

**IMPLEMENTASI ARSITEKTUR *MOBILENETV2* UNTUK
KLASIFIKASI CITRA BERAS IMPOR**



TESIS

Diajukan sebagai salah satu syarat untuk memperoleh gelar
Magister Ilmu Komputer (M.Kom)

NURUL KHASANAH

14002381

Program Studi Ilmu Komputer (S2)

Fakultas Teknologi Informasi

Universitas Nusa Mandiri

2021

PERSETUJUAN TESIS

Tesis ini diajukan oleh:

Nama : Nurul Khasanah
 NIM : 14002381
 Program Studi : Ilmu Komputer
 Jenjang : Strata Dua (S2)
 Konsentrasi : *Image Processing*
 Judul Tesis : Implementasi Arsitektur *MobileNetV2* Untuk Klasifikasi Citra Beras Impor

Telah dipertahankan pada periode 2021-1 dihadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Magister Ilmu Komputer (M.Kom) pada Program Studi Ilmu Komputer (S2) Fakultas Teknologi Informasi Universitas Nusa Mandiri.

Jakarta, 23 Agustus 2021

PEMBIMBING TESIS

Pembimbing I : Dr. Dwiza Riana, S.Si, MM,
 : M.Kom



DEWAN PENGUJI

Penguji I : Dr. Agus Subekti, M.T



Penguji II : Dr. Lindung Parningotan Manik,
 : M.T.I



Penguji III /
 Pembimbing I : Dr. Dwiza Riana, S.Si, MM,
 : M.Kom



DAFTAR ISI

	Halaman
HALAMAN SAMPUL	1
HALAMAN JUDUL.....	Error! Bookmark not defined.
SURAT PERNYATAAN ORISINALITAS DAN BEBAS PLAGIARISME	Error! Bookmark not defined.
PERSETUJUAN TESIS.....	2
LEMBAR BIMBINGAN TESIS	Error! Bookmark not defined.
KATA PENGANTAR	Error! Bookmark not defined.
SURAT PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH	Error! Bookmark not defined.
ABSTRAK	Error! Bookmark not defined.
ABSTRACT.....	Error! Bookmark not defined.
DAFTAR ISI	3
DAFTAR TABEL.....	Error! Bookmark not defined.
DAFTAR GAMBAR	Error! Bookmark not defined.
DAFTAR LAMPIRAN.....	Error! Bookmark not defined.
DAFTAR SIMBOL.....	Error! Bookmark not defined.
BAB I PENDAHULUAN	Error! Bookmark not defined.
1.1. Latar belakang	Error! Bookmark not defined.
1.2. Identifikasi Masalah	Error! Bookmark not defined.
1.3. Tujuan Penelitian.....	Error! Bookmark not defined.
1.4. Ruang Lingkup Penelitian	Error! Bookmark not defined.
1.5. Sistematika penulisan	Error! Bookmark not defined.
BAB II TINJAUAN PUSTAKA.....	Error! Bookmark not defined.
2.1. Tinjauan Pustaka	Error! Bookmark not defined.
2.1.1. Citra Digital.....	Error! Bookmark not defined.
2.1.2. <i>Dataset</i>	Error! Bookmark not defined.
2.1.3. Akuisisi Data.....	Error! Bookmark not defined.
2.1.4. <i>Data Preprocessing</i>	Error! Bookmark not defined.
2.1.5. Klasifikasi	Error! Bookmark not defined.
2.1.6. <i>Deep learning</i>	Error! Bookmark not defined.

2.1.7.	<i>Convolutional Neural Network (CNN)</i>	Error! Bookmark not defined.
2.1.8.	Ekstraksi fitur <i>CNN</i>	Error! Bookmark not defined.
2.1.9.	<i>Transfer Learning</i>	Error! Bookmark not defined.
2.1.10.	<i>MobileNet V2</i>	Error! Bookmark not defined.
2.1.11.	<i>Global Average Pooling</i>	Error! Bookmark not defined.
2.1.12.	<i>Activation function</i>	Error! Bookmark not defined.
2.1.13.	<i>Hyperparameter</i>	Error! Bookmark not defined.
2.1.14.	<i>Confusion Matrix</i>	Error! Bookmark not defined.
2.1.15.	<i>Classification Report</i>	Error! Bookmark not defined.
2.1.16.	<i>Overfitting dan Underfitting</i>	Error! Bookmark not defined.
2.1.17.	<i>Python</i>	Error! Bookmark not defined.
2.1.18.	<i>Flask</i>	Error! Bookmark not defined.
2.1.19.	<i>Google colaboratory</i>	Error! Bookmark not defined.
2.1.20.	<i>Keras dan tensorflow</i>	Error! Bookmark not defined.
2.2.	Tinjauan Studi	Error! Bookmark not defined.
2.3.	Tinjauan Objek Penelitian	Error! Bookmark not defined.
BAB III METODOLOGI PENELITIAN.....		Error! Bookmark not defined.
3.1.	Metode Penelitian.....	Error! Bookmark not defined.
3.2.	<i>Tools</i> Pendukung	Error! Bookmark not defined.
BAB IV HASIL PENELITIAN		Error! Bookmark not defined.
4.1.	Hasil akuisisi citra	13
4.2.	Hasil <i>Augmentation</i>	16
4.3.	Hasil <i>split</i> data	22
4.4.	Hasil klasifikasi	22
4.5.	Pengujian	27
4.6.	Evaluasi	29
4.7.	Implementasi	30
BAB V PENUTUP.....		Error! Bookmark not defined.
4.1.	Kesimpulan.....	41
4.2.	Saran	42
DAFTAR PUSTAKA		5

DAFTAR RIWAYAT HIDUP.....	Error! Bookmark not defined.
LAMPIRAN.....	Error! Bookmark not defined.

RINGKASAN TESIS

I. PENDAHULUAN

Pangan merupakan kebutuhan primer manusia yang harus tercukupi untuk bertahan hidup [1]. Bahan pangan pokok masyarakat Indonesia adalah beras. Jumlah produksi beras di Indonesia tahun 2019 mencapai 52,11 juta ton. Sedangkan untuk kebutuhan beras tahun 2019 adalah 33,15 juta ton. Dari jumlah produksi dan kebutuhan beras tersebut, Indonesia memiliki nilai surplus sebesar 18,95 juta ton [2]. Untuk menjaga kestabilan angka cadangan beras nasional, pemerintah Indonesia mengambil beberapa kebijakan. Salah satunya adalah melakukan impor beras dari negara penghasil beras, seperti Thailand dan Vietnam [3]. Data impor beras dari berbagai negara tahun 2019, yakni dari negara asal Vietnam sejumlah 33.133,1 ton, Thailand sejumlah 53.278 ton, Tiongkok 24,3 ton, India sejumlah 7.973,3 ton, Pakistan sejumlah 182.564,9 ton, Amerika Serikat 740,9 ton, Myanmar sejumlah 166.700,6 ton [4].

Jenis beras impor yang paling banyak masuk ke Indonesia yakni Beras *Basmati* dari Pakistan dan India, Beras *Jasmine* dari Vietnam dan Thailand, Beras *Japonica* dari Jepang, Australia, Korea, Italy, Taiwan, Beras *Bulgur* dari Timur Tengah, Beras *Kimbapssal* dari Korea [5]. Dari beberapa jenis beras impor yang ada di Indonesia masih sulit dikenali oleh masyarakat secara luas. Jenis beras impor dapat dianalisis oleh pakar industri secara manual dengan indera penglihatan. Namun, hal itu akan membutuhkan banyak waktu sehingga tidak efisien dan hasilnya belum tentu akurat [9]. Hal itu juga menyebabkan kasus beras oplosan juga banyak terjadi di Indonesia. Sebagai contoh kasus yang terjadi tahun 2016, beras oplosan antara beras lokal Demak dengan beras impor dari Thailand dijual pelaku ke pasaran sebagai beras premium [6]. Tindakan pengoplosan beras juga dilakukan oleh sejumlah wirausaha di Indonesia, seperti kasus pengoplosan yang dilakukan oleh pemilik Toko Beras Jaya Abadi di Pangkalpinang, Bangka Belitung, kasus pengoplosan beras yang dilakukan oleh pelaku usaha di Cianjur, dan tindakan pengoplosan beras dengan sabun yang terjadi di Gresik [7].

Perkembangan teknologi mampu mengklasifikasi jenis beras dari beberapa ciri yang ada. Salah satu teknologi yang dapat mengidentifikasi dengan ciri fisik yaitu dengan menggunakan teknologi pengolahan citra (*image processing*). Dalam pengolahan data citra biasanya menggunakan *deep learning* dengan algoritma *Convolutional Neural Network (CNN)*. *Deep learning* adalah cabang ilmu *machine learning* berbasis Jaringan Saraf Tiruan (JST) untuk mengklasifikasi secara langsung dari gambar atau suara. *CNN* adalah salah satu algoritma *deep learning* yang merupakan pengembangan dari *Multilayer Perceptron (MLP)* yang dirancang untuk mengolah data dalam bentuk dua dimensi, misalnya gambar atau suara [8].

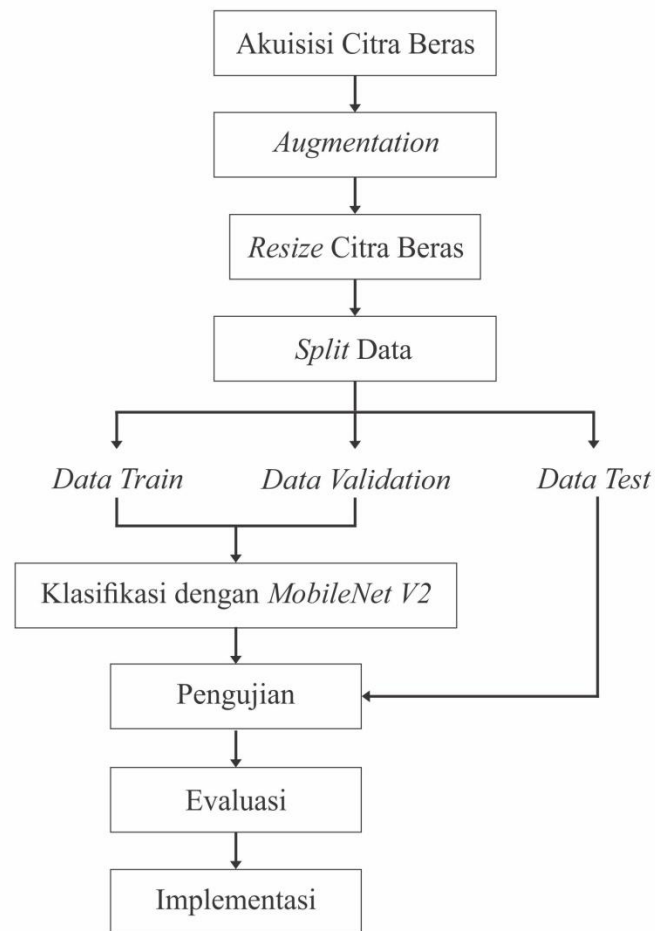
Beberapa penelitian sebelumnya mengenai identifikasi jenis beras menggunakan tahapan penelitian sebagai berikut : akuisisi data, *preprocessing*, segmentasi citra, klasifikasi dan evaluasi [8], [9], [10], [11]. Akuisisi citra dengan kamera digital dilakukan oleh Ricardo dan Gasim [8], Srimulyani dan Musdholifah [12], Suwanto, dkk [13], Aznan, dkk [14], Aprilia, dkk [15]. Sedangkan untuk akuisisi citra dengan *flatbed scanner* dilakukan oleh Nurfalah, dkk [11]. Penelitian Srimulyani dan Musdholifah [12], Aprilia, dkk [15], Suwanto, dkk [13], Sumaryanti [16] menggunakan *dataset* citra beras lokal, seperti Mentik Wangi, Pandan Wangi, Mentik Susu, Raja Lele, Ciherang, Batang Lembang, Cianjur, C4, dan Mekongga, IR64, Cilosari, Inpari 6 dan Membramo. Sedangkan penelitian Nurfalah, dkk [11], Surekha, dkk [10], Aznan, dkk [14], Dheer dan Singh [17] menggunakan citra beras impor seperti *Basmati*, *Japonica*, *Jasmine*, beras Malaysia yaitu *MR 219* dan *MR 269*, beras India yaitu *Jal Lahari*, *Kasturi*, dan lainnya. Untuk teknik segmentasi yang digunakan dalam penelitian Aprilia, dkk [15], Suwanto, dkk [13], Aznan, dkk [14] adalah segmentasi *thresholding*. Penelitian Nurfalah, dkk [11], Mohan dan Gopal [9] melakukan identifikasi citra beras dengan menggunakan algoritma *machine learning* yaitu *Neural Network* dan *Multi-SVM*. Arora, dkk [18] menggunakan 4 algoritma *machine learning* dalam proses klasifikasi, yaitu *Logistic Regression*, *Decision Tree*, *Support Vector Machine (SVM)* dan *Random Forest*. Dheer dan Singh [17] dalam penelitiannya menggunakan algoritma *machine learning* yang lebih beragam, diantaranya *Logistic Regression*, *Naïve-Bayes*, *Linear Discriminant Analysis*, *K-Nearest Neighbor's*. Klasifikasi citra beras menggunakan *deep learning* juga telah dilakukan oleh Son dan Thai-Nghe [19] yang menghasilkan

