

# Klasifikasi Jenis Beras Berbasis Citra Dengan Menggunakan Deep Learning

1<sup>st</sup> Chelsya Dwi Marnelius

Fakultas Teknik Elektro

Universitas Telkom

Bandung, Indonesia

[chelsyadwim@student.telkomuniversity.ac.id](mailto:chelsyadwim@student.telkomuniversity.ac.id)

2<sup>nd</sup> Koredianto Usman

Fakultas Teknik Elektro

Universitas Telkom

Bandung, Indonesia

[korediantousman@telkomuniversity.ac.id](mailto:korediantousman@telkomuniversity.ac.id)

3<sup>rd</sup> Nor Kumalasari Caecar Pratiwi

Fakultas Teknik Elektro

Universitas Telkom

Bandung, Indonesia

[caecarnkcp@telkomuniversity.ac.id](mailto:caecarnkcp@telkomuniversity.ac.id)

**Abstrak**—Beras merupakan salah satu bahan pangan sereal yang paling banyak dikonsumsi oleh masyarakat Asia, termasuk di Indonesia. Setiap beras memiliki kemiripan bentuk bulir dan warna yang relatif hampir sama, sehingga proses mengidentifikasi jenis beras secara visual dinilai masih cukup sulit, dan untuk mengurangi kemungkinan terjadinya *human error*. Oleh karena itu, pengolahan citra digital dapat digunakan dalam melakukan klasifikasi jenis beras menggunakan metode Convolutional Neural Network (CNN) dengan arsitektur Residual Network (ResNet-50).

Dataset yang digunakan dalam penelitian ini berjumlah 2500 citra yang diperoleh dari website Kaggle, berupa citra bulir beras putih yang terdiri dari 5 kelas yaitu Arborio, Basmati, Ipsala, Jasmine, dan Karacadag. Tahapan diawali dengan melakukan preprocessing yaitu resize ukuran citra dan normalisasi citra, lalu dilakukan pembagian dataset sebagai data latih dan data uji. kemudian selanjutnya dilakukan ekstraksi ciri dan klasifikasi menggunakan kombinasi hyperparameter input size, optimizer, learning rate, dan batch size untuk mendapatkan hasil model terbaik. Kemudian di tahap akhir hasil akan dianalisis dengan parameter akurasi dan loss.

Melalui penelitian ini diperoleh hasil akhir pengujian terbaik pada penggunaan parameter input size 64×64 piksel, optimizer Adam, learning rate 0,001, dan batch size 64, dengan hasil validation accuracy sebesar 98,20% dan loss 0,1109.

**Kata Kunci:** jenis beras, CNN, ResNet-50.

## I. PENDAHULUAN

Beras merupakan salah satu bahan pangan yang paling banyak dikonsumsi oleh penduduk di Benua Asia, terutama di Indonesia [1]. Mayoritas penduduk di Asia mengonsumsi beras sebagai sumber utama karbohidrat. Maka tak heran jika semakin banyak inovasi-inovasi baru yang bermunculan untuk memperkaya varietas beras, dengan tujuan untuk memberikan beragam pilihan kepada masyarakat dan untuk beradaptasi dengan perubahan selera serta preferensi konsumen di berbagai wilayah. Umumnya beras terbagi atas beras putih, beras coklat, beras merah, dan beras hitam, namun jenis beras yang paling sering dikonsumsi ialah beras putih [2]. Beras putih sendiri pun ada beragam jenisnya, dengan masing-masing beras memiliki bentuk, tekstur, aroma, dan rasa yang berbeda.

Namun dengan beragamnya jenis beras yang ada, maka salah satu kendala yang dirasakan masyarakat ialah masih cukup sulitnya mengidentifikasi jenis beras yang dilakukan

secara visual mata saja dinilai kurang cukup karena keterbatasan waktu manusia untuk melakukan mengidentifikasi beras satu-persatu, dan untuk mengurangi kemungkinan terjadinya *human error*.

Untuk itu diperlukan sebuah sistem klasifikasi gambar untuk mempermudah dalam mengidentifikasi varietas beras putih secara akurat dan cepat, sehingga dapat membantu masyarakat sebelum memilih beras yang akan dibeli dan menghindari kecurangan yang bisa terjadi seperti adanya campuran dengan varietas beras putih lainnya ataupun penggunaan bahan kimia yang membuat beras menjadi lebih putih.

