

**LAPORAN RESMI FINAL PROJECT KECERDASAN BUATAN
PROGRAM SISTEM KLASIFIKASI GAMBAR PEMANDANGAN
MENGUNAKAN CONVOLUTIONAL NEURAL NETWORK (CNN)**



DOSEN PENGAMPU :

Dr. Basuki Rahmat, S.Si, MT

DISUSUN OLEH :

Sabrina Putri Aulia	21081010048
Churri Asna Fatchiyah	21081010156

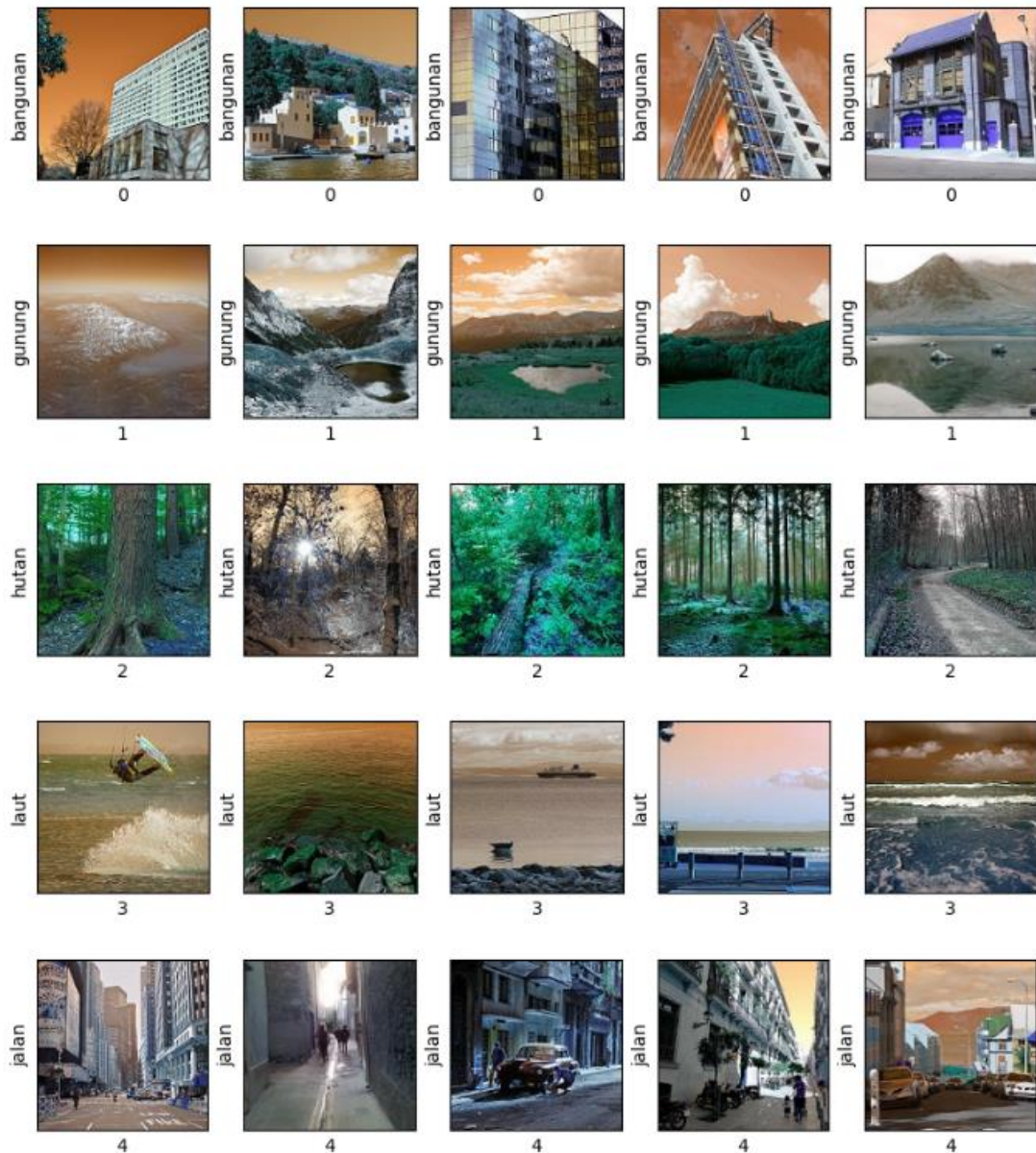
**PROGRAM STUDI INFORMATIKA FAKULTAS ILMU KOMPUTER
UNIVERSITAS PEMBANGUNAN NASIONAL "VETERAN" JAWA TIMUR**

