

Ekstraksi Fitur pada File Gambar dengan Metode Fast Fourier Transform

Maria Irmira Prasetiyowati
Universitas Multimedia Nusantara
maria@umn.ac.id

ABSTRACT

There are several methods for performing feature extraction, one of which is the Fast Fourier Transform (FFT). FFT is an effective way to convert signals or time series data to the frequency domain. This study uses the FFT method for image data. Image data is transformed using FFT and returned using the Inverse Fast Fourier Transform (IFFT). From the results of the feature extraction test on nine image files, it was found that 78% of the image data experienced a decrease in size, 11% of the images had a fixed size, and 11% of the image data experienced an increase in size.

Keyword: feature extraction, FFT, IFFT

1. Pendahuluan

Teknologi berkembang cukup pesat, memungkinkan organisasi atau perusahaan untuk menggunakan berbagai jenis data terstruktur, semi terstruktur dan tidak terstruktur [1], [2]. Data Teks, gambar, audio, dan video adalah contoh data yang tidak terstruktur, sedangkan data seperti Extensible Markup Language (XML) dan bahasa tekstual untuk bertukar data di Web merupakan contoh data semi-terstruktur[1]. Baik data terstruktur, semi terstruktur maupun data tidak terstruktur membutuhkan ruang yang cukup besar dalam media penyimpan (hardisk), terutama untuk data berdimensi tinggi. Diperlukan suatu teknik atau metode supaya dapat mengurangi ruang untuk penyimpanan data tersebut.

Ada beberapa teknik yang dapat digunakan untuk mengurangi data berdimensi tinggi, dua diantaranya adalah seleksi fitur dan ekstraksi fitur. Seleksi fitur bertujuan untuk mendapatkan jumlah fitur minimum yang menghasilkan nilai akurasi klasifikasi yang tinggi. Penelitian tentang seleksi fitur ini sudah lama dilakukan, diawali pada awal tahun 1960-an [3]. Secara garis besar seleksi fitur dibedakan dalam tiga kategori [4], [5], yaitu Filter [6]–[9], Wrapper [10], dan metode Embedded [11]. Sedangkan dalam literatur lain disebutkan bahwa selain tiga kategori tersebut ada satu kategori baru yaitu hybrid [12]. Metode filter mengevaluasi relevansi subset fitur dengan dasar informasi, korelasi, dan konsistensi [13]. Fitur yang mempunyai performa baik akan dipilih, sisanya akan dibuang dari subset fitur. Sedangkan metode wrapper bergantung pada metode klasifikasi dengan pemilihan subset fitur secara random dan dievaluasi dengan menggunakan classifier[11]. Metode lainnya adalah metode embedded yang mengintegrasikan learning dan pemilihan fitur dalam satu proses. Sedangkan metode hybrid menggunakan kombinasi metode filter dan wrapper untuk memilih fitur yang berguna [14].

Selain seleksi fitur, pengurangan fitur juga dapat dilakukan dengan menggunakan ekstraksi fitur. Berbeda dengan seleksi fitur, ekstraksi fitur tidak membuang atau memilih subset fitur, namun membuat subset fitur baru dari fitur yang sudah ada. Tujuan dari ekstraksi fitur adalah mempertahankan sebagian besar informasi yang relevan dengan melakukan kompresi. Beberapa teknik dari ekstraksi fitur adalah Principal Component Analysis (PCA)[15]–[18], Linear Discriminant Analysis (LDA)[19]–[22], dan Fast Fourier Transform (FFT)[23]–[27].