Dengan perkembangan era digital saat ini, sangat memungkinkan terciptanya sebuah komputasi yang mampu mengolah informasi dari suatu citra untuk pengenalan objek secara otomatis melalui proses *image processing*. *Computer Vision* yang terkenal dapat melakukan pengenalan citra dengan baik adalah CNN. CNN tersusun atas tiga komponen layer utama, yaitu *convolutional layer*, *pooling layer*, dan *fully connected layer*. Dengan mengadopsi cara kerja dan memiliki struktur tiruan dari neuron otak manusia. CNN mampu melakukan ekstraksi fitur dari citra secara mendetail, dan dapat menyimpan serta menjadikan acuan hasil pembelajaran yang dilakukan dengan adanya parameter weight dan bias di dalam arsitekturnya [3]. Dengan konsep itulah CNN dapat menerima input citra dengan karakteristik lebih umum.

Untuk penelitian yang membahas mengenai klasifikasi jenis/varietas beras sendiri sudah cukup banyak dilakukan dengan penggunaan metode yang berbeda-beda. Dalam penelitian yang dilakukan oleh Sofia Saidah dkk (2019) [4], dilakukan identifikasi kualitas beras menggunakan metode *k-Nearest Neighbor* dan *Support Vector Machine (SVM)*. Hasil terbaik penelitian diperoleh 96,67% untuk metode K-NN jenis *Euclidean* dengan k=1 dan hasil yang sama untuk metode SVM OAO dan OAA dengan tipe kernel Polynomial dan kernel option 7 yaitu 96,67%. Penelitian lainnya oleh Gansar Suwanto (2021) [5], dilakukan identifikasi citra digital jenis beras menggunakan metode *Adaptive Neuro Fuzzy Inference System (ANFIS)* dan Sobel. Pengujian dilakukan menggunakan 140 citra dengan memanfaatkan nilai bentuk dan tekstur citra. Hasil pengujian yang didapatkan yaitu akurasi rata-rata sebesar 85,2% dan pengaruh akurasi 3% deteksi tepi sobel sebagai metode tambahan.

Metode CNN sendiri telah terbukti keakuratannya dalam

pengenalan citra, seperti dalam penelitian Murat Koklu dkk (2021) [6], yang melakukan perbandingan 3 metode *deep learning* dalam klasifikasi jenis beras yakni metode *Artificial Neural Network* (ANN), *Deep Neural Network* (DNN), dan CNN dengan penerapan *confusion matrix* untuk mengukur performa dari masing-masing metode. Akurasi tertinggi diperoleh oleh metode CNN dengan akurasi mencapai 100%, kemudian metode DNN dengan 99,95%, dan metode ANN dengan 99,87%.

Metode CNN sendiri mempunyai berbagai macam konfigurasi arsitektur, salah satunya adalah ResNet. Arsitektur ResNet telah terbukti keakuratannya dalam hal klasifikasi citra. Seperti pada penelitian Ivan Pratama Putra (2022) [7], dilakukan klasifikasi penyakit daun jagung menggunakan arsitektur ResNet-50 dengan optimizer Adam, Nadam, dan SGD. Dengan menggunakan dataset sebanyak 4225 citra, hasil akurasi tertinggi diperoleh menggunakan optimizer Adam yaitu sebesar 98,4%, lalu optimizer Nadam dan SGD masing-masing memperoleh 98,30% dan 98%.

Aplikasi metode CNN dengan arsitektur ResNet dalam melakukan klasifikasi jenis beras masih belum banyak diteliti. Oleh karena itu, pada penelitian ini akan dilakukan klasifikasi jenis beras melalui citra gambar bulir beras menggunakan arsitektur ResNet-50.

I. KAJIAN DAN TEORI

A. Beras

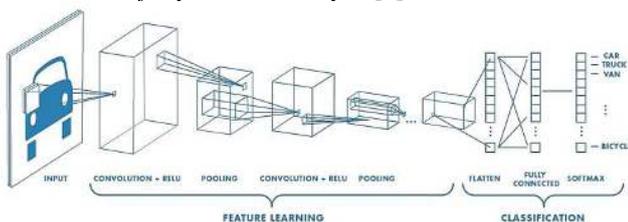
Beras merupakan bulir padi (gabah) yang telah dipisahkan dari kulit luar (*hull*), dedak (*bran*), dan bekatul (*polish*). Secara umum terdapat beberapa jenis warna pada beras, diantaranya beras putih (*milled rice*), beras merah (*red rice*), beras hitam (*black rice*), dan beras cokelat (*brown rice*). Perbedaan warna ini diatur secara genetik, karena perbedaan gen yang mengatur warna *aleurone* dan *endospermia*, serta komposisi pati pada *endospermia*. Komposisi dari pati ini menentukan tekstur beras dan warna transparan atau buram pada beras [2].

