SleepVST: Sleep Staging from Near-Infrared Video Signals using Pre-Trained Transformers

Supplementary Material

A. Broader Impact Statement

Video data is a sensitive modality and its use in healthcare must come in conjunction with patient consent and clinical assessment. In sleep medicine, video polysomnography is considered the gold-standard monitoring technique, and the use of video enables accurate diagnoses of specific conditions such as REM behaviour disorder [8]. For wider applications, the algorithms introduced in this work have been designed such that the video data can be processed into intermediate motion and cardio-respiratory signals for further processing offline. This means it is possible for camera data to be processed in real-time without being stored.

Machine learning methods have the potential to make a transformative impact in healthcare. However, they should be appropriately evaluated across diverse populations and in real-world scenarios, to understand their limitations, mitigate potential biases, and reduce clinical risk. In future work, we plan to evaluate the performance of our method across a broader population, including a greater number of older individuals and individuals with darker skin tones, and individuals with diagnosed sleep disorders.

B. Oxford Sleep Volunteers Dataset

Throughout this work, we used the video polysomnography dataset introduced by Carter *et al*. [1] for our video-based sleep staging experiments, which we refer to as the Oxford Sleep Volunteers (OSV) dataset. Figure 10 shows the two camera viewpoints and room layouts present in the dataset, along with transformed frames, and approximate head, body and outer bed regions. The homography transformations and regions were manually determined from the known camera parameters and room geometry. For more information on the dataset, including population demographics and room geometries, we refer to the original work [1].

Figure 10. **Example processing of near-infrared video frames from the OSV dataset.** (a) Real viewpoints and bed positions for each room. (b) Cropped, virtual viewpoints obtained using homography transformations. (c) Head (H), body (B), and outer (O) bed regions.

C. Additional SleepVST Results

Additional model hypnograms. Additional examples of sleep hypnograms generated using SleepVST from video data are shown in Figure 16.

Cohen's κ distribution across datasets. Figure 11 shows the distribution of Cohen's κ values with age across all three datasets. Despite being trained on contact sensor waveforms from much older subjects in the SHHS and MESA datasets, SleepVST successfully transfers to video-derived waveforms, nearly reaching parity with performance using contact sensors. This highlights the effectiveness and generality of the learnt feature space. Given the well-known changes in autonomic activity with age [15], pre-training with additional data from younger subjects may further improve transfer learning performance on the existing dataset. As may fine-tuning the entire SleepVST network on video data, rather than freezing the weights and using it as a fixed feature extractor. Conversely, more video data from older participants during the transfer learning phase would likely help to improve the performance for the existing older participants.

Pre-training confusion matrices. Figures 12 and 13 show the four-class sleep staging confusion matrices obtained after pre-training, for the SHHS and MESA test sets respectively. During pre-training, we chose to use an unbalanced cross-entropy loss. Using a class-weighted loss instead would likely help to reduce the observed rate of misclassification of (less frequent) N3 sleep.

Varying classification strategy. In Table 6, we report the performance of our method for different sleep stage classification strategies, to aid future comparisons. In each case, we use the same SleepVST model (pre-trained on four-class sleep staging) as a feature extractor, and train a new classifier using video data.

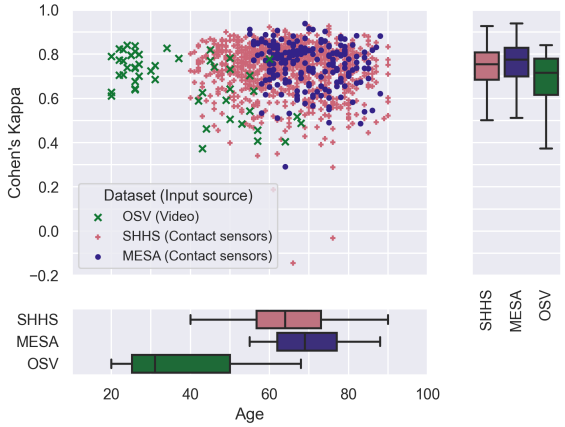


Figure 11. Scatter and box plots of Cohen's κ and age distributions across the SHHS, MESA and OSV datasets.