Beberapa penelitian telah menggunakan metode FFT untuk ekstraksi fitur, beberapa diantaranya adalah penelitian yang mengidentifikasi apakah seseorang terkena penyakit epilepsi [28]. Penelitian ini menggunakan metode FFT dan Adaptive Neuro Fuzzy Inference System (ANFIS). FFT digunakan untuk melakukan transformasi sinyal EEG dan proses ekstraksi fitur untuk mengambil ciri dari setiap sinyal untuk proses klasifikasi. Dari penelitian ini didapatkan bahwa ekstraksi fitur dengan FFT, dan klasifikasi sinyal EEG menggunakan ANFIS dengan pembagian tiga kelas menghasilkan akurasi sebesar 89.33%, sensitivitas sebesar 89.37%, dan presisi sebesar 89.33%. Begitu juga dengan penelitian yang dilakukan oleh Saduadnyana, dkk, yang menggunakan FFT untuk ekstraksi fitur pada suara gamelan rindik[29]. Penelitian ini menggunakan tiga dataset, dan menghasilkan nilai akurasi sebesar 36% untuk 1 dataset, nilai akurasi 100% untuk 2 dataset dan 45% untuk 3 dataset. Ekstraksi fitur dengan metode FFT juga digunakan pada penelitian Anshory, dkk, yang hasilnya digunakan proses klasifikasi pada kasus penyakit jantung dengan nilai akurasi klasifikasi sebesar 87% [30]. Begitu juga penelitian yang dilakukan Mironovova, dkk, yang menggunakan FFT untuk memfilter sinyal dan mendeteksi R-peak serta ekstraksi detak jantung. Penelitian ini menghasilkan performa deteksi R-peak dapat diandalkan pada perkemana EKG yang lebih konstan dengan noise rendah[31]. Sedangkan penelitian lainnya

menggabungkan metode FFT dengan metode IFFT (Invers Fast Fourier Transform), diterapkan pada sekumpulan dataset [7]. Metode IFFT digunakan untuk mengembalikan data yang ada pada dataset ke dataset aslinya.

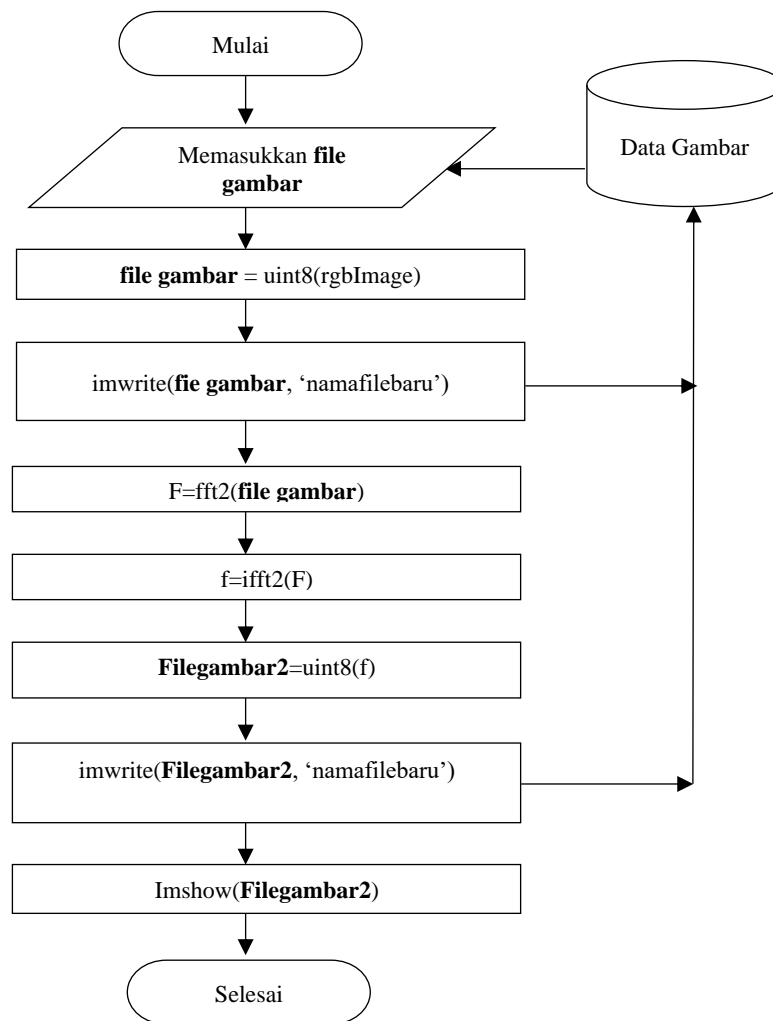
Dari beberapa penelitian yang telah diuraikan, telah terbukti bahwa Fast Fourier Transform dapat digunakan untuk melakukan ekstraksi fitur. Walaupun ekstraksi fitur yang telah dilakukan oleh peneliti-peneliti menggunakan dataset bukan untuk gambar namun bukan berarti FFT tidak dapat digunakan pada dataset gambar. Pada penelitian ini ekstraksi fitur diterapkan pada dataset gambar dengan menggunakan FFT, kemudian dikembalikan lagi ke data aslinya dengan menggunakan IFFT.