B. Citra Digital

Secara umum, citra dapat diartikan sebagai suatu gambaran atau kemiripan dari suatu objek. Citra analog tidak dapat diproses secara langsung oleh computer, sehingga harus dikonversi terlebih dahulu menjadi citra digital. Citra digital adalah sebuah larik (*array*) yang mempunyai nilai-nilai *real* ataupun kompleks yang dipresentasikan dengan deretan bit tertentu [8].

C. Convolutional Neural Network (CNN)

*Convolutional Neural Network* merupakan salah satu metode algoritma *Deep Learning* yang biasa digunakan dalam menganalisis sebuah citra. CNN dapat menerima *input* berupa citra, kemudian menentukan aspek/objek yang ada pada citra tersebut untuk selanjutnya dipelajari dan dikenali. Arsitektur CNN setidaknya dibangun oleh 4 komponen, yaitu *convolutional layer*, *pooling layer*, fungsi aktivasi, dan *fully connected layer* [9].

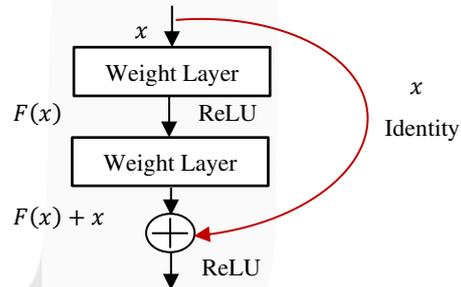


Gambar 1. Arsitektur CNN [10]

Berdasarkan Gambar 1, terlihat bahwa struktur CNN terdiri dari *input layer* (masukan), *feature extraction layer* (ekstraksi fitur), *classification layer* (klasifikasi), dan *output layer* (keluaran). Pada tahap ekstraksi fitur terdapat beberapa *hidden layer*, yaitu *convolutional layer* (lapisan konvolusi), *Rectified Liner Unit (ReLU) Activation Layer* (fungsi aktivasi ReLU), dan *pooling layer*. Pada tahap klasifikasi terdiri atas *flatten layer*, *fully connected layer*, dan *softmax* yang menghasilkan keluaran berupa hasil klasifikasi.

D. Residual Network (ResNet)

Residual Network merupakan salah satu model arsitektur dari CNN yang diperkenalkan oleh Kaiming He et al [11] dan telah berhasil memenangkan kompetisi *ImageNet Large Scale Visual Recognition Challenge (ILSVRC)* pada tahun 2015. Tiap lapis arsitektur ResNet terdapat blok yang bernama *residual block*. Blok ini menambahkan suatu jalan pintas yang akan melewati proses *training* untuk satu layer atau lebih [12], yang disebut *skip connection*. *Skip connection* merupakan proses melewati sejumlah layer untuk menghindari kehilangan gradien saat pemetaan identitas data citra dengan *layer* yang cukup dalam dilakukan [13]. *Skip connection* diimplementasikan pada dua atau tiga *layer* yang mengandung aktivasi ReLU dan *batch normalization* diantara lapisannya. Lapisan *skip connection* dapat diilustrasikan seperti Gambar 2.



Gambar 2. Residual Block [11]

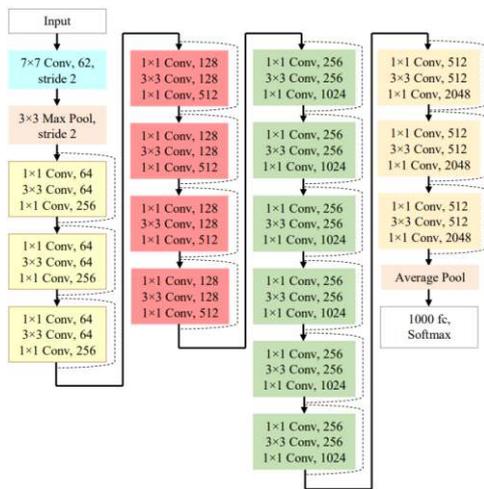
ResNet memiliki beberapa model arsitektur yang dibedakan berdasarkan jumlah *layer* yang digunakan. Setiap model arsitektur tersebut dapat dilihat pada Gambar 3.

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112x112	7x7, 64, stride 2				
		3x3 max pool, stride 2				
conv2_x	56x56	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28x28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4_x	14x14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5_x	7x7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1x1	average pool, 1000-d fc, softmax				
FLOPs		$1.8 \times 10^9$	$3.6 \times 10^9$	$3.8 \times 10^9$	$7.6 \times 10^9$	$11.3 \times 10^9$

GAMBAR 3.

Arsitektur ResNet [11].