akurasi 89.75%. Penelitian mengenai klasifikasi citra beras masih banyak menggunakan algoritma *machine learning*, sedangkan penelitian dengan *deep learning* masih sedikit. Penelitian yang telah dilakukan sebelumnya juga tidak membuat implementasi hasil pengujian, baik itu dalam bentuk *mobile* maupun *website*. Di sisi lain, penerapan *deep learning* diberbagai kasus sudah banyak ditemui dan menghasilkan hasil akurasi yang baik. Seperti penelitian yang dilakukan oleh Dai, et. al [20] dan Aningtiyas, dkk [21] dengan menggunakan arsitektur *MobileNetV2* dan menghasilkan akurasi yang tinggi yaitu 96% [20] dan 93.02% [21]. Selain itu, hasil penelitian juga dapat diimplementasikan dalam sebuah aplikasi.

Berdasarkan permasalahan mengenai kesulitan dalam klasifikasi jenis beras impor dan kekurangan penelitian sebelumnya, maka dalam penelitian ini akan dilakukan klasifikasi menggunakan *deep learning* dengan architecture *MobileNet V2*. Jenis beras yang dipakai ada 5 yaitu Basmati, Bulgur, Jasmine, Japonica dan Kimbapsal. Proses akuisisi citra akan dilakukan sendiri menggunakan flatbed scanner. Dataset yang diperoleh akan dilakukan augmentation agar jumlah dataset menjadi lebih banyak. Dataset baru akan dibagi menjadi data test, train dan validation. Untuk klasifikasi akan menggunakan arsitektur *MobileNet V2* dengan hyperparameter yang dimodifikasi untuk memperoleh hasil terbaik. Evaluasi hasil akan dilakukan menggunakan confusion matrix dan classification report yang akan digunakan untuk mengukur akurasi, precision dan recall. Hasil dari penelitian ini akan diimplementasi dalam bentuk website. Judul penelitian yang akan dilakukan yaitu “Implementasi Arsitektur *MobileNet V2* Untuk Klasifikasi Citra Beras Impor”. Dengan penelitian ini, diharapkan akan muncul hasil yang akurat untuk identifikasi jenis beras impor di Indonesia sehingga membantu masyarakat dalam mengenali jenis beras impor.

II. METODE PENELITIAN

Tahapan penelitian yang dilakukan dalam penelitian ditunjukkan pada Gambar 3.1:



Gambar 3.1
Tahapan Penelitian

Berdasarkan gambar 3.1, metode penelitian dapat dijelaskan sebagai berikut:

A. Akuisisi Citra Beras

Dataset yang digunakan dalam penelitian ini adalah citra beras impor. Terdapat 5 jenis citra beras yang digunakan yaitu beras *Basmati*, beras *Bulgur*, beras *Japonica*, beras *Jasmine* dan beras *Kimbapssal*. Citra beras yang digunakan ditunjukkan pada Gambar berikut:



Sumber: Hasil penelitian
Gambar 3.2

Beras *Basmati*



Sumber: Hasil penelitian
Gambar 3.3

Beras *Bulgur*



Sumber: Hasil penelitian
Gambar 3.4

Beras *Japonica*



Sumber: Hasil penelitian
Gambar 3.5

Beras *Jasmine*



Sumber: Hasil penelitian
Gambar 3.6

Beras *Kimbapssal*

Akuisisi citra atau pengambilan citra dilakukan menggunakan *flatbed scanner type* Canon MG2570S yang memiliki resolusi 600 x 1200 dpi. *Background* yang digunakan saat *scan* citra adalah kertas hitam polos untuk menghindari distraksi warna pada objek penelitian. Citra beras diletakkan pada bidang *scanner* secara berjarak untuk menghindari citra yang bertumpuk (*overlapping*). Berikut adalah gambaran dari proses akuisisi citra:



Gambar 3.7
Proses Akusisi citra

B. *Augmentation*

Deep learning akan menghasilkan performa secara optimal jika data yang digunakan berjumlah banyak. Untuk memperbanyak jumlah citra, dilakukan proses *augmentation*. Selain itu, proses *augmentation* dilakukan untuk mencegah terjadinya *overfitting*. Dalam proses ini dilakukan rotasi, *zoom*, *flip* terhadap citra asli, kemudian citra hasil *augmentation* akan disimpan ke media penyimpanan (*google drive*). Berikut detail variabel yang digunakan dalam proses *augmentation*:

Tabel 3.1 *Variable Augmentation*

Variabel	<i>Value in Augmentation</i>
<i>Rotation range</i>	15
<i>Width shift range</i>	0.2
<i>Height shift range</i>	0.2
<i>Zoom range</i>	0.2
<i>Horizontal flip</i>	<i>True</i>
<i>Vertical flip</i>	<i>True</i>

<i>Shear range</i>	0.2
<i>Batch size</i>	78

Sumber : Hasil penelitian

Rotasi menghasilkan gambar yang dirotasi berdasarkan derajat yang dipilih secara *random*. *Zoom* menghasilkan citra yang diperbesar secara *random*. *Flip* menghasilkan citra yang dibalik secara horizontal dan vertikal. *Batch size* digunakan untuk menentukan jumlah citra yang akan dihasilkan.

C. *Resize* Citra Beras

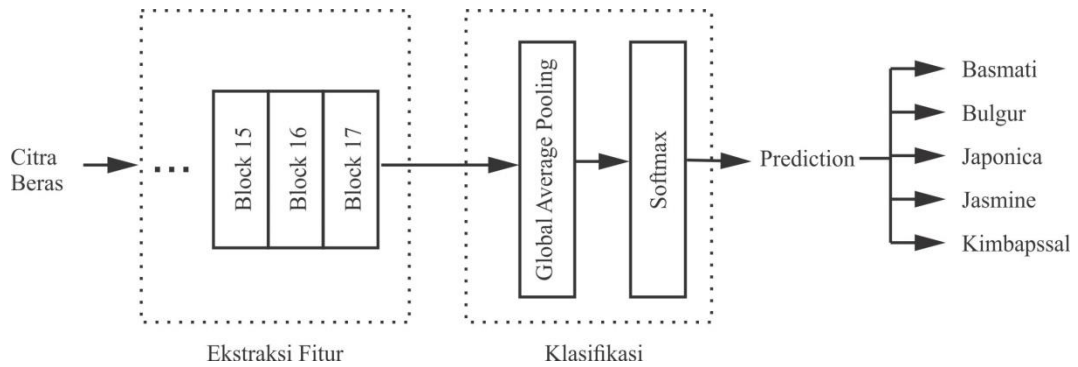
Resize adalah proses mengubah ukuran citra ke dalam sebuah *pixel* yang sama. Arsitektur *MobileNetV2* mendukung semua ukuran *input* di atas 32 x 32 *pixel*, namun semakin besar ukuran *pixel* akan memberikan hasil yang lebih baik. Dalam penelitian ini, citra akan *resize* ke dalam ukuran *default MobileNetV2* yaitu 224 x 224 *pixel*.

D. *Split* Data

Tahap selanjutnya adalah *split* data menggunakan teknik *train test split* yang didukung oleh *library sklearn*. Citra dibagi menjadi data *train* (80%), data *test* (10%) dan data *validation* (10%).

E. Klasifikasi Citra Beras dengan *MobileNet V2*

Model yang diusulkan dalam klasifikasi citra beras impor ini adalah *MobileNetV2* menggunakan *transfer learning* dengan *weights = imageNet*. Arsitektur *MobileNetV2* terdiri dari 17 blok, *Global Average Pooling*, dan *Classification Layer*. Berikut adalah rancangan arsitektur *MobileNet V2* untuk klasifikasi citra beras impor:



Sumber : Hasil penelitian

Gambar 3.8
Rancangan Arsitektur

Ukuran *input* citra yaitu (224, 224, 3). Nilai 3 mendeskripsikan bahwa citra yang memiliki 3 buah *channel* yaitu *red*, *green* dan *blue* (RGB). Pada lapisan *Bottleneck* (17 blok) terdiri dari *Ekspansion Layer*, *Depthwise Convolution* dan *Projection Convolution*. Masing-masing layer memiliki *Batch Normalization* dan fungsi aktivasi *ReLU6*, kecuali *layer projection convolution*. Hal itu dikarenakan lapisan ini menghasilkan data berdimensi rendah, sehingga jika memakai fungsi aktivasi maka informasi yang penting akan hilang. *Output* dari ekstraksi fitur adalah citra dengan ukuran 7 x 7 *pixel*. Pada layer *Global Average Pooling* ukuran citra diubah menjadi 1 x 1 *pixel*. Selanjutnya adalah *layer* klasifikasi untuk 5 kelas dengan *activation softmax*. Pada proses klasifikasi, akan dilakukan eksperimen dengan berbagai jenis *hyperparameter* yang akan disesuaikan dengan kebutuhan. Dari proses *compile* model, akan dihasilkan model terbaik dengan nilai akurasi tertinggi dan *loss* terendah. Model terbaik dari proses *fit model* akan digunakan untuk prediksi citra berdasarkan kelasnya.

F. Pengujian

Pada tahap ini dilakukan pengujian dengan model terhadap citra yang terdapat pada data *test*. Hal itu bertujuan untuk mengetahui seberapa akurat model yang telah dibentuk terhadap data *test*. Dari penerapan model ke data *train*, data *test* dan data *validation* akan diketahui terjadinya *overfitting*, *underfitting* atau *good fit* selama proses *running*. Hasil akurasi akan ditampilkan dalam bentuk grafik untuk memudahkan dalam analisa.

G. Evaluasi

Evaluasi dilakukan dengan memunculkan angka *confusion matrix* dan *classification report*. Pada dasarnya *confusion matrix* berisi perbandingan hasil klasifikasi yang dilakukan oleh model dengan hasil klasifikasi sebenarnya. Angka pada *confusion matrix* digunakan untuk mengetahui angka *TP* (*True Positif*), *TN* (*True Negatif*), *FP* (*False Positif*), dan *FN* (*False Negatif*). Sedangkan *classification report* digunakan untuk mengetahui angka *performance matrix*, seperti nilai akurasi, *precision*, *recall* dan *F1-score*.

H. Implementasi

Hasil dari penelitian ini diimplementasikan dalam bentuk *website*. Pembuatan *website* ini menggunakan *framework flask* yang terdapat pada *Python*. Dalam *website* yang dibangun akan dimasukkan model yang dihasilkan dari penelitian. *Output* yang akan dihasilkan yaitu prediksi mengenai label citra.

III. HASIL DAN PEMBAHASAN

Pada bagian ini dijelaskan mengenai hasil akuisisi citra, hasil *augmentation*, *split data*, hasil klasifikasi, pengujian, evaluasi serta implementasi dari penelitian yang telah dilakukan:

4.1. Hasil akuisisi citra

Proses akuisisi citra dilakukan menggunakan *flatbed scanner*. Citra yang dihasilkan dari proses akuisisi citra memiliki ukuran 286 x 286 *pixel*, dengan jenis citra terpisah. Jumlah citra yang diperoleh adalah 180 citra yang terbagi menjadi 5 kelas, sehingga tiap kelas terdapat 36 citra. Detail citra akan dijabarkan dalam Tabel 4.1.