2. Metode Penelitian

Metode penelitian yang dilakukan meliputi dua tahap, yaitu perancangan dan pengambilan data, dengan uraian sebagai berikut:

1. Perancangan

Sebelum melakukan penelitian, perlu adanya perancangan dari aplikasi ekstraksi fitur yang dibuat. Secara garis besar, flowchart dari aplikasi yang dibangun dapat dilihat pada Gambar.1



Gambar 1. Flowchart Ekstraksi Fitur dengan FFT

2. Pengambilan Data

Data gambar yang digunakan pada penelitian ini ada 5 jenis gambar dengan sumber sebagai berikut:

- Gambar Lenna2.JPG, diambil dari internet <https://bellard.org/bpg/lena30.jpg> dengan ukuran 512x512, 30 KB.
- Gambar Lenna3.JPG, diambil dari internet, dengan ukuran 400 x 225, 28KB <http://jayitsecurity.blogspot.com/2013/01/lenajpg-interesting-story-of-image.html>

- Gambar Minion.JPG, diambil dari foto kamera HP samsung A7, dengan ukuran 4.608 x 3.456, 1.479 KB
- Gambar Bottle.JPG, diambil dari foto kamera HP samsung A7, dengan ukuran 2.448 x 3.264, 2.086 KB
- Gambar River .JPG, diambil dari foto kamera HP samsung A30, dengan ukuran 3.456 x 4.608, 5.985 KB

3. Implementasi dan Hasil

Pada tahapan ini beberapa data gambar diubah sizenya (rezise) dengan menggunakan beberapa tools yang ada untuk dapat diujicobakan ke dalam aplikasi ekstraksi fitur. Tools yang digunakan Mozjpeg, Jpegcrush, Paint Brush, dan Photoshop. Resize dilakukan untuk mengubah file extension gambar menjadi JPG.

Detil dari data gambar yang digunakan dapat dilihat pada Tabel 1.

Tabel 1. Detil Data Gambar

Nama	Size (KB)	DPI
Lenna2.JPG	30	96
LennaE1.JPG	95	96
LennaE2.JPG	82	72
Lenna3.JPG	28	100
Minion.JPG	1.479	96
MinionEdit.JPG	100	96
Bottle.JPG	2.086	72
Gedung.JPG	7.410	72
River.JPG	5.985	72

Ujicoba ekstraksi fitur dilakukan pada komputer Intel i5-8250U CPU dan RAM 12 GB dengan sembilan data gambar berformat JPG. Hasil ujicoba didapatkan bahwa ada 7 data gambar yang mengalami penurunan ukuran, 1 data gambar tidak berubah ukuran, dan 1 data gambar mengalami kenaikan ukuran. Hasil ekstraksi fitur lebih detail dapat dilihat pada Tabel 2 dan Gambar 2. Dari Tabel 2 terlihat bahwa hasil dari ekstraksi fitur, tujuh data gambar dengan format JPG mengalami penurunan ukuran file berkisar antara 0.09 sampai 48.73KB.

Tabel 2. Tabel Hasil Ekstraksi

Nama File	Size Awal	Size Akhir	Per-bedaan
Lenna2.JPG	30 KB	31 KB	-0,01
LennaE1.JPG	95 KB	38 KB	0,57
LennaE2.JPG	82 KB	41 KB	0,41
Lenna3.JPG	28 KB	19 KB	0,09
Minion.JPG	1.479 KB	1.479 KB	0,00
MinionEdit.JPG	100 KB	41 KB	0,59
Bottle.JPG	2.086 KB	474 KB	16,12
Gedung.JPG	7.410 KB	2.537 KB	48,73
River.JPG	5.985 KB	2.799 KB	31,86

3.1. Ujicoba pertama

Ujicoba pertama dilakukan pada data gambar Lenna yang diambil dari internet dengan ukuran 29.993 byte atau sama dengan 29 KB. Data gambar Lenna dikompresi dengan menggunakan Mozjpeg dan Jpegcrush sehingga menjadi 30 KB, dengan ukuran array 512 x 512, 96 DPI (Dots Per Inch), disimpan dengan nama Lenna2.JPG. Setelah data gambar Lenna2.JPG diekstraksi fiturnya, ukuran file mengalami kenaikan ukuran 1 KB, menjadi 31 KB, dari ukuran awal 30 KB.

