EEG data alignment across devices using a neural network

Peter Johan Olsen[1,2], Andreas Tind Damgaard[1], Nannapas Banluesombatkul[1] and Kaare B. Mikkelsen[1,†]

***Abstract*— Aligning (or 'synchronizing') recordings from multiple biomedical devices is essential for accurate data analysis, but can be challenging in reality. Traditional methods, such as trigger-based synchronization or manual artifact alignment, are not always practical or reliable. In this study, we propose a neural network-based approach, to estimate the temporal offset between recordings without assuming morphological similarities. We evaluate the model on three electroencephalography (EEG) datasets: EESM17, EESM19 and Surrey-cEEGrid, each featuring different hardware setups and alignment challenges. Trained on EESM19, where perfect synchronization is available, the model generalizes well to new devices and noisy data. Our results show that this method outperforms artifact-based benchmarks and provides robust alignment. Our approach offers a promising solution for post hoc synchronization of complex biomedical data.**

I. INTRODUCTION

In many studies within biomedical engineering, multiple recording devices are used to monitor the same person. This usually results in a need to synchronize the recordings from the different devices. The need is increased by differences in startup-time between equipment and different device clock rates [1], meaning that high synchronization is hard to achieve simply by 'starting the recordings at the same time'.

In this study, 'alignment' and 'synchronization' will be used interchangeably.

A straightforward 'hardware' solution is simply to feed a trigger signal to each device; alternatively, one can simply induce simultaneous recording artifacts in all devices, which can then be found and hand-aligned during data analysis. In our experience, these approaches may sometimes fail or be unavailable — not all devices have trigger ports, and human error can cause artifact induction to fail. In such cases, methods for post hoc alignment of the datasets are needed.

The literature contains other proposed solutions to the synchronization problem, such as aligning outliers using cross correlation[2], time warping[3]–[5] and Gaussian processes[6]. However, many of these methods assume some extent of morphological similarities between the devices, or may be a bit too 'hand-held'. In this study, we investigate a more data-driven approach, wherein a neural network is trained to determine the correct lag between recordings. As such, our approach, explained in the 'methods' section, makes next to no assumptions regarding the relationship between data sets.

In this study, we specifically focus on long EEG recordings from sleep studies, because this is a particular domain where recording alignment can become an issue. Due to the nature of EEG, morphological similarity between the signals can not be assumed. Since we are focusing specifically on EEG recordings, we leverage the ability to compare similar derivations between recording devices. However, as will be seen, we also test how the method generalizes beyond this restriction.

Concretely, we focus on synchronizing polysomnography (PSG) recordings [7] with various mobile EEG data; more details can be found in the 'Data' subsection.

[1]Department of Electrical and Computer Engineering, Aarhus University, Denmark

[2]Department of Mathematics, Aarhus University, Denmark

[†]Corresponding author: mikkelsen.kaare@ece.au.dk

II. METHODS

As mentioned in the introduction, we investigate a data-driven machine learning based approach. This means feeding preprocessed data to a model (see Figure 1) and teaching it to determine whether the two data streams in question are aligned or not. By trying different offsets, the model can be used to determine what the correct offset should be.

A. Data

We train and test our approach using 3 different sleep data sets, varying the complexity and realism gradually:

1) EESM19: We trained the model using the 80 recordings with combined wearable EEG ('ear-EEG') and PSG recordings presented in Mikkelsen et al 2019 [8], [9]. This is a very suitable dataset for this task, because the wearable and PSG electrodes are combined into a single device, meaning that a ground truth is always available - the data is perfectly aligned to begin with. To mimic reality, the electrodes from each device are put into separate groups and average referenced, creating two independent groups of EEG channels.

In a realistic setting, researchers would likely be seeking to align derivations that have similar geometries. For this dataset, we represent the PSG device by the $M1-M2$ derivation, and the ear-EEG by the average of the left channels vs. the right channels: $\langle[ELE, ELI, ELT, ELA, ELB, ELC]\rangle - \langle[ERE, ERI, ERT, ERA, ERB, ERC]\rangle$.

