GI-SleepNet: A Highly Versatile Image-based Sleep Classification using a Deep Learning Algorithm

Tian Xiang Gao

Nagoya City University

Jia Yi Li

Nagoya City University

Yuji Watanabe

Nagoya City University

Chi Jung Hung

Nagoya University

Akihiro Yamanaka

Nagoya University

Kazumasa Horie

University of Tsukuba

Masashi Yanagisawa

University of Tsukuba

Masahiro Ohsawa

Nagoya City University

Kazuhiko Kume (✉ kume.kazuhiko@gmail.com)

Nagoya City University

Research Article

Keywords: sleep staging, image processing, machine learning, convolutional neural network (CNN), generative adversarial network (GAN)

Posted Date: June 10th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-586116/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

GI-SleepNet: A highly versatile image-based sleep classification using a deep learning algorithm

Tian Xiang Gao¹, Jia Yi Li¹, Yuji Watanabe², Chi Jung Hung³, Akihiro Yamanaka³, Kazumasa Horie^{4,5}, Masashi Yanagisawa⁶, Masahiro Ohsawa¹, Kazuhiko Kume^{1,*}

1. Department of Neuropharmacology, Graduate School of Pharmaceutical Sciences, Nagoya City University, Nagoya, Japan
2. Graduate School of Science, Nagoya City University, Nagoya, Japan
3. Department of Neuroscience II, Research Institute of Environmental Medicine, Nagoya University, Nagoya, Japan
4. Graduate School of Systems and Information Engineering, University of Tsukuba, Tsukuba, Japan.
5. Center for Computational Sciences, University of Tsukuba, Tsukuba, Japan.
6. International Institute for Integrative Sleep Medicine, University of Tsukuba, Tsukuba, Japan

* Correspondence should be addressed to : kume.kazuhiko@gmail.com

Tanabe 3-1, Mizuho, Nagoya 467-8603 Japan

Key words: sleep staging, image processing, machine learning, convolutional neural network (CNN), generative adversarial network (GAN)

Acknowledgements: We are grateful to the members of Kume lab for their valuable comments and encouragements. We also thank Editage (www.editage.com) for English language editing. This work was supported in part by Japan Society for the Promotion of Science (JSPS) KAKENHI Grant Numbers JP18H02481, K.K.

Conflict of interest statements: The authors declare no conflict of interest for this work.

Abstract

Sleep-stage classification is essential for sleep research. Various automatic judgment programs including deep learning algorithms using artificial intelligence (AI) have been developed, but with limitations in data format compatibility, human interpretability, cost, and technical requirements. We developed a novel program called GI-SleepNet, generative adversarial network (GAN)-assisted image-based sleep staging for mice that is accurate, versatile, compact, and easy to use. In this program, electroencephalogram and electromyography data are first visualized as images and then classified into three stages (wake, NREM, and REM) by a supervised image learning algorithm. To increase the accuracy, we adopted GAN and artificially generated fake REM sleep data to equalize the number of stages. This resulted in improved accuracy, and as few as one mouse data yielded significant accuracy. Because of its image-based nature, it is easy to apply to data of different formats, of different species of animals, and even outside of sleep research. Image data can be easily understood by humans, thus especially confirmation by experts is easy when there are some anomalies of prediction. Because deep learning of images is one of the leading fields in AI, numerous algorithms are also available.

Introduction

Sleep is a stable systemic state that in mammals is controlled by homeostasis and endogenous circadian rhythms. Sleep is characterized by electroencephalogram (EEG), which was first developed by Hans Berger in 1924¹. He called EEG a “brain mirror,” which reflects the “electrical psychic energy” within cortical tissue. He analyzed the wave phase patterns and described the α and β waves. Over a decade later, Alfred Loomis showed that the human EEG patterns dramatically changed from waking to sleep stages². Loomis initially classified sleep into five stages (A, B, C, D, and E), which are primarily manifested by the characteristic patterns of the α and spindle waves. Initially, the characterization and transition of brainwave frequencies were considered essential features. In his study, Loomis used the six-channel EEG of 30 s because the record sheets were automatically cut by a scissor every 30 s, and this marked the earliest conceptual origin of the classification epoch. These separate epochs were visually judged by researchers in a manner similar to the workflows conducted by modern polysomnography (PSG) technicians.

After the rapid eye movement (REM) sleep stage was discovered by Aserinsky and Kleitman in 1953, electrooculography (EOG) and mentalis muscle electromyography (EMG) were used for sleep classification. Rechtschaffen and Kales then set up the PSG criteria in 1968³, which has been widely used until now with only minor modifications. However, without either EOG, EMG, or the automatic integral calculation method being used for relative band powers, Loomis’s sleep classification criteria in the 1930s very closely resembled the modern one, suggesting that EEG patterns play a more critical role than any other channel. In addition, visual judgments by technicians remain important for classifying sleep stages.

In addition to the human PSG, considerable demand exists for research on rodent sleep data. The classification criteria of the sleep stages of mice are different from human PSG classification. The murine nonrapid eye movement (NREM) stage shows low EMG amplitude and high EEG δ -wave power, and NREM is classified as one stage without any further subdivision. The REM stage shows a higher θ -wave power than any other frequency band. Thus, three sleep stages, namely Wake, NREM, and REM, have typical individual features in the EEG power spectrum. Researchers of murine sleep usually use an automatic scoring commercial software such as SleepSign (Kissei Comtec Co., Ltd.)⁴, a MATLAB advanced toolbox like EEGLAB, and others⁵. However, these processing tools may present some obstacles for new researchers because of their price or high demand for programming skill.

