DAFTAR PUSTAKA

Ahmed, H. U. D. *et al.* (2021) 'Deception Detection in Videos using the Facial Action Coding System', pp. 0–2. Available at: https://arxiv.org/abs/2105.13659v1.

Akhand, M. A. H. *et al.* (2021) 'Facial emotion recognition using transfer learning in the deep CNN', *Electronics (Switzerland)*, 10(9). doi: 10.3390/electronics10091036.

Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, J. and Puigcerver, Jessica Yung, Sylvain Gelly, and N. H. (2020) 'Big transfer (bit): General visual representation learning', in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow*. UK: Springer, pp. 491–507.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, A. and Auli, M. (2020) 'wav2vec 2.0:A framework for self-supervised learning of speech representations', in *Conference on Neural Information Processing Systems (NeurIPS 2020),* pp. 12449–12460.

Alexey Dosovitskiy, Lucas Beyer, A. K., Dirk Weissenborn, Xiaohua Zhai, T. U. and Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and N. H. (2021) 'An image is worth 16x16 words: Transformers for image recognition at scale', in *International Conference on Learning Representation*, pp. 0–7.

Arsal, M., Agus Wardijono, B. and Anggraini, D. (2020) 'Face Recognition Untuk Akses Pegawai Bank Menggunakan Deep Learning Dengan Metode CNN', *Jurnal Nasional Teknologi dan Sistem Informasi*, 6(1), pp. 55–63. doi: 10.25077/teknosi.v6i1.2020.55-63.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, I. P. (2017) 'Attention Is All You Need', in *Neural Information Processing Systems (NIPS 2017),*.

Avola, D. *et al.* (2020) 'LieToMe: Preliminary study on hand gestures for deception
ion via Fisher-LSTM', *Pattern Recognition Letters*, 138, pp. 455–461. doi:
.6/j.patrec.2020.08.014.

Bablani, A. *et al.* (2021) 'Lie Detection Using Fuzzy Ensemble Approach with Novel Defuzzification Method for Classification of EEG Signals', *IEEE Transactions on Instrumentation and Measurement*, 70. doi: 10.1109/TIM.2021.3082985.

Baghel, N. *et al.* (2020) 'Truth Identification from EEG Signal by using Convolution neural network: Lie Detection', *2020 43rd International Conference on Telecommunications and Signal Processing, TSP 2020*, pp. 550–553. doi: 10.1109/TSP49548.2020.9163497.

Baldi, P. and Sadowski, P. (2014) 'The dropout learning algorithm', *Artificial Intelligence*, 210(1), pp. 78–122. doi: 10.1016/j.artint.2014.02.004.

Barsever, D., Singh, S. and Neftci, E. (2020) 'Building a Better Lie Detector with BERT: The Difference between Truth and Lies', *Proceedings of the International Joint Conference on Neural Networks*. doi: 10.1109/IJCNN48605.2020.9206937.

Bessonova, Y. V. and Oboznov, A. A. (2018) 'Eye movements and lie detection', *Advances in Intelligent Systems and Computing*, 722, pp. 149–155. doi: 10.1007/978-3-319-73888-8_25.

Bhamare, A. R., Katharguppe, S. and Silviya Nancy, J. (2020) 'Deep Neural Networks for Lie Detection with Attention on Bio-signals', *2020 7th International Conference on Soft Computing and Machine Intelligence, ISCMI 2020*, pp. 143–147. doi: 10.1109/ISCMI51676.2020.9311575.

Cai, S. *et al.* (2019) 'Effective and Efficient Dropout for Deep Convolutional Neural Networks', pp. 1–12. Available at: http://arxiv.org/abs/1904.03392.

Cakmak, R. and Zeki, A. M. (2016) 'Neuro signal based lie detection', pp. 170–174. doi: 10.1109/iris.2015.7451606.

Chou, H. C., Liu, Y. W. and Lee, C. C. (2021) 'Automatic deception detection using multiple speech and language communicative descriptors in dialogs', *APSIPA Transactions on Signal and Information Processing*, 10(May). doi: 10.1017/ATSIP.2021.6.