2022

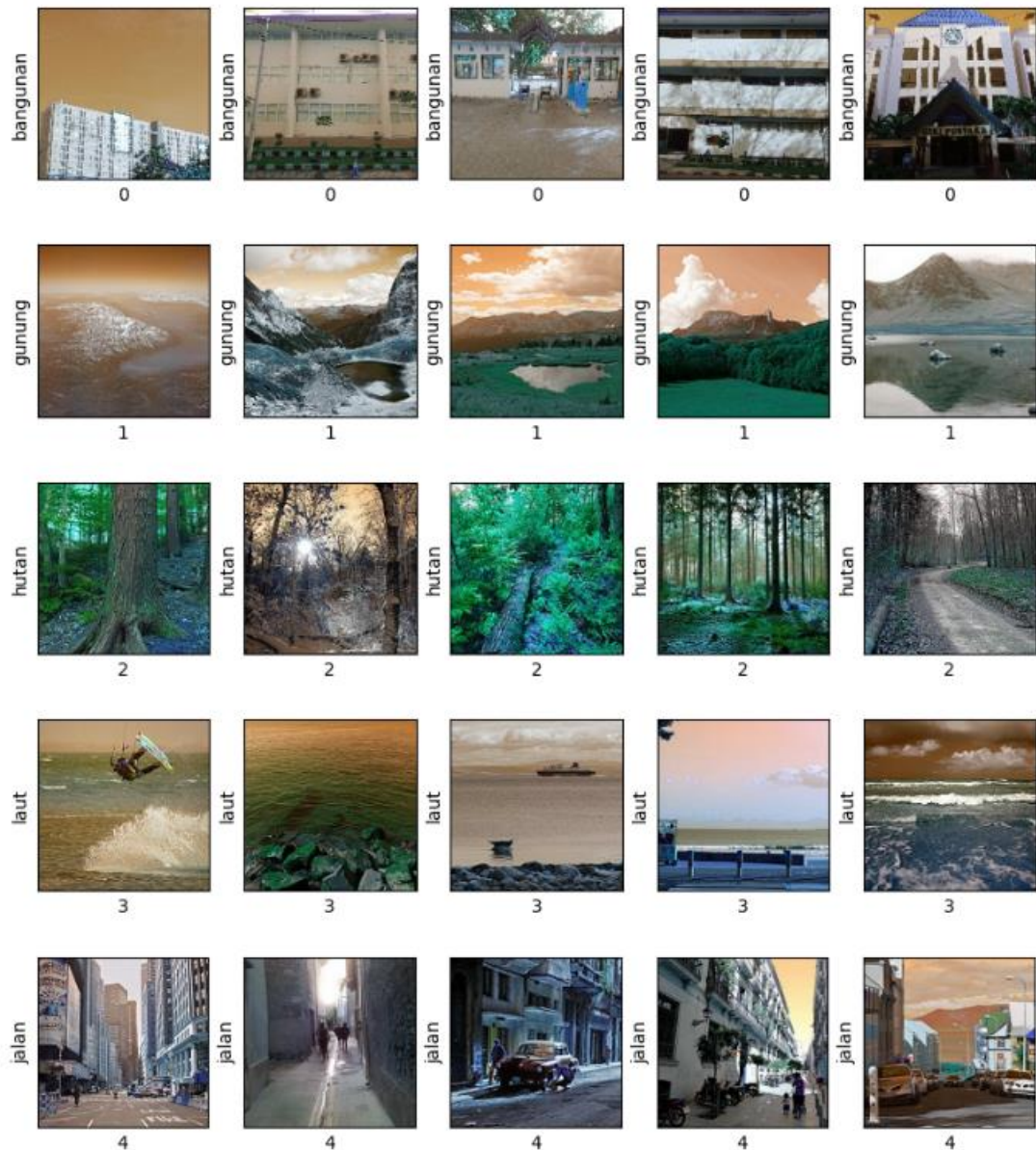
A. DATASET

Dataset yang akan digunakan adalah 250 data, yang digunakan untuk proses pelatihan (Training Data) sebanyak 200 data, dan pengujian (Test Data) sebanyak 50 data. Data tersebut dibagi rata untuk 5 (lima) jenis gambar pemandangan. Jadi untuk setiap jenis gambar pemandangan, terdapat 40 Data Latih dan 10 Data Uji.