Model arsitektur ResNet yang digunakan pada penelitian ini adalah ResNet-50, yaitu arsitektur yang memiliki 50 *layer*. Dalam arsitekturnya, ResNet-50 menggunakan lima lapisan *convolution* seperti yang terlihat pada Gambar 4.

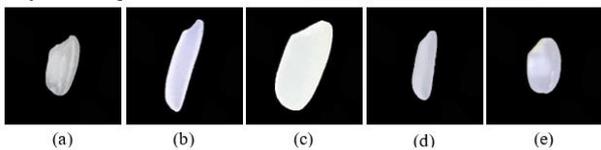


Gambar 4. Arsitektur ResNet-50 [14].

II. MODEL DAN DESAIN SISTEM

A. Dataset

Dataset yang digunakan pada penelitian ini merupakan *Rice Image Dataset* yang diperoleh dari *Kaggle Dataset* [15]. *Dataset* ini berupa citra bulir beras yang terdiri dari 5 kelas, yaitu beras Arborio, beras Basmati, beras Ipsala, beras Jasmine, dan beras Karacadag. Masing-masing kelas pada *dataset* ini terdiri dari 15000 citra, sehingga jumlah data citra keseluruhannya adalah sebanyak 75000 citra. Namun dikarenakan arsitektur ResNet-50 yang cukup kompleks dan untuk efisien waktu, maka jumlah *dataset* yang digunakan hanya sebanyak 2500 citra, dengan masing-masing kelas terdiri atas 500 citra bulir beras. Citra pada *dataset* ini berupa citra RGB berukuran 250x250 piksel dengan format .jpg. Pada Gambar 5 merupakan contoh *dataset* jenis beras dari setiap kelasnya.



Gambar 5. *Rice Image Dataset* (a) Arborio, (b) Basmati, (c) Ipsala, (d) Jasmine, (e) Karacadag [15].

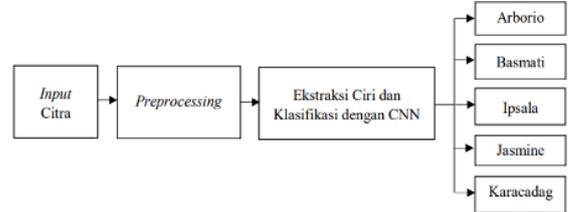
*Dataset* yang digunakan pada penelitian ini akan dibagi menjadi 2 bagian, yaitu data latih (*train*) dan data uji (*validation*). Data latih merupakan data citra yang digunakan untuk melatih pola citra agar mampu dipelajari oleh sistem, sedangkan data uji merupakan data citra yang digunakan untuk menguji performansi dari sebuah sistem. Adapun pembagian data pada penelitian ini, yakni 80% untuk data latih dan 20% untuk data uji. Sistematika pembagian data ini dapat dilihat pada Tabel 1.

Tabel 1. Sistematika Pembagian *Dataset*.

Data Latih 80%	Data Uji 20%
2000	500

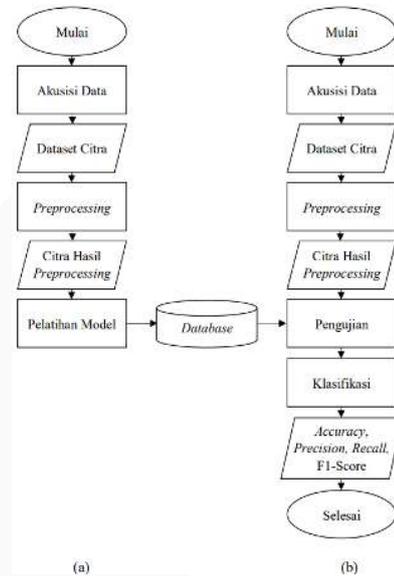
B. Desain Sistem

Pada penelitian ini dirancang sebuah sistem yang dapat melakukan klasifikasi terhadap jenis beras menggunakan metode CNN dengan arsitektur ResNet-50. Sistem akan melakukan klasifikasi inputan data berbentuk citra dari 5 jenis beras yang berbeda, yaitu beras Arborio, beras Basmati, beras Ipsala, beras Jasmine, dan beras Karacadag. Gambar 6 menunjukkan blok diagram sistem secara umum yang dilakukan pada penelitian ini.



Gambar 6. Blok Diagram Sistem.

Sistem dirancang melalui beberapa proses tahapan, yaitu *input* citra, *preprocessing*, ekstraksi ciri dan pelatihan, dan klasifikasi dengan metode CNN ke dalam 5 kelas. Adapun flowchart proses sistem yang terdiri dari tahap latih dan tahap uji berdasarkan blok diagram sistem yang terlihat pada Gambar 7.



Gambar 7.

Flowchart Sistem (a) Tahap Latih (b) Tahap Uji.