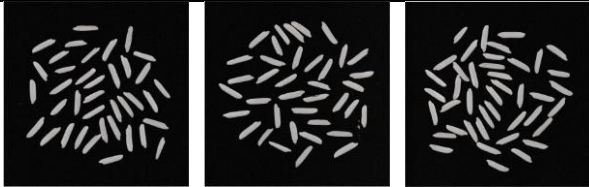
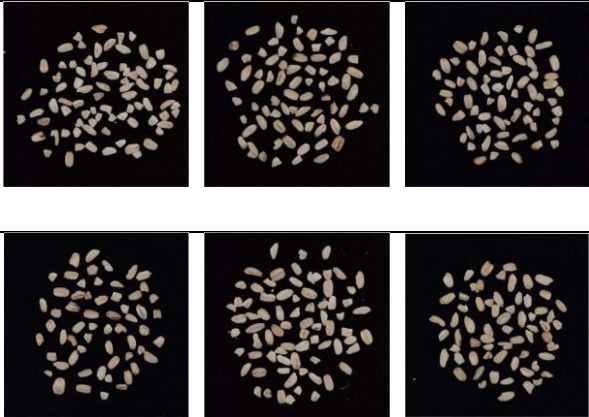
Tabel 4.1 Detail Citra Beras Impor

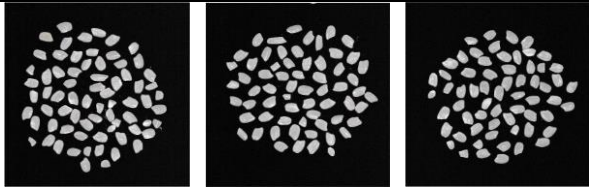
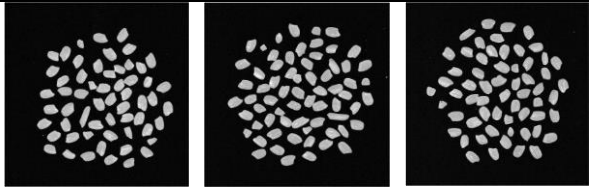
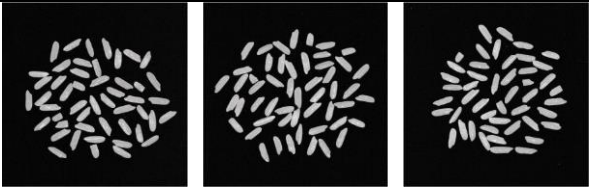
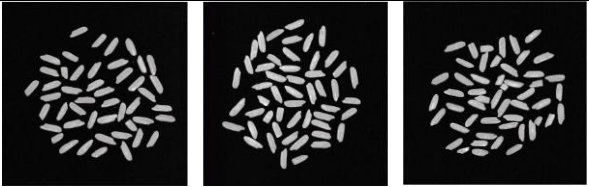
No	Kelas (label)	Jumlah citra
1	<i>Basmati</i>	36 citra
2	<i>Bulgur</i>	36 citra

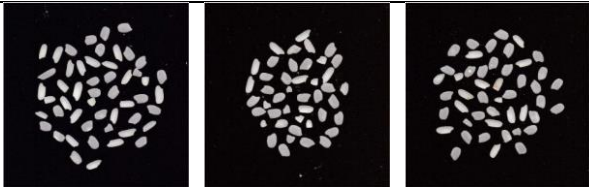
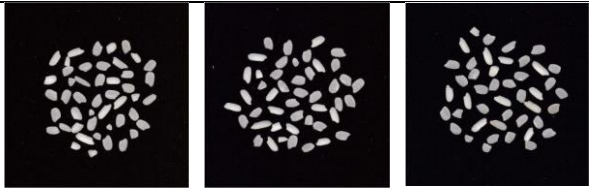
3	<i>Japonica</i>	36 citra
4	<i>Jasmine</i>	36 citra
5	<i>Kimbapssal</i>	36 citra
Total Citra		180 citra

Berikut adalah citra yang dihasilkan dari proses akuisisi citra:

Tabel 4.2 Hasil Akuisisi Citra

No	Nama File	Citra
1	<i>Basmati</i>	
No	Nama File	Citra
	<i>Basmati</i>	
2	<i>Bulgur</i>	

3	<i>Japonica</i>	
		
4	<i>Jasmine</i>	
		

No	Nama File	Citra
5	<i>Kimbapssa1</i>	
		

Sumber : Hasil penelitian

4.2. Hasil *Augmentation*

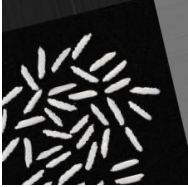
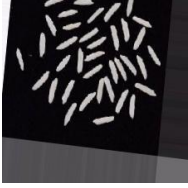
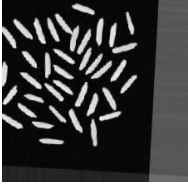
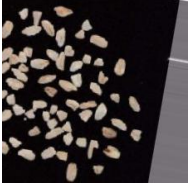
Pada proses *augmentation* dilakukan rotasi, *zoom*, *flip* vertikal dan horizontal terhadap citra asli pada semua kelas. Dari proses *augmentation* dihasilkan citra sejumlah 1980 citra. Berikut adalah rincian jumlah citra hasil *augmentation*:

Tabel 4.3 Detail Citra Hasil *Augmentation*

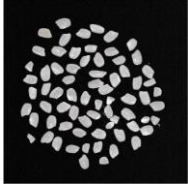
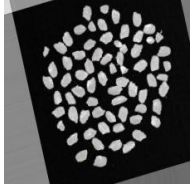
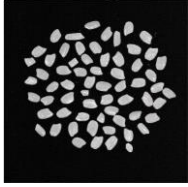
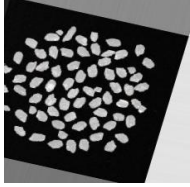
No	Kelas (<i>label</i>)	Jumlah citra
1	<i>Basmati</i>	396 citra
2	<i>Bulgur</i>	396 citra
3	<i>Japonica</i>	396 citra
4	<i>Jasmine</i>	396 citra
5	<i>Kimbapssal</i>	396 citra
Total Citra		1980 citra

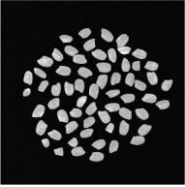
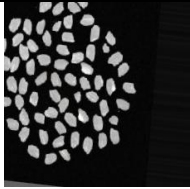
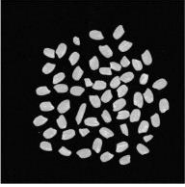
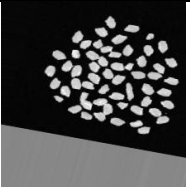
Citra tersebut disimpan dalam format *JPG*. Berikut adalah hasil dari proses *augmentation*:

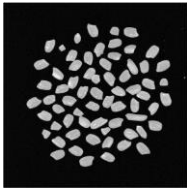
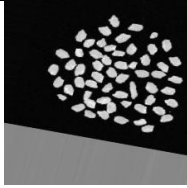
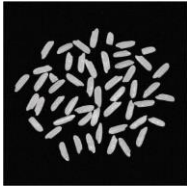
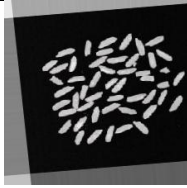
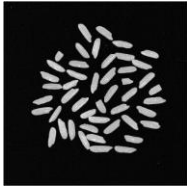
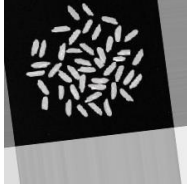
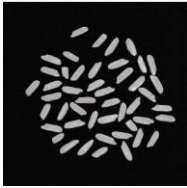
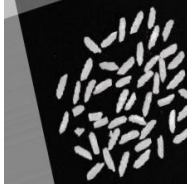
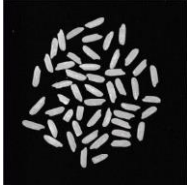
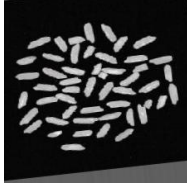
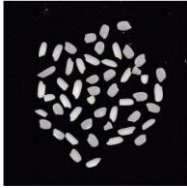
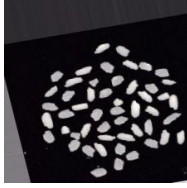
Tabel 4.4 Citra Hasil *Augmentation*

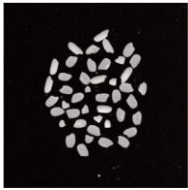
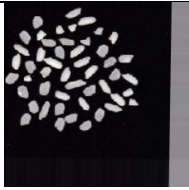
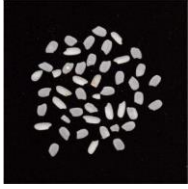
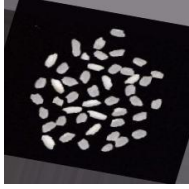
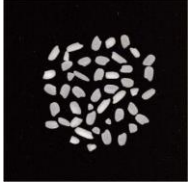
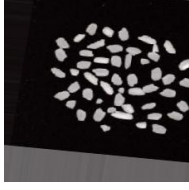
No	Jenis Beras	Citra Awal	Hasil <i>Augmentation</i>
1	<i>Basmati</i>		
2	<i>Basmati</i>		
3	<i>Basmati</i>		
4	<i>Basmati</i>		
5	<i>Basmati</i>		
6	<i>Bulgur</i>		

7	<i>Bulgur</i>		
---	---------------	---	---

No	Jenis Beras	Citra Awal	Hasil <i>Augmentation</i>
8	<i>Bulgur</i>		
9	<i>Bulgur</i>		
10	<i>Bulgur</i>		
11	<i>Japonica</i>		
12	<i>Japonica</i>		

13	<i>Japonica</i>		
14	<i>Japonica</i>		

No	Jenis Beras	Citra Awal	Hasil <i>Augmentation</i>
15	<i>Japonica</i>		
16	<i>Jasmine</i>		
17	<i>Jasmine</i>		
18	<i>Jasmine</i>		
19	<i>Jasmine</i>		
20	<i>Jasmine</i>		
21	<i>Kimbapssal</i>		

No	Jenis Beras	Citra Awal	Hasil <i>Augmentation</i>
22	<i>Kimbapssal</i>		
23	<i>Kimbapssal</i>		
24	<i>Kimbapssal</i>		
25	<i>Kimbapssal</i>		

4.3. Hasil *split* data

Proses *split* data adalah membagi data menjadi data *train* (80%), data *test* (10%) dan data *validation* (10%) sehingga menghasilkan data *train* berjumlah 1584 citra, data *test* 198 citra, dan data *validation* sebanyak 198 citra.

4.4. Hasil klasifikasi

Proses klasifikasi dilakukan menggunakan arsitektur *MobileNetV2* dengan *transfer learning* menggunakan *ImageNet*. Berdasarkan eksperimen yang telah dilakukan penulis menggunakan *dataset* yang sama, arsitektur *MobileNetV2* menghasilkan akurasi yang lebih baik daripada arsitektur *CNN* lainnya. Pada tahap eksperimen, penulis juga menguji beberapa *hyperparameter* sebagai berikut:

Tabel 4.5 *Hyperparameter MobileNetV2*

Variabel	Value in MobileNetV2
<i>Weight</i>	<i>ImageNet</i>
<i>Input_shape</i>	<i>(Image size, Image size, 3)</i>
<i>Image size</i>	224
<i>Batch_size</i>	32, 64
<i>Training epoch</i>	10
<i>Optimizer</i>	<i>Adam, RMSProp, SGD</i>
<i>Learning rate</i>	0.001
<i>Loss</i>	<i>categorical_crossentropy</i>
<i>Activation</i>	<i>Softmax</i>
<i>Model save</i>	<i>model.hdf5</i>

Sumber : Hasil penelitian

Tahap selanjutnya adalah klasifikasi dengan data *train* dan data *validation*. Proses klasifikasi akan menghasilkan model terbaik yang akan disimpan dalam *model.hdf5* untuk selanjutnya akan dipakai dalam tahap implementasi. Berikut adalah hasil eksperimen yang telah dilakukan dengan menggunakan beberapa jenis *hyperparameter*:

Tabel 4.6 Hasil Eksperimen tanpa *augmentataion* data

No	Arsitektur	<i>Train</i>	<i>Test</i>	<i>Val</i>	<i>Optimizer</i>	<i>Batch size</i>	<i>Epoch</i>	Akurasi	<i>Loss</i>
1	<i>MobileNetV2</i>	60%	20%	20%	<i>Adam</i>	32	10	100%	4%