Dataset 1

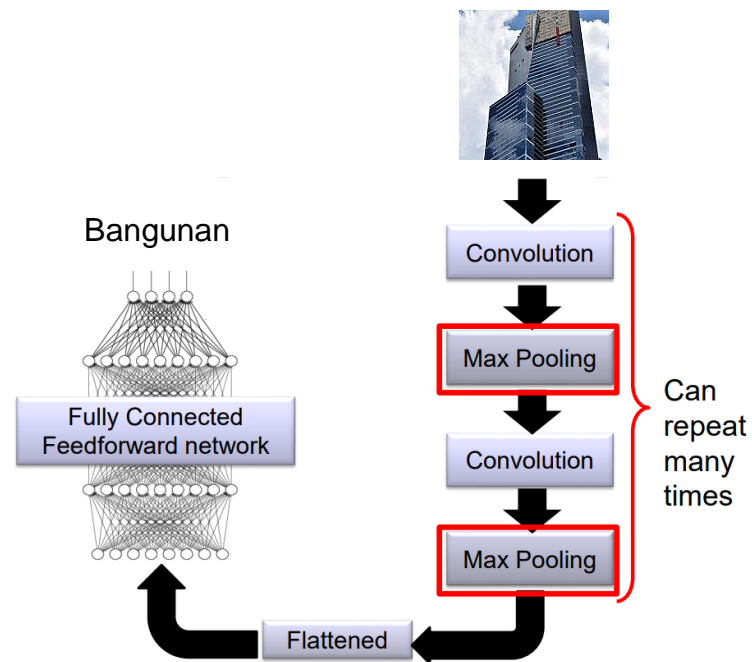


Dataset 2



Kami melakukan dua kali pelatihan dan pengujian untuk klasifikasi gambar pemandangan. Dalam kedua pelatihan dan pengujian tersebut, terdapat perbedaan substansial pada data uji/testing sub kategori bangunan. Pada pengklasifikasian pertama, kami menggunakan data uji sub bangunan yang diambil dari sumber Kaggle, sementara pada pengklasifikasian kedua, kami menggunakan data uji sub bangunan yang didapat dari UPN Veteran Jawa Timur.

B. CNN ARCHITECTURE



Model: "sequential_1"

Layer (type)	Output Shape	Param #
conv2d_5 (Conv2D)	(None, 148, 148, 100)	2800
max_pooling2d_4 (MaxPooling 2D)	(None, 74, 74, 100)	0
conv2d_6 (Conv2D)	(None, 72, 72, 200)	180200
max_pooling2d_5 (MaxPooling 2D)	(None, 36, 36, 200)	0
conv2d_7 (Conv2D)	(None, 34, 34, 200)	360200
max_pooling2d_6 (MaxPooling 2D)	(None, 17, 17, 200)	0
conv2d_8 (Conv2D)	(None, 15, 15, 200)	360200
max_pooling2d_7 (MaxPooling 2D)	(None, 7, 7, 200)	0
conv2d_9 (Conv2D)	(None, 5, 5, 200)	360200
flatten_1 (Flatten)	(None, 5000)	0
dense_2 (Dense)	(None, 200)	1000200
dense_3 (Dense)	(None, 5)	1005

=====
 Total params: 2,264,805
 Trainable params: 2,264,805
 Non-trainable params: 0

