

KLASIFIKASI KELUARGA TUMBUHAN TEMU-TEMUAN DENGAN CONVOLUTIONAL NEURAL NETWORK DAN GLCM

Kristian Davidson Runtu

Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Tarumanagara,
Jln. Letjen S. Parman No. 1, Jakarta, 11440, Indonesia
E-mail: kristian.535180135@stu.untar.ac.id

ABSTRAK

Dalam keluarga tumbuhan temu-temuan terdapat berbagai banyak jenis yang masing-masing mempunyai berbagai keuntungan jika dikonsumsi atau digunakan. Namun seringkali susah membedakan tumbuhan-tumbuhan ini karena bentuknya yang mirip. Dengan penelitian ini yang menggunakan sebuah algoritma *Convolutional Neural Network* (CNN) untuk membuat program klasifikasi antara citra tumbuhan jahe yang akan membantu membedakan jenis-jenis jahe. Penelitian ini juga membandingkan hasil dari GLCM. Data yang digunakan adalah citra temu-temuan yang bertotal 350 gambar yang dibagi menjadi 6 kelas. Hasil yang diperoleh dari penelitian ini didapatkan tingkat akurasi tertinggi sebesar 51% pada model InceptionV3 di model lainnya didapatkan 21% untuk Decision Tree, 24% untuk Logistic Regression, 23% untuk Bagging Classifier, 21% untuk Gradient Boosting, 31% untuk ResNet50, dan 31% untuk Xception.

Kata kunci: Klasifikasi, Temu-termuan, CNN, GLCM, Python.

ABSTRACT

There are many types of plants in the ginger family, each of which has various benefits when consumed or used. However, it is often difficult to distinguish between these plants because of their similar appearance. This study uses a Convolutional Neural Network (CNN) algorithm to create a classification program for ginger images that will help distinguish between different types of ginger. This research also compares the results of GLCM. The data used consists of 350 images of ginger plants, divided into 6 classes. The results obtained from this research show that the highest accuracy level of 51% was achieved by the InceptionV3 model, while the other models achieved 21% for Decision Tree, 24% for Logistic Regression, 23% for Bagging Classifier, 21% for Gradient Boosting, 31% for ResNet50, and 31% for Xception.

Keywords: Classification, Herbal Medicine, CNN, GLCM, Python.

1. PENDAHULUAN

Latar Belakang

Dalam dunia masakan, pengobatan bahkan kecantikan pasti seringkali bertemu dengan temu-temuan sebagai salah satu bahan untuk membuat sesuatu. Temu-temuan atau *Zingiberaceae* seringkali ditemukan di daerah tropis salah satunya di Indonesia. Meskipun seringkali dilihat, masih banyak orang yang kesulitan dalam membedakan diantara tumbuhan-tumbuhan ini karena bentuknya yang sangat mirip. Maka dengan itu, dibutuhkan sebuah sistem untuk membantu membedakan bentuk diantara temu-temuan. Sistem menggunakan algoritma CNN untuk melakukan klasifikasi antara citra temu-temuan yang didapatkan.

Tujuan Sistem

Sistem yang dibuat dalam penelitian ini bertujuan untuk mengklasifikasi jenis temu-temuan dengan menggunakan CNN.

Manfaat Sistem

Sistem dibuat untuk membantu orang-orang yang seringkali kesulitan dalam membedakan antara jenis temu-temuan sehingga tidak salah memilih.

Penelitian Relevan

Penelitian ini didukung dengan beberapa literature / penelitian relevan yang membahas mengenai permasalahan yang di angkat. Berdasarkan hasil eksplorasi, berikut telah ditemukan beberapa penelitian relevan tersebut.

Penelitian pertama “Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101” oleh Soelaiman R., Suartika E.P., dan Wijaya A. (2016) [1]. Penelitian ini bertujuan untuk membuat perangkat lunak yang dapat membuat klasifikasi data Caltech 101 pada suatu citra / gambar umum dengan menggunakan metode CNN. Melalui pengujian, model CNN 3-layer (Convolution, Subsampling, Fully Connected) yang telah dibuat peneliti menghasilkan tingkat akurasi yang bervariasi untuk setiap kategori pada data tersebut, sekitar 20% - 50%.