1	<i>MobileNetV2</i>	60%	20%	20%	<i>Adam</i>	32	10	90%	20%
2	<i>MobileNetV2</i>	70%	15%	15%	<i>Adam</i>	32	10	96%	14%
3	<i>MobileNetV2</i>	80%	10%	10%	<i>Adam</i>	32	10	98%	5%

Dari eksperimen dengan *split* data, diperoleh hasil terbaik yaitu dengan perbandingan *split* data *train* (80%), data *test* (10%) dan data *validation* (10%) yang menghasilkan akurasi 98% dan nilai *loss* 5%. Selanjutnya akan dilakukan eksperimen dengan menggunakan jenis *optimizer*. Berikut adalah hasil eksperimen yang telah dilakukan:

Tabel 4.7 Hasil Eksperimen dengan *Optimizer*

No	Arsitektur	<i>Train</i>	<i>Test</i>	<i>Val</i>	<i>Optimizer</i>	<i>Batch size</i>	<i>Epoch</i>	Akurasi	<i>Loss</i>
1	<i>MobileNetV2</i>	80%	10%	10%	<i>Adam</i>	32	10	98%	5%
2	<i>MobileNetV2</i>	80%	10%	10%	<i>RMSProp</i>	32	10	98%	18%
3	<i>MobileNetV2</i>	80%	10%	10%	<i>SGD</i>	32	10	94%	14%

Dari eksperimen dengan *optimizer*, diperoleh hasil terbaik yaitu dengan *optimizer Adam* yang menghasilkan akurasi 98% dan nilai *loss* 5%. Selanjutnya akan dilakukan eksperimen dengan menggunakan *batch size*. Berikut adalah hasil eksperimen yang telah dilakukan:

Tabel 4.8 Hasil Eksperimen dengan *Batch size*

No	Arsitektur	<i>Train</i>	<i>Test</i>	<i>Val</i>	<i>Optimizer</i>	<i>Batch size</i>	<i>Epoch</i>	Akurasi	<i>Loss</i>
1	<i>MobileNetV2</i>	80%	10%	10%	<i>Adam</i>	32	10	98%	5%
2	<i>MobileNetV2</i>	80%	10%	10%	<i>Adam</i>	64	10	97%	7%

Sumber : Hasil penelitian

Dari penelitian yang telah dilakukan, hasil terbaik yaitu nilai akurasi sebesar 98% dan nilai *loss* 5%. Penelitian tersebut diperoleh dari eksperimen dengan prosentase *split data train* (80%), data *test* (10%) dan data *validation* (10%) dan menggunakan *optimizers Adam*, *batch size* = 32 dan *epoch* = 10. Hasil penelitian yang diperoleh lebih baik dari penelitian sebelumnya [10][11] [19].

Dari eksperimen yang dilakukan, dihasilkan *summary model MobileNetV2* sebagai berikut:

Tabel 4.9 *Summary model MobileNetV2*

<i>Input</i>	<i>Operator</i>	<i>t</i>	<i>c</i>	<i>n</i>	<i>s</i>
224 x 224 x 3	<i>Conv2D</i>	-	32	1	2
112 x 112 x 32	<i>Bottleneck</i>	1	16	1	1
112 x 112 x 16	<i>Bottleneck</i>	6	24	2	2
56 x 56 x 24	<i>Bottleneck</i>	6	32	3	2
28 x 28 x 32	<i>Bottleneck</i>	6	64	4	2
14 x 14 x 64	<i>Bottleneck</i>	6	96	3	1
14 x 14 x 96	<i>Bottleneck</i>	6	160	3	2
7 x 7 x 160	<i>Bottleneck</i>	6	320	1	1
7 x 7 x 320	<i>Conv2D 1 x 1</i>	-	1280	1	1
7 x 7 x 1280	<i>Global avg pooling</i>	-	1280	1	-
1 x 1 x 1280	<i>Dense</i>	-	5	-	-

Sumber : Hasil penelitian

4.5. Pengujian

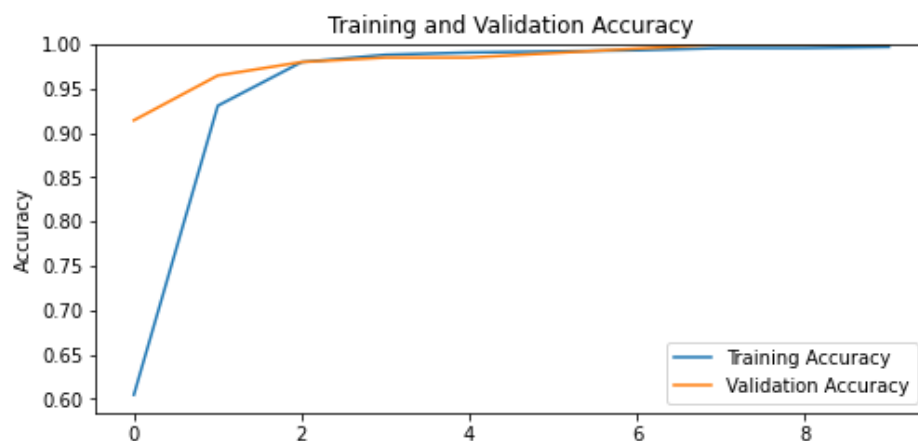
Proses pengujian dilakukan dengan membandingkan hasil akurasi dan nilai *loss* dari data *test* dengan data *train*. Data *test* tidak digunakan dalam proses klasifikasi, sehingga data ini dapat dijadikan acuan untuk menentukan kelayakan model yang telah dibangun. Berikut adalah hasil akurasi data tersebut:

Tabel 4.10 Hasil Akurasi dan Nilai *Loss*

Data	Hasil Akurasi	Nilai <i>Loss</i>
Data <i>Test</i>	98%	5%
Data <i>Train</i>	99%	4%

Sumber : Hasil penelitian

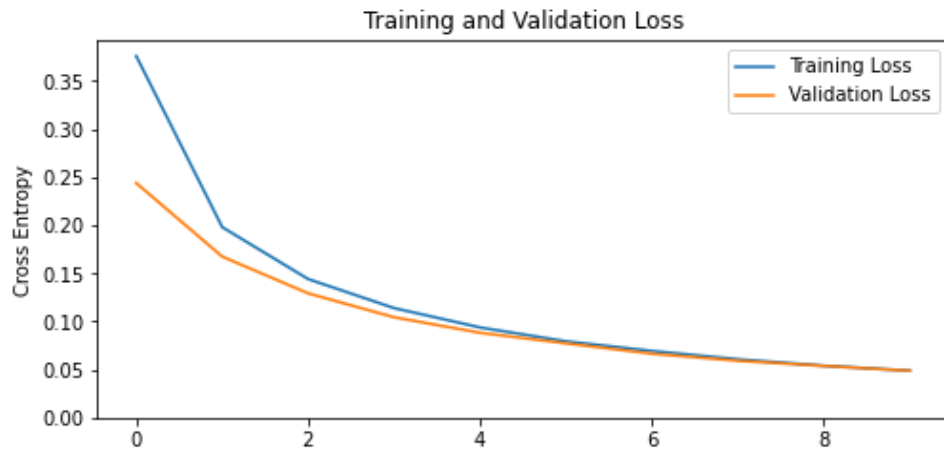
Dari hasil akurasi dan nilai *loss* pada data *test* dan data *train* dapat diketahui bahwa model tersebut memiliki nilai yang seimbang jika diterapkan pada data yang berbeda. Selanjutnya akan ditampilkan grafik akurasi dan nilai *loss* antara data *train* dan data *validation* untuk mengetahui model yang dibentuk mengalami *overfitting* atau *underfitting*. Berikut adalah grafik akurasi dan nilai *loss* pada data *train* dan data *validation*:



Sumber : Hasil penelitian

Gambar 4.3

Grafik akurasi data *train* dan data *validation* dengan *augmentation* data



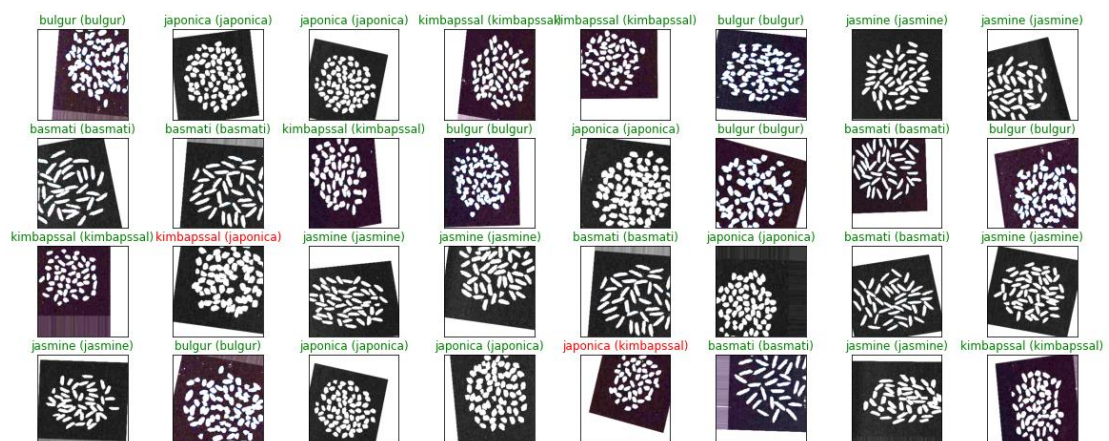
Sumber : Hasil penelitian

Gambar 4.4

Grafik nilai *loss* data *train* dan data *validation* dengan *augmentation* data

Grafik perbandingan nilai akurasi dan nilai *loss* pada data *train* dan data *validation* menunjukkan grafik yang seimbang. Artinya nilai akurasi dan nilai *loss* pada data *train* sebanding dengan nilai akurasi dan nilai *loss* pada data *validation*, sehingga dapat dikatakan bahwa model yang dibentuk tidak mengalami *overfitting* maupun *underfitting* dan model layak untuk diterapkan dalam proses implementasi.

Model yang dihasilkan dari proses klasifikasi digunakan untuk prediksi citra pada data *testing*. Berikut adalah hasil prediksi citra beras:



Sumber : Hasil penelitian

Gambar 4.5

Hasil Prediksi Citra

Dari 32 citra pada data *testing* yang diprediksi di *Google Collaboratory* menggunakan model yang telah dibangun, hasil prediksi sebanyak 30 citra sesuai dengan label asli. Sedangkan hasil prediksi 2 citra tidak sesuai dengan prediksi.

4.6. Evaluasi

Proses evaluasi dilakukan menggunakan *confusion matrix*. Angka *confusion matrix* dapat digunakan untuk menghitung akurasi, *recall* dan *precision*. Berikut adalah angka *confusion matrix* yang dihasilkan:

Tabel 4.11 *Confusion Matrix*

	<i>Basmati</i>	<i>Bulgur</i>	<i>Japonica</i>	<i>Jasmine</i>	<i>Kimbapssal</i>	
<i>Basmati</i>	40	0	0	0	0	40
<i>Bulgur</i>	0	39	0	0	0	39
<i>Japonica</i>	0	0	34	0	2	36
<i>Jasmine</i>	0	0	0	40	0	40
<i>Kimbapssal</i>	0	0	1	0	42	43
	40	39	35	40	44	198

Sumber : Hasil penelitian

Dari hasil *confusion matrix*, dapat diperoleh kesimpulan sebagai berikut:

1. 40 citra *Basmati* diprediksi dengan benar (*True Positif*)
2. 39 citra *Bulgur* diprediksi dengan benar (*True Positif*)
3. 34 citra *Japonica* diprediksi dengan benar (*True Positif*), 2 citra benar tetapi diprediksi salah (*False Negatif*), 1 citra salah tetapi diprediksi benar (*False Positif*)
4. 40 citra *Jasmine* diprediksi dengan benar (*True Positif*)
5. 42 citra *Kimbapssal* diprediksi dengan benar (*True Positif*), 2 citra benar tetapi diprediksi salah (*False Negatif*), 1 citra salah tetapi diprediksi benar (*False Positif*)

Hasil evaluasi ditampilkan lebih detail menggunakan *classification report*:

Tabel 4.12 *Classification Report*

	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>	<i>Support</i>
<i>Basmati</i>	100%	100%	100%	40
<i>Bulgur</i>	100%	100%	100%	39
<i>Japonica</i>	97%	94%	96%	36
<i>Jasmine</i>	100%	100%	100%	40
<i>Kimbapssal</i>	95%	98%	97%	43
<i>Accuracy</i>			98%	198
<i>Macro avg</i>	99%	98%	98%	198
<i>Weighted avg</i>	98%	96%	98%	198

Sumber : Hasil penelitian

Dari tabel 4.7 dapat dilihat nilai *precision*, *recall* dan *F1-score* untuk masing-masing kelas. Untuk kelas *Basmati*, nilai *precision* = 100%, *recall* = 100% dan *F1-score* = 100%. Untuk kelas *Bulgur*, nilai *precision* = 100%, *recall* = 100% dan *F1-score* = 100%. Untuk kelas *Japonica*, nilai *precision* = 97%, *recall* = 94% dan *F1-score* = 96%. Untuk kelas *Jasmine*, nilai *precision* = 100%, *recall* = 100% dan *F1-score* = 100%. Untuk kelas *Kimbapssal*, nilai *precision* = 95%, *recall* = 98% dan *F1-score* = 97%. Hasil akurasi yang diperoleh adalah 98%, *recall* adalah 96% dan *precision* 98%.