. *et al.* (2019) 'Accuracy, confidence, and experiential criteria for lie ion through a videotaped interview', *Frontiers in Psychiatry*, 9(January), 14. doi: 10.3389/fpsyt.2018.00748.

David Silver, Aja Huang, Chris J Maddison, A. G., Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, M. and Lanctot, et al (2016) 'Mastering the game of Go with deep neural networks and tree search', *Nature*, 529, pp. 484–489. doi: 10.1038/nature16961.

Delgado-Herrera, M., Reyes-Aguilar, A. and Giordano, M. (2021) 'What Deception Tasks Used in the Lab Really Do: Systematic Review and Meta-analysis of Ecological Validity of fMRI Deception Tasks', *Neuroscience*, 468, pp. 88–109. doi: 10.1016/j.neuroscience.2021.06.005.

Deng, X. *et al.* (2016) 'An improved method to construct basic probability assignment based on the confusion matrix for classification problem', *Information Sciences*, 340–341, pp. 250–261. doi: 10.1016/j.ins.2016.01.033.

Ding, M. *et al.* (2019a) 'Face-focused cross-stream network for deception detection in videos', *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June(2), pp. 7794–7803. doi: 10.1109/CVPR.2019.00799.

Ding, M. *et al.* (2019b) 'Face-focused cross-stream network for deception detection in videos', in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 7794–7803. doi: 10.1109/CVPR.2019.00799.

Dodia, S. *et al.* (2020) 'Lie detection using extreme learning machine: A concealed information test based on short-time Fourier transform and binary bat optimization using a novel fitness function', *Computational Intelligence*, 36(2), pp. 637–658. doi: 10.1111/coin.12256.

Esraa Yasser, etall E. A. F. (2020) 'Deception Detection', in. Ain Shams University Faculty of Computer & Information Sciences Computer Science Department, pp. 3–3. doi: 10.1145/3132847.3137174.

Fathima Bareeda, E. P., Shajee Mohan, B. S. and Ahammed Muneer, K. V. (2021) 'Lie Detection using Speech Processing Techniques', *Journal of Physics: Conference Series*, 1921(1). doi: 10.1088/1742-6596/1921/1/012028.

M. D. Y. and Rochmawati, N. (2022) 'Optimisasi Hyperparameter CNN gunakan Random Search Untuk Deteksi COVID-19 Dari Citra X-Ray

Dada', *Journal of Informatics and Computer Science (JINACS)*, 4(01), pp. 10–18. doi: 10.26740/jinacs.v4n01.p10-18.

Gallardo-Antolín, A. and Montero, J. M. (2021) 'Detecting deception from gaze and speech using a multimodal attention LSTM-based framework', *Applied Sciences (Switzerland)*, 11(14). doi: 10.3390/app11146393.

Gao, J. *et al.* (2014) 'A novel algorithm to enhance P300 in single trials: Application to lie detection using F-score and SVM', *PLoS ONE*, 9(11). doi: 10.1371/journal.pone.0109700.

Gorunescu, I. S. R. L. V. 12 (2011) *Intelligent Systems Reference Library,Volume 12*. Volume 12. Springer. Available at: https://www.ptonline.com/articles/how-to-get-better-mfi-results.

Gu, J. *et al.* (2018) 'Recent advances in convolutional neural networks', *Pattern Recognition*, 77, pp. 354–377. doi: 10.1016/j.patcog.2017.10.013.

Guermazi, R., Abdallah, T. B. and Hammami, M. (2021) 'Facial micro-expression recognition based on accordion spatio-temporal representation and random forests', *Journal of Visual Communication ...*. Available at: https://www.sciencedirect.com/science/article/pii/S1047320321001164.

Gumelar, A. B. *et al.* (2019) 'Human Voice Emotion Identification Using Prosodic and Spectral Feature Extraction Based on Deep Neural Networks', *2019 IEEE 7th International Conference on Serious Games and Applications for Health, SeGAH 2019*, pp. 1–8. doi: 10.1109/SeGAH.2019.8882461.