Thanks to the technical advances in machine learning, we have the opportunity to utilize artificial neural networks to study the wake-sleep cycle activities generated by natural neural networks from the past 10 years. An unsupervised

43 algorithm known as FASTER⁶ (Fully Automated Sleep sTaging method via EEG/EMG Recordings) attained
44 prominence even before the first TensorFlow beta version was released in 2015. FASTER calculates the power
45 spectrum of both EEG and EMG and performs clustering for power spectrum values using principal component
46 analysis. The sensitivity performances of NREM and Wake states is comparatively fine. However, because the
47 clustering of rare events (REM) for “hard” rule classical clustering analysis is complex, the sensitivity of REM is low
48 and unstable in different experimental environments.

49 After TensorFlow was released, most of the algorithms were aimed at human PSG, but later these human-based
50 approaches were found to be instructive for other mammalian sleep studies. In 2017, Guo et al. open-sourced the
51 DeepSleepNet model for EEG single-channel-based sleep stage scoring⁷, which was trained by the Sleep-EDF dataset
52 for humans. Before DeepSleepNet, most classification methods were dependent on complex calculations for
53 extracting band power features. However, the DeepSleepNet model works without utilizing any hand-engineered
54 features by merging the two branches (EEG & EMG) of a convolutional neural network (CNN) and bidirectional
55 long short-term memory (Bi-LSTM) cells.

56 Recently, MC-SleepNet was created for sleep stage scoring in mice⁸, which DeepSleepNet inspired with the
57 addition of EMG. The results of performance analysis, excluding the low precision of REM on small-scale datasets,
58 revealed that MC-SleepNet was superior. However, for laboratory-level studies, particularly for some rare transgenic
59 strains that are not easily propagated, performance on small-scale datasets is also important. This is also true of large-
60 scale datasets, particularly for research related to REM sleep anomalies in mice⁹. The problem with the one-
61 dimensional CNN is its weakness in outlier detection, especially when applied to sleep studies. This is considered a
62 cause of DeepSleepNet’s low sensitivity for N1, as the N1 stage is short and contains various noises.

63 The CNN was originally developed to analyze two-dimensional (2D) image data. Thus far, 2D image-based CNN-
64 LSTM networks have been applied very successfully in the field of medical diagnostic imaging^{10,11}. This strategy has
65 also been shown to be successful in the analysis of phenotypic convergence for taxonomy of species such as
66 butterflies¹². Analysis of the features of the images extracted by a 2D CNN even showed that identifying the migration
67 patterns between phases was possible. This type of parsing method not only can classify all discrete data, it can also
68 provide a visual interpretation of the transformations between various stages and the data relationships within each
69 stage group.

70 In this study, to utilize the full power of a 2D CNN for sleep stage classification, we developed a novel method

called GI-SleepNet, a GAN-assisted Image-based program, to process EEG and EMG data. In the first step, we produce an image file corresponding to each classification epoch, which is composed of an EEG power spectrum plot and EMG raw wave data graph of that epoch. We then manually classify these epoch images into three stages: Wake, NREM, and REM. We then use a 2D CNN for supervised learning of the designated images. GI-SleepNet precisely follows the logic and workflow of expert technicians, and identifying what the machine is learning is easy. In addition, accumulated knowledge about and numerous methods on deep learning of images are readily applicable.

Methods

Training hardware and software

We used the following computer hardware and software environment.

Computer 1: (i7 9700K/GeForce RTX2070 super/64 GB) Anaconda Python 3.7; TensorFlow 2.1.0;

Computer 2: (i7 9850HK/Quadro RTX3000/64 GB) Anaconda Python 3.7; TensorFlow 2.1.0;

Computer 3: (i9 10900KF/GeForce RTX3090/128 GB) Anaconda Python 3.8; TensorFlow nightly version (tf-nightly-gpu).

Computers 1 and 2 were mainly used for image dataset preparation and classification training. Computer 3 was used for image dataset preparation and GAN training to generate fake REM images.

The classification training strategy involved using the TensorBoard to monitor the metrics and perform early stopping when necessary.

Animal experiment

Experiments using mice were approved by the ethical committee board of Nagoya City University and were conducted by following the guidelines of the Animal Care and Use Committee of Nagoya City University and the National Institutes of Health and the Japanese Pharmacological Society. This manuscript is written following the recommendations in the ARRIVE guidelines¹³.

Sleep data

Two separate mouse datasets were used in this study. We used our own datasets for preliminary algorithm development. EEG and EMG signals from male *C57BL/6J* mice 10 to 14 weeks old (Clea Japan Inc., Japan) were recorded as described previously⁹. Briefly, mice were anesthetized with isoflurane, and stainless-steel screws and wires were surgically implanted in the skull and into the trapezius muscle, respectively. They served as electrodes and were connected to a microtip amplifier (Intan, RHD2216, 16-channel amplifier chip with bipolar inputs) and an Open Ephys acquisition board for recording¹⁴. All data were recorded at least one week after the electrode implant surgery. EEG and EMG signals from four mice were saved in a digital format file for further processing. For the performance test of our new algorithm, we used a small-scale dataset (Tsukuba-14 dataset) from a previous study⁸. The Tsukuba-14 dataset contains data segments from 14 mice, and each segment contains four days of data (17,280 epochs of 20 s) for a single mouse.

