

KLASIFIKASI CITRA IKAN DENGAN MENGGUNAKAN METODE CNN DAN GLCM

Bryan Immanuel

Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Tarumanagara
Jl. Letjen S. Parman No.1, Jakarta Barat, DKI Jakarta, Indonesia 11410
e-mail: bryan.535180090@stu.untar.ac.id

ABSTRAK

Penelitian ini berisi tentang klasifikasi citra dari enam jenis ikan dengan total 300 gambar pada masing-masing jenis dengan menggunakan metode *Convolutional Neural Network* (CNN) dan *Gray Level Co-occurrence Matrix* (GLCM). Penelitian ini menggunakan Python sebagai bahasa pemrograman utama, namun dibantu juga dengan library lainnya. Arsitektur CNN yang digunakan adalah *InceptionV3*, *ResNet50* dan *Xception*. Sedangkan, hasil dengan metode GLCM diperoleh melalui *Decision Tree*, *Logistic Regression*, *Neural Network*, *Bagging Classifier*, dan *Gradient Boosting*. Dengan tujuan membantu user untuk membedakan suatu jenis ikan dari ikan yang lainnya dan memberikan analisis yang lengkap berdasarkan citranya. Hasil penelitian ini memberi tingkat keberhasilan dari model CNN dengan arsitektur *InceptionV3* dengan total *precision* mencapai 83%, *recall* sebesar 72%, *F1-score* dan akurasi dengan nilai yang sama yaitu 72%.

Kata kunci: Klasifikasi, Pengolahan Citra, CNN, GLCM, Ikan

ABSTRACT

This research involves the classification of images of six types of fish, with a total of 300 images for each type, using the Convolutional Neural Network (CNN) and Gray Level Co-occurrence Matrix (GLCM) methods. This research uses Python as the main programming language, but is also supported by other libraries. The CNN architectures used are InceptionV3, ResNet50, and Xception. Meanwhile, the results using the GLCM method were obtained through Decision Tree, Logistic Regression, Neural Network, Bagging Classifier, and Gradient Boosting. The aim is to help users distinguish one type of fish from another and provide a complete analysis based on the image. The results of this study show the success rate of the CNN model with the InceptionV3 architecture, with a total precision of 83%, recall of 72%, F1-score and accuracy with the same value of 72%.

Keywords: Classification, Image Processing, CNN, GLCM, Fish.

1. PENDAHULUAN

Sebagai negara yang berpulau, Indonesia sangat bersandar kepada ikan sebagai bahan dasar pangan. Pemerintah juga sangat mendorong pergerakan mengonsumsi ikan karena sangat dipercaya bahwa ikan memiliki khasiat-khasiat yang sangat bermanfaat, terutama di bagian otak dan kepintaran. Selain itu, bisnis jual beli ikan juga mendukung ekonomi bangsa ini, dan juga memberi hidup yang lebih baik untuk para nelayan dan orang-orang yang terlibat didalam pekerjaan ini. Namun, sebaik-baiknya khasiat ikan, tentu ada ikan yang juga tidak baik untuk dikonsumsi, dan itu bisa sangat membahayakan, apalagi dengan berbagai banyak jenis ikan di perairan kita. Mengingat era sekarang, dimana tidak semua orang bisa mengenali satu jenis ikan dari yang lainnya, dibutuhkan sebuah alternatif dimana mereka bisa mengenali satu jenis ikan dari yang lainnya.

Maka dari itu, penelitian ini ditujukan untuk membuat sebuah program untuk memberi klasifikasi dan pembedaan dari satu ikan dengan yang lainnya, sehingga memberi hasil yang memuaskan bagi mereka yang mencari jawaban yang berhubungan dengan topik ini. Proses penelitian ini menggunakan beberapa model *Convolutional Neural Network* (CNN) dengan arsitektur *InceptionV3*, *ResNet50*, dan *Xception* serta menggunakan metode *Gray Level Co-occurrence Matrix* (GLCM) melalui algoritma *Decision Tree*, *Logistic Regression*, *Neural Network* sederhana, *Bagging Classifier*, dan *Gradient Boosting*.

2. TINJAUAN LITERATUR

Setelah melakukan pencarian terhadap penelitian lain yang relevan dengan topik penelitian yang dilakukan ini, beberapa penelitian berikut menjadi referensi dan contoh rujukan pada penelitian ini, sehingga menarik untuk dianalisis kembali.

