

DAFTAR PUSTAKA

- [1] Y. Gultom, A. M. Arymurthy, and R. J. Masikome, “Batik Classification using Deep Convolutional Network Transfer Learning,” *J. Ilmu Komput. dan Inf.*, vol. 11, no. 2, p. 59, Jun. 2018, doi: 10.21609/jiki.v11i2.507.
- [2] I. J. Goodfellow *et al.*, “Generative Adversarial Networks,” Jun. 2014, [Online]. Available: <http://arxiv.org/abs/1406.2661>
- [3] “hindupuravinash/the-gan-zoo: A list of all named GANs!” <https://github.com/hindupuravinash/the-gan-zoo> (accessed Jan. 28, 2022).
- [4] H.-W. Dong, W.-Y. Hsiao, L.-C. Yang, and Y.-H. Yang, “MuseGAN: Multi-track Sequential Generative Adversarial Networks for Symbolic Music Generation and Accompaniment”, Accessed: Jan. 28, 2022. [Online]. Available: <https://salu133445.github.io/musegan/>
- [5] T. Karras, S. Laine, and T. Aila, “A Style-Based Generator Architecture for Generative Adversarial Networks,” Dec. 2018, [Online]. Available: <http://arxiv.org/abs/1812.04948>
- [6] “agusekominarno/Batik: Batik Data set, Image Retrieval, Image Classification, Pattern Recognition.” <https://github.com/agusekominarno/Batik> (accessed Jan. 28, 2022).
- [7] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium.”
- [8] M. Seitzer, “mseitzer/pytorch-fid: Compute FID scores with PyTorch.,” 2020. <https://github.com/mseitzer/pytorch-fid> (accessed Sep. 02, 2022).
- [9] A. B. Prizilla and A. Sachari, “Teknik Klowong dalam Upaya Pengembangan Model Pewarisan Tradisi Membatik Warga Rifa’iyah di Desa Kalipucang Wetan Jawa Tengah,” *Fak. Ilmu Seni Rupa Dan Desain Inst. Teknol. Bandung*, vol. 7, no. 10, pp. 1–12, 2019.
- [10] R. C. Gonzalez and R. E. Woods, “Digital Image Processing Third Edition

Pearson,” p. 954, 2008.

- [11] M. Irfan, “Pengenalan Visi Komputer.”
<https://repository.unikom.ac.id/62692/1/CV-1-introduction.pdf>
- [12] I. Maliki, “Grafika Komputer”.
- [13] A. N. T. RD. Kusumanto, “PENGOLAHAN CITRA DIGITAL UNTUK MENDETEKSI OBYEK MENGGUNAKAN PENGOLAHAN WARNA MODEL NORMALISASI RGB,” *Stud. Environ. Sci.*, vol. 17, no. C, pp. 329–332, 1981, doi: 10.1016/S0166-1116(08)71924-1.
- [14] S. Nesi, “Pengenalan Pola Untuk Deteksi Uang Koin,” *Semin. Nas. Teknol. Inf. Komun. dan Ind. III*, pp. 18–24, 2011.
- [15] T. Karras, T. Aila, S. Laine, and J. Lehtinen, “Progressive Growing of GANs for Improved Quality, Stability, and Variation,” Oct. 2017, [Online]. Available: <http://arxiv.org/abs/1710.10196>
- [16] T. Paristech, “Understanding and Organising the Latent Space of Autoencoders Alasdair Newson,” 2020.
- [17] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” *32nd Int. Conf. Mach. Learn. ICML 2015*, vol. 1, pp. 448–456, 2015.
- [18] D. Ulyanov, A. Vedaldi, and V. Lempitsky, “Instance Normalization: The Missing Ingredient for Fast Stylization,” no. 2016, 2016, [Online]. Available: <http://arxiv.org/abs/1607.08022>
- [19] X. Huang and S. Belongie, “Arbitrary Style Transfer in Real-time with Adaptive Instance Normalization,” Mar. 2017, [Online]. Available: <http://arxiv.org/abs/1703.06868>
- [20] J. Lederer, “Activation Functions in Artificial Neural Networks: A Systematic Overview,” 2021.
- [21] J. Brownlee, “Probability for Machine Learning : Discover how to harness uncertainty with Python,” *Mach. Learn. Mastery*, p. 319, 2020.

- [22] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein GAN,” 2017, [Online]. Available: <http://arxiv.org/abs/1701.07875>
- [23] D. P. Kingma and J. L. Ba, “A : a m s o,” pp. 1–15, 2015.
- [24] “Learning PyTorch with Examples — PyTorch Tutorials 1.11.0+cu102 documentation.”
https://pytorch.org/tutorials/beginner/pytorch_with_examples.html
(accessed Jun. 21, 2022).
- [25] “A Gentle Introduction to torch.autograd — PyTorch Tutorials 1.11.0+cu102 documentation.”
https://pytorch.org/tutorials/beginner/blitz/autograd_tutorial.html (accessed Jun. 27, 2022).
- [26] D. Ulyanov, V. Lebedev, A. Vedaldi, and V. Lempitsky, “Texture Networks: Feed-forward Synthesis of Textures and Stylized Images,” Mar. 2016, [Online]. Available: <http://arxiv.org/abs/1603.03417>
- [27] J. Brownlee, “Better Deep Learning : Train Faster, Reduce Overfitting, and Make Better Predictions,” *Mach. Learn. Mastery With Python*, vol. 1, no. 2, p. 539, 2018.