Penelitian kedua “Image Classification with Deep Learning and Comparison between Different Convolutional Neural Network Structures using Tensorflow and Keras” oleh Chauhan K. dan Ram S. (2018) [2]. Penelitian ini bertujuan untuk membuat klasifikasi dua kategori hewan (kucing dan anjing) dari suatu citra menggunakan model CNN berbasis library keras dan tensorflow dalam bahasa pemrograman Python. Pengujian dilakukan dengan membandingkan antara beberapa model CNN dengan kombinasi fungsi aktivasi dan classifier yang berbeda. Ditemukan bahwa model dengan fungsi aktivasi Relu dan classifier Sigmoid memberikan akurasi tertinggi, yaitu sebesar 90,54%

Penelitian ketiga “Food Image Classification with Convolutional Neural Network” oleh Islam Md.et al. (2018) [3]. Penelitian ini bertujuan untuk membuat klasifikasi makanan berdasarkan suatu citra dengan CNN. Model CNN yang telah dibuat dengan 5-layer (Convolutional, Max Pooling, Dropout, Fully Connected, Softmax) dapat membuat klasifikasi 11 jenis kategori makanan (dari produk roti hingga sayur-sayuran). Pengujian model dilakukan dengan 100 epoch, 0.01 Adam learning rate dan menghasilkan akurasi sebesar 74,70%.

Penelitian keempat “Indonesian Food Image Recognition Using Convolutional Neural Network” oleh Giovany S. et al. (2019) [4]. Penelitian ini bertujuan untuk Michael Jose Dharmali, et al. Sistem Klasifikasi Kerapihan Kamar Hotel membuat sistem klasifikasi yang dapat memisahkan antara 5 jenis makanan populer di Indonesia (bakso, ayam bakar, sate, gado-gado dan rendang) menggunakan model CNN dengan arsitektur standard dan model Inception-V4 dengan fungsi optimisasi Adam. Hasil pengujian memberikan tingkat akurasi sebesar 95,2%

Penelitian kelima “DeepFood: Automatic Multi-Class Classification of Food Ingredients Using Deep Learning” oleh Pan L. et al. 2017 [5]. Penelitian DeepFood dievaluasi pada kumpulan data multi-kelas yang mencakup 41 kelas bahan makanan dan 100 gambar untuk setiap kelas. Hasil eksperimen menggambarkan efektivitas kerangka DeepFood untuk klasifikasi multi-kelas bahan makanan. Model yang mengintegrasikan set fitur mendalam ResNet, pemilihan fitur Information Gain (IG), dan pengklasifikasi SMO ini telah menunjukkan keunggulannya dalam pengenalan bahan makanan dibandingkan dengan beberapa pekerjaan yang ada di area ini.

2. METODE PENELITIAN

2.1 Data

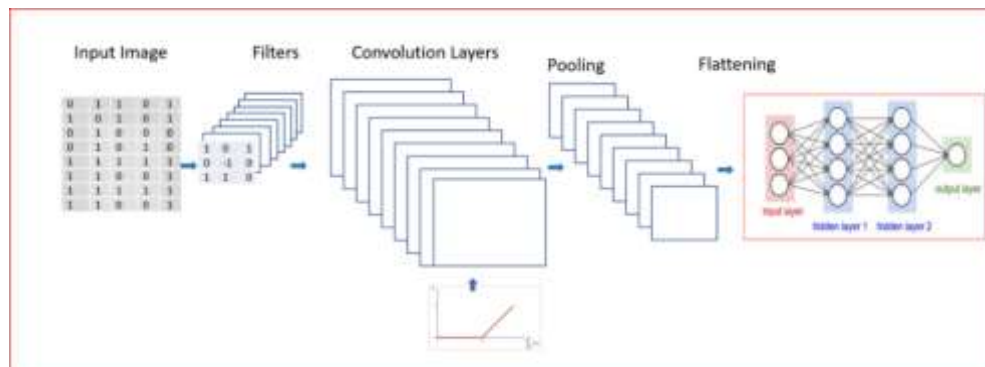
Dataset yang digunakan diperoleh dengan cara *web scraping* menggunakan ekstensi *Google Chrome Fatkun Batch Download*. Dengan pergi ke website *Google Image* kemudian masukan kata kunci misalnya “jahe” maka ekstensi tersebut akan otomatis menarik semua gambar yang ditemukan kemudian diunduh. Dataset yang dikumpulkan ada 6 kelas yang tiap kelasnya terdapat kurang lebih 50 gambar temu-temuan.