Pada Gambar 7 menunjukkan tahapan umum proses klasifikasi jenis beras pada penelitian ini. Akuisisi data merupakan proses pengumpulan *dataset* yang bersumber dari *Kaggle Dataset* yang terdiri dari 5 kelas yaitu beras Arborio, beras Basmati, beras Ipsala, beras Jasmine, dan beras Karacadag dengan jumlah total citra *dataset* yang digunakan adalah 2500 citra RGB. Kemudian pada tahap *preprocessing* dilakukan proses *resize* dan normalisasi pada data citra. Pada penelitian ini, citra akan di *resize* dengan empat ukuran yang berbeda, yaitu 64x64 piksel, 128x128 piksel, 224x224 piksel dan 250x250 piksel. Setelah proses *resize* dan normalisasi citra selesai dilakukan, maka didapatkan hasil data citra yang siap diproses ke tahap pengolahan citra selanjutnya. Selanjutnya data citra yang telah melalui tahap *preprocessing*, masuk ke proses pelatihan model dengan arsitektur ResNet-50.

C. Parameter Performansi Sistem

Performansi sistem dapat dilakukan setelah tahapan pada *training* data dan *testing* data telah selesai dilakukan. Parameter performansi ini dilakukan untuk menentukan seberapa baik kualitas dari sistem yang telah dirancang. Performansi kerja sistem dapat diukur dari empat parameter, yaitu *accuracy*, *precision*, *recall*, dan *F1-Score* [16]. Dalam mengukur keempat parameter ini dapat dibantu dengan menggunakan *confusion matrix* seperti yang terlihat pada Tabel 2. Terdapat empat istilah yang merupakan representasi hasil proses klasifikasi dari *confusion matrix*, yaitu:

1. *True Positive* (TP) adalah data yang bernilai positif dan dideteksi benar oleh system sebagai data positif (klasifikasi yang benar dari kelas positif).
2. *False Negative* (FN) adalah data yang bernilai positif tetapi dideteksi oleh sistem sebagai data negative.
3. *False Positive* (FP) adalah data yang bernilai negatif tetapi dideteksi oleh sistem sebagai data positif.
4. *True Negative* (TN) adalah data yang bernilai negatif dan dideteksi benar oleh sistem sebagai data negatif (klasifikasi yang benar dari kelas negatif).

Tabel 2. *Confusion Matrix*

Actual Class	Predicted Class	
	True	False
True	TP	FN
False	FP	TN

Adapun persamaan dari *accuracy*, *precision*, *recall*, dan *F1-Score* yang dapat dilihat melalui persamaan (3.1), persamaan (3.2), persamaan (3.3), dan persamaan (3.4).

$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN} \times 100\% \tag{3.1}$$

$$Precision = \frac{TP}{TP+FP} \times 100\% \tag{3.2}$$

$$Recall = \frac{TP}{TP+FN} \times 100\% \tag{3.3}$$

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \times 100\% \tag{3.4}$$

III. HASIL DAN PEMBAHASAN

A. Skenario Pengujian I : *Input Size*

Pada skenario pertama, dilakukan pengujian terhadap pemilihan ukuran citra masukan yang akan digunakan. Terdapat empat *input size* yang akan dibandingkan hasil pengujianya, yaitu 64×64 piksel, 128×128 piksel, 224×224 piksel, dan 250×250 piksel.

Kemudian untuk *hyperparameter* lainnya menggunakan *hyperparameter* yang telah ditentukan sebelumnya, yakni *Optimizer* menggunakan Adam, *learning rate* bernilai 0,001, dan *batch size* yang dipakai

adalah 32.

Hasil pengujian ini dapat dilihat melalui nilai akurasi dan *loss* dari data *training* dan *validation* pada kinerja masing-masing pengujian.

Tabel 3. Hasil Pengujian *Input Size*

Input Size	Train Accuracy	Train Loss	Validation Accuracy	Validation Loss
64×64	98,35%	0,0783	98,00%	0,1087
128×128	98,10%	0,0611	97,00%	0,0899
224×224	98,60%	0,0458	97,40%	0,0794
250×250	96,90%	0,0939	96,20%	0,1456

Berdasarkan Tabel 3, diperoleh hasil akurasi terbaik pada pengujian ukuran citra 64×64 piksel dengan nilai *validation accuracy* mencapai 98,00% dengan *loss* 0,1087. Melalui hasil ini, maka penggunaan ukuran citra 64×64 piksel dinilai sudah baik dalam mengenali lima kelas jenis beras karena mencapai akurasi tertinggi, fitur yang ada pada citra pun menjadi lebih baik dan tidak membuat beban komputasi menjadi terlalu berat.