Penelitian pertama yang sesuai berjudul “*A Deep Convolution Neural Network Model for Vehicle Recognition and Face Recognition*” [1]. Tujuan dari penelitian ini adalah untuk mencari alternatif baru untuk meningkatkan akurasi dari efisiensi pengenalan akan berbagai citra dan data, khususnya dalam data pengenalan akan kendaraan bermotor. Algoritma tradisional *machine learning* memiliki banyak kekurangan, sehingga teknik *deep learning* dikembangkan sebagai alternatif solusi. Oleh karena itu, sebuah jaringan sembilan-lapis terbangun, dan mencapai hasil yang memecahkan rekor, yaitu 92.2% dengan menggunakan *framework deep learning* bernama *Caffe*. Metode ini mampu memberi definisi dari model, pengaturan pengoptimalan dan beban latihan, sehingga dapat menjalankan model terbaik dan data dalam jumlah yang sangat besar.

Penelitian kedua berjudul “*Food Image Classification with Convolutional Neural Network*” [2]. Tujuan dari penelitian ini adalah untuk menyortir dan mengklasifikasi berbagai macam gambar makanan yang ada di sosial media, sekaligus membantu restoran untuk mengiklankan jualan mereka ke pasar yang sesuai, ke orang-orang yang memang menyukai makanan yang serupa. Dari hal ini, manfaat secara finansial dan teknis bagi pemilik restoran bisa tercapai. Data yang digunakan berupa 16643 citra makanan yang dibagi menjadi 11 kelompok penting. Kelompok-kelompok itu adalah Produk susu, Roti, Makanan Penutup, Telur, Goreng, Makanan, Daging, Mie, Nasi, Sop, *Seafood*, dan Buah. Metode yang digunakan dalam penelitian ini adalah CNN untuk mengklasifikasi dengan arsitektur *InceptionV3* yang terlebih dahulu menjalani *pre-processing*, dimana setiap citra diperlakukan secara *fine-tuning* untuk menyesuaikan input pada model yang dibangun dan memaksimalkan outputnya..

3. METODE PENELITIAN

3.1 Pengolahan Data

Dalam penelitian ini, data yang digunakan berupa enam jenis ikan yang masing-masing dikumpulkan sebanyak 50 gambar. Dengan begitu, total gambar yang dikumpulkan ada sebanyak 300 gambar. Semua data gambar diunduh dari *platform Google Images* dengan format data adalah jpg. Sebelum melakukan pengolahan lebih lanjut, data terlebih dahulu akan melalui tahap pra-pemrosesan, dimana semua gambar akan melewati diatur sedemikian rupa agar memiliki karakteristik yang sama dan dapat diolah dengan mudah oleh model yang akan dibangun. Pra-pemrosesan biasanya berisi *rescale*, *rotation range*, *shear range*, dan lainnya.

Setelah pra-pemrosesan selesai dilakukan, setiap gambar dapat diolah dengan pendekatan GLCM dan CNN. GLCM merupakan teknik analisis tekstur pada sebuah citra karena setiap gambar memiliki pola dan tekstur yang berbeda-beda, GLCM juga mampu mengekstraksi informasi yang relevan dari fitur pada gambar dan menghitung nilai probabilitas dari hasil perhitungan hubungan ketetanggaan antara dua pixel pada jarak dan orientasi sudut tertentu. [3].

3.2 GLCM

Dalam proses GLCM, citra matriks berisi 3 lapisan warna dasar, yaitu dasar warna alias RGB namun dikonversi menjadi 1 lapisan matriks *grayscale*. Setelah melewati berbagai proses GLCM lainnya, kemudian ciri fitur statistik GLCM digunakan untuk mendapatkan hasil akhir proses algoritma. Fitur-fitur statistik pada GLCM dengan sudut-sudutnya yang ditunjukkan pada Gambar 1 serta dapat disimbolkan dengan persamaan berikut ini [4][6]:

a. *Entropy*

Entropy digunakan untuk mengukur keteracakan dari distribusi intensitas. Persamaan *entropy*:

$$E2 = - \sum_i^m \sum_j^n p(i,j) \log\{p(i,j)\}$$

Dimana:

P = Matriks GLCM hasil normalisasi

i = Indeks baris matriks P

j = indeks kolom matriks P

b. *Energi*

Energi merupakan fitur GLCM yang digunakan untuk mengukur konsentrasi pasangan intensitas pada matriks GLCM dan didefinisikan dalam persamaan berikut ini:

$$H = \sum_i^m \sum_j^n \frac{p(i,j)}{1 + (i-j)^2}$$

c. *Homogeneity*

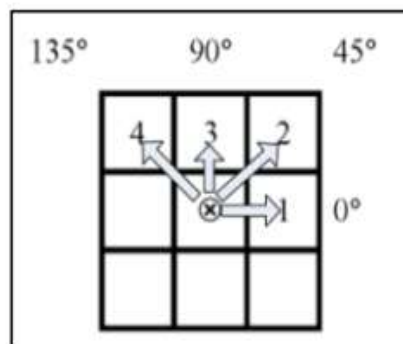
Menunjukkan homogenitas variasi intensitas dalam citra yang dalam persamaannya dapat didefinisikan sebagai berikut:

$$E1 = \sum_i^m \sum_j^n P(i,j)^2$$

d. *Contrast*

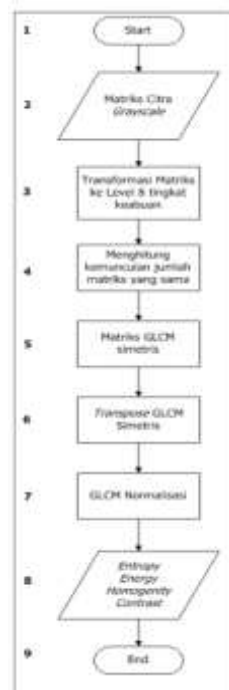
Contrast adalah perhitungan perbedaan intensitas antara pixel yang satu dan pixel yang berdekatan di seluruh gambar. Kontras bernilai nol untuk gambar yang konstan. Persamaannya dapat dilihat sebagai berikut:

$$C2 = \sum_{i=0}^m \sum_{j=0}^n P_{(i,j)}(i,j)^2$$



Gambar 1. Ilustrasi Sudut GLCM

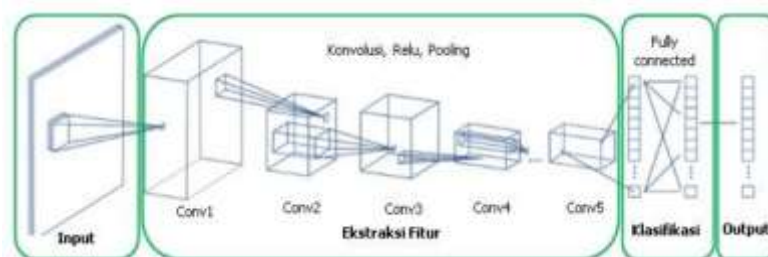
Berikut pada Gambar 2 adalah *flowchart* untuk proses dengan GLCM.



Gambar 3. Alur Pengolahan dengan Metode GLCM

3.3 CNN

Pendekatan selanjutnya adalah *deep learning* dengan model CNN. Metode ini digunakan atas dasar dilakukannya ekstraksi fitur melalui proses konvolusi namun dapat langsung melakukan klasifikasi dari data citra. CNN merupakan hasil operasi konvolusi yang menggabungkan beberapa lapisan jaringan *neural* dalam pemrosesannya dengan elemen yang beroperasi secara paralel karena terinspirasi oleh sistem saraf biologis [5]. Pada Gambar 2 ditunjukkan salah satu contoh cara kerja dari model CNN.



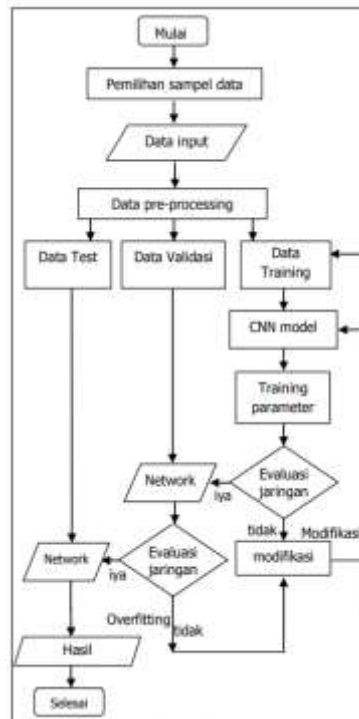
Gambar 3. Arsitektur CNN

Struktur CNN terdiri dari input, ekstraksi fitur, konvolusi, dan output yang bekerja secara hierarki. *Output* dari lapisan konvolusi pertama akan digunakan sebagai input untuk lapisan konvolusi kedua, dan begitu seterusnya. Hal ini dikarenakan proses dari ekstraksi fitur cukup panjang, sehingga banyak dari bagiannya dimudahkan menjadi berbagai macam, seperti konvolusi, *ReLU*, *Pooling*, dan *Flatten* [7].