Prediction and calculations

The prediction model is presented in Fig1, Fig2, and Fig3, and all the raw prediction results are presented in the Supplemental Table (Microsoft Excel file). Values of scoring valuation scale (accuracy, recall, F1-score, etc.) we showed in the data table are the average values of the 14 (or the 10 for the ting dataset valuation) individual mice. The customized calculation codes are performed based on the Python library Scikit-learn.

Results

Data image production

To develop an image-based process, we first determined the format of a single image. All images were generated using Python's Matplotlib and Seaborn libraries. The upper and lower parts of a single data image were the EMG raw data graph and heatmap of the EEG power spectrum, respectively (Fig. 1A). The latter was calculated by fast Fourier transform (FFT) and normalized by Python's Scikit-learn library. The size of the original image was 800×800 pixels. The image had 32-bit color depth, although all the images were produced at gray scale. All of the marks on the horizontal and vertical coordinates as well as the color bar of the heatmap remained on the images,

which help human visual perception and do not interfere with machine learning, as they are identical in all images. The values of both the horizontal and vertical coordinates were set to a constant between images in advance.

We created two image datasets with different data period lengths (Fig. 1B). One of them contained one epoch (20 s) of EEG/EMG information, whereas the other contained two epochs (40 s) consisting of the epoch of interest and the preceding epoch. For machine learning, we scaled down the image size.

Selection of the appropriate network structure from pretrained models

For preliminary work, to confirm whether sleep scoring using the created images worked effectively, we constructed our own small image dataset using EEG and EMG data from *C57BL/6J* mice. In this trial, the input size of the images was set to 800×800 pixels. After trying some transfer learning models such as DenseNet (acc = 53%), MobileNet (acc = 67%), and ResNet (acc = 78%) on our dataset, we found that VGG-19 (acc = 94%) had good potential. In order to reduce the amount of data to be calculated, we tried to reduce the input size and found that the performance could still be maintained at 180×180 . The structure is quite similar to vgg19, both have five blocks of 2D-CNN to extract images information. We then added four Dense layers and two Dropout layers at the ends of the networks to prevent overfitting. (Fig. 1C).

Expansion of the dataset by GAN

The ratio of the three stages of an ordinary mouse is approximately (Wake: NREM: REM) 10:10:1 under conventional experimental conditions. Thus, we suspected that the low precision of REM by the existing algorithm was due to an imbalance in the number of stages in sleep datasets. The small sample size of the REM might have reduced the precision of REM, particularly on the small-scale dataset⁸, a problem that must be solved. Thus, we decided to increase the number of REM epochs.

Instead of increasing the size of the actual dataset, which is time-consuming and laborious, we increased the size of the REM epoch with artificially produced fake REM data by designing a REM data generator using a deep convolutional generative adversarial network (DCGAN). GANs for data augmentation with medical image data have been widely used¹⁵. Because low-resolution images are difficult to check, we tried to improve the resolution of the generated image to 512×512 (Fig. 2A). Because it is difficult for a standard DCGAN model to generate high resolution images, we chose an advanced Wasserstein GAN with gradient penalty (WGANGP) model, which was

originally described for that famous CelebA face-dataset training in the O'Reilly series book "Generative Deep Learning" (Chapter 4.6)¹⁶. The Generator of WGANP could be considered as a reverse version of our classifier. In this book, the original version only has five blocks with the 128×128 output size. We modified this structure and add another two blocks to make it accommodate our high- resolution output demand (Fig. 2B). Accordingly, we also improved the depth of discriminator depth. (Fig. 2C).

Performance of the newly developed algorithm and its comparison with previous algorithms

After debugging on our small dataset, we evaluated the model's fitting performance on another dataset compared to current advanced models such as MC-SleepNet. We thus created images using Tsukuba-14 datasets. Because we expected that redundant information would be beneficial to discriminate data in sleep stage transition, we created both one- and two-epoch datasets. This strategy is regarded as an extremely simplified version of LSTM, in which the "short memory" has only one previous set of epoch data. We also increased the REM data using the WGANP. We examined three datasets, namely the one and two-epoch datasets and the WGANP-adjusted two-epoch dataset. Overall, our model performed nearly as well as or even slightly better in terms of accuracy, and Cohen's κ as compared with MC-SleepNet (Fig. 3A). The huge improvement in F1 score is thought to benefit from the higher recall of REM. WGANP adjustment with fake REM images increased the overall accuracy. Even without this adjustment, the precision of REM on the two-epoch version maintained a high level similar to that of MC-SleepNet on large-scale data. We believed this is because the image spectral features of REM are conducive to being identified.

To understand more intuitively how our classifier recognizes the interrelationships between the three stages within the model, we extracted the output information from both the first (128) and last (64) dense layers. The first dense layer is thought to collect all the feature information extracted from the CNN, while the last dense layer would integrate all the information for the final classification. Because that information is high-dimensional, to clearly see the distribution of each stage within the classification processing, dimensionality reduction is necessary. UMAP(Uniform Manifold Approximation and Projection) is an effective dimension reduction and clustering tool, we embedded the output information of these dense layers into three components and visualized them in 3D space. We changed the parameters of `n_neighbors` from 5 to 100 (Supplemental Fig. 1). The results were insightful. When `n_neighbors` were set at 75 (Fig. 3C), on the first dense layer, the three datasets showed variant distributions for the network and behaved consistently with scoring performance such as precision or recall. For the one-epoch dataset,

the wake and sleep stages were completely separate, but many cases of NREM near to REM were observed, which is why the precision of sleep on the one-epoch dataset was the highest (Fig. 3C, left). On the two-epoch dataset, REM exhibited a stick-like clustering and was connected with NREM, which matched the actual situation, as all phase-transition points of the sleep-activity cycle could be presented. NREM is always ahead of REM in time series, so the REM is only connected to NREM but not Wake (Fig. 3C, center). The most exciting aspect was the performance on the WGANP-adjusted dataset. Because the fake REM data balanced the entire dataset, even the NREM and wake stages were in remarkable balance with each other, whereas the REM closely resembled a small branch growing on NREM, which was consistent with to reality. Wake also became closer to the REM, which could be regarded as some mid-wake points during the sleep that often occur after REM (Fig. 3C, right).

