Uncertainty-informed On-device Personalisation Using Early Exit Networks on Sensor Signals

Terry Fawden*, Lorena Qendro*†, Cecilia Mascolo*
*University of Cambridge †Nokia Bell Labs, Cambridge

***Abstract*—As their computational capabilities improve, attention has turned towards deploying deep learning models on edge devices to process the locally generated sensor signals. While these devices remain comparatively resource-constrained, early exit approaches have been shown to reduce the computational demands of on-device model personalisation training, which improves the accuracy and latency of a generalised model by fitting it to a specific use scenario. However, existing methods provide no mechanism to select the most informative signals for training. This work aims to improve prior approaches by interpreting the early exits as an ensemble of models trained with a joint loss function, retaining prior approaches' energy and latency savings while improving the accuracy. Additionally, it provides a principled mechanism to choose the signals that introduce higher uncertainty to the prediction due to the distributional shift and include them in the personalisation procedure, reducing energy consumption and latency. The key findings are a 42% energy saving with exit-only retraining versus a standard (without intermediate exits) model, which increases up to 79% when a subset of training samples was chosen according to the uncertainty estimation, alongside a 4.23pp increase in F1 score.**

***Index Terms*—Personalisation, Uncertainty, Early Exit, Sensor Signal**

I. Introduction

In systems dealing with bioelectrical or motion signals, the variation across users is particularly high due to the innate differences in body composition or altered functions caused by medical conditions [1], [2]. A deep learning model that more closely fits the distribution of a specific user is acutely desirable since it can improve accuracy and therefore functionality in the given application scenario [3]. This may be achieved by retraining the model using signals collected from the specific user in a process referred to as *personalisation* [4]. Due to their application context, many devices typically collect a multitude of specific user data; hence performing the personalisation process on-device is preferred [5]. Nonetheless, this is challenging since these devices lack the computational resources for training neural network models effectively, particularly alongside sustained use. This may be avoided with a distributed computing approach, processing the signals on an external server, but this adds latency and complexity to the system and unnecessary sharing of sensitive user data [6].

There exists a selection of methods for performing model personalisation on-device. Several transfer learning approaches have been used to personalise EEG-based affective models for brain-computer interfacing, leading to a more accurate personalised model [7]. Further, few-shot learning in gaze estimation achieved increased accuracy while running on-device in real time [8]. However, in general, these meta-learning methods require high volumes of training data, and the full adaptation of the model to the distributional shift tends to be a relatively gradual process [9].

Early exit neural networks add confidence-based intermediate classifiers to the main neural network model, which allows the inference process to complete prematurely if the early model layers produce a prediction above a given threshold [10]. State-of-the-art approaches employ various exiting strategies, including dynamic threshold adjustment and breaking down the network into smaller sequentially-executed sub-networks [11], [12]. However, these aim to optimise model execution as opposed to model training, which is the main challenge for performing personalisation on-device.

Despite the small added memory and computation cost of these classifiers, alongside the additional hyperparameter to tune caused by their application-specific placement, the inherent structure of early exit neural networks is especially suitable for performing time and energy-efficient model personalisation on-device. This has been demonstrated in recent work, which achieved higher accuracy and faster training by freezing the main model ('backbone') and just training the exits [5]. However, this method does not include a principled way of differentiating and selecting the best samples for personalisation training. Additionally, it assumes that the last exit is the most accurate, which may not be true due to network overthinking - the situation in which a correct prediction is reached before the final layer and potentially changes to an incorrect classification over the following layers [13]. Further accuracy improvements may be made by treating the model as an ensemble of early exits [14], [15].