C. TRAINING

Training 1

Hasil proses pelatihan CNN dan akurasi validasi dengan Data Uji sebagai berikut:

```
Epoch 1/20
7/7 [=====] - 18s 2s/step - loss: 1.6437 - accuracy: 0.1900 - val_loss: 1.6042 - val_accuracy: 0.2000
Epoch 2/20
7/7 [=====] - 20s 3s/step - loss: 1.6076 - accuracy: 0.2000 - val_loss: 1.6047 - val_accuracy: 0.2000
Epoch 3/20
7/7 [=====] - 21s 3s/step - loss: 1.6002 - accuracy: 0.2000 - val_loss: 1.5682 - val_accuracy: 0.2000
Epoch 4/20
7/7 [=====] - 17s 2s/step - loss: 1.5658 - accuracy: 0.2850 - val_loss: 1.4179 - val_accuracy: 0.4400
Epoch 5/20
7/7 [=====] - 22s 3s/step - loss: 1.3831 - accuracy: 0.4100 - val_loss: 1.3781 - val_accuracy: 0.4000
Epoch 6/20
7/7 [=====] - 20s 3s/step - loss: 1.3669 - accuracy: 0.4400 - val_loss: 1.2090 - val_accuracy: 0.6000
Epoch 7/20
7/7 [=====] - 17s 2s/step - loss: 1.2945 - accuracy: 0.4250 - val_loss: 1.0951 - val_accuracy: 0.5200
Epoch 8/20
7/7 [=====] - 17s 2s/step - loss: 1.1322 - accuracy: 0.5200 - val_loss: 0.8615 - val_accuracy: 0.6400
Epoch 9/20
7/7 [=====] - 17s 2s/step - loss: 1.1128 - accuracy: 0.5250 - val_loss: 0.8919 - val_accuracy: 0.7000
Epoch 10/20
7/7 [=====] - 17s 2s/step - loss: 1.0513 - accuracy: 0.6000 - val_loss: 0.7483 - val_accuracy: 0.7400
Epoch 11/20
7/7 [=====] - 18s 3s/step - loss: 1.0085 - accuracy: 0.5800 - val_loss: 0.7503 - val_accuracy: 0.6400
Epoch 12/20
7/7 [=====] - 17s 2s/step - loss: 1.0215 - accuracy: 0.5250 - val_loss: 0.7966 - val_accuracy: 0.5800
Epoch 13/20
7/7 [=====] - 18s 3s/step - loss: 0.9318 - accuracy: 0.6250 - val_loss: 0.7324 - val_accuracy: 0.8600
Epoch 14/20
7/7 [=====] - 17s 2s/step - loss: 0.8786 - accuracy: 0.6650 - val_loss: 0.7177 - val_accuracy: 0.7200
Epoch 15/20
7/7 [=====] - 16s 2s/step - loss: 0.8762 - accuracy: 0.6550 - val_loss: 0.8570 - val_accuracy: 0.6200
Epoch 16/20
7/7 [=====] - 16s 2s/step - loss: 1.0213 - accuracy: 0.5950 - val_loss: 0.6135 - val_accuracy: 0.8600
Epoch 17/20
7/7 [=====] - 17s 2s/step - loss: 0.8501 - accuracy: 0.7000 - val_loss: 0.7413 - val_accuracy: 0.6800
Epoch 18/20
7/7 [=====] - 17s 2s/step - loss: 0.7680 - accuracy: 0.6850 - val_loss: 0.4958 - val_accuracy: 0.7600
Epoch 19/20
7/7 [=====] - 17s 2s/step - loss: 0.7076 - accuracy: 0.7600 - val_loss: 0.5016 - val_accuracy: 0.7800
Epoch 20/20
7/7 [=====] - 17s 2s/step - loss: 0.8478 - accuracy: 0.7000 - val_loss: 0.6782 - val_accuracy: 0.8600
```