Gunadi, I. G. A. and Hartati, S. (2013) 'Manual Assessment Derajat Kebohongan Pada Adegan Video Berdasarkan Naïve Bayesian', *Snati*, pp. 41–45.

Gupta, V. *et al.* (2019a) 'Bag-of-lies: A multimodal dataset for deception detection', *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2019-June(ii), pp. 83–90. doi: 10.1109/CVPRW.2019.00016.

Gupta, V. *et al.* (2019b) 'Bag-of-lies: A multimodal dataset for deception detection', *IEEE Computer Society Conference on Computer Vision and Pattern nition Workshops*, 2019-June, pp. 83–90. doi:)9/CVPRW.2019.00016.

Gupta, V. *et al.* (2019c) 'Bag-of-lies: A multimodal dataset for deception detection', in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 83–90. doi: 10.1109/CVPRW.2019.00016.

Van Huynh, N. *et al.* (2021) 'DeepFake: Deep Dueling-Based Deception Strategy to Defeat Reactive Jammers', *IEEE Transactions on Wireless Communications*, 20(10), pp. 6898–6914. doi: 10.1109/TWC.2021.3078439.

Immanuel, J., Joshua, A. and Thomas George, S. (2018) 'A Study on Using Blink Parameters from EEG Data for Lie Detection', in *2018 International Conference on Computer Communication and Informatics, ICCCI 2018*. IEEE, pp. 1–5. doi: 10.1109/ICCCI.2018.8441238.

John Jumper, Richard Evans, Alexander Pritzel, T. G. and Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvu nakool, Russ Bates, Augustin Z˘´ıdek, Anna Potapenko, et (2021) 'Highly accurate protein structure prediction with alphafold. Nature', *Nature*, 596(7873), pp. 583–589. doi: 10.1038/s41586-021-03819-2.

Jun-Teng Yang Guei-Ming Liu Scott C.-H Huang (2020) 'EMOTION TRANSFORMATION FEATURE : NOVEL FEATURE FOR DECEPTION DETECTION IN VIDEOS Jun-Teng Yang Institute of Communications Engineering , National Tsing Hua University , Hsinchu , Taiwan', in. doi: 10.1109/ICIP40778.2020.9190846.

Kamran Haider, S. *et al.* (2017) 'Evaluation of P300 based Lie Detection Algorithm', *Electrical and Electronic Engineering*, 2017(3), pp. 69–76. doi: 10.5923/j.eee.20170703.01.

Khan, W. *et al.* (2021a) 'Deception in the eyes of deceiver: A computer vision and machine learning based automated deception detection', *Expert Systems with Applications*, 169(February 2020), p. 114341. doi: 10.1016/j.eswa.2020.114341.

Khan, W. *et al.* (2021b) 'Deception in the eyes of deceiver: A computer vision and machine learning based automated deception detection', *Expert Systems with Applications*, 169(November 2020), p. 114341. doi: .6/j.eswa.2020.114341.

Krishna, S. T. and Kalluri, H. K. (2019) 'Deep learning and transfer learning approaches for image classification', *International Journal of Recent Technology and Engineering*, 7(5), pp. 427–432.

Krishnamurthy, G. *et al.* (2023) 'A Deep Learning Approach for Multimodal Deception Detection', *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 13396 LNCS, pp. 87–96. doi: 10.1007/978-3-031-23793-5_8.

Kusnadi, A., Widyantara, I. M. O. and Linawati, L. (2021) 'Deteksi Kebohongan Berdasarkan Fitur Fonetik Akustik', *Majalah Ilmiah Teknologi Elektro*, 20(1), p. 113. doi: 10.24843/mite.2021.v20i01.p13.

Labibah, Z., Nasrun, M. and Setianingsih, C. (2018) 'Lie Detector With The Analysis Of The Change Of Diameter Pupil and The', *The 2018 IEEE International Conference on Internet of Things and Intelligence System (IoTaIS) Lie*, pp. 214–220.