3.2. Ujicoba kedua

Ujicoba kedua dilakukan pada data gambar Lenna2.JPG yang sudah dikompresi kemudian dibuka pada software Paintbrush, dan disimpan kembali dengan nama LennaE1.JPG, ukuran array 512 x 512. Ukuran file dari hasil penyimpanan ini adalah 95 KB. Ukuran file menurun 57 KB menjadi 38 KB setelah data gambar LennaE1.JPG diekstraksi fiturnya.

3.3. Ujicoba ketiga

Ujicoba ketiga, data gambar Lenna2.JPG disimpan kembali dengan format JPG menggunakan Photoshop (resize), dengan nama LennaE2.JPG. Ukuran file LennaE2.JPG menjadi 82 KB. Data gambar LennaE2 diekstraksi fiturnya menurun setengahnya, menjadi 41 KB.

3.4. Ujicoba keempat

Ujicoba keempat, menggunakan data Lenna berbeda, yang diambil langsung dari internet dengan ukuran 28 KB, ukuran array 400 x 225, dengan nama Lenna3.JPG. Setelah diekstraksi fitur, ukuran file dari data gambar Lenna3.JPG mengalami penurunan 9 KB menjadi 19 KB.

3.5. Ujicoba kelima

Ujicoba kelima dilakukan pada data gambar Minion.JPG. Data gambar ini diambil dari kamera HP Samsung A7, dengan ukuran file 1.479 KB, array 4.608 x 3.456. Setelah fitur diekstraksi ukuran file tetap sama dengan ukuran file semula yaitu 1.479 KB.

3.6. Ujicoba keenam

Ujicoba keenam, data Minion.JPG di-resize dengan menggunakan software Paintbrush, dan disimpan dengan nama MinionEdit.JPG, dengan ukuran 100 KB. Setelah data gambar MinionEdit.JPG diekstraksi fiturnya, ukuran file menurun 59 KB menjadi 41 KB.

3.7. Ujicoba ketujuh

Ujicoba ketujuh, dilakukan pada data gambar Bottle.JPG. Data gambar ini diambil dari kamera HP Samsung Galaxy A7, dengan ukuran file 2.086 KB, array 2.448 x 3.264. Setelah fitur diekstraksi ukuran file menurun 1.612 KB menjadi 474 KB.

3.8. Ujicoba kedelapan

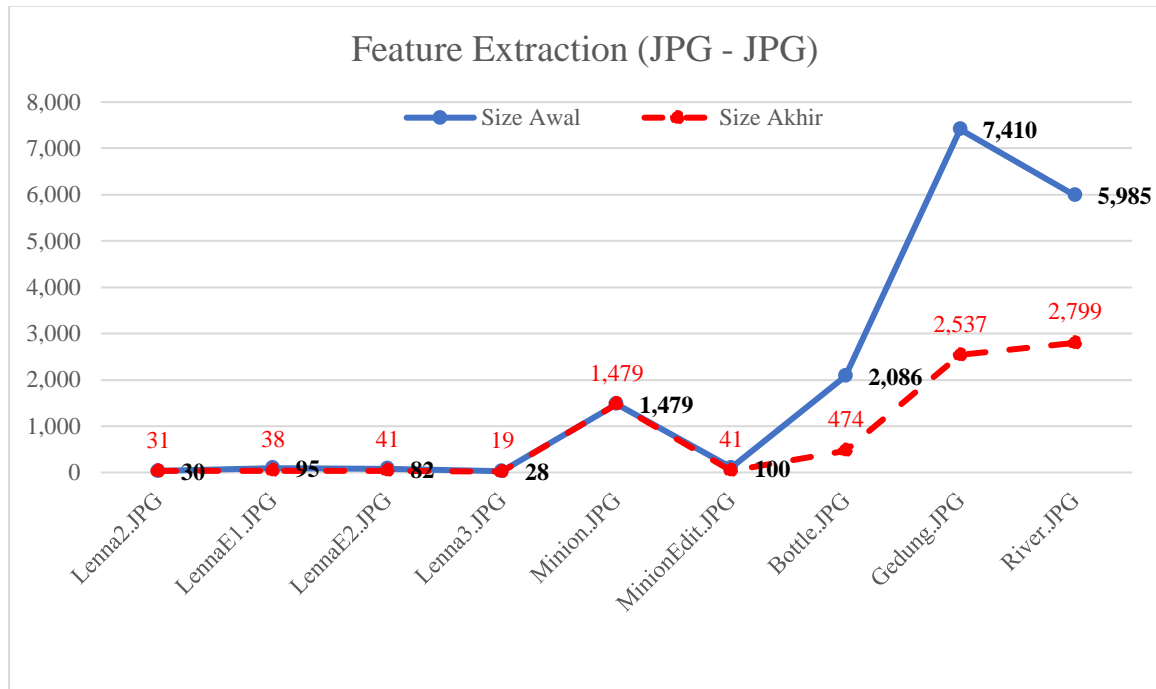
Ujicoba kedelapan, dilakukan pada data gambar Gedung.JPG. Data gambar ini diambil dari kamera HP Samsung Galaxy A30, dengan ukuran file 7.410 KB, array 4.608 x 3.456. Setelah fitur diekstraksi ukuran file menurun 4.873 KB, menjadi 2.357 KB.