Training 2

Hasil proses pelatihan CNN dan akurasi validasi dengan Data Uji sebagai berikut:

```
Epoch 1/20
7/7 [=====] - 157s 21s/step - loss: 1.7266 - accuracy: 0.2050 - val_loss: 1.6176 - val_accuracy: 0.2000
Epoch 2/20
7/7 [=====] - 144s 21s/step - loss: 1.6149 - accuracy: 0.2000 - val_loss: 1.6013 - val_accuracy: 0.2000
Epoch 3/20
7/7 [=====] - 150s 22s/step - loss: 1.5799 - accuracy: 0.2600 - val_loss: 1.4000 - val_accuracy: 0.4800
Epoch 4/20
7/7 [=====] - 155s 22s/step - loss: 1.5110 - accuracy: 0.3750 - val_loss: 1.4271 - val_accuracy: 0.4200
Epoch 5/20
7/7 [=====] - 137s 19s/step - loss: 1.3616 - accuracy: 0.4650 - val_loss: 1.3345 - val_accuracy: 0.4400
Epoch 6/20
7/7 [=====] - 135s 19s/step - loss: 1.2233 - accuracy: 0.4850 - val_loss: 1.3142 - val_accuracy: 0.4600
Epoch 7/20
7/7 [=====] - 135s 19s/step - loss: 1.2301 - accuracy: 0.4550 - val_loss: 1.1508 - val_accuracy: 0.5600
Epoch 8/20
7/7 [=====] - 136s 19s/step - loss: 1.0797 - accuracy: 0.5350 - val_loss: 1.0930 - val_accuracy: 0.5200
Epoch 9/20
7/7 [=====] - 135s 19s/step - loss: 0.9412 - accuracy: 0.6050 - val_loss: 1.3044 - val_accuracy: 0.5000
Epoch 10/20
7/7 [=====] - 134s 19s/step - loss: 0.9255 - accuracy: 0.5950 - val_loss: 1.0702 - val_accuracy: 0.6200
Epoch 11/20
7/7 [=====] - 134s 19s/step - loss: 0.9143 - accuracy: 0.6650 - val_loss: 1.1008 - val_accuracy: 0.5600
Epoch 12/20
7/7 [=====] - 133s 19s/step - loss: 0.9859 - accuracy: 0.5750 - val_loss: 1.1570 - val_accuracy: 0.6000
Epoch 13/20
7/7 [=====] - 133s 19s/step - loss: 0.8768 - accuracy: 0.6250 - val_loss: 1.0653 - val_accuracy: 0.6000
Epoch 14/20
7/7 [=====] - 132s 19s/step - loss: 0.8899 - accuracy: 0.6300 - val_loss: 1.2754 - val_accuracy: 0.5200
```

```




Epoch 15/20
7/7 [=====] - 135s 19s/step - loss: 0.8550 - accuracy: 0.6550 - val_loss: 1.0604 - val_accuracy: 0.580
0
Epoch 16/20
7/7 [=====] - 140s 20s/step - loss: 0.7356 - accuracy: 0.6750 - val_loss: 1.4206 - val_accuracy: 0.540
0
Epoch 17/20
7/7 [=====] - 139s 20s/step - loss: 0.8430 - accuracy: 0.6550 - val_loss: 0.9959 - val_accuracy: 0.580
0
Epoch 18/20
7/7 [=====] - 135s 19s/step - loss: 0.6757 - accuracy: 0.7400 - val_loss: 1.1221 - val_accuracy: 0.580
0
Epoch 19/20
7/7 [=====] - 133s 19s/step - loss: 0.6125 - accuracy: 0.7500 - val_loss: 1.1286 - val_accuracy: 0.580
0
Epoch 20/20
7/7 [=====] - 132s 19s/step - loss: 0.6484 - accuracy: 0.7400 - val_loss: 1.0861 - val_accuracy: 0.600
0



```

D. TESTING

Testing 1



Hasil dari Klasifikasi gambar pemandangan menggunakan CNN sebagai berikut:




No	Gambar Pemandangan		Hasil Prediksi Benar	Hasil Prediksi Salah	Akurasi (Persentasi Keberhasilan) (%)
	Gambar	Jenis			
1		Bangunan	8	2	80
2		Gunung	9	1	90
3		Hutan	10	0	100

4		Laut	8	2	80
5		Jalan	8	2	80
Rata - rata					86

Testing 2

Hasil dari Klasifikasi gambar pemandangan menggunakan CNN sebagai berikut:

No	Gambar Pemandangan		Hasil Prediksi Benar	Hasil Prediksi Salah	Akurasi (Persentasi Keberhasilan) (%)
	Gambar	Jenis			
1		Bangunan	5	5	50
2		Gunung	9	1	90

3		Hutan	8	2	80
4		Laut	2	8	20
5		Jalan	6	4	60
Rata - rata					60

E. KESIMPULAN

Klasifikasi dan pengenalan objek pemandangan merupakan proses untuk mengklasifikasikan gambar pemandangan ke dalam kategori tertentu berdasarkan karakteristik visualnya. Penelitian ini menggunakan metode Convolutional Neural Network (CNN) untuk mengidentifikasi dan mengklasifikasikan jenis pemandangan. Dari hasil percobaan yang dilakukan, diperoleh tingkat keakuratan sebesar 84 persen dalam pengujian klasifikasi gambar pemandangan pertama dan 60 persen dalam pengujian klasifikasi gambar pemandangan kedua. Perbedaan dalam menggunakan sumber data uji sub kategori bangunan yang digunakan memberikan variasi dan keberagaman dalam pengujian. Hal ini membantu kami memahami bagaimana sistem klasifikasi CNN bekerja dalam situasi yang berbeda.