Lakshan, I. *et al.* (2019) 'Real Time Deception Detection for Criminal Investigation', *2019 National Information Technology Conference, NITC 2019*, pp. 8–10. doi: 10.1109/NITC48475.2019.9114422.

Lasniari, S. *et al.* (2022) 'Pengaruh Hyperparameter Convolutional Neural Network Arsitektur ResNet-50 Pada Klasifikasi Citra Daging Sapi dan Daging Babi', 5(3), pp. 474–481. doi: 10.32672/jnkti.v5i3.4424.

Li, J. *et al.* (2022) 'Speech Emotion Recognition Using a Dual-Channel Complementary Spectrogram and the CNN-SSAE Neutral Network', *Applied Sciences (Switzerland)*, 12(19). doi: 10.3390/app12199518.

Lindasalwa Muda, M. B. and I. E. (2010) 'Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques', *Journal of computing and information ...*, 2(3). doi: 10.48550/arXiv.1003.4083.

Llamas, J. *et al.* (2017) 'Classification of architectural heritage images using deep learning techniques', *Applied Sciences (Switzerland)*, 7(10). doi: 10.3390/app7100992.

Luján-García, J. E. *et al.* (2020) 'A transfer learning method for pneumonia classification and visualization', *Applied Sciences (Switzerland)*, 10(8). doi: 10.3390/APP10082908.

Luque, A. *et al.* (2019) 'The impact of class imbalance in classification performance metrics based on the binary confusion matrix', *Pattern Recognition*, 91, pp. 216–231. doi: 10.1016/j.patcog.2019.02.023.

Martin Graciarena, Elizabeth Shriberg, A. S. and Frank Enos, Julia Hirschberg, and S. K. (2006) 'Combining prosodic lexical and cepstral systems for deceptive speech detection', in *In 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, pp. 0–7.

Masrur Ahmed, A. A. *et al.* (2021) 'Deep learning hybrid model with Boruta-Random forest optimiser algorithm for streamflow forecasting with climate mode indices, rainfall, and periodicity', *Journal of Hydrology*, 599(May), p. 126350. doi: 10.1016/j.jhydrol.2021.126350.

Mathur, L. and Matarić, M. J. (2020) 'Introducing Representations of Facial Affect in Automated Multimodal Deception Detection', *ICMI 2020 - Proceedings of the 2020 International Conference on Multimodal Interaction*, pp. 305–314. doi: 10.1145/3382507.3418864.

Md. Rakibul Hasan, Md.Mahbub Hasan, M. Z. H. (2021) 'How many Mel-frequency cepstral coefficients to be utilized in speech recognition? A study with the Bengali language', *The Journal of Engineering*, 2021(issue 12), pp. 817–827. doi: 10.1049/tje2.12082.

Miranda, N. D. *et al.* (2020) 'Convolutional Neural Network Pada Klasifikasi Sidik Jari Menggunakan Resnet-50 Classification of Fingerprint Pattern Using Convolutional Neural Network in Clahe Image', *Jurnal Teknik Informatika (JUTIF)*, 1(2), pp. 61–68. doi: DOI: 10.20884/1.jutif.2020.1.2.18.

Mohamed Abouelenien, Veronica P ´ erez-Rosas, R. M.-´ and Cea, and M. B. (2016) 'Detecting Deceptive Behavior via Integration of Discriminative Features from Multiple Modalities', in *IEEE Transactions on Information Forensics and ty*. ieeexplore.ieee.org, pp. 1042–1055. doi: 10.1109/TIFS.2016.2639344.

Nandini, G. S., Kumar, A. P. S. and K, C. (2021) 'Dropout technique for image classification based on extreme learning machine', *Global Transitions Proceedings*, 2(1), pp. 111–116. doi: 10.1016/j.gltp.2021.01.015.

Nasri, H., Ouarda, W. and Alimi, A. M. (2016) 'ReLiDSS: Novel lie detection system from speech signal', *Proceedings of IEEE/ACS International Conference on Computer Systems and Applications, AICCSA*, 0. doi: 10.1109/AICCSA.2016.7945789.