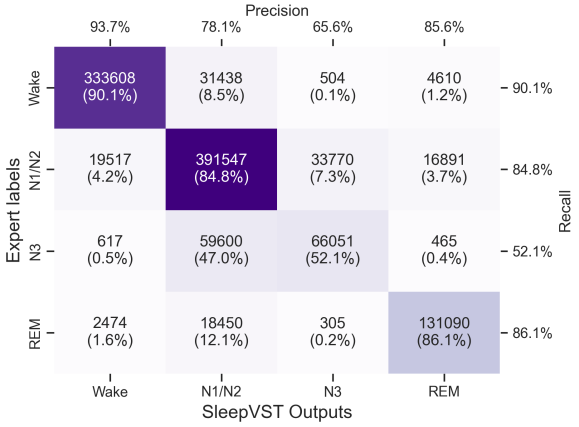


Figure 12. SleepVST confusion matrix with expert labels, evaluated on the SHHS test set.

Table 6. Video-based sleep staging performance using SleepVST with different classification strategies.

	κ_μ	κ_T	Acc$_\mu$ / %	Acc$_T$ / %
Sleep–Wake	0.782±0.197	0.857	93.3±5.4	93.7
W–NREM–REM	0.760±0.139	0.795	87.4±6.6	87.9
W–N1/N2–N3–REM	0.677±0.133	0.708	77.7±8.9	78.8
W–N1–N2–N3–REM	0.646±0.139	0.674	74.1±10.7	75.3

Additional motion feature ablations. Table 7 shows the effectiveness of the remaining components of our motion feature set. Higher thresholds ($\delta = 1$) measure the elapsed time since large movements, which are more important for overall agreement κ_T. Lower thresholds ($\delta = 0.01$) are more sensitive to smaller movements e.g. brief awakenings, leading to greater agreement around fragmented sleep

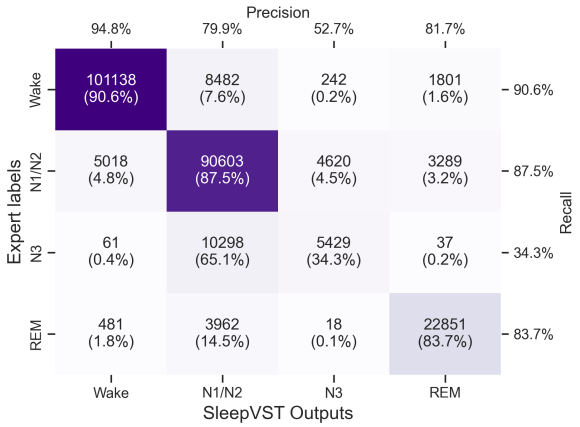


Figure 13. SleepVST confusion matrix against expert labels, evaluated on the MESA test set.

κ_F.

Table 7. Ablation of motion feature parameters T, δ and Δ.

			Cohen's κ	
Ablation	Parameter Set	n_f^*	κ_T	κ_F
Threshold(s)	$\delta \in \{0.01\}$	54	0.698	0.485
	$\delta \in \{0.1\}$	54	0.702	0.486
	$\delta \in \{1\}$	54	0.705	0.471
	$\delta \in \{0.01, 0.1, 1\}$	90	**0.708**	**0.491**
Time shift(s)	$T \in \{0\}$	30	0.705	0.464
	$T \in \{\text{-}90, 0, 90\}$	90	**0.708**	**0.491**
Window(s)	$\Delta \in \{30\}$	72	0.706	0.486
	$\Delta \in \{300\}$	72	0.705	0.472
	$\Delta \in \{30, 300\}$	90	**0.708**	**0.491**

*No. features.

D. SleepVST Architecture Details

Convolutional encoder. Each convolutional layer, denoted 'Kx1 Conv, M' in Figure 4, consisted of 1D convolutions with kernel size K, stride 1, and M output channels, followed by batch normalisation [3], and ReLU activation.

Transformer encoder. The parameters of our transformer encoder are detailed in Table 8, these values were informed by the original design of Vaswani *et al.* [12]. To improve training stability, we employed pre-layer normalisation [14] within each transformer encoder layer.

Training. Each training run was performed using a single NVIDIA A10 GPU with 24 GB RAM. Using a sequence length of two hours (N=240) and our default architecture, we used a batch size of 128, the largest power of two that could fit on the GPU. We employed early-stopping to terminate training once there had been no improvement in the

Table 8. SleepVST transformer encoder parameters.