B. Skenario Pengujian II : *Optimizer*

Pada skenario kedua, dilakukan pengujian terhadap ketiga *optimizer* yang akan digunakan yaitu Adam, RMSprop, dan SGD. Untuk *hyperparameter* lainnya menggunakan *input size* 64×64 piksel yang diperoleh dari hasil terbaik pengujian sebelumnya, *learning rate* 0,001, dan *batch size* 32.

Sama seperti pengujian sebelumnya, hasil pengujian ini juga dilihat melalui nilai akurasi dan *loss* dari data *training* dan *validation* pada kinerja masing-masing pengujian.

Tabel 4. Hasil Pengujian *Optimizer*

Optimizer	Train Accuracy	Train Loss	Validation Accuracy	Validation Loss
Adam	98,35%	0,0783	98,00%	0,1087
RMSprop	95,50%	0,1400	96,00%	0,1549
SGD	88,50%	0,4600	87,00%	0,4891

Berdasarkan Tabel 4, diperoleh akurasi terbaik pada pengujian *optimizer* Adam dengan nilai *validation accuracy* mencapai 98,00% dan *loss* 0,1087. Melalui hasil tersebut, maka *optimizer* yang akan digunakan pada pengujian selanjutnya adalah Adam.

C. Skenario Pengujian III : *Learning Rate*

Pada scenario ketiga, dilakukan pengujian terhadap *learning rate* dengan data citra asli. Adapun nilai *learning rate* yang digunakan yaitu 0,1, 0,01, 0,001, dan 0,0001. Untuk *hyperparameter* lainnya menggunakan *input size* 64×64

piksel dan *optimizer* Adam yang diperoleh dari hasil pengujian terbaik sebelumnya, serta *batch size* 32. Sama seperti pengujian sebelumnya, pengujian ini juga dilihat dari hasil nilai akurasi dan *loss* dari data *training* dan *validation* yang dihasilkan dari masing-masing pengujian.

Tabel 5.  
Hasil Pengujian *Learning Rate*

Learning Rate	Train Accuracy	Train Loss	Validation Accuracy	Validation Loss
0,1	93,95%	0,3873	94,40%	0,4181
0,01	93,55%	0,1540	93,60%	0,2016
0,001	98,35%	0,0783	98,00%	0,1087
0,0001	95,95%	0,2845	96,00%	0,3057

Berdasarkan Tabel 5, diperoleh akurasi terbaik pada pengujian *learning rate* 0,001 dengan nilai *validation accuracy* mencapai 98,00% dan *loss* 0,1087. Berdasarkan hasil tersebut, maka untuk pengujian selanjutnya akan menggunakan nilai *learning rate* 0,001. Pada pengujian ini memperlihatkan bahwa nilai *learning rate* yang terlalu tinggi akan menyebabkan *loss* yang cukup besar.

D. Skenario Pengujian IV : *Batch Size*

Pada skenario keempat, dilakukan pengujian terhadap *batch size* pada data citra asli. Nilai *batch size* yang digunakan adalah 8, 16, 32, dan 64. Sedangkan untuk *hyperparameter* lainnya menggunakan *input size* 64x64 piksel, *optimizer* Adam, dan *learning rate* 0,001 yang merupakan hasil terbaik dari pengujian sebelumnya. Sama seperti pengujian sebelumnya, pengujian ini dilihat dari hasil akurasi dan *loss* dari data *training* dan *validation* dari masing-masing pengujian.

Tabel 6.  
Hasil Pengujian *Batch Size*

Batch Size	Train Accuracy	Train Loss	Validation Accuracy	Validation Loss
8	98,00%	0,0675	97,40%	0,0944
16	98,50%	0,0633	97,60%	0,0962
32	98,35%	0,0783	98,00%	0,1087
64	98,50%	0,0864	98,20%	0,1109

Berdasarkan hasil pada Tabel 6, diperoleh akurasi terbaik pada pengujian *batch size* 64 dengan nilai *validation accuracy* 98,20% dan *loss* 0,1109. Pada pengujian ini *batch size* memberikan dampak peningkatan akurasi dari skenario sebelumnya.

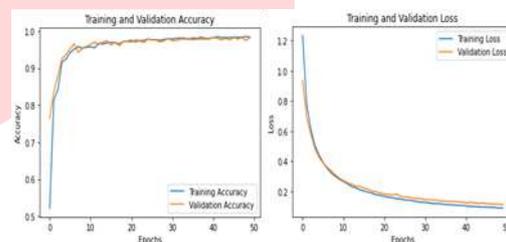
E. Hasil Pengujian Terbaik

Setelah keempat skenario pengujian *input size*, *optimizer*, *learning rate*, dan *batch size* dilakukan, maka hal selanjutnya adalah melakukan pembahasan mengenai kesimpulan hasil dan analisa dari hasil pengujian terbaik.