This work aims to improve on previous approaches to on-device personalisation by interpreting the early exit neural network as an ensemble of models with a shared backbone. Training the exits with a joint loss function takes into account all neighbouring exits rather than just the accuracy of the last one as in previous work, therefore boosting the resulting accuracy while retaining the latency and energy benefits. In our evaluation, we explore the contribution of each exit to the overall accuracy, noticing that the last exit (backbone) often underperforms. This finding further highlights that the assumption considering the last exit as the most accurate is not valid and, therefore, not the best way to select the most informative signals for personalisation. To solve this issue, we

propose a novel method to further reduce energy consumption and latency by selecting a smaller set of training sensor signals according to their predictive uncertainty. Our experiments show that our approach can guarantee up to 42% energy savings when training a model with early exits compared to a standard state-of-the-art deep learning model. Further, reducing the training sample size according to predictive uncertainty resulted in energy savings of up to 79% with accuracy increases of up to 4.23pp.

II. Methods

Early exit ensembles are a collection of weight-sharing sub-networks created by adding exit branches to any backbone neural network architecture [15]. These sub-networks form an implicit ensemble of models from which uncertainty can be quantified. With only minor architectural modifications, any multi-layered feed-forward neural network $f_\theta(\cdot)$ (composed of B blocks/layers) can be converted into an implicit ensemble of networks by adding early exit blocks. The early exit block is a neural network (NN) $g_{\phi_i}(\cdot)$ with parameters ϕ_i which takes as input the intermediary output $\mathbf{h}^{(i)}$ from the i-th block of the backbone neural network $f_\theta(\cdot)$. As such, any NN can output a set $\mathcal{M}$ containing up to $B-1$ outputs from early exits blocks, in addition to the standard output from its final block

$$\mathcal{M} = \{p_{\phi_1}(y|\mathbf{x}), \ldots, p_{\phi_{B-1}}(y|\mathbf{x}), p_\theta(y|\mathbf{x})\} \tag{1}$$

where $|\mathcal{M}| = B$ denotes the number of exit blocks and, consequently, the ensemble size, $\mathbf{x}$ is the input data and $y \in \{1, \ldots, C\}$ the corresponding class labels.

We use two training paradigms to train an early exit ensemble for on-device personalisation: *end-to-end* and *exit-only* training. The end-to-end training represents a joint training of the network as a whole, including the backbone and exit blocks. In this procedure, the loss function is a composition of the individual predictive losses of each exit:

$$\mathcal{L}_{g_\phi} = \sum_{i=1}^{B-1} L_{CE}(y, g_{\phi_i}(y|\mathbf{x})) \tag{2}$$

$$\mathcal{L} = L_{CE}(y, f_\theta(y|\mathbf{x})) + \mathcal{L}_{g_\phi} \tag{3}$$

where $L_{CE}(\cdot,\cdot)$ is the cross-entropy loss function. The end-to-end training provides a general model which would cater to a large population of users, assuming that data is independently and identically distributed across different users and devices.

The exit-only training procedure includes a frozen backbone (i.e. the weights will be unchanged) of the general model while the exits are trained via Eq. 2. This training provides a personalised model and is performed on-device, preserving user data privacy. For exit-only training, we propose an uncertainty-aware sample selection approach using the uncertainty provided by the early exit ensemble of the general (non-personalised) model. For each sample, the early exit ensemble and prediction provide predictive entropy $H(y|\mathbf{x})$. This measures the uncertainty of the input personalisation data according to the distribution on which the model was initially trained, and is used to assess the informativeness of the incoming sample. Formally,

$$H(y|\mathbf{x}) = -\sum_y p(y|\mathbf{x}) \log p(y|\mathbf{x}). \tag{4}$$

Our uncertainty-aware approach provides a principled way to select the most informative samples for personalisation, and concurrently adjusts the number of chosen samples considering the device's energy requirements.