Owayjan, M., Kashour, A., Haddad, N. Al, *et al.* (2012) 'Desain dan Pengembangan Deteksi Kebohongan Sistem menggunakan Ekspresi Mikro Wajah', pp. 33–38.

Owayjan, M., Kashour, A., Al Haddad, N., *et al.* (2012) 'The design and development of a lie detection system using facial micro-expressions', *2012 2nd International Conference on Advances in Computational Tools for Engineering Applications, ACTEA 2012*, pp. 33–38. doi: 10.1109/ICTEA.2012.6462897.

Pasquali, D. *et al.* (2021) 'Detecting Lies is a Child (Robot)'s Play: Gaze-Based Lie Detection in HRI', *International Journal of Social Robotics*, 15(4), pp. 583–598. doi: 10.1007/s12369-021-00822-5.

Pérez-Rosas, V. *et al.* (2015) 'Deception detection using real-life trial data', in *ICMI 2015 - Proceedings of the 2015 ACM International Conference on Multimodal Interaction*, pp. 59–66. doi: 10.1145/2818346.2820758.

Prananta1, L. *et al.* (2022) 'The Effectiveness of Time Stretching for Enhancing Dysarthric Speech for Improved Dysarthric Speech Recognition'. doi: 10.48550/arXiv.2201.04908.

Prasetyo, E., Suciati, N. and Fatichah, C. (2022) 'Multi-level residual network VGGNet for fish species classification', *Journal of King Saud University - Computer and Information Sciences*, 34(8), pp. 5286–5295. doi: 10.1016/j.jksuci.2021.05.015.

Rachburee, N. and Punlumjeak, W. (2022) 'Lotus species classification using transfer learning based on VGG16, ResNet152V2, and MobileNetV2', *IAES International Journal of Artificial Intelligence*, 11(4), pp. 1344–1352. doi: 591/ijai.v11.i4.pp1344-1352.

Rayhan Ahmed, M. *et al.* (2023) 'An ensemble 1D-CNN-LSTM-GRU model with data augmentation for speech emotion recognition', *Expert Systems with Applications*, 218(June 2021), p. 119633. doi: 10.1016/j.eswa.2023.119633.

Sang, D. V. and Cuong, L. T. B. (2019) 'Improving CRNN with EfficientNet-like feature extractor and multi-head atention for text recognition', *ACM International Conference Proceeding Series*, pp. 285–290. doi: 10.1145/3368926.3369689.

Saputra, W. A. (2022) 'Implementation of Spatial-Level Augmentation on Pneumonia Classification with Convolutional Neural Network', 4(2), pp. 1042–1050. doi: 10.47065/bits.v4i2.2270.

Sepp Hochreiter, J. S. (1997) 'LSTM CAN SOLVE HARD LO G TIME LAG PROBLEMS', in *Advances in neural information processing systems*, pp. 473-479.

Shallu and Mehra, R. (2018) 'Breast cancer histology images classification: Training from scratch or transfer learning?', *ICT Express*, 4(4), pp. 247–254. doi: 10.1016/j.icte.2018.10.007.

Siji George, C. G. and Sumathi, B. (2020) 'Grid search tuning of hyperparameters in random forest classifier for customer feedback sentiment prediction', *International Journal of Advanced Computer Science and Applications*, 11(9), pp. 173–178. doi: 10.14569/IJACSA.2020.0110920.

Singh, B., Rajiv, P. and Chandra, M. (2015) 'Lie detection using image processing', *ICACCS 2015 - Proceedings of the 2nd International Conference on Advanced Computing and Communication Systems*, pp. 3–7. doi: 10.1109/ICACCS.2015.7324092.

Singh, L. (2022) 'Deep bi-directional LSTM network with CNN features for human emotion recognition in audio-video signals', *International Journal of Swarm Intelligence*. doi: 10.1504/IJSI.2022.121102.

Soldner, F., Pérez-Rosas, V. and Mihalcea, R. (2019) 'Box of Lies®: Multimodal deception detection in dialogues', *NAACL HLT 2019 - 2019 Conference of the American Chapter of the Association for Computational Linguistics: n Language Technologies - Proceedings of the Conference*, 1, pp. 1768– doi: 10.18653/v1/n19-1175.