Langkah pertama dari proses ini adalah mengubah data citra dari yang tadinya berwarna menjadi *grayscale* atau monokromatik, alias abu-abu. Kemudian dihitung matriksnya untuk mendapatkan nilai terbaik. Kemudian dihitung juga ciri-ciri fitur GLCM pada masing-masing data citra.

Flowchart dari proses CNN yang ditunjukkan oleh Gambar 4 dapat dibilang sedikit lebih kompleks namun proses ini adalah bagian integral dari jalannya CNN. Seperti pada umumnya, data

harus dikumpulkan dan kemudian memasuki tahap pra-pemrosesan, dimana data dilatih dan diuji untuk mencapai hasil terbaik dari proses tersebut. Setelah terbentuknya model dari proses algoritma itu, jaringan akan terbentuk dan memberikan hasil akurasi yang sesuai dengan kebutuhan.



Gambar 4. Alur Pengolahan dengan Metode CNN

3.4 Metode Evaluasi

Metode evaluasi yang digunakan adalah melihat hasil yang muncul di *classification report*, dimana nilai-nilai dari fitur itu digunakan sebagai metode evaluasi. Fitur yang digunakan adalah *precision*, *recall*, *F1-score* dan *support*. Nilai yang juga dilihat untuk menentukan hasil tersebut baik atau tidak adalah akurasi karena akan sangat menentukan apakah proses itu berhasil atau tidak.

4. HASIL DAN PEMBAHASAN

Dalam proses penelitian data citra yang diolah dengan metode GLCM, hasil akhir dari datanya akan dibagi berdasarkan algoritma klasifikasi seperti yang sudah disebutkan sebelumnya.:

Berikut ditampilkan hasil dari masing-masing algoritma pada Gambar 5, Gambar 6, Gambar 7, Gambar 8, dan Gambar 9.

	precision	recall	f1-score	support
Bawal	0.13	0.15	0.14	13
Kembung	0.13	0.14	0.14	14
Lele	0.23	0.21	0.22	14
Mujair	0.17	0.13	0.15	15
Tengiri	0.17	0.29	0.21	14
Tongkol	0.09	0.05	0.06	20
accuracy			0.16	90
macro avg	0.15	0.16	0.15	90
weighted avg	0.15	0.16	0.15	90

Gambar 5. Hasil Evaluasi *Decision Tree*

	precision	recall	f1-score	support
Bawal	0.00	0.00	0.00	13
Kembung	0.31	0.29	0.30	14
Lele	0.15	0.29	0.20	14
Mujair	0.28	0.47	0.35	15
Tengiri	0.12	0.21	0.15	14
Tongkol	0.00	0.00	0.00	20
accuracy			0.20	90
macro avg	0.14	0.21	0.17	90
weighted avg	0.14	0.20	0.16	90

Gambar 6. Hasil Evaluasi *Logistic Regression*

	precision	recall	f1-score	support
Bawal	1.00	0.08	0.14	13
Kembung	0.00	0.00	0.00	14
Lele	0.00	0.00	0.00	14
Mujair	0.23	0.40	0.29	15
Tengiri	0.09	0.21	0.13	14
Tongkol	0.39	0.35	0.37	20
accuracy			0.19	90
macro avg	0.29	0.17	0.16	90
weighted avg	0.28	0.19	0.17	90

Gambar 7. Hasil Evaluasi *Neural Network*

	precision	recall	f1-score	support
Bawal	0.19	0.38	0.25	13
Kembung	0.18	0.14	0.16	14
Lele	0.00	0.00	0.00	14
Mujair	0.09	0.13	0.11	15
Tengiri	0.25	0.21	0.23	14
Tongkol	0.27	0.15	0.19	20
accuracy			0.17	90
macro avg	0.16	0.17	0.16	90
weighted avg	0.17	0.17	0.16	90

Gambar 8. Hasil Evaluasi *Bagging Classifier*

	precision	recall	f1-score	support
Bawal	0.23	0.23	0.23	13
Kembung	0.36	0.36	0.36	14
Lele	0.17	0.14	0.15	14
Mujair	0.14	0.20	0.17	15
Tengiri	0.19	0.21	0.20	14
Tongkol	0.36	0.25	0.29	20
accuracy			0.23	90
macro avg	0.24	0.23	0.23	90
weighted avg	0.25	0.23	0.24	90