2) EESM17: A smaller data set from the same research group, presented in Mikkelsen et al 2017 [10], [11], the primary usefulness from this dataset is the fact that the recordings were made using completely different equipment from EESM19, meaning that we can test out-of-domain model performance. Additionally, the researchers also used a shared-amplifier setup, meaning that a ground truth is available. For this data set, the 'left-right' derivations were used again, meaning that the PSG device is represented by the $A1-A2$ derivation and the ear-EEG device by $\langle[ELA, ELE, ELI, ELB1, ELB, ELG, ELK]\rangle - \langle[ERA, ERE, ERI, ERB1, ERB, ERG, ERK]\rangle$.

The original data set is 9 subjects, however, on initial inspection, one subject was found to be so noisy that it was skipped for this analysis. This rejection was done prior to running any analysis.

3) Surrey-cEEGrid: Medium-sized dataset with 20 recordings done with yet another hardware setup ('cEEGrid'), presented in Mikkelsen et al 2019[2], [12]. This is the most realistic and challenging of the three datasets, because no ground truth is available.

Besides the preprocessing mentioned below, for this dataset, we also applied a 62 Hz notch filter, removing an artifact specific to this data set.

For this dataset, the PSG was represented by 6 derivations: F3-M2, C3-M2, O1-M2, F4-M1, C4-M1 and O2-M1, and the wearable EEG was represented by $\langle[L1,L2,L3,L4,L4A,L4B,L5,L6,L7,L8]\rangle - \langle[R1,R2,R3,R4,R4A,R4B,R5,R6,R7,R8]\rangle$. For some recordings, some cEEGrid channels were missing, and were simply ignored.

For this dataset, the 'spatial ensembling', described below, was used to find the optimal offset.

In all three datasets, as an initial preprocessing step, we made sure to set all NaN-values to 0.

B. Model

The model architecture is presented in Figure 1. The input is two vectors ('A' and 'B'), which may or may not have an offset relative to each other. The first step in the model reduces the dimensionality of the input data, using a convolutional layer with a kernel size of 64 and a stride of 16. This is then followed by a max pooling layer with a kernel size of 4. This results in both input vectors being roughly 16 times shorter, and helps extracting important features in the data. Next, the two resulting vectors are stacked and fed to a Gated Recurrent Unit (GRU)[13], which functions as an encoding layer, outputting a hidden state. This is passed through three fully connected layers of sizes 128, 128 and 64, with ReLU activations between them. The output layer is of size 1 and uses a sigmoid activation function. This results in a single output value between 0 and 1, which represents the estimated probability that the two input vectors are aligned.

C. Preprocessing

Prior to feeding the data into the model, it is pre-processed in the following manner:

1) Apply a high-pass filter using a FIR filter of length 33 seconds with a lower passband edge at 0.10 Hz and a transition bandwidth of 0.10 Hz.
2) Downsample the data to 200 Hz (if necessary). This was done using the 'resample' function from the MNE library.
3) Standardize to have zero mean and unit variance for each 15-minute window.
4) If the two time series are of different lengths, pad the shorter time series with zeros.

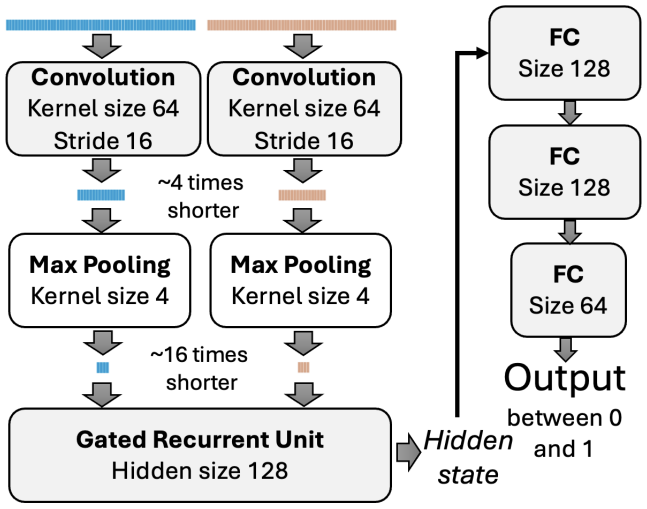

Fig. 1. Model architecture. The blue rectangle represents the input vector A and the orange rectangle represents the other input vector B. "FC" denotes a fully connected layer.