III. Experiments

Model architecture. To evaluate our on-device personalisation technique, we consider two state-of-the-art architectures, ResNet18 [16] and VGG16 [17] and openly available datasets. These implementations included 18 blocks (69 layers) with 3,846,982 trainable parameters, and 14 blocks (53 layers) with 23,822,918 trainable parameters, respectively. 4 early exits were inserted after layers 8, 24, 40 and 56 for ResNet18 and layers 4, 11, 18 and 25 for VGG16 following a 'Semantic' exit strategy and capacity factor $\gamma = 0.2$ [14]. Each exit block included average pooling, a linear layer, a ReLU activation layer, and another linear output layer to return the predictions.
Datasets. The *Epileptic Seizure* dataset [18] contains EEG signals from 500 subjects recorded for 23.6 seconds. This time series data is sampled into 4097 data points and then split into 23 segments of 178 data points (i.e. 1 second). Each of these 23 segments is labelled 1-5 according to the state of the subject, with 1 corresponding to seizure activity, 2 corresponding to a recording from a tumour location, 3 from a healthy location where the subject had a tumour elsewhere, and then 4 and 5 for the subject's eyes closed and open respectively.

The *Human Activity Recognition (HAR)* [19] dataset contains 6-axis IMU sensor signals from 30 participants recorded for several seconds each. These signals were filtered and sampled in fixed-width sliding windows of 2.56 sec and 50% overlap, and then 561 time and frequency domain variables were derived to form the training and testing vectors. Each feature vector is labelled in {walking, walking_upstairs, walking_downstairs, sitting, standing or laying} (1-6).

These datasets were further split into an 80/10/10 train/validate/test ratio. The ResNet18 model was trained and evaluated on both EEG and HAR datasets, while for the larger VGG16 model, this was done on the latter. The personalised datasets were created by separating the signal segments from 10% of the subjects from the rest of the dataset and excluding them from the initial model training, then performing the retraining with only these samples. This was chosen to make the personalisation dataset encompass 10% of the available data, which provided a reasonable balance between initial training and personalisation.
Baselines. The baselines used for comparison were standard ResNet18 and VGG16 models with no early exits, both with and without additional personalisation training. Also, each model with the early exits added was trained with all samples in the personalisation training set and used as a baseline to compare with those trained only with selected samples chosen by predictive uncertainty.

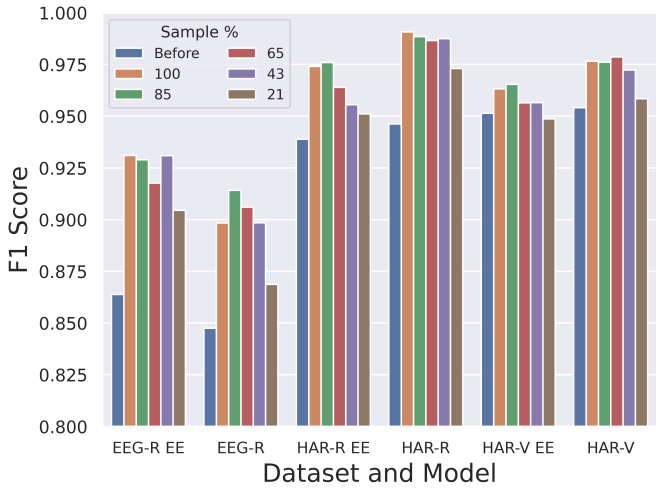
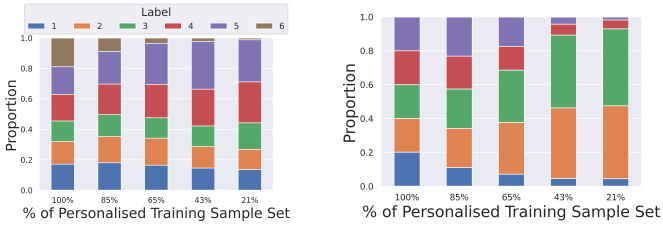


Fig. 1: The F1 score from retraining each model and dataset by the proportion of samples used for personalisation. EE represents our approach with early exits; the others are the baselines.



(a) HAR dataset. (b) EEG dataset.