Table 1. Dataset

Temukunci	
Temuireng	
Jahe	
Kencur	
Kunyit	
Lengkuas	

2.2 Algoritma

CNN atau *convolutional neural network* adalah jaringan saraf yang digunakan untuk memproses data dengan format input seperti matriks 2D seperti gambar. CNN biasanya digunakan untuk pengenalan citra dan klasifikasi citra. Gambar tersebut dikenali oleh CNN sebagai matriks 2D. Arsitektur jaringan saraf convolutional adalah sebagai berikut:



Gambar 5 Convolutional Neural Network

2.2.1 Convolution

Pada lapisan ini, ekstraksi fitur dilakukan dari input dan peta fitur dihasilkan. Ekstraksi fitur menggunakan kernel (saluran), yaitu matriks yang digunakan untuk melakukan operasi produk titik pada input. Perhitungan dot product dilakukan dari kiri atas ke kanan bawah. Kernel bekerja dari kiri ke kanan melalui serangkaian pergeseran piksel (langkah). Jika Anda datang ke kanan, coba lagi dari kolom kiri ke bawah. Setelah lapisan konvolusi, fungsi aktivasi digunakan untuk menentukan apakah neuron yang dihasilkan aktif. Unit linier penyearah (ReLU) yang umum digunakan. ReLU mengembalikan nilai 0 jika neuron negatif dan menghasilkan nilainya sendiri jika positif.

2.2.2 Pooling

Lapisan ini membantu mengurangi ukuran peta fitur. Ada dua jenis pooling yang umum digunakan: average pooling dan maximum pooling. Mean pooling mengambil nilai rata-rata dari matriks, dan maximum pooling mengambil nilai maksimum.

2.2.3 Fully Connected

Setelah ekstraksi fitur, data dibagi menjadi berbagai kelas menggunakan lapisan yang terhubung penuh. Lapisan yang terhubung penuh adalah lapisan di mana semua input dari satu lapisan terhubung ke semua unit aktivasi di lapisan berikutnya.

2.3 Rancangan eksperimen

Berikut adalah flowchart pelatihan dan pengenalan menggunakan CNN.



Gambar 1. Flowchart pelatihan dan pengenalan

2.4 Metode evaluasi

Dalam penelitian hasil dievaluasi menggunakan nilai recall, precision, F1-score dan dihitung nilai akurasinya.

$$\begin{aligned} \text{Nilai True Positive (TP)} &= \frac{TP}{\sum \text{ semua kelas positif}} & (1) \\ \text{Nilai True Negative (TN)} &= \frac{TN}{\sum \text{ semua kelas negatif}} & (2) \\ \text{Nilai Presisi} &= \frac{TP}{TP + FP} & (3) \\ \text{Nilai Akurasi} &= \frac{TP + TN}{P + N} & (4) \\ \text{Nilai Recall} &= \frac{TP}{P} & (5) \\ \text{Nilai F - measure} &= \frac{2}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}} & (6) \end{aligned}$$

3. HASIL DAN PEMBAHASAN

3.1 Hasil Penelitian dengan GLCM

Setelah dilakukan beberapa percobaan pada GLCM, didapatkan hasil untuk *precision*, *recall* dan F1-Score dibawah ini.

3.1.1 Decision Tree Classifier

	precision	recall	f1-score	support
1Temukunci	0.22	0.17	0.20	23
2Temuirenh	0.27	0.30	0.29	20
3abe	0.25	0.21	0.23	24
Kencur	0.04	0.06	0.05	17
Kunyit	0.26	0.25	0.26	24
Lengkuas	0.27	0.26	0.27	23
accuracy			0.21	131
macro avg	0.22	0.21	0.21	131
weighted avg	0.23	0.21	0.22	131

Gambar 2 Hasil Evaluasi Decision Tree Classifier

3.1.2 Bagging Classifier

	precision	recall	f1-score	support
1Temukunci	0.20	0.04	0.07	23
2Temuirenh	0.18	0.35	0.24	20
Jahe	0.43	0.25	0.32	24
Kencur	0.24	0.47	0.31	17
Kunyit	0.28	0.33	0.30	24
Lengkuas	0.18	0.09	0.12	23
accuracy			0.24	131
macro avg	0.25	0.26	0.23	131
weighted avg	0.25	0.24	0.22	131

Gambar 3. Hasil Evaluasi Bagging Classifier

3.1.3 Gradient Boosting Classifier

	precision	recall	f1-score	support
1Temukunci	0.29	0.17	0.22	23
2Temuirenh	0.16	0.25	0.20	20
Jahe	0.26	0.25	0.26	24
Kencur	0.16	0.18	0.17	17
Kunyit	0.21	0.21	0.21	24
Lengkuas	0.20	0.17	0.19	23
accuracy			0.21	131
macro avg	0.21	0.21	0.20	131
weighted avg	0.22	0.21	0.21	131

Gambar 4. Hasil Evaluasi Gradient Boosting Classifier

3.2 Hasil Penelitian CNN

3.2.1 InceptionV3

Hasil dengan metode InceptionV3 menunjukkan akurasi untuk data training dan validation yang seimbang, dan nilai loss untuk training dan validation menunjukkan nilai rendah. Artinya tidak banyak data yang hilang Ketika dilakukan pengtesan sebanyak 50 kali.