Architecture Parameter	Value
Encoder layers	6
Self-attention heads	8
Encoder dropout probability	0.1
MLP size	512

validation loss for three consecutive epochs, restoring the model checkpoint that achieved the minimum value. The training run which produced our best model took 6.4 h. All models were implemented using the PyTorch [5] framework.

E. Transformer Output Tiling

Because of the quadratic complexity of the original Transformer [12], we used an input sequence length of N=240 epochs, i.e. two hours, to the SleepVST model. This is much shorter than the sleep recordings within each dataset, which typically last between 8 and 12 hours. To apply SleepVST to these longer sequences, we re-applied the model at up to 30-minute steps, resulting in multiple outputs for timesteps away from the start and end of the recording. Using this approach, the model can be applied to 10 hours of waveform data in ≈ 0.8 s.

When directly applying the pre-trained SleepVST model, e.g. to SHHS and MESA test sets, we took the mode of the overlapping classifications as the output classification. This is illustrated in Figure 14.

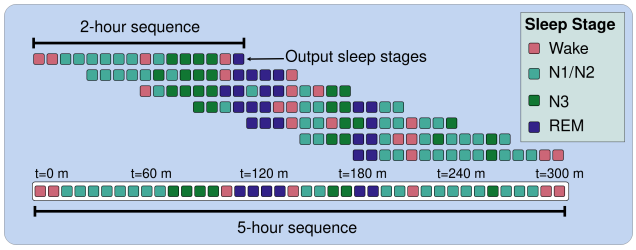


Figure 14. **Classifying longer input sequences using SleepVST.** After applying the model to overlapping two-hour sub-sequences, the modal classification at each timestep is used as the output classification.

Similarly, when using SleepVST as a feature extractor on video-derived waveforms, we re-applied the model at intervals to produce overlapping two-hour sequences of feature vectors. For timesteps with multiple feature vectors, we used the feature vector which was closest to the middle of a sequence. This is illustrated in Figure 15.

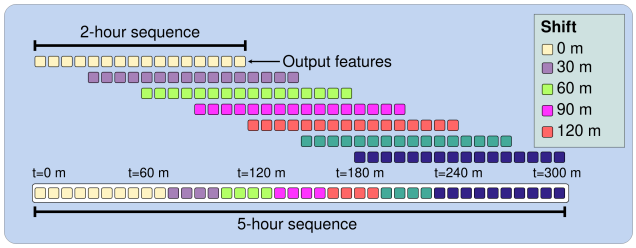


Figure 15. **Extracting features from longer input sequences using SleepVST.** The model is applied to two-hour input sub-sequences, producing overlapping output feature sequences. For each timestep, we use the feature vector which is closest to the middle of a sequence.

F. Optical Flow Estimation

We used the Dense Inverse Search algorithm [4] to calculate the optical flow field at a frequency of 4 Hz from the homography-transformed frames, following the same procedure as [1].

G. Cohen's Kappa Calculation

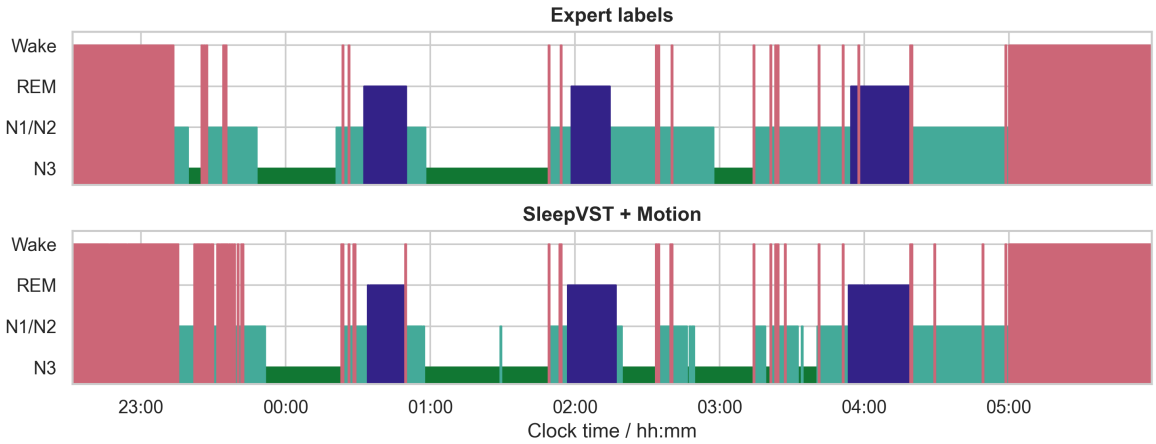
From a confusion matrix $M \in \mathbb{R}^{C\times C}$ with elements m_{ij}, the function $K : \mathbb{R}^{C\times C} \to \mathbb{R}$ which calculates Cohen's κ statistic is given by:

$$K(M) = 1 - \frac{\sum_{i,j} w_{ij} m_{ij}}{\sum_{i,j} w_{ij} e_{ij}} \tag{9}$$

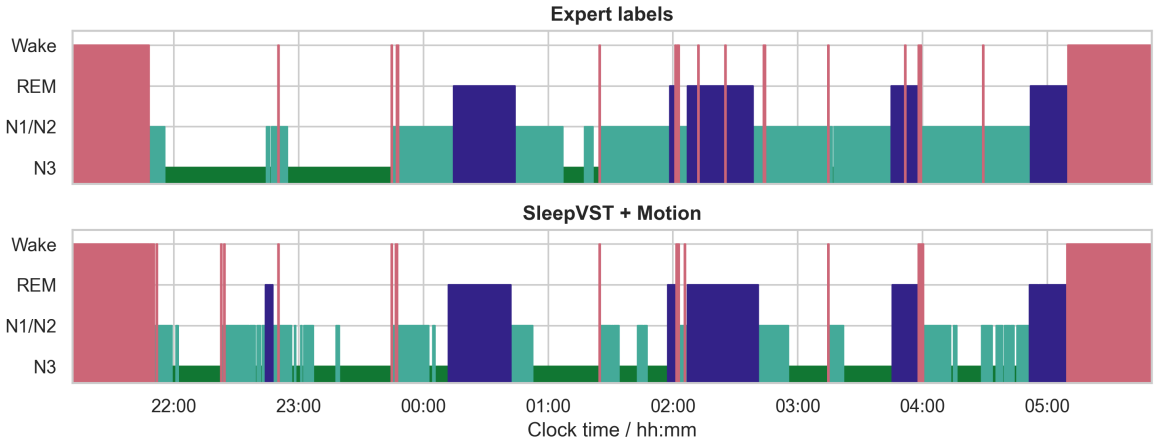
where w_{ij} and e_{ij} are defined as follows:

$$w_{ij} = \begin{cases} 0 & \text{if } i = j \\ 1 & \text{otherwise} \end{cases} \tag{10}$$

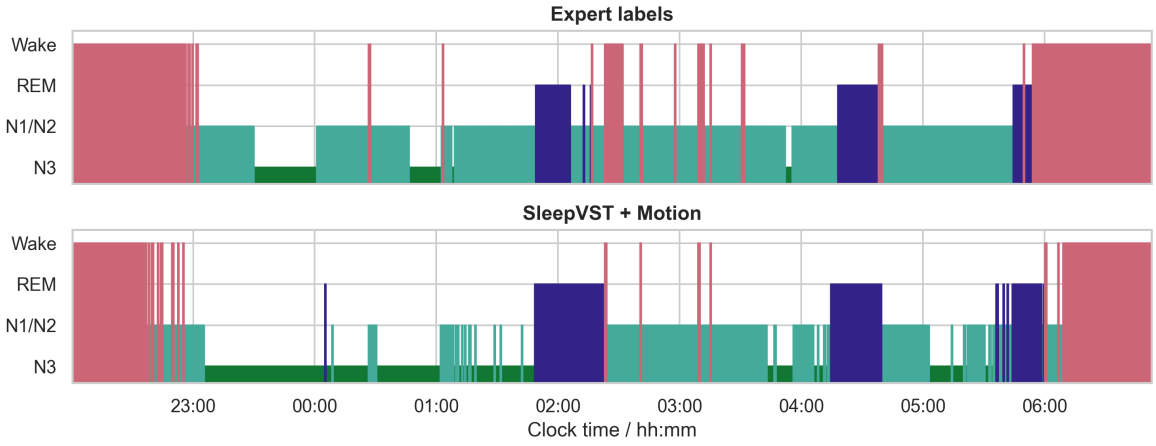
$$e_{ij} = \frac{\left(\sum_{i'} m_{i'j}\right) \cdot \left(\sum_{j'} m_{ij'}\right)}{\sum_{i'j'} m_{i'j'}} \tag{11}$$



(a) Cohen's $\kappa = 0.84$ between model and expert labels.