Fig. 2: Sample proportions used for retraining when selecting sensor signals by uncertainty.

Setup. The initial model training was performed on the Peta4-Skylake cluster using PyTorch [20] for 800 epochs. The personalised model was trained on an Nvidia Jetson TX2, with Dual-Core NVIDIA Denver 2 64-Bit CPU Quad-Core ARM® Cortex®-A57 MPCore, 8GB 128-bit LPDDR4 Memory 1866 MHx - 59.7 GB/s [21]. We run experiments using CPU and GPU, and CPU-only to further restrict computational capacity and closely replicate the resource constraints of most tiny devices. The personalisation training was performed for 150 epochs, as this was empirically where a consistent level of validation F1 and accuracy was reached across all models. The models were trained with the Adam optimiser and a learning rate of 10^{-3}. The Jetson TX2 platform is equipped with a power monitoring chip (TI INA226 [22]) which measures voltage supply, and current draw, accessible by software - the power and energy were computed from their logs. Energy is calculated as $E = \sum_{t=0}^{T} I_t V_t$ where I_t and V_t are current and voltage measured at time t and T is the total training time in seconds. The training time was measured using the onboard clock. We perform 10 runs (with the whole test set) and average the results.

After the initial method of model retraining, the process was repeated with only a subset of training samples used, selected by their predictive uncertainty measured by predictive entropy. This was calculated before the retraining process by passing each sample through the model and calculating its predictive entropy, then subsequently reordering the samples from highest to lowest. The subset was then taken for training, with the top 21%, 43%, 65%, and 85% samples taken and trained the same way as for the full training set.

IV. Results

Classification accuracy. From inspecting Figure 1, it is clear that the F1 score obtained by the personalised model is greater than the non-personalised model for all models and signal types used by an average of 4.19pp. This indicates that the personalisation process improves the model performance independent of the model structure. However, the accuracy attained from retraining just the early exits is relatively similar to retraining the entire backbone, with the models trained on the EEG dataset performing slightly better. This was accentuated in the VGG16 model, where the backbone complexity is greater. Even so, the maximum difference across models was 3.27pp for a given sample size. The classification accuracy varies with the number of samples used for personalisation, and the general trend shows a drop-off in accuracy as this decreases. However, the point of maximum F1 varies depending on the model and dataset - this occurs when using all samples for the EEG-R EE and HAR-V models, 85% of samples for the EEG-R, HAR-R EE and HAR-V EE models, and 65% for the HAR-R model. The range in F1 scores by the number of samples was relatively small for all setups, with the biggest range of 4.54pp in the EEG-R and the lowest of 1.67pp in HAR-V EE. The ranges were 2.66pp and 2.48pp for the EEG-R EE and HAR-R EE models, respectively. Crucially, the F1 score with just 21% of the samples used for personalisation was superior to that without personalisation in all cases, aside from a slight decrease with HAR-V EE.

Sample proportions. When selecting samples based on uncertainty, it is clear that while the initial retraining set samples are evenly distributed, being increasingly selective by uncertainty favours certain labels. Inspection of Figures 2a and 2b shows that labels 2 (recording from tumour location) and 3 (recording from healthy location) appear more on the EEG dataset. In contrast, label 6 (laying) is nearly eliminated on the HAR dataset. Additionally, this is a crucial finding because it indicates that the personalised model can focus on critical classes, which would increase trust in the prediction, such as in the case of the EEG dataset. Instead, for the HAR dataset, it can easily identify activities which are very different from the others of the cohort such as laying vs walking/sitting, avoiding retraining on non-informative samples. Finally, our approach focuses the sample selection on those the model finds harder to classify and gives more useful retraining information for adapting to the user dataset.

Per-exit accuracy. Figures 3a, 3b and 3c show the F1 scores attained by each individual exit after personalisation. As a general trend, the later model exits suffer the most from personalisation, with fewer samples for the EEG-R and HAR-R. In contrast, the HAR-V shows a lower reduction in the F1 score for all exits aside from a sharp decrease in the first.