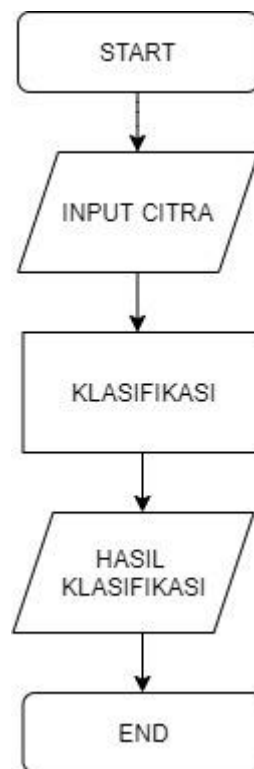
4.7. Implementasi

Penelitian yang dilakukan arsitektur *MobileNetV2* menghasilkan sebuah model dalam format .hdf5. Hasil penelitian diimplementasikan dalam bentuk *website* menggunakan *framework flask* dengan bahasa pemrograman *Python*. *Website* yang dibangun hanya bisa klasifikasi 5 jenis beras impor saja (*Basmati*, *Bulgur*,

Japonica, *Jasmine* dan *Kimbapssal*) dengan citra terpisah. Proses identifikasi dilakukan menggunakan *sample* 1 citra per kelas untuk mengetahui nilai *probability* masing-masing kelas. Citra yang digunakan dalam implementasi adalah citra yang berbeda dengan citra yang digunakan saat klasifikasi. Hal tersebut dilakukan untuk menguji tingkat akurasi model terhadap citra baru. Nilai *probability* tertinggi akan dibandingkan dengan kelas asli, jika hasilnya sesuai maka *website* layak untuk digunakan.

A. Flow Chart (Diagram Alur)

Berikut adalah diagram alur sistem berjalannya *website* yang dibangun:



Sumber : Hasil penelitian

Gambar 4.6

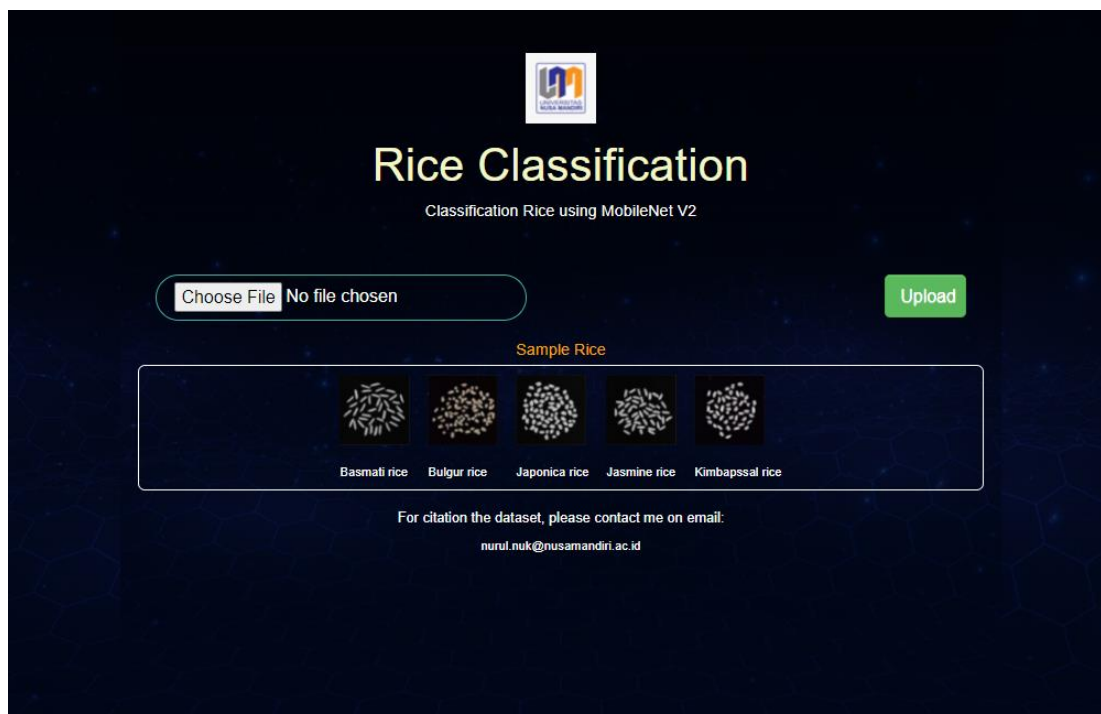
Flow Chart

B. Website

Dalam *website* pengujian, terdapat 2 halaman yaitu halaman awal (*index*) dan halaman hasil prediksi (*result*). Berikut adalah tampilan *website*:

1. Halaman awal

Halaman awal (*index*) berisi tentang *sample* beras impor yang terdiri dari *Basmati*, *Bulgur*, *Japonica*, *Jasmine* dan *Kimbapssal*. Di halaman ini terdapat *button* untuk *upload* citra beras yang akan diklasifikasi. Untuk jenis *file* yang diupload pada halaman ini harus berbentuk *JPG* atau format gambar lainnya. Jika *file* yang diupload bukan format gambar, maka sistem tidak akan memproses dan harus dilakukan *input* citra kembali.



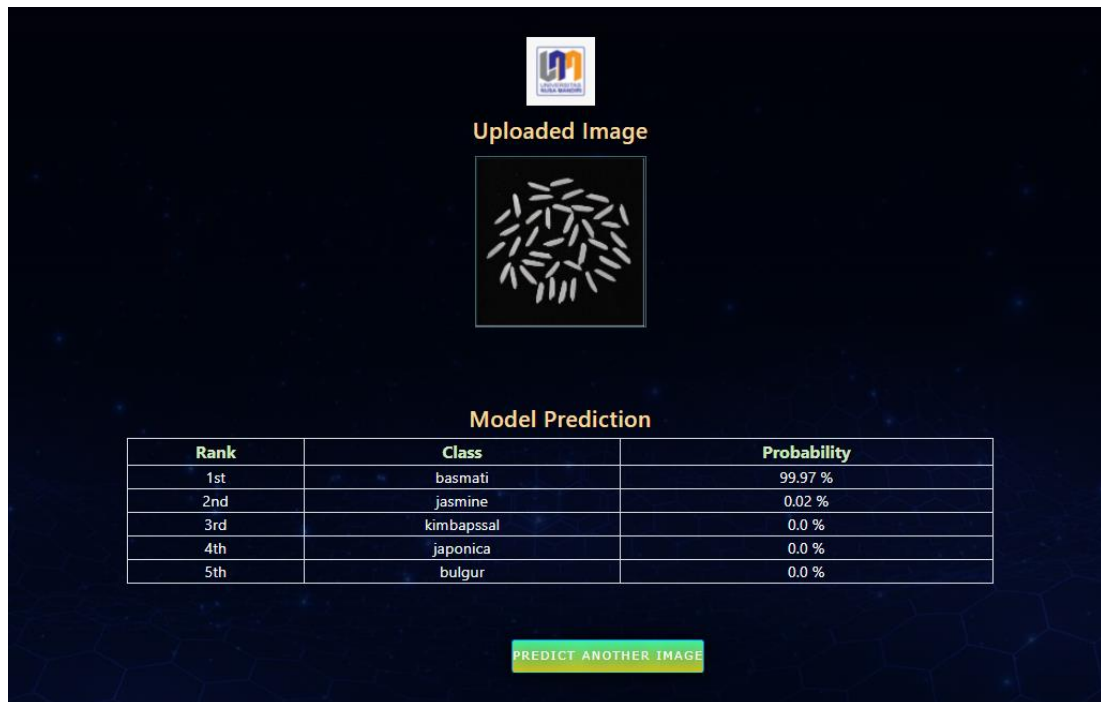
Sumber : Hasil penelitian

Gambar 4.7

Halaman *index*

2. Halaman *result*

Halaman *result* berisi hasil identifikasi citra beras yang telah diupload. Identifikasi berisi nama kelas dan nilai *probability* citra. Nilai *probability* per citra akan diurutkan dari nomor 1 sampai nomor 5. Berikut adalah halaman *result* untuk 5 jenis beras:

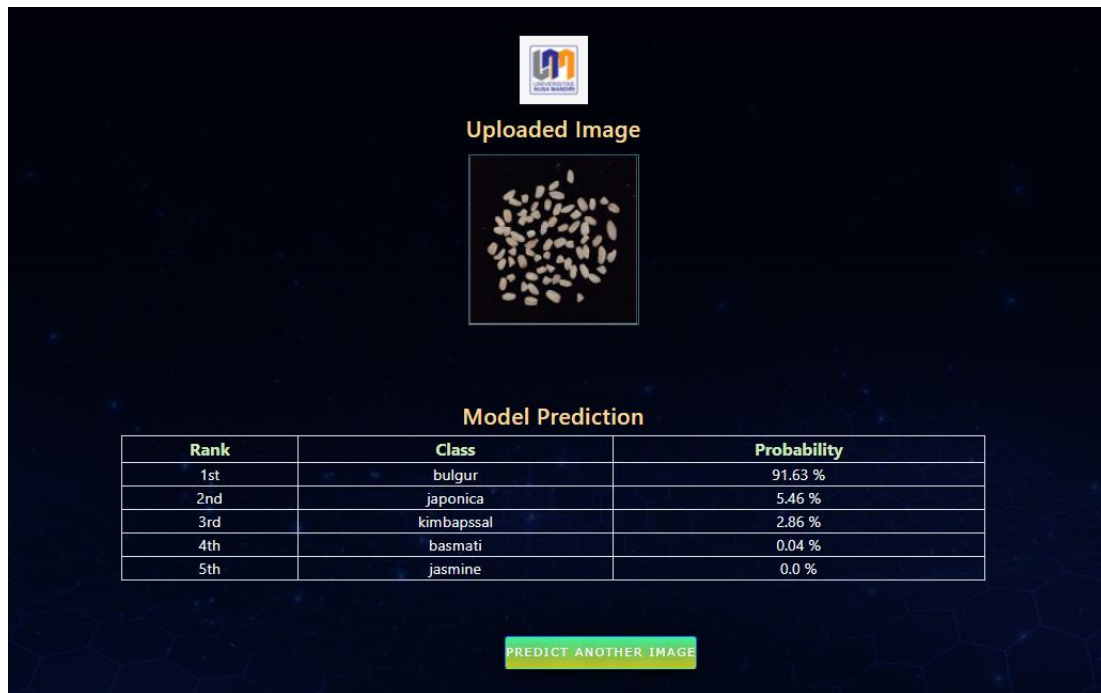


Sumber : Hasil penelitian

Gambar 4.8

Halaman *result* beras *Basmati*

Pada gambar 4.6. ditunjukkan hasil *result* dari citra *Basmati* yang dimasukkan ke dalam *website*. Dalam halaman ini ditampilkan citra yang telah diinput dan nilai *probability* yang dihasilkan.