Srivastava, N. *et al.* (2014) 'Dropout: A simple way to prevent neural networks from overfitting', *Journal of Machine Learning Research*, 15, pp. 1929–1958.

Srivastava, N. and Dubey, S. (2020) 'Moth Monarch Optimization-Based Deep Belief Network in Deception Detection System', *Sadhana - Academy Proceedings in Engineering Sciences*, 45(1). doi: 10.1007/s12046-020-01354-w.

Sun, H. *et al.* (2020) 'Spectral-Spatial Attention Network for Hyperspectral Image Classification', *IEEE Transactions on Geoscience and Remote Sensing*, 58(5), pp. 3232–3245. doi: 10.1109/TGRS.2019.2951160.

Svm, A. *et al.* (2015) 'Eksperimen Deteksi Kebohongan berbasis EEG-P300', pp. 29–30.

Talaat, F. M. (2023) 'Explainable Enhanced Recurrent Neural Network for lie detection using voice stress analysis', *Multimedia Tools and Applications*, (0123456789). doi: 10.1007/s11042-023-16769-w.

Turnip, A. *et al.* (2017) 'Lie detection based EEG-P300 signal classified by ANFIS method', *Journal of Telecommunication, Electronic and Computer Engineering*, 9(1–5), pp. 107–110.

Vrij, A. *et al.* (2004) 'Detecting deceit via analyses of verbal and nonverbal behavior in children and adults', *Human Communication Research*, 30(1), pp. 8–41. doi: 10.1093/hcr/30.1.8.

Wang, J., Li, J. and Shi, X. (2021) 'Integrated design system of voice-visual VR based on multi-dimensional information analysis', *International Journal of Speech Technology*. doi: 10.1007/s10772-020-09696-w.

Wei-Ning Hsu, Benjamin Bolte, Y.-H. H. T., Kushal Lakhotia, Ruslan Salakhutdinov, and A. and Mohamed (2021) 'Representation, Hubert: Self-supervised speech Units, learning by masked prediction of hidden', in *IEEE/ACM Transactions on Audio, Speech, and Language Processing*. ieeexplore.ieee.org, pp. 3451–3460.

Wu, Z. *et al.* (2018) 'Deception detection in videos', *32nd AAAI Conference on ial Intelligence, AAAI 2018*, pp. 1695–1702. doi:)9/aaai.v32i1.11502.

Xiaobao Guo, Nithish Muthuchamy Selvaraj, Zitong Yu, Adams Wai-Kin Kong, Bingquan Shen, A. K. (2023) 'Audio-Visual Deception Detection: DOLOS Dataset and Parameter-Efficient Crossmodal Learning', in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22135–22145.

Ying-Li Tian, T. K. and and Jeffrey F.Cohn (2013) 'Chapter 11. Facial Expression Analysis', *Journal of Infectious Diseases*, 174(4), pp. 835–838.

Yohannes, Y. and Wijaya, R. (2021) 'Klasifikasi Makna Tangisan Bayi Menggunakan CNN Berdasarkan Kombinasi Fitur MFCC dan DWT', *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, 8(2), pp. 599–610. doi: 10.35957/jatisi.v8i2.470.

Yong Fang, Jian Gao, Cheng Huang, Hua Peng, R. W. (2019) 'Self Multi-Head Attention-based Convolutional Neural Networks for fake news detection', *PLoS ONE*. doi: 10.1371/journal.pone.0222713.

Younessi Heravi, M. A. *et al.* (2023) 'A New Approach for Lie Detection Using Non-Linear and Dynamic Analysis of Video-Based Eye Movement', *Frontiers in Biomedical Technologies*, 10(1), pp. 88–95. doi: 10.18502/fbt.v10i1.11516.

Yu, T. and Zhu, H. (2020) 'Hyper-Parameter Optimization: A Review of Algorithms and Applications', pp. 1–56. Available at: http://arxiv.org/abs/2003.05689.