To evaluate how our fake REM images are compared to the actual data, we performed visualization using the above processing. On the first and the last middle dense layer, with the the `n_neighbors` of 75, we could observe that the fake REM and REM are completely combined and widespread in UMAP 3D space (Supplemental Fig. 2AB). This distribution indicated two things: one is that the neural network successfully classifies real and fake images into one category. The other is our fake REM has relatively high diversity, which is considered to be handled with care when using DCGAN.

197

198 **Effects of different epoch lengths**

Another advantage of using images for judgment is that even images with some missing information can also be recognized. To test the performance on non-standard datasets, we created images of different epoch lengths. As Fig. 4A shows, we shortened the epoch length from 40 s (two epochs) to 20 s (one epoch) by 5 s and then classified the images using our algorithm. The accuracy gradually decreased with the shortening of the epoch length, but even the shortest version (20 s) showed good performance ($\text{acc} = 93.88\%$, $\kappa = 0.8954$) (Fig. 4BC). This indicated the robustness of our algorithm, and it could be helpful in some specific situations, such as small-scale data breakage or short-term sampling instability. All of the shortened version results revealed a high grade of recall for NREM and wake states, whereas the recall for REM also showed an expected gradual decline with the shortening of length.

207

208 **Performance on a tiny dataset**

Practically speaking, for most laboratory-level studies, performance on small-scale datasets is essential, as

customizing the algorithm for different mouse strains to improve the accuracy of quantification is required. To investigate the performance on tiny datasets, we randomly selected a single-mouse data, two mice, three mice and four-mice data segments for use as the training dataset and then applied the trained model to the other 10 mouse segments for use as the evaluation dataset. Although the results from using a single training segment dataset was not satisfactory, the performances of two, three, and four training segment datasets were considered satisfactory, and the recall of REM also remained at an acceptable level (Fig. 5). In addition, we expanded the REM data of the single-mouse dataset using WGANP. This resulted in higher accuracy but was still inferior to the two-mice datasets.

To improve the single-mouse data performance, we increased the number of images in all three stages by generating fake images of the wake, NREM, and REM stages (Fig. 6A). on a tiny dataset of a single-mouse segment, the accuracy increased when we expanded all three-stage data with fake images (Fig. 6BC). Although the accuracy of 93.96% remained unsatisfactory, this value was quite impressive for such a tiny dataset. In order to confirm if GAN is better than simple increase of REM images, we tested another dataset with simple addition of the 10 times number of REM images within a single mouse dataset, and the result was not improved at all.

Post-prediction filters for further improvement of accuracy

In the final step, we found that when the accuracy was higher than 95%, certain points of error became conspicuous and common errors could be corrected. For example, because the wake-stage EEG power spectrum was sometimes quite similar to the REM stage, some “REM stages” could be predicted within a continuous wake stage period, particularly when the mouse was just resting (i.e., not sleeping). To solve this problem, we designed a smoothing filter to remove atypical short REM periods from wake periods. The filters could also be highly customizable when having to adhere to some special experimental requirements. With these filters, the scoring performance was higher than previously (Supplemental Fig. 3). It should note that this processing does not work on algorithms with low accuracy or the datasets of poor labelling quality, and can be counterproductive.

Discussion

GI-SleepNet, the novel image-based learning algorithm we developed in this study, has several advantages over the conventional numerical data-based algorithm. First, the format of the data has excellent flexibility. In our case,

one image has both raw EMG data and EEG frequency power spectrum data. EMG and EEG data differ between laboratories with different sampling rates and different data formats. Epoch lengths also differ among animal species. For human PSG, EOG should be included along with EMG and EEG. However, these differences do not matter once all the data are formatted as a single image. Thus, this method is readily applicable to any species and even outside of sleep research. Second, the image data that the machine learns has high interpretability. It is instinctively comprehensive, and each image contains sufficient visual information to classify it into three categories by researchers. Thus, it is easy to create training datasets manually and perform post-prediction analysis. Following automatic classification by machine, it is also easy to confirm the results and find and solve errors. Third, because image recognition is one of the most advanced fields in AI machine learning, it is easy to find sophisticated algorithms and find recent progress. Therefore, we included the GAN method to adjust the REM data by producing fake REM images. Fourth, because the size of one set of data is limited to a 2D-image extent and because the image processing algorithms are optimized, our method requires relatively low computing power and short processing time. Fifth, our method exhibits high accuracy with small datasets, making it useful in practice. It performed well on both our own dataset and those from different laboratories, even though these data were recorded using different types of acquisition equipment. This means it could be re-used easily on small datasets of specific strains or transgenic animals, which may exhibit atypical EEG patterns. Thus, applying it to precious animal strains from which only a limited amount of data is available can also be advantageous to researchers.