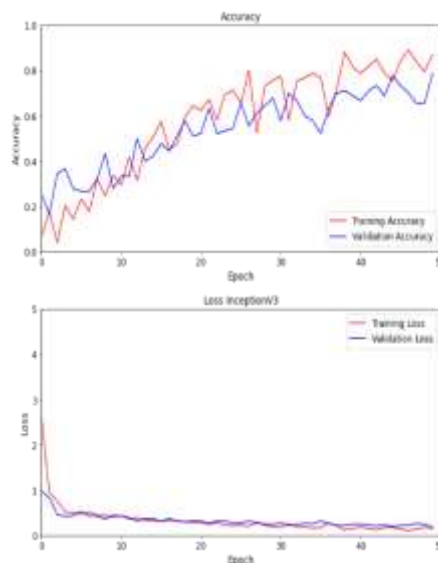
Gambar 9. Hasil Evaluasi *Gradient Boosting*

Dapat dilihat bahwa hasil yang diperoleh dari masing-masing algoritma yang didahului oleh proses GLCM memberikan hasil yang kurang maksimal. Hasil terbaik yang bisa diperoleh dari metode ini hanya dapat ditemukan pada algoritma *Gradient Boosting* dan *Logistic Regression*. Hal

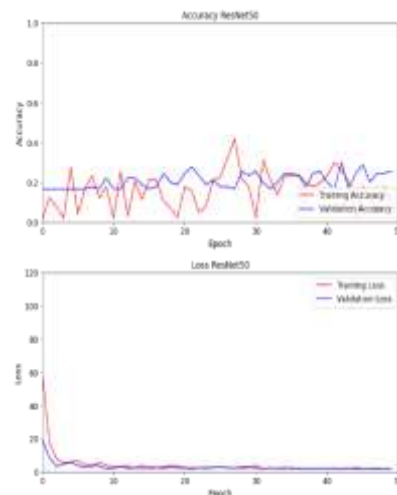
ini bisa terjadi karena adanya kemungkinan dalam memproses setiap fitur pada citra, *Gradient Boosting* memperhitungkan nilai matriks *grayscale* dan tidak menghilangkan lebih banyak fitur pada citra namun tetap menghasilkan nilai terbaik dengan *vanishing gradient*, sehingga hasil yang diperoleh dapat lebih maksimal.

Sedangkan, pada proses data citra dengan algoritma *Logistic Regression*, setiap fitur masih tetap dapat dipertahankan tanpa menghilangkan nilai matriks yang mengangkut informasi data yang penting, sehingga ketika melalui proses regresi nilai tersebut masih tersimpan dan otentik. Oleh karena itu, *output* yang dihasilkan masih memiliki informasi yang penting namun berhasil diperbarui tanpa kehilangan esensinya.

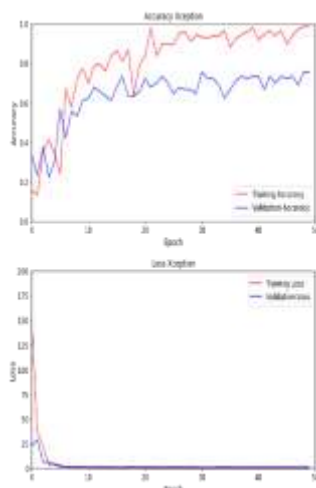
Dalam pengolahan data citra yang dimiliki dengan pendekatan *deep learning*, data citra yang ada akan terlebih dahulu dilatih. Setelah dilatih, data kemudian menjadi input ke dalam arsitektur untuk diolah lebih lanjut oleh model. Beberapa arsitektur yang digunakan dalam klasifikasi citra ikan dari penelitian ini adalah: *InceptionV3*, *ResNet50* dan *Xception*. Setiap metode ini merupakan model yang sama yaitu *Sequential*. Berikut adalah hasil dari masing-masing metode yang ditunjukkan oleh Gambar 10, Gambar 11, dan Gambar 12.



Gambar 10. Hasil Evaluasi dengan Arsitektur *InceptionV3*



Gambar 11. Hasil Evaluasi dengan Arsitektur *ResNet50*



Gambar 12. Hasil Evaluasi dengan Arsitektur *Xception*

Dari grafik yang ditampilkan, dapat disimpulkan bahwa arsitektur *InceptionV3* dan *Xception* memberikan performa yang tidak jauh berbeda namun cukup memuaskan. Namun, untuk *ResNet50* grafiknya menunjukkan akurasi yang kurang baik yaitu hanya di bagian tengah saja dan akurasinya tidak berada pada level terbaik. Jika dilihat dari hasil secara angka, nilai terbaik tetap didapat oleh *InceptionV3*. Seluruh hasil yang sudah diperoleh tersebut ditampilkan pada Tabel 1.