D. Model training

During training, the model is fed 15-minute data bites (180 000 samples sampled at 200 Hz) from both devices/recordings. The training process involves a mixture of aligned and artificially misaligned ('positive' and 'negative') examples, which the model must learn to differentiate. This setup shares similarities with simple implementations of contrastive learning [14], as it involves comparing pairs of inputs to learn a meaningful distinction. However, unlike typical contrastive learning approaches, the objective here is not to learn a general-purpose data representation. Instead, the classification task of determining alignment is the end goal itself, and the model is explicitly trained using supervised labels and binary cross-entropy loss.

Samples are always drawn from EESM19, for which perfect alignment is possible. For unaligned samples, an offset between 10 and 300 seconds is added between the two devices. To increase the robustness of the final model, a small offset between 0 and 0.5 seconds is used for 'aligned' examples, since we found that the model would otherwise be too prone to incorrectly assign the 'unaligned' label in real-world testing.

The model was trained using cross-entropy loss and the Adam optimizer with parameters $(\beta_1,\beta_2) = (0.9, 0.999)$, as well as a batch size of 512 and a learning rate of 0.002.

E. Evaluation

1) Search algorithm: To realistically evaluate the trained model, we must use it to actually align time series. This means devising a search algorithm that efficiently detects the region in offset-space containing the correct lag, thereby achieving automatic alignment.

We have chosen a simple, batched algorithm that works without gradient information:

(a) Try K different offsets, $\{d_j\}_{j=1}^{K}$ in a large window centered around 0.

(b) For each offset, j, get the mean of the estimated probabilities of alignment, averaged across s segments with that offset: $p_j = \frac{1}{s}\sum_{i=1}^{s} y_i$, y_i being individual model outputs. Generate weights based on the averaged probabilities: $w_j = e^{\theta(1-p_j)}$.
(c) Use the weights to estimate the most likely offset, $D = \frac{\sum_{j=1}^{K} w_j d_j}{\sum_{j=1}^{K} w_j}$.
(d) Zoom into a smaller window centered around D.
(e) Repeat, decreasing the window size each time.

In our implementation, $K = 2500, \theta = 150$, $s =$ recording duration$/15$ minutes. A sequence of four window sizes (defining the search spaces) were used: 20 minutes, 10 minutes, 1 minute, and 10 seconds.

2) Model confidence: After the last zoom, the model is evaluated at D (for which the average model output is not otherwise known). The average output is interpreted as the model's 'confidence' in the alignment.

3) Spatial ensembling: In the Surrey-cEEGrid dataset, a 'left-right' derivation is not available for the PSG recording. Instead, we estimate the correct lag for multiple derivations (all compared to the left-right cEEGrid derivation) and then choose the lag with the highest confidence.

F. Benchmark

As a realistic alternative approach, we use the 'artifact alignment' approach outlined in Mikkelsen et al 2019[2]: Artifacts are identified in each recording as those points in time where the signal amplitude exceeds a certain threshold (In practice, 2 standard deviations is used). By storing the artifact locations in binary vectors, i.e., an element is 1 if it contains an artifact and otherwise 0, the two recordings can be aligned by finding the highest cross-correlation between the two binary vectors. This corresponds to aligning the recordings by aligning their artifacts.

To be as realistic as possible, we only consider offsets of +/- 1 hour, since it seems unlikely to be performing recordings with less timing knowledge than that. In the case of the Surrey-cEEGrid dataset for which multiple derivations are used, the peak cross correlation was used as 'confidence'.