Zabcikova, M., Koudelkova, Z. and Jasek, R. (2022) 'Concealed information detection using EEG for lie recognition by ERP P300 in response to visual stimuli: A review', *WSEAS Transactions on Information Science and Applications*, 19, pp. 171–179. doi: 10.37394/23209.2022.19.17.

Zhipeng, F. and Gani, H. (2021) 'Cultural Events Classification using Hyper-parameter Optimization of Deep Learning Technique', *International Journal of Advanced Computer Science and Applications*, 12(5), pp. 603–609. doi: 10.14569/IJACSA.2021.0120572.

Zhou, Y. and Bu, F. (2023) 'An Overview of Advancements in Lie Detection ology in Speech', *International Journal of Information Technologies and ıs Approach*, 16(2), pp. 1–24. doi: 10.4018/IJITSA.316935.

Zoubida, L. and Adjoudj, R. (2017) 'Integrating face and the both irises for personal authentication', *International Journal of Intelligent Systems.* j.mecs-press.net. Available at: https://j.mecs-press.net/ijisa/ijisa-v9-n3/IJISA-V9-N3-2.pdf.

Lampiran 1 Source code model untuk data image

```
from tensorflow.keras.models import load_model
from tensorflow.keras.applications.mobilenet import preprocess_input
import numpy as np
import cv2
# Muat dataset
dataset = np.load("dataset_image.npy", allow_pickle=True)
images = []
labels = []
# Pisahkan gambar dan label dari dataset, dan periksa validitas gambar
for data in dataset:
    image, label = data
    if image is not None and not image.size == 0:
        images.append(image)
        labels.append(label)
images = np.array(images)
labels = np.array(labels)
# Kurangi ukuran gambar menjadi 64x64 piksel dan konversi ke tipe data float32
resized_images = np.array([cv2.resize(image, (64, 64),
interpolation=cv2.INTER_LINEAR).astype(np.float32) for image in images])
# Load model MobileNet yang telah Anda latih
model_mobilenet = load_model("mobilenet82.h5")
# Normalisasi gambar menggunakan preprocess_input dari MobileNet
images_preprocessed = preprocess_input(resized_images)
# Ekstraksi fitur dari data gambar yang telah diubah ukurannya menggunakan
obileNet yang telah Anda latih
atures_mobilenet = model_mobilenet.predict(images_preprocessed)
```

```
# Pastikan untuk menyimpan hasil ekstraksi fitur dan label untuk digunakan
dalam pelatihan model fusi Anda
np.save("image_features_mobilenet.npy", image_features_mobilenet)
np.save("labels.npy", labels)
```

Lampiran 2 Source code model untuk data numeric

```
import pandas as pd
from tensorflow.keras.models import load_model
from tensorflow.keras.models import Model
import numpy as np
from tensorflow.keras.utils import to_categorical
# Load data from CSV
data = pd.read_csv("dataset.csv")
# Simpan kolom "Status" sebagai label numerik
labels_numerik = data['Status']
# Hapus kolom yang tidak diperlukan
data = data.drop(columns=['Video ID','Frame','Status','Pupil left Count', 'Pupil
Right Count', 'Pupil Center Count', 'Mouth Open Count', 'Mouth Close Count'])
# Load model MobileNet yang telah Anda latih
model_spasial = load_model("model_spasial82.h5")
# Proses data perubahan ekspresi wajah sesuai kebutuhan model
processed_data_perubahan_ekspresi = data  # Ganti dengan kode pemrosesan data
yang sesuai.
# Ekstraksi fitur dari model CNN spasial
feature_layer = Model(inputs=model_spasial.input,
outputs=model_spasial.layers[-2].output)  # Sesuaikan indeks lapisan yang sesuai
features_spasial = feature_layer.predict(processed_data_perubahan_ekspresi)
# Konversi label numerik menjadi format one-hot encoding (jika diperlukan)
# Misalnya, jika label hanya berupa 0 dan 1, Anda dapat menggunakan
                                                    ɔrical
                                                    ıe_hot = to_categorical(labels_numerik)
                                                    . hasil ekstraksi ke dalam file .npy
```

```
np.save("features_spasial.npy", features_spasial)
np.save("labels_one_hot.npy", labels_one_hot)
```