Tabel 7.  
Hasil Pengujian Terbaik

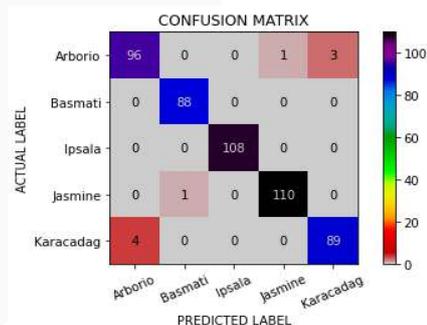
Hyperparameter	Train Accuracy	Train Loss	Validation Accuracy	Validation Loss
Input Size : 64x64 piksel	98,50%	0,0864	98,20%	0,1109
Optimizer : Adam				
Learning Rate : 0,001				
Batch Size : 64				

Tabel 7 menunjukkan hasil pengujian terbaik dari setiap skenario yang telah dilakukan. Berdasarkan tabel tersebut, diperoleh bahwa *hyperparameter* terbaik dalam melakukan klasifikasi jenis beras pada penelitian ini adalah dengan menggunakan *input size* 64x64 piksel, *optimizer* Adam, *learning rate* 0,001, dan *batch size* 64. Kemudian jika ditinjau dari grafik *accuracy* dan *loss* pada Gambar 8, penggunaan kombinasi *hyperparameter* ini memiliki hasil grafik yang baik tanpa adanya *overfitting* ataupun *underfitting*.



Gambar 8.  
Grafik Pengujian *Hyperparameter* terbaik.

Kemudian pada Gambar 9 memperlihatkan hasil *confusion matrix* dari model sistem dengan *hyperparameter* terbaik. Melalui *confusion matrix* tersebut dapat dilihat bahwa secara keseluruhan model sistem ini sudah dapat melakukan klasifikasi jenis beras dengan baik. Meski pada beberapa kelas masih terdapat kesalahan prediksi, seperti yang terjadi pada kelas Arborio, Basmati, Jasmine, dan Karacadag.



Gambar 9.  
*Confusion Matrix Best Model.*

Selanjutnya melakukan analisis terhadap hasil parameter performansi yang telah diperoleh, yakni berupa *accuracy*, *precision*, *recall*, *F1-Score* seperti yang terlihat pada Tabel 8.

Tabel 8.  
Hasil Parameter Performansi

Class	Precision	Recall	F1-Score
Arborio	96,00%	96,00%	96,00%

Basmati	98,88%	100%	99,44%
Ipsala	100%	100%	100%
Jasmine	99,10%	99,10%	99,10%
Karacadag	96,74%	95,70%	96,22%
<b>Accuracy</b>	98,20%		

Hasil pada tabel tersebut menunjukkan bahwa masing-masing kelas memiliki nilai *precision*, *recall*, dan *F1-Score* tersendiri. Dengan tingkat akurasi yang diperoleh secara keseluruhan mencapai 98,20%. dimana hasil antara parameter performansi dan *confusion matrix* memiliki hasil yang saling mempengaruhi. Oleh karena itu, dapat dikatakan bahwa model system ini sudah memiliki tingkat performansi yang cukup baik dalam melakukan klasifikasi jenis beras Arborio, Basmati, Ipsala, Jasmine, dan Karacadag.

#### IV. KESIMPULAN

Berdasarkan hasil simulasi dan pengujian sistem yang telah dilakukan, dapat disimpulkan bahwa klasifikasi jenis beras menggunakan CNN dengan arsitektur ResNet-50, dapat berjalan dengan optimal dalam mengklasifikasikan jenis beras Arborio, Basmati, Ipsala, Jasmine, dan Karacadag.

Hasil tersebut didapatkan berdasarkan pada empat skenario pengujian yang telah dilakukan, dimana hampir di setiap skenario memperoleh hasil akurasi diatas 90%. Hasil akurasi terbaik dengan menggunakan arsitektur ResNet-50 yang diperoleh pada penelitian ini adalah dengan menggunakan kombinasi dari *hyperparameter input size* 64×64 piksel, *optimizer* Adam, *learning rate* 0,001, dan *batch size* 64, dengan nilai *validation accuracy* 98,20% dan *loss* 0,1109.