(a) EEG-R (b) HAR-V (c) HAR-R

Fig. 3: The F1 score for each exit for varying retraining sample sizes.

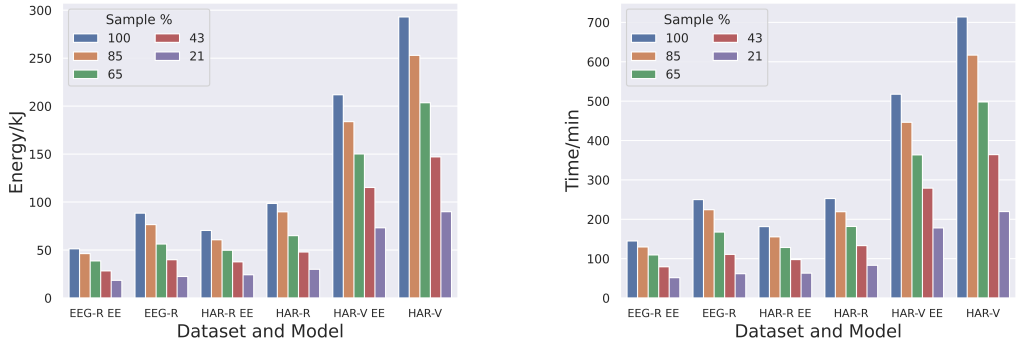


Fig. 4: CPU only: The energy and time required for retraining each model and dataset by the proportion of samples used.

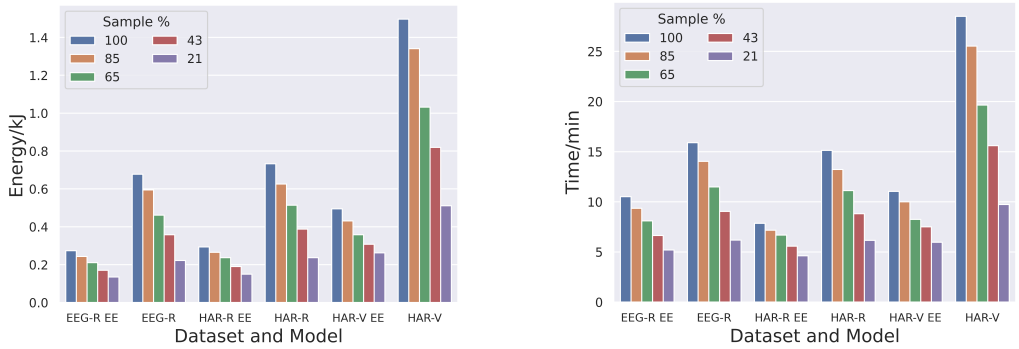


Fig. 5: CPU + GPU: The energy and time required for retraining each model and dataset by the proportion of samples used.

The poor performance of the first exit in the VGG16 model likely reflects its relatively earlier placement compared to the ResNet18 model and may explain why the overall F1 score after personalisation is relatively lower on this than the others. Overall, it would seem that the overall F1 score is determined more by the cluster of high-performing exits than by a single poor-performing exit. Inspecting the plots shows the last exit of the backbone is not always the best-performing one, and in fact performs quite poorly. We attribute this to the model overthinking issue identified in previous literature [13]. Most importantly, these findings show that choosing the samples for personalisation assuming that the last exit is always more accurate, like in previous works [5], is not valid.