(b) Cohen's $\kappa = 0.64$ between model and expert labels.



(c) Cohen's $\kappa = 0.39$ between model and expert labels.

Figure 16. Example four-class sleep hypnograms from the OSV dataset for various model–expert Cohen's κ agreement values. Within each subfigure, the top hypnogram shows expert labels annotated using signals from the vPSG recording, and the bottom hypnogram shows labels automatically generated from near-infrared video using our method.

H. SHHS Test Set

The 500 randomly sampled participant IDs used to form our test set are as follows:

203949, 204944, 200702, 203956, 202106, 201917, 203231, 204150, 205275, 203034, 201936, 205025, 200885, 201204, 204594, 201308, 204960, 205608, 204379, 203354, 204125, 204330, 203384, 201213, 201598, 204495, 204590, 203944, 203945, 202921, 201287, 203495, 205462, 204068, 203423, 202981, 203505, 204079, 204939, 203390, 204179, 204885, 202435, 202157, 202834, 200825, 203684, 205299, 200897, 200152, 202828, 203530, 203312, 205398, 203984, 202801, 203264, 201453, 203652, 200460, 202820, 204638, 203367, 205565, 200839, 203213, 204016, 204473, 200751, 201206, 200927, 201608, 201102, 205494, 201399, 200730, 202948, 200293, 204865, 203520, 204795, 200953, 203125, 205340, 205009, 204825, 200752, 202794, 203165, 203306, 203490, 204041, 202383, 204656, 203772, 203829, 204517, 201557, 202650, 203308, 203564, 202210, 200991, 205663, 203559, 204303, 200513, 205664, 202152, 204928, 203974, 200851, 205169, 205645, 204256, 201783, 201414, 204540, 204661, 203894, 201373, 202496, 200718, 202227, 200243, 203826, 202946, 203446, 202123, 200604, 201058, 205704, 204798, 205346, 204823, 203748, 203824, 200680, 200698, 203166, 204140, 205072, 203512, 200646, 205744, 200679, 204295, 201670, 204486, 203316, 203511, 200998, 204335, 200886, 202968, 204409, 205575, 202943, 205082, 203200, 200321, 205226, 200899, 203845, 202940, 200687, 200948, 203311, 200769, 200782, 203925, 200666, 202480, 201521, 205593, 200507, 203282, 204001, 205601, 203372, 204988, 204269, 203946, 205257, 200176, 204231, 204530, 204963, 200145, 200888, 202187, 200088, 203882, 203286, 204418, 200578, 204273, 200823, 203065, 205349, 203106, 200217, 202912, 203566, 200154, 201323, 200662, 204289, 205651, 203252, 205044, 201024, 203716, 204232, 200632, 200102, 205450, 200566, 202521, 202563, 200584, 203192, 201298, 205485, 205739, 204617, 204642, 201331, 205596, 203202, 203381, 204956, 203522, 200303, 203534, 204190, 201130, 201268, 200191, 201513, 200955, 203689, 203157, 201401, 201517, 202489, 203018, 203232, 205146, 202963, 202821, 204287, 204422, 204472, 200334, 200925, 203135, 200516, 202442, 204171, 203392, 201503, 202458, 205587, 203502, 203626, 204702, 204599, 200150, 205605, 202566, 204296, 202221, 203296, 205537, 203860, 200842, 200318, 204023, 202463, 201628, 200858, 205419, 201068, 205312, 202663, 202444, 200579, 201629, 203198, 204283, 204846, 204699, 200178, 200981, 203237, 200564, 204340, 202785, 202938, 204506, 201493, 203610, 201353, 203769, 200105, 204522, 204343, 204086, 200320, 204425, 202842, 204459, 202226, 203528, 204978, 204461, 204691, 200111, 201083, 200950, 200108, 203455, 204647, 200952, 204443, 204435, 204504, 205064, 203476, 203695, 203721, 202201, 200841, 200935, 204093, 201223, 201470, 204747, 205350, 204871, 203303, 204132, 201219, 204676, 201402, 205722, 204115, 200744, 205772, 203281, 204926, 205086, 202990, 203235, 204384, 201432, 203039, 200387, 203961, 203456, 203462, 202546, 203895, 205595, 204496, 201241, 205004, 200712, 204565, 203671, 200936, 201271, 203734, 202428, 203260, 200653, 204405, 201982, 200703, 203314, 202405, 200668, 205702, 204431, 202608, 203056, 204544, 203754, 205761, 203942, 200835, 200192, 205539, 200945, 203254, 204177, 200219, 205486, 202642, 202417, 203498, 203347, 204759, 202361, 205255, 201064, 201544, 200295, 204233, 204187, 202150, 204224, 204370, 204952, 200571, 203976, 204304, 200347, 205126, 203224, 200591, 203060, 201543, 204923, 201778, 204914, 205222, 204907, 200766, 205305, 204235, 200406, 205661, 200209, 204778, 201640, 204236, 200901, 205356, 200853, 200210, 204898, 203269, 203451, 202825, 201299, 204690, 203557, 203589, 203037, 204337, 204460, 201316, 201312, 204856, 203138, 203412, 200242, 203460, 200920, 200887, 201918, 203395, 205530, 201349, 200829, 203208, 201519, 200386, 203117, 200466, 202605, 203121, 200624, 205721, 204323, 204554, 205289, 204934, 200233, 203149, 204170, 203966, 205252, 205548, 203006, 202902, 203818, 202942, 201018, 205588, 202395, 203709, 205591, 205532, 204763, 202829, 205626, 201538.