We anticipate that researchers themselves can customize our model. To make our image-based scoring system easier to use, we developed a graphical user interface (GUI) for research purposes based on the Python binding GUI toolkit Tkinter (Supplemental Fig. 4). Our GUI includes semi-automatic data-preprocessing, the large-scale output of plotting images, neural network training, and prediction functions. We also released several trained h5 files for immediate use without a training dataset. The only action for individual researchers who wish to use our model is to create their own datasets and fit them to our shared network structure. Finally, the DCGAN-created images and forced filters based on our own dataset have also been packaged in the GUI. If this system can be used by a greater number of researchers, we hope to collect considerably more data from different devices and further improve the noise resistance of the algorithm.

Image-based scoring systems may be applied to identify not only sleep activity but also other physiological and pathological events such as epileptic seizures or preclinical Alzheimer's disease symptoms. For example, one study

266 has been conducted showing the differences in the prefrontal cortex EEG power spectrum during Y-maze tests
267 between normal mice and Alzheimer's disease models. A better understanding and design of experiments may be
268 beneficial if creating image datasets of different performance groups in behavioural tests.

269 Our method does have some limitations. First, a priori knowledge is required to design images. Second, producing
270 images is initially time-consuming. We used EMG and EEG signals of nearly the same size in the case of our dataset.
271 However, the ratio of sizes can be changed, which affects the accuracy. However, to change the ratio, ensuring that
272 all datasets contain all images for all epochs is required, a process that is time-consuming and laborious. Despite
273 these limitations, we believe that our novel algorithm will provide a versatile tool for future research in many fields.

274

275

276 **References**

- 277 1. Tudor, M., Tudor, L. & Tudor, K. I. [Hans Berger (1873-1941)--the history of electroencephalography]. *Acta*
278 *Med. Croatica* **59**, 307–13 (2005).
- 279 2. Loomis, A. L., Harvey, E. N. & Hobart, G. A. Cerebral states during sleep, as studied by human brain potentials.
280 *J. Exp. Psychol.* **21**, 127–144 (1937).
- 281 3. Rechtschaffen, A. & Kales, A. *A manual of standardized terminology, techniques and scoring system of sleep*
282 *stages in human subjects*. (Public Health Service, US Government Printing Office, 1968).
- 283 4. Chowdhury, S. *et al.* GABA neurons in the ventral tegmental area regulate non-rapid eye movement sleep in
284 mice. *Elife* **8**, 1–27 (2019).
- 285 5. Rytkönen, K.-M., Zitting, J. & Porkka-Heiskanen, T. Automated sleep scoring in rats and mice using the naive
286 Bayes classifier. *J. Neurosci. Methods* **202**, 60–4 (2011).
- 287 6. Sunagawa, G. A., Séi, H., Shimba, S., Urade, Y. & Ueda, H. R. FASTER: an unsupervised fully automated sleep

288 staging method for mice. *Genes Cells* **18**, 502–18 (2013).

289 7. Supratak, A., Dong, H., Wu, C. & Guo, Y. DeepSleepNet: A Model for Automatic Sleep Stage Scoring Based on
290 Raw Single-Channel EEG. *IEEE Trans. Neural Syst. Rehabil. Eng.* **25**, 1998–2008 (2017).

291 8. Yamabe, M. *et al.* MC-SleepNet: Large-scale Sleep Stage Scoring in Mice by Deep Neural Networks. *Sci. Rep.*
292 **9**, 1–12 (2019).

293 9. Izawa, S. *et al.* REM sleep-active MCH neurons are involved in forgetting hippocampus-dependent memories.
294 *Science* **365**, 1308–1313 (2019).

295 10. Shahzadi, I., Tang, T. B., Meriadeau, F. & Quyyum, A. CNN-LSTM: Cascaded Framework For Brain Tumour
296 Classification. in *2018 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES)* 633–637
297 (IEEE, 2018). doi:10.1109/IECBES.2018.8626704.

298 11. Ko, H., Chung, H., Lee, H. & Lee, J. Feasible Study on Intracranial Hemorrhage Detection and Classification
299 using a CNN-LSTM Network. in *2020 42nd Annual International Conference of the IEEE Engineering in*
300 *Medicine & Biology Society (EMBC)* vols 2020-July 1290–1293 (IEEE, 2020).

301 12. Hoyal Cuthill, J. F., Guttenberg, N., Ledger, S., Crowther, R. & Huertas, B. Deep learning on butterfly
302 phenotypes tests evolution’s oldest mathematical model. *Sci. Adv.* **5**, eaaw4967 (2019).

303 13. Kilkenny, C., Browne, W. J., Cuthill, I. C., Emerson, M. & Altman, D. G. Improving bioscience research
304 reporting: the ARRIVE guidelines for reporting animal research. *PLoS Biol.* **8**, e1000412 (2010).

305 14. Black, C. *et al.* Open Ephys electroencephalography (Open Ephys + EEG): a modular, low-cost, open-source
306 solution to human neural recording. *J. Neural Eng.* **14**, 035002 (2017).

- 307 15. Chaitanya, K. *et al.* Semi-supervised task-driven data augmentation for medical image segmentation. *Med. Image*
308 *Anal.* **68**, 101934 (2021).
- 309 16. Foster, D. *Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play*. (O'Reilly Media,
310 Incorporated, 2019).
- 311
- 312

Figure Legends

Fig. 1 Image production for image-based machine learning and learning network structure

A: Sample images of three sleep stages: Wake, NREM, and REM. The upper part of the data image is the EMG. The vertical coordinates are fixed between all images. The lower part is the heatmap of the EEG power spectrum (1–20 Hz) of 1-s bins. The brightness of the heatmap was normalized by Python's Scikit-learn library.