3.9. Ujicoba kesembilan

Ujicoba kesembilan, dilakukan pada data gambar River.JPG. Data gambar ini diambil dari kamera HP Samsung Galaxy A30, dengan ukuran file 5.985 KB, array 3.456 x 4.608. Setelah fitur diekstraksi ukuran file menurun 3.186 KB menjadi 2.799 KB.

4. Pembahasan

Pada Tabel 2 dan Gambar 2 tampak bahwa 7 dari 9 data gambar (78%) yang digunakan mengalami penurunan ukuran (size). Hanya satu gambar (11%) yang mengalami kenaikan ukuran menjadi 31 KB, dari ukuran awal 30 KB, dan satu data gambar (11%) tidak mengalami penurunan maupun kenaikan ukuran (size). Dari hasil ujicoba yang dilakukan terlihat bahwa data gambar yang mengalami penurunan ukuran file, adalah data gambar yang telah mengalami manipulasi atau penyimpanan kembali (resize). Fungsi FFT di sini akan mengembalikan complex double array. Sehingga ketika file berekstensi .JPG dibaca, file tersebut mempunyai tipe array UINT8. Dan ketika IFFT dieksekusi, maka data akan dikembalikan menjadi UINT8. UINT8 merupakan tipe data yang berisi semua bilangan bulat dari 0 (nol) sampai 255. UINT8 banyak digunakan dalam grafik. Hasil lengkap dari pengujian yang telah dilakukan dapat dilihat pada Gambar 2.



Gambar 2. Ekstraksi Fitur

Hal ini membuka peluang untuk mengenali apakah suatu data gambar adalah data yang sudah dimanipulasi atau tidak

5. Kesimpulan

Dalam penelitian ini, didapatkan bahwa dari hasil uji ekstraksi ciri pada sembilan file citra diketahui bahwa 78% data citra mengalami penurunan ukuran, 11% citra memiliki ukuran tetap, dan 11% data citra mengalami peningkatan ukuran. Selain itu, hasil ujicoba didapatkan bahwa data gambar yang mengalami penurunan ukuran file adalah data gambar yang sudah dimanipulasi.

Sedangkan untuk saran penelitian selanjutnya dapat menambahkan ujicoba dengan menggunakan data gambar dengan format (extension) data yang berbeda seperti PNG dan TIF, dan hasil ekstraksi fitur yang dihasilkan diubah ke dalam format yang berbeda. Selain itu dapat dilihat pada hasil ujicoba bahwa sebagian besar data gambar yang dimanipulasi mengalami penurunan ukuran file. Hal ini membuka peluang untuk mengenali apakah suatu data gambar adalah data yang sudah dimanipulasi atau tidak

Ucapan Terima Kasih

Ucapan terima kasih kepada Universitas Multimedia Nusantara (UMN) atas dukungan dan sumber daya yang telah diberikan kepada peneliti.