III. Results

Figure 2 shows an example run of the algorithm using a recording from the test split in Surrey-cEEGrid. The black dots each represent the output from a single run of the model (so, the estimated probability of alignment based on a single 15-minute section). The red line is is p_j as defined above, and the vertical line indicates estimated best offset.

Figure 3 shows an overview of the distribution of alignment errors for all three datasets, using both our approach and the benchmark method. Focusing on EESM17 and EESM19, we see that our model reliably performs correct alignment, whereas the benchmark method occasionally stumbles. Moving on to the Surrey-cEEGrid dataset, quantifying errors is a bit more challenging, since there is no ground truth. The authors of the dataset have proposed a manual synchronization, obtained via a procedure similar to

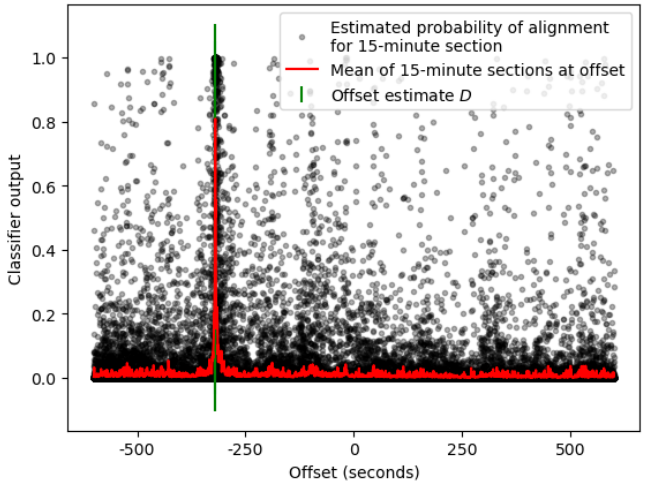

Fig. 2. Example of the output from the first iteration of the algorithm. The red line shows the average classifier output for a given lag, and we see that the model has a quite high confidence for an offset of about -260 seconds. As described above, the algorithm can be rerun with a smaller field of view to increase the resolution.

our benchmark method. Out of 17 recordings, our method agrees with the manual alignment in 12 cases (in this case, 'agreement' means a difference less than 5 seconds). Upon closer inspection of the remaining 5 recordings, we actually believe that our approach gives a more convincing alignment than that suggested by the authors in 4 cases. Figure 4 shows an example of manual and automatic alignment. We see that the motion artifacts for this 30-minute segment align better using the proposed offset based on our automatic alignment. In the remaining recording, we concede that the manual alignment is more convincing, meaning that our approach gives an error of 23 seconds.

IV. Discussion and Conclusion

We consider the results very promising. The Surrey-cEEGrid data set is the most challenging, but it is also an overall hard dataset to work with (there are many recording artifacts), and we are missing a reliable ground truth. Apart from these problematic examples, our approach is reliable and outperforms the benchmark method.

A. Future directions

While the present model largely 'gets the job done', there are multiple interesting directions to take the work further:

How dissimilar can the EEG sources be before the approach breaks down? And what if we were even comparing dissimilar modalities, such as EEG and actigraphy?

Can we learn what the network is basing its decisions on? We went into the project expecting a high reliance on movement artifacts, but the model also manages correct decisions for data sequences without any artifacts.

Along these lines, we have not yet implemented a flag for 'indecisiveness' in the model output, i.e., how to know when *not* to trust the output. We expect something simple like the area under the curve in Figure 2 might be a good first start,