Energy and latency. The training times for model personalisation by number of samples are shown in Figures 4 and 5. The speedup from retraining only the early exits is significant compared to the backbone, being 1.7×, 1.4×, and 1.4× faster when all samples are used for EEG-R EE, HAR-R EE, and HAR-V EE vs their baseline equivalents. Using CPU+GPU, these are 1.5× for EEG-R EE and an accentuated 1.9× and 2.6× for HAR-R EE and HAR-V EE. Notably, exit-only training saves 3h15' for the HAR-V model on the CPU. The speedup is approximately linear to the number of samples - since the time per operation is unchanged for a given dataset shape, the number of operations performed is the primary determinant of retraining time. The latency reduction across the models was 4.8×/3.05×, 4.0×/3.3× and 4.0×/4.8× when using 21% of the samples versus all samples on the CPU/CPU+GPU. This equates to savings of 3h18' (4h9' down to 51.7'), 3h9' (4h12' to 63'), and 8h55'(11h53' to 2h58') across the models on the CPU and up to 22.5' (28.5' down to 6') on the CPU+GPU. The speedups for the HAR-V models were greater than the HAR-R models using CPU+GPU but approximately the same when using just the CPU.

The energy usage trend by the number of samples largely mirrors that of training time since the power per operation was approximately constant for each model and dataset. The energy required per epoch was greater for the HAR dataset and the VGG16 model, with the Jetson TX2 device operating at a higher current. When all samples were used, exit-only retraining required on average 42%, 29% and 28% less energy on CPU and 60%, 60% and 67% less on CPU+GPU. Meanwhile, the savings ranged down to 79%, 75% and 75% when only 21% of the samples were used on CPU, and this was even greater using CPU+GPU with 80%, 80% and 82% energy savings. In absolute terms, this is a significant reduction from 88.4kJ to 18.3kJ (saving 70.1kJ), 98.5kJ to 24.2kJ (saving 74.3kJ) and 293kJ to 73.2kJ (saving 230kJ) on the CPU. Similarly, the energy saved is greatest for the HAR-V model when using the CPU+GPU. These savings come with a slight reduction in the F1 score for HAR-R (1.5pp) and HAR-V (1.1pp) and a 3.27pp improvement for the EEG-R. Using the sample number with the highest F1 score for each model and dataset saved 37kJ, 38kJ, and 109kJ respectively on the CPU.

V. Conclusion

This paper puts forward a new method for on-device neural network model personalisation with early exits performed on sensor signals. This method treats the early exits as an ensemble of models trained with a joint loss function, with the effect of improving model accuracy while minimising the time and energy costs of training. Additionally proposed is a technique to reduce the number of signal samples used for retraining by selecting those with the highest predictive uncertainty to save computation. The key findings were that personalisation with early exits reduced energy cost by up to 42% versus a standard neural network model. Additionally, analysis of the per-exit accuracy showed that it is not valid to assume the last exit is the most accurate and hence not the best strategy for selecting samples for personalisation. Furthermore, selecting the number of sensor signals for personalisation by predictive uncertainty resulted in energy savings of up to 79% with an increase in accuracy of 4.23pp in terms of F1 score, showing that principled selection of samples effectively provides both improved accuracy and efficiency. Finally, our approach provides scope to improve the performance of wearable devices for continuous monitoring of highly individualised human motion and biosignals.

Acknowledgement

This work was supported by the EPSRC CDT in Sensor Technologies and Applications (EP/S023046/1).

References

[1] Y. N. Singh and P. Gupta, "Correlation-based classification of heartbeats for individual identification," *Soft Computing*, vol. 15, no. 3, pp. 449–460, 2011.

[2] J. M. Hausdorff, "Gait variability: methods, modeling and meaning," *Journal of neuroengineering and rehabilitation*, vol. 2, no. 1, pp. 1–9, 2005.

[3] S. Matsui, N. Inoue, Y. Akagi, G. Nagino, and K. Shinoda, "User adaptation of convolutional neural network for human activity recognition," in *2017 25th European Signal Processing Conference (EUSIPCO)*, 2017, pp. 753–757.