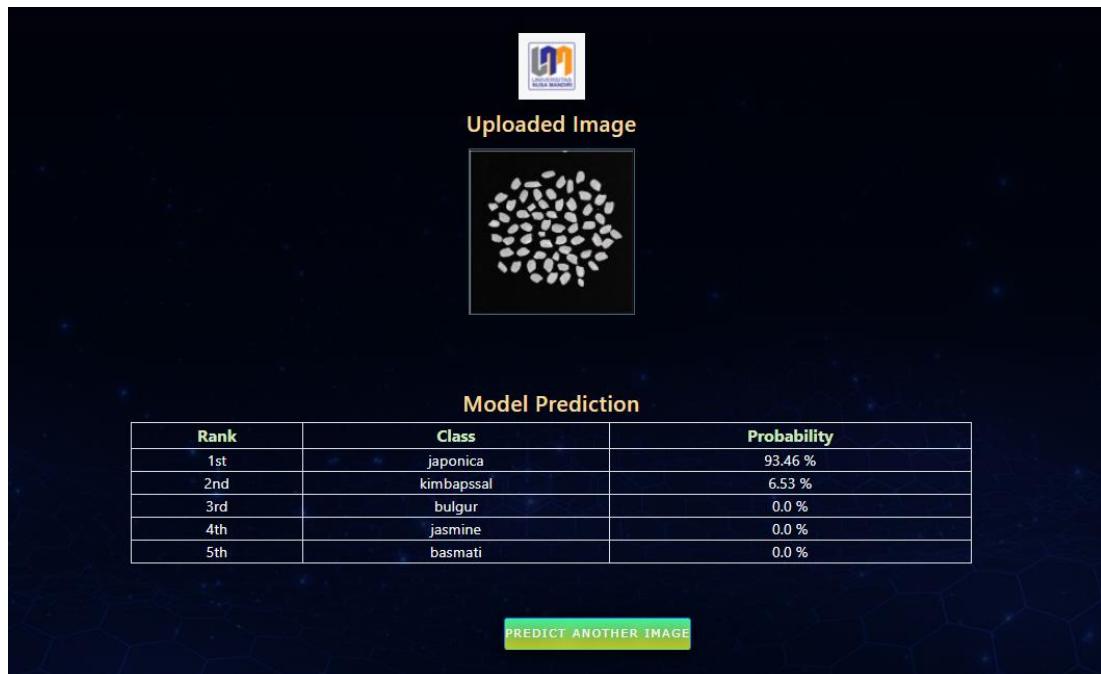


Sumber : Hasil penelitian

Gambar 4.9

Halaman *result* beras *Bulgur*

Pada gambar 4.7. ditunjukkan hasil *result* dari citra *Bulgur* yang dimasukkan ke dalam *website*. Dalam halaman ini ditampilkan citra yang telah diinput dan nilai *probability* yang dihasilkan.

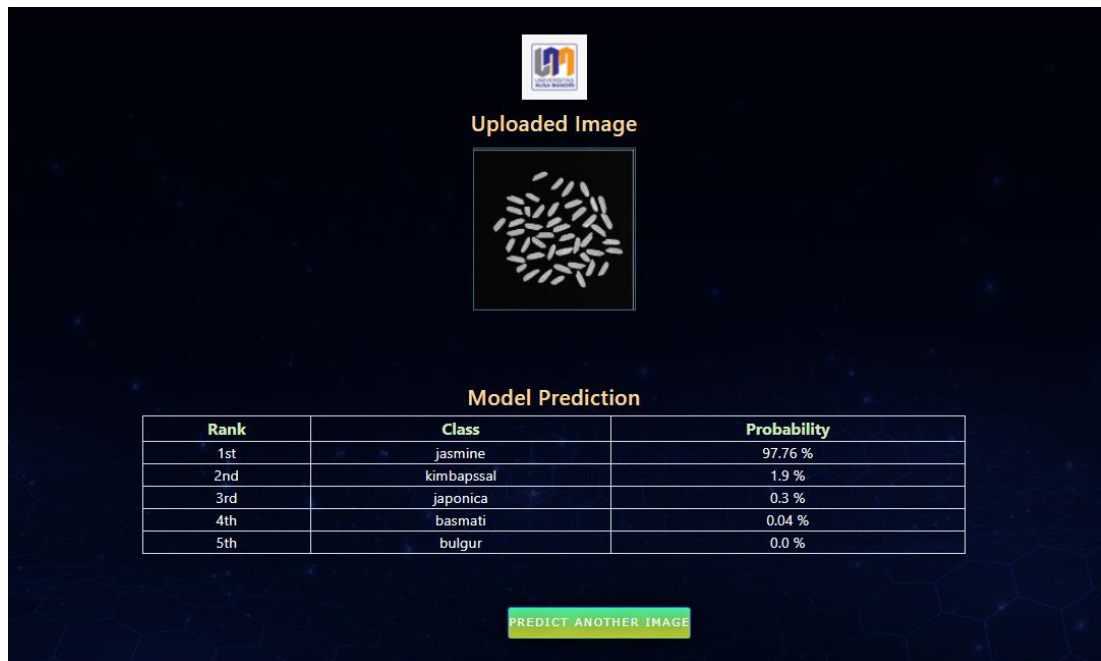


Sumber : Hasil penelitian

Gambar 4.10

Halaman *result* beras *Japonica*

Pada gambar 4.8. ditunjukkan hasil *result* dari citra *Japonica* yang dimasukkan ke dalam *website*. Dalam halaman ini ditampilkan citra yang telah diinput dan nilai *probability* yang dihasilkan.

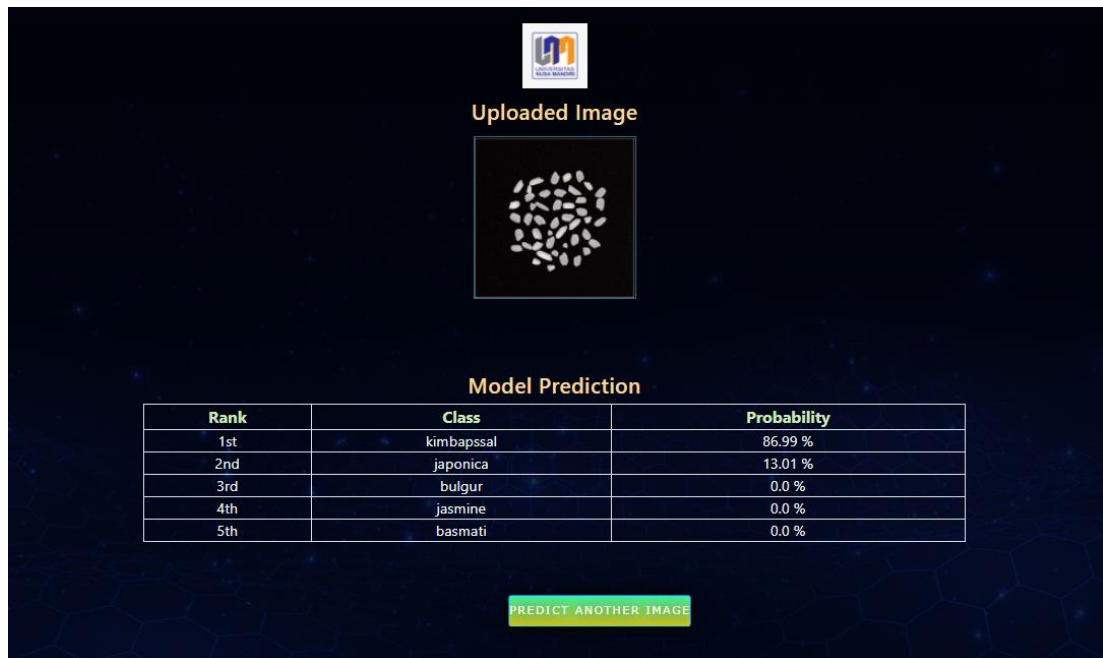


Sumber : Hasil penelitian

Gambar 4.11

Halaman *result* beras *Jasmine*

Pada gambar 4.9. ditunjukkan hasil *result* dari citra *Jasmine* yang dimasukkan ke dalam *website*. Dalam halaman ini ditampilkan citra yang telah *diinput* dan nilai *probability* yang dihasilkan.



Sumber : Hasil penelitian

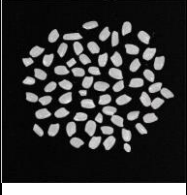
Gambar 4.12

Halaman *result* beras *Kimbapssal*

Pada gambar 4.10. ditunjukkan hasil *result* dari citra *Kimbapssal* yang dimasukkan ke dalam *website*. Dalam halaman ini ditampilkan citra yang telah diinput dan nilai *probability* yang dihasilkan.

Dari proses implementasi yang telah dilakukan, dapat diketahui nilai *probability* tiap kelas. Nilai *probability* yang paling besar menunjukkan bahwa citra termasuk dalam kelas tersebut. Berikut adalah hasil implementasi masing-masing citra setiap kelas:

Tabel 4.13 Nilai *probability* citra yang sesuai

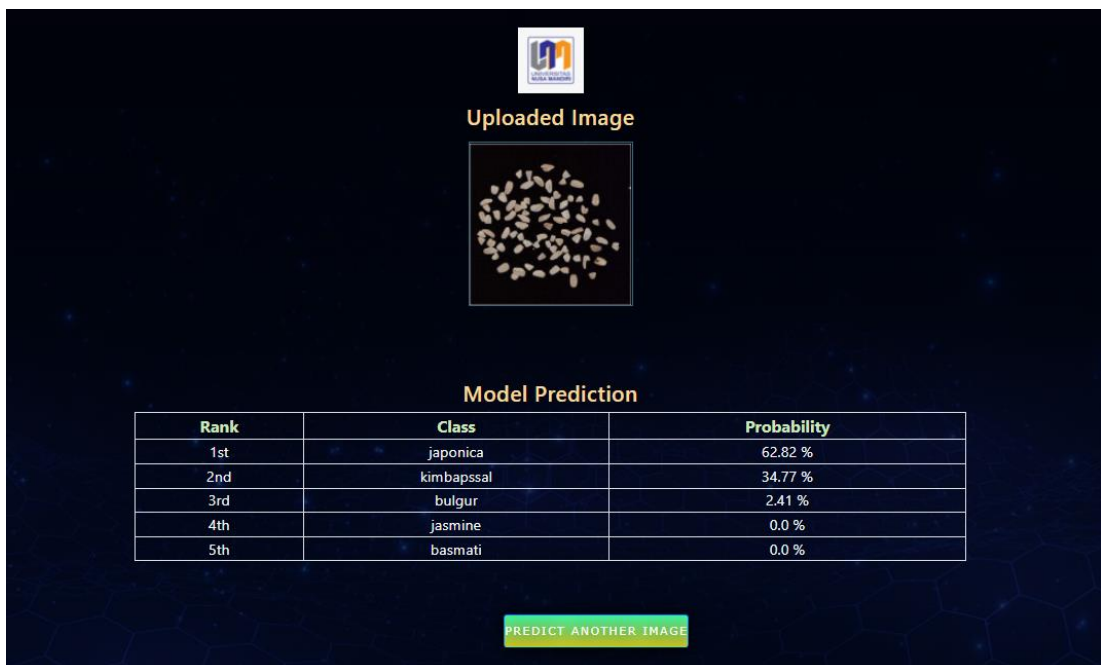
Sample Citra	Kelas Asli	Model Prediction					Kesimpulan
		<i>Basmati</i>	<i>Bulgur</i>	<i>Japonica</i>	<i>Jasmine</i>	<i>Kimhapssal</i>	
	<i>Basmati</i>	98.42%	0.01%	0.0%	1.27%	0.3%	Sesuai
	<i>Bulgur</i>	0.04 %	91.63 %	5.46 %	0.0%	2.86 %	Sesuai
	<i>Japonica</i>	0.0%	0.0%	93.46%	0.0%	6.53%	Sesuai
	<i>Jasmine</i>	0.04%	0.0%	0.3%	97.76%	1.9%	Sesuai
	<i>Kimhapssal</i>	0.0%	0.0%	13.01%	0.0%	86.99%	Sesuai

Sumber : Hasil penelitian

Nilai *probability* citra *Basmati* adalah 99.97% (*Basmati*), 0.02% (*Bulgur*), 0.0% (*Japonica*), 0.0% (*Jasmine*), 0.0% (*Kimhapssal*). Angka ini menunjukkan bahwa citra *Basmati* dapat diidentifikasi dengan benar. Nilai *probability* citra *Bulgur* adalah 0.04 % (*Basmati*), 91.63 % (*Bulgur*), 5.46 % (*Japonica*), 0.0%

(*Jasmine*), 2.86 % (*Kimbapssal*). Angka ini menunjukkan bahwa citra *Bulgur* dapat diidentifikasi dengan benar. Nilai *probability* citra *Japonica* adalah 0.0% (*Basmati*), 0.0% (*Bulgur*), 93.46% (*Japonica*), 0.0% (*Jasmine*), 6.53% (*Kimbapssal*). Angka ini menunjukkan bahwa citra *Japonica* dapat diidentifikasi dengan benar. Nilai *probability* citra *Jasmine* adalah 0.04% (*Basmati*), 0.0% (*Bulgur*), 0.3% (*Japonica*), 97.76% (*Jasmine*), 1.9% (*Kimbapssal*). Angka ini menunjukkan bahwa citra *Jasmine* dapat diidentifikasi dengan benar. Nilai *probability* citra *Kimbapssal* adalah 0.0% (*Basmati*), 0.0% (*Bulgur*), 13.01% (*Japonica*), 0.0% (*Jasmine*), 86.99% (*Kimbapssal*). Angka ini menunjukkan bahwa citra *Kimbapssal* dapat diidentifikasi dengan benar. Dari nilai *probability* yang dihasilkan, dapat disimpulkan bahwa model dapat mengidentifikasi citra dengan benar sesuai dengan kelas asli.