Daftar Pustaka

- [1] A. Gandami dan M. Haider, "Beyond the hype: Big data concepts, methods, and analytics," *International Journal of Information Management*, vol. 35, no. 2, hlm. 137–144, Apr 2015, doi: 10.1016/j.ijinfomgt.2014.10.007.
- [2] J. Wu dan S. Coggeshall, *Foundations of Predictive Analytics*, 0 ed. Chapman and Hall/CRC, 2012. doi: 10.1201/b11639.
- [3] P. Lewis, "The characteristic selection problem in recognition systems," *IEEE Trans. Inform. Theory*, vol. 8, no. 2, hlm. 171–178, Feb 1962, doi: 10.1109/TIT.1962.1057691.
- [4] I. Guyon dan A. Elisseeff, "An Introduction to Variable and Feature Selection".
- [5] A. Zakeri dan A. Hokmabadi, "Efficient feature selection method using real-valued grasshopper optimization algorithm," *Expert Systems with Applications*, vol. 119, hlm. 61–72, Apr 2019, doi: 10.1016/j.eswa.2018.10.021.
- [6] Z. Liu, B. Chang, dan F. Cheng, "An interactive filter-wrapper multi-objective evolutionary algorithm for feature selection," *Swarm and Evolutionary Computation*, vol. 65, hlm. 100925, Agu 2021, doi: 10.1016/j.swevo.2021.100925.
- [7] M. I. Prasetyowati, N. U. Maulidevi, dan K. Surendro, "Determining threshold value on information gain feature selection to increase speed and prediction accuracy of random forest," *J Big Data*, vol. 8, no. 1, Art. no. 1, Des 2021, doi: 10.1186/s40537-021-00472-4.
- [8] J. Ba, P. Wang, X. Yang, H. Yu, dan D. Yu, "Glee: A granularity filter for feature selection," *Engineering Applications of Artificial Intelligence*, vol. 122, hlm. 106080, Jun 2023, doi: 10.1016/j.engappai.2023.106080.