[4] E. Sangineto, G. Zen, E. Ricci, and N. Sebe, "We are not all equal: Personalizing models for facial expression analysis with transductive parameter transfer," *MM 2014 - Proceedings of the 2014 ACM Conference on Multimedia*, pp. 357–366, Nov. 2014. [Online]. Available: http://dx.doi.org/10.1145/2647868.2654916

[5] I. Leontiadis, S. Laskaridis, S. I. Venieris, and N. D. Lane, "It's always personal: Using Early Exits for Efficient On-Device CNN Personalisation," *HotMobile 2021 - Proceedings of the 22nd International Workshop on Mobile Computing Systems and Applications*, pp. 15–21, Feb. 2021. [Online]. Available: https://doi.org/10.1145/3446382.3448359

[6] A. K. Talukder, L. Zimmerman, and P. H. A, *Cloud Economics: Principles, Costs, and Benefits*. Springer, London, 2010, pp. 343–360. [Online]. Available: https://link.springer.com/chapter/10.1007/978-1-84996-241-4{_}20

[7] W.-L. Zheng and B.-L. Lu, "Personalizing EEG-Based Affective Models with Transfer Learning," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)*, 2016, pp. 2732–2738. [Online]. Available: http://bcmi.sjtu.edu.cn/

[8] J. He, K. Pham, N. Valliappan, P. Xu, and C. Roberts, "On-device Few-shot Personalization for Real-time Gaze Estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019.

[9] N. Mairittha, T. Mairittha, and S. Inoue, "On-Device Deep Personalization for Robust Activity Data Collection," *Sensors*, vol. 21, no. 41, 2020. [Online]. Available: https://dx.doi.org/10.3390/s21010041

[10] S. Teerapittayanon, B. McDanel, and H. T. Kung, "BranchyNet: Fast inference via early exiting from deep neural networks," *Proceedings - International Conference on Pattern Recognition*, vol. 0, pp. 2464–2469, Jan. 2016.

[11] M. Wang, J. Mo, J. Lin, Z. Wang, and L. Du, "Dynexit: A dynamic early-exit strategy for deep residual networks," in *2019 IEEE International Workshop on Signal Processing Systems (SiPS)*. IEEE, 2019, pp. 178–183.

[12] K. Neshatpour, F. Behnia, H. Homayoun, and A. Sasan, "Exploiting energy-accuracy trade-off through contextual awareness in multi-stage convolutional neural networks," in *20th International Symposium on Quality Electronic Design (ISQED)*, 2019, pp. 265–270.

[13] Y. Kaya, S. Hong, and T. Dumitras, "Shallow-Deep Networks: Understanding and Mitigating Network Overthinking," in *International Conference on Machine Learning*. PMLR, May 2019, pp. 3301–3310. [Online]. Available: https://proceedings.mlr.press/v97/kaya19a.html

[14] L. Qendro, A. Campbell, P. Lio, and C. Mascolo, "High Frequency EEG Artifact Detection with Uncertainty via Early Exit Paradigm," in *ICML 2021 Workshop on Human In the Loop Learning*, 2021.

[15] ——, "Early exit ensembles for uncertainty quantification," in *Machine Learning for Health*. PMLR, 2021.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[18] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[19] D. Anguita, A. Ghio, L. Oneto, X. Parra Perez, and J. L. Reyes Ortiz, "A public domain dataset for human activity recognition using smartphones," in *Proceedings of the 21th international European symposium on artificial neural networks, computational intelligence and machine learning*, 2013, pp. 437–442.

[20] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, and T. Killeen, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*. Curran Associates, Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html

[21] A. A. Suzen, B. Duman, and B. Sen, "Benchmark Analysis of Jetson TX2, Jetson Nano and Raspberry PI using Deep-CNN," *HORA 2020 - 2nd International Congress on Human-Computer Interaction, Optimization and Robotic Applications, Proceedings*, Jun. 2020.

[22] B. Giovino and M. Electronics, "Making Sense of Current Sensing," *White Paper*, 2015.