	precision	recall	f1-score	support
0	0.32	1.00	0.48	18
1	0.71	0.26	0.38	19
2	0.50	0.10	0.17	20
3	1.00	0.05	0.09	22
4	0.59	0.91	0.71	22
5	0.65	0.62	0.64	45
accuracy			0.51	146
macro avg	0.63	0.49	0.41	146
weighted avg	0.64	0.51	0.45	146

Gambar 5. Hasil Evaluasi Akurasi InceptionV3

3.2.2 ResNet50

Hasil dengan metode ResNet50 menunjukkan akurasi untuk data training dan validation yang sangat rendah dan tidak stabil dan nilai loss untuk training dan validation menunjukkan nilai rendah. Artinya metode ini tidak cocok untuk digunakan dalam pengklasifikasian terhadap gambar karena banyak data yang kabur Ketika dilakukan pengtesan sebanyak 50 kali.

	precision	recall	f1-score	support
0	0.00	0.00	0.00	18
1	0.00	0.00	0.00	19
2	0.00	0.00	0.00	20
3	0.00	0.00	0.00	22
4	0.00	0.00	0.00	22
5	0.31	1.00	0.47	45
accuracy			0.31	146
macro avg	0.05	0.17	0.08	146
weighted avg	0.09	0.31	0.15	146

Gambar 6. Hasil Evaluasi Akurasi ResNet50

3.2.3 Xception

Hasil dengan metode Xception menunjukkan akurasi untuk data *training* dan *validation* yang seimbang, dan nilai loss untuk training dan validation menunjukkan nilai rendah. Artinya tidak banyak data yang hilang Ketika dilakukan pengtesan sebanyak 50 kali.

	precision	recall	f1-score	support
0	0.00	0.00	0.00	18
1	0.00	0.00	0.00	19
2	0.00	0.00	0.00	20
3	0.00	0.00	0.00	22
4	0.00	0.00	0.00	22
5	0.31	1.00	0.47	45
accuracy			0.31	146
macro-avg	0.05	0.17	0.08	146
weighted avg	0.09	0.31	0.15	146

Gambar 7. Perhitungan Hasil Evaluasi Akurasi Xception

Tabel 2. Perbandingan

	Precision (%)	Recall (%)	F1-Score (%)	Accuracy (%)
Decision Tree	0.23	0.2	0.22	0.21
Logistic Regression	0.25	0.26	0.23	0.24
Bagging Classifier	0.24	0.24	0.22	0.23
Gradient Boosting	0.21	0.21	0.21	0.21
InceptionV3	0.64	0.51	0.45	0.51
ResNet50	0.09	0.31	0.15	0.31
Xception	0.09	0.31	0.15	0.31

Hasil Tabel diatas menunjukkan bahwa dengan penggunaan model InceptionV3 menghasilkan nilai tertinggi dibandingkan model yang lain. Nilai yang diperoleh InceptionV3 yaitu 0.64 untuk precision, 0.51 untuk recall, 0.45 untuk f1-score, dan akurasi 0.51.

4. KESIMPULAN

Dari Penelitian perhitungan yang telah dilakukan, maka dapat disimpulkan bahwa penggunaan Algoritma Deep Learning dengan metode InceptionV3 menghasilkan hasil akurasi yang maksimal 51% dibandingkan dengan Algoritma yang lain rata-rata dibawah 30%.

DAFTAR PUSTAKA

- [1] K. Chauhan and S. Ram, "Image Classification with Deep Learning and Comparison between Different," International Journal of Advance Engineering and Research, vol. 5, no. 2, pp. 533-538, 2018.
- [2] I. W. Suartika E. P, A. Y. Wijaya and R. Soelaiman, "Klasifikasi Citra Menggunakan Convolutional Neural Network (Cnn) pada Caltech 101," JURNAL TEKNIK ITS, vol. 5, no. 1, pp. A65-A69, 2016.
- [3] T. Islam, N. K. Siddique, S. Rahman and T. Jabid , "Food Image Classification with Convolutional Neural Network," in International Conference on Intelligent Informatics and BioMedical Sciences, Bangkok, 2018.
- [4] S. Giovany, A. Putra, A. S. Hariawan, L. A. Wulandhari and E. Irwansyah, "Indonesian Food Image Recognition Using Convolutional Neural Network," in CSOC 2019. Advances in Intelligent Systems and Computing, Switzerland, 2019.
- [5] L. Pan, S. Pouyanfar, H. Chen, J. Qin, and S.-C. Chen, "DeepFood: Automatic multi-class classification of food ingredients using Deep Learning," 2017 IEEE 3rd International Conference on Collaboration and Internet Computing (CIC), 2017.