Lampiran 3 source code data suara

```
import os
import numpy as np
import librosa
from tensorflow.keras.models import load_model
from tensorflow.keras.models import Model
from tensorflow.keras.utils import to_categorical
import warnings
warnings.filterwarnings("ignore", message="Error in loading the saved optimizer
state*", category=UserWarning)
# Function for sound feature extraction using MFCC
def extract_mfcc(file_path, n_mfcc=20, n_fft=2048, hop_length=512,
max_length=182):
    y, sr = librosa.load(file_path)
    mfccs = librosa.feature.mfcc(y=y, sr=sr, n_mfcc=n_mfcc, n_fft=n_fft,
hop_length=hop_length)
    if mfccs.shape[1] > max_length:
        mfccs = mfccs[:, :max_length]
    else:
        mfccs = np.pad(mfccs, ((0, 0), (0, max_length - mfccs.shape[1])),
mode='constant')
    return mfccs
# Path to the folder containing sound data to extract features
sound_data_folder = 'Dataset'
# Path to the pre-trained sound model (replace with your actual model path)
sound_model_path = 'model.h5'
```

```
e pre-trained sound model
odel = load_model(sound_model_path)
n to predict label using the sound model and extract features
```

```
def predict_and_extract_feature(file_path, model, layer_name):
    mfccs = extract_mfcc(file_path)
    mfccs = np.expand_dims(mfccs, axis=0)  # Adjust dimensions for the model
    prediction = model.predict(mfccs)
# Replace 'layer_name' with the desired layer name in your model
    feature_extraction_model = Model(inputs=model.input,
outputs=model.get_layer(layer_name).output)
    extracted_features = feature_extraction_model.predict(mfccs)
    return extracted_features, prediction
# Extract sound features and make predictions for each sound file in the folder
all_features = []
all_labels = []
for label_name in os.listdir(sound_data_folder):
    label_dir = os.path.join(sound_data_folder, label_name)
        if os.path.isdir(label_dir):
        # Iterate through files in the subfolder
        for filename in os.listdir(label_dir):
            if filename.endswith(".wav"):
                file_path = os.path.join(label_dir, filename)
                  # Extract sound features and predict using the sound model
                extracted_features, prediction = predict_and_extract_feature(file_path,
sound_model, 'dense')
                # Assign label 1 for "Lie" and 0 for "Truth" based on folder name
                label = 1 if label_name.lower() == 'lie' else 0
                all_features.append(extracted_features)
                all_labels.append(label)
# Convert to numpy array
all_features = np.array(all_features)
all_labels = np.array(all_labels)
                e extracted features and labels for further testing or fusion data
                sound_features.npy", all_features)
                sound_labels.npy", all_labels)
```

Lampiran 4 Source code gabungan tiga model

```
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.models import Model
from tensorflow.keras.layers import Input, Dense, concatenate
import tensorflow as tf
# Load models
model1 = load_model('model.h5')
model2 = load_model('mobilenet82.h5')
model3 = load_model("model_spasial82.h5")
# Pisahkan output yang diinginkan dari masing-masing model
output1 = model1.get_layer('dense').output  # Ganti 'dense_1' dengan nama lapisan Dense yang sesuai di model pertama
output2 = model2.get_layer('global_average_pooling2d').output  # Ganti 'global_average_pooling2d' dengan nama lapisan yang sesuai di model kedua
output3 = model3.get_layer('dense_308').output  # Ganti 'layer_name' dengan nama lapisan yang sesuai di model ketiga
# Gabungkan output-output tersebut dengan operasi konkatenasi
merged = concatenate([output1, output2, output3])
# Buat model gabungan dengan tiga input (sesuaikan dengan input dari masing-masing model)
fusion_model = Model(inputs=[model1.input, model2.input, model3.input], outputs=merged)
# Compile model
fusion_model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
# Tampilkan ringkasan model
fusion_model.summary()
```