- [9] J. Wen, H. Chu, Z. Lai, T. Xu, dan L. Shen, "Enhanced robust spatial feature selection and correlation filter learning for UAV tracking," *Neural Networks*, vol. 161, hlm. 39–54, Apr 2023, doi: 10.1016/j.neunet.2023.01.003.
- [10] S. Jadhav, H. He, dan K. Jenkins, "Information gain directed genetic algorithm wrapper feature selection for credit rating," *Applied Soft Computing*, vol. 69, hlm. 541–553, Agu 2018, doi: 10.1016/j.asoc.2018.04.033.
- [11] R. Kohavi dan G. H. John, "Wrappers for feature subset selection," *Artificial Intelligence*, vol. 97, no. 1–2, hlm. 273–324, Des 1997, doi: 10.1016/S0004-3702(97)00043-X.
- [12] K. Jha dan S. Saha, "Incorporation of multimodal multiobjective optimization in designing a filter based feature selection technique," *Applied Soft Computing*, vol. 98, hlm. 106823, Jan 2021, doi: 10.1016/j.asoc.2020.106823.
- [13] H. Liu dan R. Setiono, "A Probabilistic Approach to Feature Selection—a filter solution," *ICML*, vol. 96, hlm. 319–327, 1996.
- [14] S. Das, "Filters, Wrappers and a Boosting-Based Hybrid for Feature Selection," *Proceeding of the Eighteenth International Conference on Machine Learning (ICML 2001)*, Jul 2001.
- [15] Q. Liu, Z. Gui, S. Xiong, dan M. Zhan, "A principal component analysis dominance mechanism based many-objective scheduling optimization," *Applied Soft Computing*, vol. 113, hlm. 107931, Des 2021, doi: 10.1016/j.asoc.2021.107931.
- [16] K. Gupta, "An integrated batting performance analytics model for women's cricket using Principal Component Analysis and Gini scores," *Decision Analytics Journal*, vol. 4, hlm. 100109, Sep 2022, doi: 10.1016/j.dajour.2022.100109.
- [17] B. Yesilkaya, E. Sayilgan, Y. K. Yuçe, M. Perc, dan Y. Isler, "Principal component analysis and manifold learning techniques for the design of brain-computer interfaces based on steady-state visually evoked potentials," *Journal of Computational Science*, vol. 68, hlm. 102000, Apr 2023, doi: 10.1016/j.jocs.2023.102000.
- [18] P. Balasubramanian dan P. T. Praharaj, "Principal component analysis revealed the key influencing factors of kombucha bacterial cellulose yield and productivity," *Bioresource Technology Reports*, vol. 23, hlm. 101539, Sep 2023, doi: 10.1016/j.biteb.2023.101539.
- [19] J. D. C. Souza, S. F. C. Soares, L. C. M. De Paula, C. J. Coelho, M. C. U. De Araújo, dan E. C. D. Silva, "Bat algorithm for variable selection in multivariate classification modeling using linear discriminant analysis," *Microchemical Journal*, vol. 187, hlm. 108382, Apr 2023, doi: 10.1016/j.microc.2022.108382.
- [20] T. Bahraini, S. M. Hosseini, M. Ghasempour, dan H. S. Yazdi, "Density-oriented linear discriminant analysis," *Expert Systems with Applications*, vol. 187, hlm. 115946, Jan 2022, doi: 10.1016/j.eswa.2021.115946.
- [21] K. T. Le, C. Chau, F. J. P. Richard, dan E. Guedj, "An adapted linear discriminant analysis with variable selection for the classification in high-dimension, and an application to medical data," *Computational Statistics & Data Analysis*, vol. 152, hlm. 107031, Des 2020, doi: 10.1016/j.csda.2020.107031.
- [22] A. Tantin, E. Bou Assi, E. Van Asselt, S. Hached, dan M. Sawan, "Predicting urinary bladder voiding by means of a linear discriminant analysis: Validation in rats," *Biomedical Signal Processing and Control*, vol. 55, hlm. 101667, Jan 2020, doi: 10.1016/j.bspc.2019.101667.
- [23] M. I. Prasetyowati, N. U. Maulidevi, dan K. Surendro, "The accuracy of Random Forest performance can be improved by conducting a feature selection with a balancing strategy," *PeerJ Computer Science*, vol. 8, hlm. e1041, Jul 2022, doi: 10.7717/peerj-cs.1041.
- [24] A. M. Grigoryan dan S. S. Agaian, "Split manageable efficient algorithm for Fourier and Hadamard transforms," *IEEE Trans. Signal Process.*, vol. 48, no. 1, hlm. 172–183, Jan 2000, doi: 10.1109/78.815487.
- [25] S. G. Johnson dan M. Frigo, "A Modified Split-Radix FFT With Fewer Arithmetic Operations," *IEEE Trans. Signal Process.*, vol. 55, no. 1, hlm. 111–119, Jan 2007, doi: 10.1109/TSP.2006.882087.
- [26] Y. Zhou, W. Cao, L. Liu, S. Agaian, dan C. L. P. Chen, "Fast Fourier transform using matrix decomposition," *Information Sciences*, vol. 291, hlm. 172–183, Jan 2015, doi: 10.1016/j.ins.2014.08.022.
- [27] E. E. Swartzlander dan H. H. M. Saleh, "FFT Implementation with Fused Floating-Point Operations," *IEEE Trans. Comput.*, vol. 61, no. 2, hlm. 284–288, Feb 2012, doi: 10.1109/TC.2010.271.
- [28] S. Suwanto, M. H. Bisri, D. C. R. Novitasari, dan A. H. Asyhar, "Classification of EEG Signals using Fast Fourier Transform (FFT) and Adaptive Neuro Fuzzy Inference System (ANFIS)," *JMM*, vol. 5, no. 1, hlm. 35–44, Mei 2019, doi: 10.15642/mantik.2019.5.1.35-44.
- [29] I. K. K. Saduadnyana, I. K. G. Suhartana, L. A. A. R. Putri, I. G. N. A. C. Putra, I. K. A. Mogi, dan I. B. M. Mahendra, "Implementasi Double Frequency Modulation dan FFT dalam Sintesis Suara Rindik," *JLK*, vol. 11, no. 1, hlm. 93, Jul 2022, doi: 10.24843/JLK.2022.v11.i01.p10.
- [30] I. Anshory dan A. Efiyanti, "Feature Extraction of Heart Signals using Fast Fourier Transform".
- [31] M. Mironovova dan J. Bila, "Fast fourier transform for feature extraction and neural network for classification of electrocardiogram signals," dalam *2015 Fourth International Conference on Future Generation Communication Technology (FGCT)*, Luton, United Kingdom: IEEE, Jul 2015, hlm. 1–6. doi: 10.1109/FGCT.2015.7300244.