Dari penelitian yang dilakukan, terdapat citra yang memiliki nilai *probability* tidak sesuai dengan kelasnya. Berikut adalah halaman *result* untuk jenis beras yang tidak sesuai dengan kelasnya:



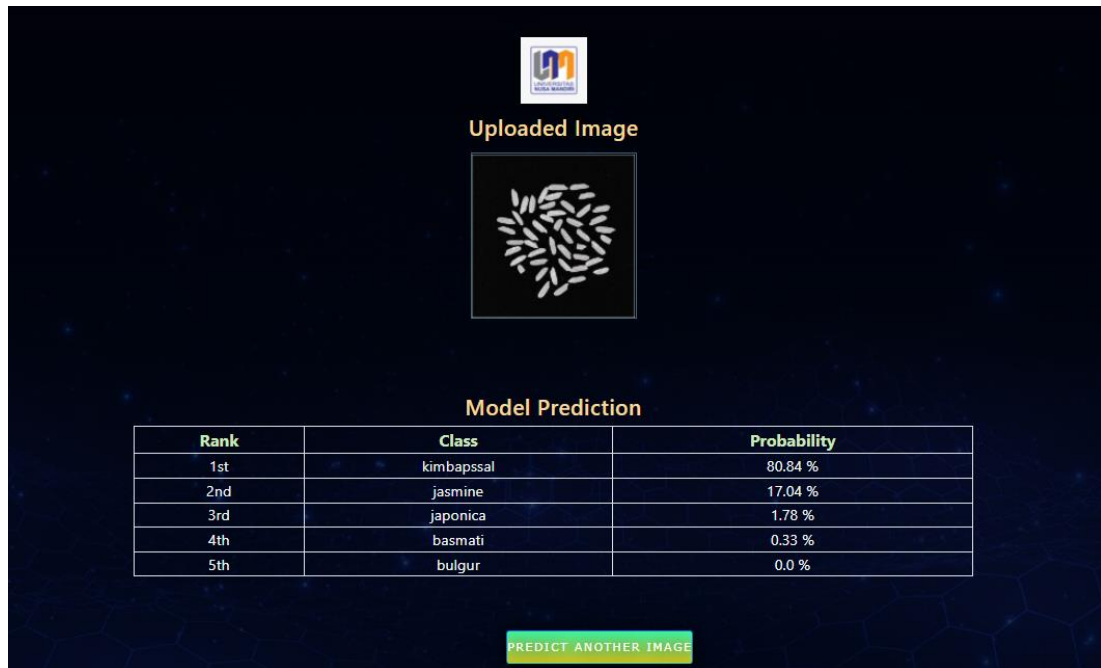
Rank	Class	Probability
1st	japonica	62.82 %
2nd	kimbapssal	34.77 %
3rd	bulgur	2.41 %
4th	jasmine	0.0 %
5th	basmati	0.0 %

Sumber : Hasil penelitian

Gambar 4.13

Halaman *result* beras *Bulgur* prediksi tidak sesuai kelas

Pada gambar 4.11. ditunjukkan hasil *result* dari citra *Basmati* yang dimasukkan ke dalam *website*. Dalam halaman ini ditampilkan citra yang telah diinput dan nilai *probability* yang dihasilkan.



Sumber : Hasil penelitian

Gambar 4.14

Halaman *result* beras *Jasmine* prediksi tidak sesuai kelas

Pada gambar 4.12. ditunjukkan hasil *result* dari citra *Jasmine* yang dimasukkan ke dalam *website*. Dalam halaman ini ditampilkan citra yang telah diinput dan nilai *probability* yang dihasilkan.

Citra yang memiliki nilai *probability* tidak sesuai dengan kelasnya ditunjukkan dalam Tabel 4.12.

Tabel 4.14 Nilai *probability* citra yang tidak sesuai

Sample Citra	Kelas Asli	Model Prediction					Kesimpulan
		<i>Basmati</i>	<i>Bulgur</i>	<i>Japonica</i>	<i>Jasmine</i>	<i>Kimbapssal</i>	
	<i>Bulgur</i>	0 %	2.41%	62.82 %	0.0 %	34.77 %	Tidak sesuai
	<i>Jasmine</i>	0.33 %	0.0 %	1.78 %	17.04 %	80.84 %	Tidak sesuai

Nilai *probability* citra *Bulgur* adalah 0% (*Basmati*), 2.41% (*Bulgur*), 62.82% (*Japonica*), 0.0% (*Jasmine*), 34.77% (*Kimbapssal*). Angka ini menunjukkan bahwa citra *Bulgur* tidak dapat diidentifikasi dengan benar. Nilai *probability* citra *Jasmine* adalah 0.33% (*Basmati*), 0.0% (*Bulgur*), 1.78% (*Japonica*), 17.04% (*Jasmine*), 80.84% (*Kimbapssal*). Angka ini menunjukkan bahwa citra *Jasmine* tidak dapat diidentifikasi dengan benar. Dari 5 jenis beras yang diprediksi, hanya 2 jenis beras saja yang ditemukan salah prediksi yaitu beras *Bulgur* dan beras *Jasmine*. Hal itu terjadi dikarenakan adanya kesamaan bentuk antara beras *Bulgur* dan beras *Japonica*, serta beras *Jasmine* dan beras *Kimbapssal*.

IV. PENUTUP

Pada bab ini akan dijelaskan mengenai kesimpulan dari hasil penelitian yang telah dilakukan dan beberapa saran yang dapat diaplikasikan untuk penelitian selanjutnya.

4.1. Kesimpulan

Berdasarkan permasalahan yang ada dan diidentifikasi sebelumnya, maka penulis dapat mengambil kesimpulan dari penelitian yang telah dilakukan bahwa:

1. Citra beras impor yang dihasilkan dalam penelitian ini adalah 180 citra yang terbagi dalam 5 kelas, yaitu 36 citra *Basmati*, 36 citra *Bulgur*, 36 citra *Japonica*, 36 citra *Jasmine*, 36 citra *Kimbapssal*. Jumlah citra ini lebih banyak dan beragam kelasnya dibanding dengan penelitian terdahulu [11].
2. Teknik *preprocessing* yang diterapkan pada proses klasifikasi citra beras yaitu *augmentation*, *resize* citra dan *split* data. Proses *augmentation* memiliki pengaruh terhadap hasil penelitian, yaitu membuat hasil penelitian tidak *overfitting*.
3. Klasifikasi citra beras menggunakan arsitektur *MobileNetV2* menghasilkan nilai akurasi sebesar 96,4% dan nilai *loss* 13,6%. Susunan *hyperparameter* yang digunakan yaitu *optimizer Adam*, ukuran *batch size* 32, jumlah *epoch* 10. Hasil klasifikasi dalam penelitian ini lebih baik dibanding dengan penelitian terdahulu [11].
4. Hasil penelitian diimplementasikan dalam bentuk *website*. Dalam *website* yang dibangun, pengguna dapat mengidentifikasi citra beras *Basmati*, *Bulgur*, *Japonica*, *Jasmine* dan *Kimbapssal*. Tahapan yang dilakukan, yaitu pengguna memasukkan citra beras. Kemudian citra tersebut akan diprediksi sesuai kelasnya berdasarkan nilai *probability* yang dihasilkan.

4.2. Saran

Berikut adalah beberapa saran yang dapat diaplikasikan untuk penelitian selanjutnya agar menghasilkan hasil penelitian yang lebih baik:

1. Dalam penelitian selanjutnya diharapkan dapat menambah jenis beras sebagai *dataset* penelitian, mengingat masih banyak jenis beras yang ada.
2. Menambah teknik *preprocessing* citra lainnya, mengingat masih banyak teknik *preprocessing* yang ada.
3. Menggunakan *self* arsitektur dan arsitektur *CNN* lainnya dalam proses klasifikasi jenis beras impor untuk menghasilkan nilai akurasi terbaik.
4. Mengimplementasikan hasil penelitian dalam bentuk aplikasi *mobile* sehingga lebih *user friendly*.

DAFTAR PUSTAKA

- [1] F. Abadi and H. Herwin, "Pengaruh Harga, Kualitas Produk Dan Kualitas Pelayanan Terhadap Keputusan Pembelian Beras Organik Di Jakarta," *J. Ris. Manaj. dan Bisnis Fak. Ekon. UNIAT*, vol. 4, no. 1, pp. 1–8, 2019.
- [2] Wahyuningsih, "Pusat Data dan Sistem Informasi Pertanian Sekretariat Jenderal Kementerian Pertanian," *Bul. Konsumsi Pangan*, vol. 09, no. 01, pp. 32–42, 2019.
- [3] S. E. Rahayu and H. Febriaty, "Analisis Perkembangan Produksi Beras," *Proseding Semin. Nas. Kewirausahaan*, vol. 1, no. 1, pp. 219–226, 2019.
- [4] Badan Pusat Statistik, "BPS 2019-Data import beras." Badan Pusat Statistik, Indonesia, p. 1, 2019.
- [5] Hermanto dan Saptana, "KEBIJAKAN HARGA BERAS DITINJAU DARI DIMENSI PENENTU HARGA Rice Price Policy Reviewed from the Dimensions of Price Diterminations," *Forum Penelit. Agro Ekon. Vol. 35 No. 1, Juli 2017 31-43*, vol. 35, no. 1, pp. 31–43, 2017.
- [6] A. W. Putri, S. JK, and A. N. Rahmanto, "KOMUNIKASI KRISIS KEMENTERIAN PERTANIAN PADA KASUS PENGGEREBEKAN GUDANG BERAS PT IBU (Analisis Isi Kualitatif Menggunakan Situational Crisis Communication Theory)," *J. Stud. Komun. dan Media*, vol. 23, no. 1, p. 53, 2019.
- [7] Inne Christina, "PENGOPLOSAN BERAS Inne Christina Ermanto Fahamsyah A . Latar Belakang Kegiatan perekonomian di era globalisasi pada sekarang ini berkembang semakin pesat . Kegiatan perekonomian berkembang seiring dengan dengan populasi masyarakat yang semakin meningkat .," *J. Huk. Adigama*, no. 20, 2018.
- [8] D. Ricardo and G. Gasim, "Perbandingan Akurasi Pengenalan Jenis Beras dengan Algoritma Propagasi Balik pada Beberapa Resolusi Kamera," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 3, no. 2, pp. 131–140, 2019.
- [9] D. Mohan and M. Gopal Raj, "Quality analysis of rice grains using ANN and SVM," *J. Crit. Rev.*, vol. 7, no. 1, pp. 395–402, 2020.
- [10] R. Surekha, R. Shobarani, G. V. S. George, and C. Tamilselvi, "Rice Breed Classification with a Hybrid Segmentation Method using Image processing," vol. 29, no. 8, pp. 2997–3005, 2020.
- [11] A. Ridan Nurfalah, Dwiza Riana, "Identifikasi Citra Beras Menggunakan Algoritma Multi-SVM Dan Neural," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 1, no. 10, 2021.
- [12] W. Srimulyani and A. Musdholifah, "Identification of Rice Variety Using