Tabel 1. Hasil Akhir Rata-rata Data Citra Ikan

	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>	<i>Accuracy</i>
<i>Decision Tree Classifier</i>	0.15	0.16	0.15	0.16
<i>Logistic Regression</i>	0.14	0.21	0.17	0.20
<i>Neural Network</i>	0.29	0.17	0.16	0.19
<i>Bagging Classifier</i>	0.16	0.17	0.16	0.17
<i>Gradient Boosting</i>	0.24	0.23	0.23	0.23
<i>Inception V3</i>	0.83	0.72	0.72	0.72
<i>ResNet50</i>	0.24	0.26	0.19	0.26
<i>Xception</i>	0.24	0.26	0.19	0.26

Jika dilihat dari hasil yang diterima, banyak dari hasil akhir tidak mencapai nilai standar, yang berarti tidak semua metode berhasil memberi performa yang baik untuk data citra yang diteliti. Hal ini dikarenakan mungkin sedikitnya data yang ada dan hanya terdapat 300 gambar saja per masing-masing kelas. Mungkin juga karena kualitas dari gambar yang tidak memadai, sehingga memberi hasil yang tidak memuaskan.

5. KESIMPULAN

Kesimpulan yang didapatkan dari penelitian ini menunjukkan betapa pentingnya memiliki jumlah data yang banyak dan kualitas data. Apabila dilihat dari tabel yang ada, dapat diketahui bahwa nilai dari *precision*, *recall*, dan *F1-score* dari setiap kelompok data citra yang dioleh dengan delapan metode yang berbeda memberikan nilai yang sangat rendah. Akan tetapi karena nilai dari yang lainnya merupakan nilai yang buruk, maka dapat disimpulkan bahwa data yang diteliti belum memberikan hasil yang baik, sehingga dibutuhkan data yang lebih banyak dan pra-pemrosesan yang perlu diulang.

Disarankan juga untuk penelitian selanjutnya untuk menggunakan metode yang lebih baik dalam mengolah data dalam jumlah sedikit namun mampu menghasilkan klasifikasi citra yang baik. Metode-metode yang dimaksudkan adalah seperti LSTM atau GRU menjadi

saran ke depannya. Selain itu, diperlukan tahapan seperti *transfer learning* atau *fine-tuning* agar dalam pengolahan citra data lebih mudah diproses.

DAFTAR PUSAKA

- [1] R. S. J. H. J. D. L. H. Q. G. Xingcheng Luo, "A Deep Convolution Neural Network Model for Vehicle," p. 6, 2017.
- [2] S. R. B. N. K. S. T. J. Md Tohidul Islam, "Food Image Classification with Convolutional Neural," *ICIIBMS 2018, Track 2: Artificial Intelligent, Robotics, and Human-Computer Interaction, Bangkok, Thailand*, p. 6, 2018.
- [3] X. Z. Y. Y. Z. W. G. W. Tianxiao Zhang, "Efficient Golf Ball Detection and Tracking Based on Convolutional Neural Networks and Kalman Filter," p. 10, 2021.
- [4] T. A. L. B. S. S. M. O. H. B. P. F. F. B. E. Devvi Sarwinda, "Automatic Multi-class Classification of Indonesian Traditional Food using Convolutional Neural Networks," *2020 3rd International Conference on Computer and Informatics Engineering (IC2IE)*, p. 5, 2020.
- [5] A. Y. W. d. R. S. I Wayan Suartika E. P, "Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101," *JURNAL TEKNIK ITS Vol. 5, No. 1, (2016) ISSN: 2337-3539 (2301-9271 Print)*, p. 5, 2016.
- [6] A. A. S. H. Mira Musrini B, "Implementasi Algoritma GLCM Dan MED pada Aplikasi Pendeteksi Kolesterol Melalui Iris Mata," *MIND Journal*, vol. 2, p. 19, 2017.
- [7] H. Erlyna Nour Arrofiqoh, "IMPLEMENTASI METODE CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI TANAMAN PADA CITRA RESOLUSI TINGGI," p. 8, 2018.