B: Schematic of the one- and two-epoch data image generation. Images are labeled according to sleep stage, and two-epoch images are classified according to the designation of the latter half of the 20-s epoch.

C: Modified network structure based on VGG-19. As shown in panels, all the Conv2D layers filters' strides were set to 1×1, size of filters was 3×3, activation functions were rectified linear unit (ReLU). All the MaxPooling2D layers had the same parameter of pool_size (2,2) and strides (2,2). The padding mode of MaxPooling2D is "valid". Both of them have similar five blocks. The Output was purposely designed much longer than the Vgg19, because we need to exact those hidden layers information for further study of clustering visualization.

Fig. 2 Expansion of the dataset by fake images

A. Schematic representation of WGANGP-based image expansion.

B. Overview of GAN structure, containing the generator and discriminator. In this study, all the generated fake images were discriminated against the real images from tsukuba-14 dataset.

C. The true REM images (left) and the fake REM images (right) generated based on the dataset.

D, E. Modified generator and discriminator structure of WGANGP. The generator is a reverse process of classifier. The specific parameters of WGANGP settings are unchanged and can be confirmed online on O'Reilly series book official GitHub pages. For example, both the generator Conv2DTranspose layers and discriminator Conv2D layers filters' strides were 2×2, and the size of filters was 5×5.

Fig. 3 Performance of image-based sleep classification

A. Scoring performance on Tsukubai-14 datasets compared to the original MC-SleepNet algorithm. Overall evaluation by three scales of accuracy, F1-score, and Cohen's κ shows the improved performance with the additional one epoch and the assistance of the GAN-generated fake REM images.

341 The scaled data of the MC-SleepNet are from the original work.
342 B. Comparison bar graph of three parameters between different algorithms.
343 C. Visualization of the dense layer of the model using the UMAP clustering algorithms.
344 The different visible clustering separations display the scoring performance, particularly for the REM stage.

345

346 **Fig. 4 Effect of epoch length on the performance**

347 A. Schematic of the change in epoch length.
348 B. Comparison bar graph of three parameters between different epoch lengths.
349 C. Data table

350

351 **Fig. 5 Performance of novel algorithm on tiny datasets**

352 A. Schematic of the protocol to test the performance on tiny datasets. The 14-segment dataset was divided into two
353 parts: one to four segments were used for training, and the remaining 10 segments were used as the evaluation dataset.
354 B. Data table
355 C. Comparison bar graph of three parameters under different conditions.

356

357 **Fig. 6 Expansion of all three stage images by WGANGP**

358 A. Schematic of the protocol to create fake images for all three stages.
359 B. Comparison bar graph of three parameters under different conditions.
360 C. Data table

361

362

363

364 **Supplemental Figure Legends**

365

366 **Supplemental Fig.1. Visualization of the dense layer of the model using the UMAP clustering algorithms.**

367 The distribution of all epoch data of the middle and last dense layers with various n_neighbor parameters set from 5
368 to 100. Figure 3C displays the middle dense data of the n_neighbor at 75.

369

370 **Supplemental Fig. 2. Visualization of the dense layer of the GAN model using the UMAP clustering algorithms.**

371 The distribution of all epoch data of the first middle dense layer (A) and last middle dense layer (B) with n_neighbor
372 parameters set at 75.

373

374 **Supplemental Fig. 3. Scoring performance with the forced correction filters.**

375 The filters can determine the epoch that we consider to be anomalies and fix those points. These exceptions include
376 the REM epoch (for only 1–2 times) or the NREM epoch (for only 1–2 times) isolated over a long period of the wake
377 stage. In these cases, they are corrected for the wake stage.

378

379 **Supplemental Fig. 4**

380 The simply designed GUI is based on the standard Python interface Tkinter. It consists of three main functions:
381 creating datasets based on customized requirements, training the labeled datasets, and predicting previous datasets.
382 Currently, dat, edf, and csv data types can be processed. The DCGANs and forced automatic filter options are also
383 open for users to create their own datasets for their experimental systems.

384

385

386 **Supplemental Table.**

387

388 **Confusion matrix of prediction results for all segment datasets**

389 Excel files have three sheets that contain the results of the confusion matrix for the entire dataset prediction, non-
390 standard (shortened) dataset prediction, and single-mouse dataset-based prediction. The precision (or recall) for each
391 stage is calculated using the confusion matrices. One matrix contains three rows (actual number of Wake, NREM,
392 and REM) and three columns (predicted number of Wake, NREM, and REM).

393

Fig. 1

A

B

Dataset of 1 epoch

Dataset of 2 epoch

Fig. 1
C Vgg19

Input
(224*224)

Block1
Conv2D layer
filters:64
filter size: 3*3
stride:1*1
activation:ReLU

Block2
Conv2D layer
filters:128
filter size: 3*3
stride:1*1
activation:ReLU

Block3
Conv2D layer
filters:256
filter size: 3*3
stride:1*1
activation:ReLU

Block4
Conv2D layer
filters:512
filter size: 3*3
stride:1*1
activation:ReLU

Block5
Conv2D layer
filters:512
filter size: 3*3
stride:1*1
activation:ReLU

Scoring
Output

Dense units:
25088 → 4096 → 3
(default 1000)
activation:
ReLU or
Softmax
(last layer only)

Our model

Input
(180*180)

