\rangle}\right) \tag{2}$$

Let $\hat{g}^{(m)}$ be per-task gradients, $m \in \{1, 2, 3\}$, for three learning tasks in our case, obtained by back-propagation through their specific loss separately (so per-task gradients contain many zeros in other parts of the network), we then aim to blend them into a single vector with better generalization behavior.

Assuming $v^{(m)}$ represents a collection of estimations for $\Delta\mathcal{L}_{\diamond}$ which is prone to overfitting $\mathbb{E}[\langle \Delta\mathcal{L}_{train} - \Delta\mathcal{L}_{\diamond}, v^{(m)}\rangle\langle \Delta\mathcal{L}_{train} - \Delta\mathcal{L}_{\diamond}, v^{(j)}\rangle] = 0$, for $j \neq m$. Given the constraint $\sum_m w^{(m)} = 1$, the optimal weights $w^{(m)}_{optimal} \in \mathbb{R}$ for the optimization problem.

$$w_{optimal} = \underset{w}{\operatorname{argmin}}\, \mathbb{E}\left[\left(\frac{\langle \Delta\mathcal{L}_{train} - \Delta\mathcal{L}_{\diamond}, \sum_m w^{(m)} v^{(m)}\rangle}{\langle \Delta\mathcal{L}_{\diamond}, \sum_m w^{(m)} v^{(m)}\rangle}\right)^2\right] \tag{3}$$

are given by

$$w^{(m)}_{optimal} = \frac{1}{Z}\frac{\langle \Delta\mathcal{L}_{\diamond}, v^{(m)}\rangle}{\sigma^{(m)2}}, \tag{4}$$

where $\sigma^{(m)2} \equiv \mathbb{E}[\langle \Delta\mathcal{L}_{train} - \Delta\mathcal{L}_{\diamond}, v^{(m)} \rangle^2]$ and $Z = \sum_m \frac{\langle \Delta\mathcal{L}_{\diamond}, v^{(m)} \rangle}{2\sigma^{(m)2}}$ is a normalizing factor where the proof is found in the work by [1].

The answer to the optimization problem mentioned above is approximated using the multi-task architecture of MixNet. In each back-propagation step, we compute the gradients $\Delta\mathcal{L}^{(m)}$, $m \in \{1, 2, 3\}$ for individual tasks. This enables us to calculate the gradient of the weighted loss as follows:

$$\mathcal{L}(n) = \sum_{m \in \{1,2,3\}} w^{(m)}(n)\mathcal{L}^{(m)}(n) \tag{5}$$

where the blended gradient can be obtained from $\sum_{m \in \{1,2,3\}} w^{(m)} \Delta\mathcal{L}^{(m)}$. Hence, assigning suitable values to $w^{(m)}$ will provide a convenient method for implementing gradient blending through loss re-weighting.

Table S1: Classification performance (Accuracy ± SD and F1-score ± SD) in % of MixNet on BCIC IV 2b dataset using the subject-dependent and subject-independent manners comparisons on six different margins (α). Bold denotes the best numerical values.

Margins	**Subject-dependent**		**Subject-independent**	
	Accuracy	**F1-score**	**Accuracy**	**F1-score**
0.1	**76.64 ± 14.32**	**76.41 ± 14.42**	72.78 ± 10.78	72.03 ± 11.48
0.5	76.55 ± 13.89	76.17 ± 14.11	74.00 ± 10.78	73.41 ± 11.28
1	76.58 ± 13.93	76.33 ± 14.03	74.00 ± 11.04	73.55 ± 11.30
5	75.98 ± 14.66	75.58 ± 14.92	**75.02 ± 11.30**	**74.48 ± 11.78**
10	75.44 ± 14.55	74.94 ± 14.84	74.03 ± 11.25	73.28 ± 11.87
100	75.70 ± 14.81	75.23 ± 15.15	74.35 ± 10.56	73.78 ± 11.07

Table S2: Classification performance (Accuracy ± SD and F1-score ± SD) in % of MixNet on BCIC IV 2b dataset using the subject-dependent and subject-independent manners comparisons on seven different sizes of latent vector (z). Bold denotes the best numerical values.

# of latent vectors	**Subject-dependent**		**Subject-independent**	
	Accuracy	**F1-score**	**Accuracy**	**F1-score**
4	**76.72 ± 14.78**	**76.48 ± 14.82**	74.75 ± 11.02	74.26 ± 11.30
8	75.89 ± 13.94	75.56 ± 14.05	74.35 ± 11.22	73.66 ± 11.83
$U \times N_f$	76.64 ± 14.32	76.41 ± 14.42	**75.02 ± 11.30**	**74.48 ± 11.78**
32	75.57 ± 14.00	75.18 ± 14.16	74.66 ± 11.17	74.15 ± 11.51
64	76.36 ± 14.27	76.11 ± 14.36	74.22 ± 11.22	73.36 ± 12.65
128	75.30 ± 13.45	75.02 ± 13.55	74.73 ± 11.06	74.21 ± 11.50
256	74.66 ± 14.65	74.39 ± 14.75	73.66 ± 10.80	73.28 ± 11.09

Table S3: Classification performance (Accuracy ± SD and F1-score ± SD) in % of MixNet on BCIC IV 2b dataset using the subject-dependent and subject-independent manners comparisons on five different sizes of warm-up period (W). Bold denotes the best numerical values.

Warm-up	**Subject-dependent**		**Subject-independent**	
	Accuracy	**F1-score**	**Accuracy**	**F1-score**
2	76.31 ± 14.44	76.04 ± 14.53	74.86 ± 11.37	74.09 ± 12.17
3	76.58 ± 14.11	76.31 ± 14.19	**75.66 ± 10.49**	**75.23 ± 10.78**
5	76.72 ± 14.78	76.48 ± 14.82	75.02 ± 11.30	74.48 ± 11.78
7	**77.07 ± 14.59**	**76.84 ± 14.68**	74.28 ± 10.85	73.69 ± 11.23
9	76.91 ± 14.86	76.67 ± 14.95	74.09 ± 11.05	73.64 ± 11.34

References

[1] W. Wang, D. Tran, and M. Feiszli, "What makes training multi-modal classification networks hard?" in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 12 695–12 705.

[2] H. Phan, O. Y. Chén, M. C. Tran, P. Koch, A. Mertins, and M. De Vos, "Xsleepnet: Multi-view sequential model for automatic sleep staging," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 9, pp. 5903–5915, 2022.