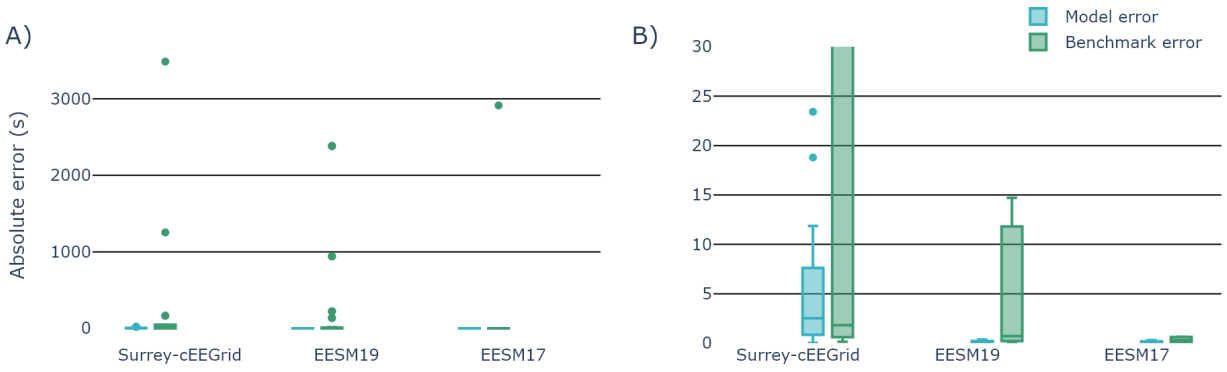

Fig. 3. Distributions of alignment errors for our proposed model and the benchmark method. A: full picture, with outliers all the way up to 3500 seconds. B: zoomed in to show the main distribution. Largest error for the model is 23 seconds, 5 values are above 5 seconds.

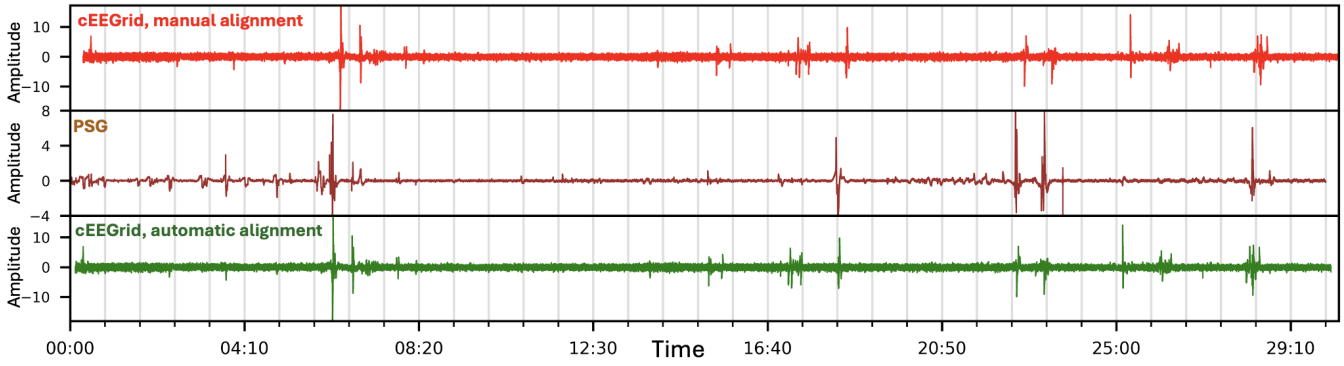

Fig. 4. Comparison of manual and proposed automatic alignment. Vertical lines added as visual guide. Comparing the alignment of artifacts between the two proposed alignments, we see a better match to the bottom plot, indicating a better synchronization / alignment.

but at the time of writing we have not had the chance to investigate further.

Finally, it seems likely that for optimal performance, the model should incorporate time-varying offsets. This would allow handling situations where the devices have markedly different clock rates, or where one device is dropping samples, for instance.

V. Code availability

We have packaged our work here, both the final product and the components needed to perform local development, in a GitLab repository, available at `https://gitlab.au.dk/tech_ear-eeg/sleep-code/signal-alignment`.

VI. Acknowledgements

The authors acknowledge the Danish e-infrastructure Consoritum for awarding this project access to the LUMI supercomputer, owned by the EuroHPC Joint Undertaking, hosted by CSC (Finland), under grant DeiC-AU-L5-0024.