- Geometric Features and Neural Network,” *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 13, no. 3, p. 301, 2019.
- [13] Gansar Suwanto, R. Ibnu Adam, and Garno, “Identifikasi Citra Digital Jenis Beras Menggunakan Metode Anfis dan Sobel,” *J. Inform. Polinema*, vol. 7, no. 2, pp. 123–128, 2021.
 - [14] A. A. Aznan, R. Ruslan, I. H. Rukunudin, F. A. Azizan, and A. Y. Hashim, “Rice seed varieties identification based on extracted colour features using image processing and artificial neural network (ANN),” *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 7, no. 6, pp. 2220–2225, 2017.
 - [15] D. Aprilia, ; Jajam, H. Jaman, ; Riza, and I. Adam, “Application of Backpropagation Neural Network Algorithm for Ciherang Rice Image Identification,” pp. 141–148, 2019.
 - [16] Lilik Sumaryanti, “ANALISIS CITRA DIGITAL UNTUK KLASIFIKASI KUALITAS BERAS,” *J. Ilm. Mustek*, vol. 151, no. 2, pp. 10–17, 2018.
 - [17] P. Dheer and R. K. Singh, “Identification of Indian rice varieties using machine learning classifiers,” *Plant Arch.*, vol. 19, no. 1, pp. 155–158, 2019.
 - [18] B. Arora, N. Bhagat, L. Saritha, and S. Arcot, “Rice Grain Classification using Image Processing Machine Learning Techniques,” *Proc. 5th Int. Conf. Inven. Comput. Technol. ICICT 2020*, pp. 205–208, 2020.
 - [19] N. Hong Son and N. Thai-Nghe, “Deep Learning for Rice Quality Classification,” in *Proceedings - 2019 International Conference on Advanced Computing and Applications, ACOMP 2019*, 2019, pp. 92–96.
 - [20] W. Dai, Y. Dai, K. Hirota, and Z. Jia, “A Flower Classification Approach with MobileNetV2 and Transfer Learning,” in *The 9th International Symposium on Computational Intelligence and Industrial Applications*, 2020, pp. 1–5.
 - [21] A. S. dan S. W. Prisky Ratna Aningtiyas, “Pembuatan Aplikasi Deteksi Objek Menggunakan TensorFlow Object Detection API dengan Memanfaatkan SSD MobileNet V2 Sebagai Model Pra - Terlatih,” *J. Ilm. Komputasi*, vol. 19, no. 3, pp. 421–430, 2020.
 - [22] D. Rohpandi, A. Sugiharto, and M. Y. S. Jati, “Klasifikasi Citra Digital Berbasis Ekstraksi Ciri Berdasarkan Tekstur Menggunakan GLCM Dengan Algoritma K-Nearest Neighbor,” *J. Inform.*, vol. 3, no. 2, pp. 79–86, 2017.
 - [23] F. Riandari, “Implementasi Metode Geometric Mean Filter Untuk Perbaikan Dengan Reduksi Noise Pada Citra Digital,” *J. Mantik Penusa*, vol. 2, no. 2, pp. 175–179, 2018.
 - [24] T. Suparwati, “Perbaikan Citra Menggunakan Metode Contrast Stretching,” *J. Siger Mat.*, vol. Vol. 02 No, no. 2721–6853, pp. 13–18, 2021.
 - [25] Suyanto, *Machine Learning Tingkat Dasar dan Lanjut*. Bandung:

Informatika Bandung, 2018.

- [26] B. Purnama, *Pengantar Machine Learning*, 1st ed. Bandung: Informatika Bandung, 2019.
- [27] P. S. C. Moonallika, K. Q. Fredlina, and I. B. K. Sudiatmika, “Penerapan Data Mining Untuk Memprediksi Kelulusan Mahasiswa Menggunakan Algoritma Naive Bayes Classifier (Studi Kasus STMIK Primakara),” *J. Ilm. Komput.*, vol. 6, no. 1, pp. 47–56, 2020.
- [28] E. Y. Purwani and I. P. Wardana, “Karakteristik Fisiko-kimia Varietas Beras Khusus untuk Pangan Inovatif,” *J. Penelit. Pertan. Tanam. Pangan*, vol. 2, no. 3, p. 165, 2019.
- [29] G. Mahajan, A. Matloob, R. Singh, V. P. Singh, and B. S. Chauhan, “Basmati Rice in the Indian Subcontinent: Strategies to Boost Production and Quality Traits,” *Adv. Agron.*, vol. 151, pp. 159–213, 2018.
- [30] Wisnu Adi Yulianto, *Teknologi Pengolahan Beras Pratanak*. Dee Publish, 2021.
- [31] D. V. Panjaitan, S. R. Ingot, A. Mardiansyah, and L. M. Christoffel, “Analisis Dampak Export Restriction Komoditi Pertanian Terhadap Indonesia,” *J. Agribisnis Indones.*, vol. 8, no. 1, pp. 1–16, 2020.
- [32] M. P. Yuliansari Arita, “Proses Pembuatan Masker Bunga Rosella Dan Tepung Beras Sebagai Pencerahan Kulit Wajah,” *J. Tata Rias*, vol. 09, no. Vol 9, No 2 (2020), pp. 367–376, 2020.
- [33] A. H. Hasugian and I. Zufria, “Perancangan Sistem Restorasi Citra Dengan Metode Image Inpainting,” *Algoritma. J. Ilmu Komput. dan Inform.*, vol. 6341, no. November, p. 1, 2018.
- [34] C. D. Sindua, V. C. Poekoel, P. D. K. Manembu, T. Elektro, and U. Sam, “Monitoring dan Akuisisi Data Sistem Pertanian Pintar Berbasis Web,” *J. Tek. Elektro dan Komput.*, vol. 9, no. 2, pp. 61–72, 2020.
- [35] R. R. Rerung, “Penerapan Data Mining dengan Memanfaatkan Metode Association Rule untuk Promosi Produk,” *J. Teknol. Rekayasa*, vol. 3, no. 1, p. 89, 2018.
- [36] F. T. A. Ridho Aji Pangestu, Basuki Rahmat, “Implementasi algoritma cnn untuk klasifikasi citra lahan dan perhitungan luas,” *J. Inform. dan Sist. Inf.*, vol. 1, no. 1, pp. 166–174, 2020.
- [37] J. Sanjaya and M. Ayub, “Augmentasi Data Pengenalan Citra Mobil Menggunakan Pendekatan Random Crop, Rotate, dan Mixup,” *J. Tek. Inform. dan Sist. Inf.*, vol. 6, no. 2, pp. 311–323, 2020.
- [38] Y. D. Pramudita, S. S. Putro, and N. Makhmud, “Klasifikasi Berita Olahraga Menggunakan Metode Naïve Bayes dengan Enhanced Confix Stripping Stemmer,” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 3, p. 269, 2018.

- [39] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Comput. Electron. Agric.*, vol. 147, no. July 2017, pp. 70–90, 2018.
- [40] R. J. S. Raj, S. J. Shobana, I. V. Pustokhina, D. A. Pustokhin, D. Gupta, and K. Shankar, "Optimal feature selection-based medical image classification using deep learning model in internet of medical things," *IEEE Access*, vol. 8, pp. 58006–58017, 2020.
- [41] S. Ilahiyah and A. Nilogiri, "Implementasi Deep Learning Pada Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun Menggunakan Convolutional Neural Network," *JUSTINDO (Jurnal Sist. dan Teknol. Inf. Indones.)*, vol. 3, no. 2, pp. 49–56, 2018.
- [42] M. Z. Alom *et al.*, "The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches," 2018.
- [43] H. L. G. Tianmei Guo, Jiwen Dong, "Simple Convolutional Neural Network on Image Classification," *IEEE 2nd Int. Conf. Big Data Anal.*, pp. 721–724, 2017.
- [44] A. T. W. M. Raihan Rafiiful Allaam, "KLASIFIKASI GENUS TANAMAN ANGGREK MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN)," in *e-Proceeding of Engineering*, 2021, vol. 8, no. 2, pp. 3147–3179.
- [45] S. M. Suyanto, Kurniawan Nur Ramadhani, *Deep Learning Modernisasi Mchine Learning untuk Big Data*. Bandung: Penerbit Informatika, 2019.
- [46] M. R. Alwanda, R. Putra, K. Ramadhan, D. Alamsyah, P. Studi, and T. Informatika, "Implementasi Metode Convolutional Neural Network Menggunakan Arsitektur LeNet-5 untuk Pengenalan Doodle," *J. Algoritm.*, vol. 1, no. 1, 2020.
- [47] I. Cholissodin, Sutrisno, A. A. Soebroto, U. Hasanah, and Y. I. Febiola, *AI , MACHINE LEARNING & DEEP LEARNING (Teori & Implementasi)*, Versi 1.01., no. July 2019. Malang, 2021.
- [48] Jan Wira Gotama Putra, *Pengenalan Konsep Pembelajaran Mesin dan Deep Learning*, 1.4., vol. 4. 2020.
- [49] Hendriyana and Y. H. Maulana, "Identifikasi Jenis Kayu menggunakan Convolutional Neural Network dengan Arsitektur Mobilenet," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 4, no. 1, pp. 70–76, 2020.
- [50] P. N. Srinivasu, J. G. Sivasai, M. F. Ijaz, A. K. Bhoi, W. Kim, and J. J. Kang, "Classification of Skin Disease Using Deep Learning Neural Networks with MobileNet V2 and LSTM," *sensors*, pp. 1–27, 2021.
- [51] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 4510–4520, 2018.

- [52] A. Souid, N. Sakli, and H. Sakli, "Classification and predictions of lung diseases from chest x- rays using mobilenet v2," *Appl. Sci.*, vol. 11, no. 6, 2021.
- [53] K. Kuan, G. Manek, J. Lin, Y. Fang, and V. Chandrasekhar, "Region average pooling for context-aware object detection," in *Proceedings - International Conference on Image Processing, ICIP*, 2018, vol. 2017-Septe, pp. 1347–1351.
- [54] J. Brownlee, *Deep Learning for Natural Language Processing*. 2017.
- [55] Nikhil Ketkar, *Deep Learning with Python*. Bangalore, India: Apress, 2017.
- [56] A. P. Adedigba, S. A. Adeshina, O. E. Aina, and A. M. Aibinu, "Optimal hyperparameter selection of deep learning models for COVID-19 chest X-ray classificatio," *Intell. Med.*, vol. 5, no. January, 2021.
- [57] H. Pan, Z. Pang, Y. Wang, Y. Wang, and L. Chen, "A New Image Recognition and Classification Method Combining Transfer Learning Algorithm and MobileNet Model for Welding Defects," *IEEE Access*, vol. 8, pp. 119951–119960, 2020.
- [58] Python Software Foundation, "Python," 2021. [Online]. Available: <https://www.python.org/>. [Accessed: 23-Jun-2021].
- [59] R. Irsyad, "Penggunaan Python Web Framework Flask Untuk Pemula," 2018.
- [60] T. Carneiro, R. V. M. Da Nobrega, T. Nepomuceno, G. Bin Bian, V. H. C. De Albuquerque, and P. P. R. Filho, "Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications," *IEEE Access*, vol. 6, pp. 61677–61685, 2018.
- [61] F. CHOLLET, *Deep Learning With Python*. 2018.
- [62] R. S. Eko Supriyadi, Achmad Basuki, "DETEKSI KUALITAS BERAS MENGGUNAKAN SEGMENTASI CITRA BERDASARKAN PECAHAN BULIR DAN SEBARAN WARNA," *JIRE (Jurnal Inform. Rekayasa Elektronik.*, vol. 3, no. 1, pp. 20–29, 2020.
- [63] N. Nagoda and L. Ranathunga, "Rice Sample Segmentation and Classification Using Image Processing and Support Vector Machine," in *2018 13th International Conference on Industrial and Information Systems, ICIIS 2018 - Proceedings*, 2018, no. 978, pp. 179–184.
- [64] G. Geetharamani and A. P. J., "Identification of plant leaf diseases using a nine-layer deep convolutional neural network," *Comput. Electr. Eng.*, vol. 76, pp. 323–338, 2019.
- [65] Q. Wu, Y. Chen, and J. Meng, "Dcgan-based data augmentation for tomato leaf disease identification," *IEEE Access*, vol. 8, pp. 98716–98728, 2020.