Block1
Conv2D layer
filters:200
filter size: 3*3
stride:1*1
activation:ReLU

Block2
Conv2D layer
filters:128
filter size: 3*3
stride:1*1
activation:ReLU

Block3
Conv2D layer
filters:64
filter size: 3*3
stride:1*1
activation:ReLU

Block4
Conv2D layer
filters:32
filter size: 3*3
stride:1*1
activation:ReLU

Block5
Conv2D layer
filters:16
filter size: 3*3
stride:1*1
activation:ReLU

Scoring
Output

Dense units:
128 → 64 → 3
activation:
ReLU or
Softmax
(last layer only)
Dropout rate:0.5

Fig. 2

A

B

C

Fig. 2 GAN Generator

D

GAN discriminator

E

Fig. 3

A

	Wake		NREM		REM		Sleep		Acc	f1-score	k stat
	Rec.	Prec.	Rec.	Prec.	Rec.	Prec.	Rec.	Prec.			
1epoch	97.57%	96.99%	96.23%	96.61%	87.56%	88.20%	95.22%	95.63%	96.11%	0.961	0.936
2epoch	95.41%	98.72%	98.41%	94.97%	81.02%	90.35%	97.00%	94.64%	96.32%	0.961	0.938
2epoch_gan	96.63%	98.28%	97.98%	95.94%	82.26%	90.17%	96.70%	95.52%	96.67%	0.966	0.945
MC-SleepNet	99.3%	98.6%	93.90%	99.6%	99.5%	62.6%			96.4%	0.919	0.94

B

C

Fig. 4

C

	Wake		NREM		REM		Sleep		Acc	f1-score	k stat
	Rec.	Prec.	Rec.	Prec.	Rec.	Prec.	Rec.	Prec.			
shorten(20s)	91.94%	97.94%	98.34%	90.35%	52.92%	94.63%	94.64%	90.53%	93.88%	0.935855	0.895405
shorten(25s)	93.19%	98.43%	98.63%	91.89%	62.20%	94.69%	95.66%	92.03%	94.86%	0.9467	0.912981
shorten(30s)	94.62%	98.41%	98.44%	93.35%	68.52%	93.09%	96.01%	93.34%	95.42%	0.952985	0.924189
shorten(35s)	95.61%	98.26%	98.29%	94.42%	72.99%	93.21%	96.23%	94.34%	96.08%	0.959936	0.934644
2epoch(40s)	95.41%	98.72%	98.41%	94.97%	81.02%	90.35%	97.00%	94.64%	96.32%	0.938434	0.961254

Fig. 5

A

B

C

	Wake		NREM		REM		Sleep		Acc	f1-score	k stat
	Rec.	Prec.	Rec.	Prec.	Rec.	Prec.	Rec.	Prec.			
1mouse	98.29%	76.90%	76.36%	97.22%	71.02%	81.58%	75.93%	95.82%	87.19%	0.864979	0.784807
1mouse_gan	97.63%	85.78%	87.06%	96.21%	71.76%	84.97%	85.85%	95.38%	90.80%	0.90661	0.84607
2mouse	96.01%	95.06%	95.41%	94.95%	75.33%	88.24%	93.77%	94.48%	95.52%	0.954497	0.923603
2mouse_gan	95.86%	97.70%	96.94%	95.16%	78.17%	81.87%	95.46%	94.17%	95.53%	0.953359	0.928123
3mouse	96.37%	97.40%	96.88%	95.85%	82.58%	84.69%	95.72%	94.97%	95.72%	0.956983	0.927713
3mouse_gan	95.92%	97.58%	95.46%	95.95%	86.27%	71.06%	94.71%	93.53%	95.14%	0.952748	0.921037
4mouse	97.47%	97.08%	97.04%	96.57%	81.27%	89.48%	95.76%	96.04%	95.78%	0.95726	0.927185
4mouse_gan	97.29%	97.28%	97.02%	96.54%	82.78%	87.74%	95.86%	95.86%	95.68%	0.95641	0.926398

Fig. 6
A

B

C

	Wake		NREM		REM		Sleep		Acc	f1-score	k stat
	Rec.	Prec.	Rec.	Prec.	Rec.	Prec.	Rec.	Prec.			
one mouse	98.29%	76.90%	76.36%	97.22%	71.02%	81.58%	75.93%	95.82%	87.19%	0.8650	0.7848
one mouse(REM*10)	98.41%	79.28%	78.63%	97.48%	76.21%	73.08%	78.44%	78.44%	86.83%	0.8627	0.7842
one mouse(+fake GAN*9)	97.63%	85.78%	87.06%	96.21%	71.76%	84.97%	85.85%	95.38%	90.80%	0.9066	0.8416
one mouse(ALL Stage GAN)	97.80%	91.95%	92.19%	97.03%	79.33%	79.48%	91.18%	95.58%	93.96%	0.9395	0.8920

Figures

Fig. 1

Fig. 1

Figure 1

Image production for image-based machine learning and learning network structure A: Sample images of three sleep stages: Wake, NREM, and REM. The upper part of the data image is the EMG. The vertical coordinates are fixed between all images. The lower part is the heatmap of the EEG power spectrum (1–20 Hz) of 1-s bins. The brightness of the heatmap was normalized by Python's Scikit-learn library. B: Schematic of the one- and two-epoch data image generation. Images are labeled according to sleep stage, and two 320 epoch images are classified according to the designation of the latter half of the 20-s epoch. C: Modified network structure based on VGG-19. As shown in panels, all the Conv2D layers filters' strides were set to 1×1, size of filters was 3×3, activation functions were rectified linear unit (ReLU). All the MaxPooling2D layers had the same parameter of pool_size (2,2) and strides (2,2). The padding mode of MaxPooling2D is "valid". Both of them have similar five blocks. The Output was purposely designed much longer than the Vgg19, because we need to exact those hidden layers information for further study of clustering visualization.