References

[1] N. Schütz, A. Botros, M. Single, A. C. Naef, P. Buluschek, and T. Nef, "Deep canonical correlation alignment for sensor signals," 2021.

[2] K. B. Mikkelsen, J. K. Ebajemito, M. A. Bonmati-Carrion, N. Santhi, V. L. Revell, G. Atzori, C. della Monica, S. Debener, D.-J. Dijk, A. Sterr, and M. de Vos, "Machine-learning-derived sleep–wake staging from around-the-ear electroencephalogram outperforms manual scoring and actigraphy," *Journal of Sleep Research*, vol. 28, no. 2, p. e12786, 2019. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/jsr.12786

[3] A. Fu, E. Keogh, L. Lau, C. Ratanamahatana, and R. Wong, "Scaling and time warping in time series querying," *The VLDB Journal — The International Journal on Very Large Data Bases*, vol. 17, pp. 899–921, 2008.

[4] D. Folgado, M. Barandas, R. Matias, R. Martins, M. Carvalho, and H. Gamboa, "Time alignment measurement for time series," *Pattern Recognition*, vol. 81, pp. 268–279, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031320318301286

[5] T. Chen, M. Abdelmaseeh, and D. Stashuk, "Affine and regional dynamic time warping," *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, pp. 440–448, 2015. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/ICDMW.2015.124

[6] N. Suematsu and A. Hayashi, "Time series alignment with gaussian processes," *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pp. 2355–2358, 2012.

[7] R. B. Berry, R. Brooks, C. Gamaldo, S. M. Harding, R. M. Lloyd, C. L. Marcus, and B. V. Vaughn, *The AASM Manual for the Scoring*

of Sleep and Associated Events: Rules, Terminology and Technical Specifications, 2nd ed. American Academy of Sleep Medicine, 2016.

[8] K. B. Mikkelsen, Y. R. Tabar, S. L. Kappel, C. B. Christensen, H. O. Toft, M. C. Hemmsen, M. L. Rank, M. Otto, and P. Kidmose, "Accurate whole-night sleep monitoring with dry-contact ear-eeg," *Scientific Reports*, vol. 9, no. 1, p. 16824, 2019.

[9] K. Bjarke Mikkelsen, Y. Rezai Tabar, L. Rævsbæk Birch, S. Lind Kappel, C. Bech Christensen, L. Dalskov Mosgaard, M. Otto, M. Christian Hemmsen, M. Lind Rank, and P. Kidmose, "Ear-EEG sleep monitoring data sets," *Scientific Data*, vol. 12, no. 1, p. 301, Feb. 2025, publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41597-025-04579-8

[10] K. B. Mikkelsen, D. B. Villadsen, M. Otto, and P. Kidmose, "Automatic sleep staging using ear-eeg," *BioMedical Engineering OnLine*, vol. 16, no. 1, p. 111, 2017.

[11] K. B. Mikkelsen, D. B. Villadsen, L. Birch, M. Otto, and P. Kidmose, "Ear-EEG Sleep Monitoring 2017 (EESM17)," 2024. [Online]. Available: https://openneuro.org/datasets/ds004348/versions/1.0.5

[12] K. B. Mikkelsen, J. K. Ebajemito, M. A. Bonmati-Carrion, Nayantara Santhi, V. L. Revell, G. Atzori, L. Birch, C. D. Monica, S. Debener, Derk-Jan Dijk, A. Sterr, and M. De Vos, "Surrey cEEGrid sleep data set," 2024. [Online]. Available: https://openneuro.org/datasets/ds005207/versions/1.0.0

[13] J. Chung, Ç. Gülçehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *NIPS 2014 Deep Learning and Representation Learning Workshop*, 2014.

[14] P. H. Le-Khac, G. Healy, and A. F. Smeaton, "Contrastive representation learning: A framework and review," *IEEE Access*, vol. 8, pp. 193 907–193 934, 2020.