References

[1] Jonathan Carter, João Jorge, Bindia Venugopal, Oliver Gibson, and Lionel Tarassenko. Deep Learning-Enabled Sleep Staging From Vital Signs and Activity Measured Using a Near-Infrared Video Camera. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 5940–5949, Vancouver, BC, Canada, 2023. IEEE. 1, 3

[2] C. Iber. The AASM Manual for the Scoring of Sleep and Associated Events: Rules, Terminology, and Technical Specification. 2007. 1

[3] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015. 2

[4] Till Kroeger, Radu Timofte, Dengxin Dai, and Luc Van Gool. Fast Optical Flow using Dense Inverse Search, 2016. 3

[5] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*, 2019. 3

[6] Aakash K. Patel, Vamsi Reddy, and John F. Araujo. *Physiology, Sleep Stages*. StatPearls Publishing, 2022. 4

[7] Huy Phan, Oliver Y. Chén, Minh C. Tran, Philipp Koch, Alfred Mertins, and Maarten De Vos. XSleepNet: Multi-View Sequential Model for Automatic Sleep Staging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9): 5903–5915, 2022. 1

[8] Ambra Stefani and Birgit Högl. Sleep in Parkinson's disease. *Neuropsychopharmacology*, 45(1):121–128, 2020. 8, 1

[9] U.S. Food and Drug Administration (FDA). De Novo Classification Request for Oxehealth Vital Signs, 2020, https://www.accessdata.fda.gov/cdrh_docs/reviews/DEN200019.pdf. 2, 3

[10] U.S. Food and Drug Administration (FDA). De Novo Classification Request for Gili Pro Biosensor, 2020, https://www.accessdata.fda.gov/cdrh_docs/reviews/DEN200038.pdf. 2

[11] Mark van Gastel, Sander Stuijk, Sebastiaan Overeem, Johannes P. van Dijk, Merel M. van Gilst, and Gerard de Haan. Camera-Based Vital Signs Monitoring During Sleep – A Proof of Concept Study. *IEEE Journal of Biomedical and Health Informatics*, 25(5):1409–1418, 2021. 2

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. *arXiv:1706.03762 [cs]*, 2017. arXiv: 1706.03762. 2, 3

[13] Qiongyan Wang, Hanrong Cheng, and Wenjin Wang. Feasibility of Exploiting Physiological and Motion Features for Camera-based Sleep Staging: A Clinical Study. In *2023 45th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 1–5, 2023. ISSN: 2694-0604. 2

[14] Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. On layer normalization in the transformer architecture. In *International Conference on Machine Learning*, pages 10524–10533. PMLR, 2020. 2

[15] Dan Ziegler, G. Laux, K. Dannehl, M. Spüler, H. Mühlen, P. Mayer, and F.a. Gries. Assessment of Cardiovascular Autonomic Function: Age-related Normal Ranges and Reproducibility of Spectral Analysis, Vector Analysis, and Standard Tests of Heart Rate Variation and Blood Pressure Responses. *Diabetic Medicine*, 9(2):166–175, 1992. 1