Fig. 2

Fig. 2 GAN Generator

GAN discriminator

Figure 2

Expansion of the dataset by fake images A. Schematic representation of WGANP-based image expansion. B. Overview of GAN structure, containing the generator and discriminator. In this study, all the generated fake images were discriminated against the real images from tsukuba-14 dataset. C. The true REM images (left) and the fake REM images (right) generated based on the dataset. D, E. Modified generator and discriminator structure of WGANP. The generator is a reverse process of classifier. The specific parameters of WGANP settings are unchanged and can be confirmed online on O'Reilly series book official GitHub pages. For example, both the generator Conv2DTranspose layers and discriminator Conv2D layers filters' strides were 2×2, and the size of filters was 5×5.

Fig. 3

A

	Wake		NREM		REM		Sleep		Acc	f1-score	k stat
	Rec.	Prec.	Rec.	Prec.	Rec.	Prec.	Rec.	Prec.			
1epoch	97.57%	96.99%	96.23%	96.61%	87.56%	88.20%	95.22%	95.63%	96.11%	0.961	0.936
2epoch	95.41%	98.72%	98.41%	94.97%	81.02%	90.35%	97.00%	94.64%	96.32%	0.961	0.938
2epoch_gan	96.63%	98.28%	97.98%	95.94%	82.26%	90.17%	96.70%	95.52%	96.67%	0.966	0.945
MC-SleepNet	99.3%	98.6%	93.90%	99.6%	99.5%	62.6%			96.4%	0.919	0.94

B

C

Figure 3

Performance of image-based sleep classification A. Scoring performance on Tsukubai-14 datasets compared to the original MC-SleepNet algorithm. Overall evaluation by three scales of accuracy, F1-score, and Cohen's K shows the improved performance with the additional one epoch and the assistance of the GAN-generated fake REM images. The scaled data of the MC-SleepNet are from the original work. B. Comparison bar graph of three parameters between different algorithms. C. Visualization of the dense

layer of the model using the UMAP clustering algorithms. The different visible clustering separations display the scoring performance, particularly for the REM stage.

Figure 4

Effect of epoch length on the performance A. Schematic of the change in epoch length. B. Comparison bar graph of three parameters between different epoch lengths. C. Data table

Fig. 5
A

B

C

	Wake		NREM		REM		Sleep		Acc	f1-score	k stat
	Rec.	Prec.	Rec.	Prec.	Rec.	Prec.	Rec.	Prec.			
1mouse	98.29%	76.90%	76.36%	97.22%	71.02%	81.58%	75.93%	95.82%	87.19%	0.864979	0.784807
1mouse_gan	97.63%	85.78%	87.06%	96.21%	71.76%	84.97%	85.85%	95.38%	90.80%	0.90661	0.84607
2mouse	96.01%	95.06%	95.41%	94.95%	75.33%	88.24%	93.77%	94.48%	95.52%	0.954497	0.923603
2mouse_gan	95.86%	97.70%	96.94%	95.16%	78.17%	81.87%	95.46%	94.17%	95.53%	0.953359	0.928123
3mouse	96.37%	97.40%	96.88%	95.85%	82.58%	84.69%	95.72%	94.97%	95.72%	0.956983	0.927713
3mouse_gan	95.92%	97.58%	95.46%	95.95%	86.27%	71.06%	94.71%	93.53%	95.14%	0.952748	0.921037
4mouse	97.47%	97.08%	97.04%	96.57%	81.27%	89.48%	95.76%	96.04%	95.78%	0.95726	0.927185
4mouse_gan	97.29%	97.28%	97.02%	96.54%	82.78%	87.74%	95.86%	95.86%	95.68%	0.95641	0.926398

Figure 5

Performance of novel algorithm on tiny datasets A. Schematic of the protocol to test the performance on tiny datasets. The 14-segment dataset was divided into two parts: one to four segments were used for training, and the remaining 10 segments were used as the evaluation dataset. B. Data table C. Comparison bar graph of three parameters under different conditions.

Fig. 6
A

B

C

	Wake		NREM		REM		Sleep		Acc	f1-score	k stat
	Rec.	Prec.	Rec.	Prec.	Rec.	Prec.	Rec.	Prec.			
one mouse	98.29%	76.90%	76.36%	97.22%	71.02%	81.58%	75.93%	95.82%	87.19%	0.8650	0.7848
one mouse(REM*10)	98.41%	79.28%	78.63%	97.48%	76.21%	73.08%	78.44%	78.44%	86.83%	0.8627	0.7842
one mouse(+fake GAN*9)	97.63%	85.78%	87.06%	96.21%	71.76%	84.97%	85.85%	95.38%	90.80%	0.9066	0.8416
one mouse(ALL Stage GAN)	97.80%	91.95%	92.19%	97.03%	79.33%	79.48%	91.18%	95.58%	93.96%	0.9395	0.8920

Figure 6

Expansion of all three stage images by WGANGP A. Schematic of the protocol to create fake images for all three stages. B. Comparison bar graph of three parameters under different conditions. C. Data table

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [GaoGISleepNetMouseSupFigs20210601.pdf](#)
- [GaoGISleepNetMouseSupTable20210601.xlsx](#)