Oleh sebab itu, model sistem pada penelitian ini telah memiliki performansi yang baik dalam melakukan klasifikasi jenis beras Arborio, Basmati, Ipsala, Jasmine, dan Karacadag menggunakan CNN dengan arsitektur ResNet-50, sehingga dapat dijadikan sebagai referensi awal untuk dilakukan pengembangan pada penelitian selanjutnya.

#### REFERENSI

- [1] B. E. Sawe, "Top 10 Rice Consuming Countries," *www.WorldAtlas.com*, 2019. <https://www.worldatlas.com/articles/top-10-rice-consuming-counties.html> (accessed Jun. 27, 2022).
- [2] D. G. Patria, Sukamto, and Sumarji, *Rice Science and Technology (Ilmu dan Teknologi Beras)*, vol. 53, no. 7. 2021.
- [3] Faiz Nashrullah, Suryo Adhi Wibowo, and Gelar Budiman, "Investigasi Parameter Epoch Pada Arsitektur ResNet-50 Untuk Klasifikasi Pornografi," *Journal of Computer, Electronic, and Telecommunication*, vol. 1, no. 1, pp. 1–8, 2020, doi: 10.52435/complete.v1i1.51.
- [4] S. Saidah, M. B. Adinegara, R. Magdalena, N. Kumalasari, C. Pratiwi, and F. T. Elektro, "Identifikasi Kualitas Beras Menggunakan Metode k-Nearest Neighbor dan Support Vector Machine Rice Quality Identification using k-Nearest Neighbor and Support Vector Machine Method," *TELKA*, vol. 5, no. 2, pp. 114–121, 2019.
- [5] Gansar Suwanto, R. Ibnu Adam, and Garno, "Identifikasi Citra Digital Jenis Beras Menggunakan Metode Anfis dan Sobel," *Jurnal Informatika Polinema*, vol. 7, no. 2, pp. 123–128, 2021, doi: 10.33795/jip.v7i2.406.
- [6] M. Koklu, I. Cinar, and Y. S. Taspinar, "Classification of Rice Varieties with Deep Learning Methods," *Comput Electron Agric*, vol. 187, no. June, p. 106285, 2021, doi: 10.1016/j.compag.2021.106285.
- [7] I. P. Putra, R. Rusbandi, and D. Alamsyah, "Klasifikasi Penyakit Daun Jagung Menggunakan Metode Convolutional Neural Network," *Jurnal Algoritme*, vol. 2, no. 2, pp. 102–112, 2022, doi: 10.35957/algoritme.v2i2.2360.
- [8] D. Putra, *Pengolahan Citra Digital*. Yogyakarta: Andi, 2010. [Online]. Available: <https://books.google.co.id/books?id=NectMutqXJAC&printsec=frontcover#v=onepage&q&f=false>
- [9] N. K. C. Pratiwi, N. Ibrahim, Y. N. Fu'Adah, and S. Rizal, "Deteksi Parasit Plasmodium pada Citra Mikroskopis Hapusan Darah dengan Metode Deep Learning," *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, vol. 9, no. 2, p. 306, Apr. 2021, doi: 10.26760/elkomika.v9i2.306.
- [10] N. A. Batubara and R. M. Awangga, *Tutorial Object Detection Plate Number With Convolutional Neural Network (CNN)*. Bandung: Kreatif Industri Nusantara, 2020.
- [11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-Decem, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.
- [12] K. H. Mahmud, Adiwijaya, and S. al Faraby, "Klasifikasi Citra Multi-Kelas Menggunakan Convolutional Neural Network," *e-Proceeding of Engineering*, vol. 6, no. 1, pp. 2127–2136, 2019.
- [13] R. Erwandi and S. Suyanto, "Klasifikasi Kanker Payudara Menggunakan Residual Neural Network," *Indonesia Journal on Computing (Indo-JC)*, vol. 5, no. 1, pp. 45–52, 2020, doi: 10.21108/indojc.2020.5.1.373.
- [14] Sandhopi, Lukman Zaman P.C.S.W, and Yosi Kristian, "Identifikasi Motif Jepara pada Ukiran dengan Memanfaatkan Convolutional Neural Network," *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*, vol. 9, no. 4, pp. 403–413, 2020, doi: 10.22146/jnteti.v9i4.541.
- [15] M. Koklu, "Rice Image Dataset," *www.kaggle.com*, 2021. <https://www.kaggle.com/datasets/muratkokludataset/rice-image-dataset> (accessed Apr. 22, 2022).
- [16] N. K. C. Pratiwi, Y. N. Fu'adah, and E. Edwar, "Early Detection of Deforestation through Satellite Land Geospatial Images based on CNN Architecture," *JURNAL INFOTEL*, vol. 13, no. 2, pp. 54–62, May 2021, doi: 10.20895/infotel.v13i2.642