



IDENTIFIKASI KEMATANGAN BUAH SIRSAK BERDASARKAN TEKSTUR DAN WARNA MENGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK

Disusun oleh:
Kharisma Nur'aisyah
NIM 18305141031

Dosen Pembimbing:
Dr. Sri Andayani, S.Si., M.Kom.
NIP 197204261997022001

**PROGRAM STUDI MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI YOGYAKARTA
2022**



Buah Sirsak

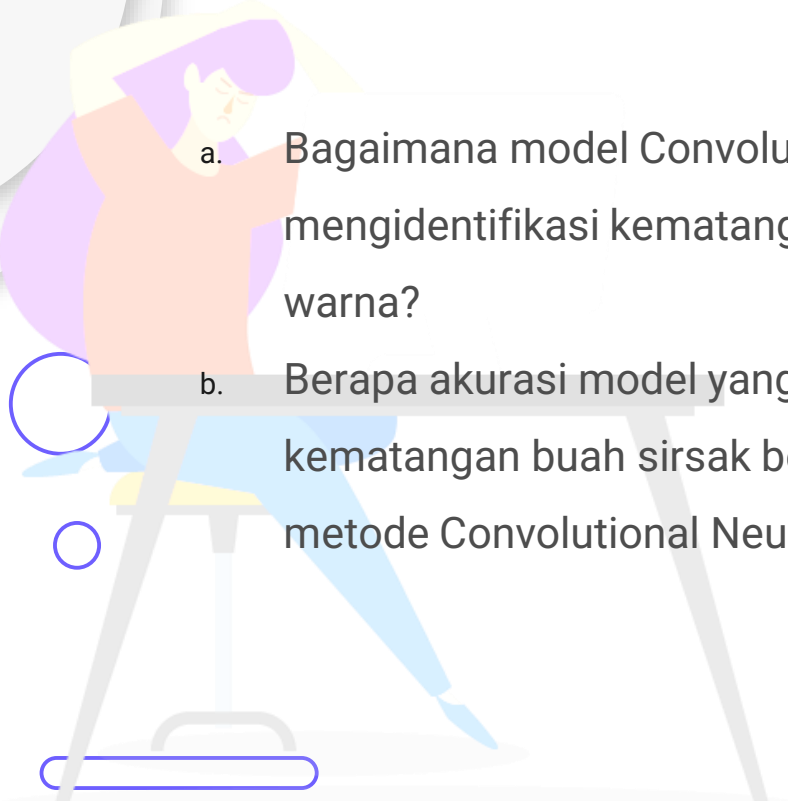


Sirsak (*Annona Muricata L*) merupakan salah satu tanaman yang berasal dari Karibia, Amerika Selatan, dan Amerika Tengah. Buah sirsak sangat cocok dikonsumsi karena kandungan gizi buah sirsak yang banyak dan bermanfaat bagi kesehatan tubuh (Elidar, 2017).



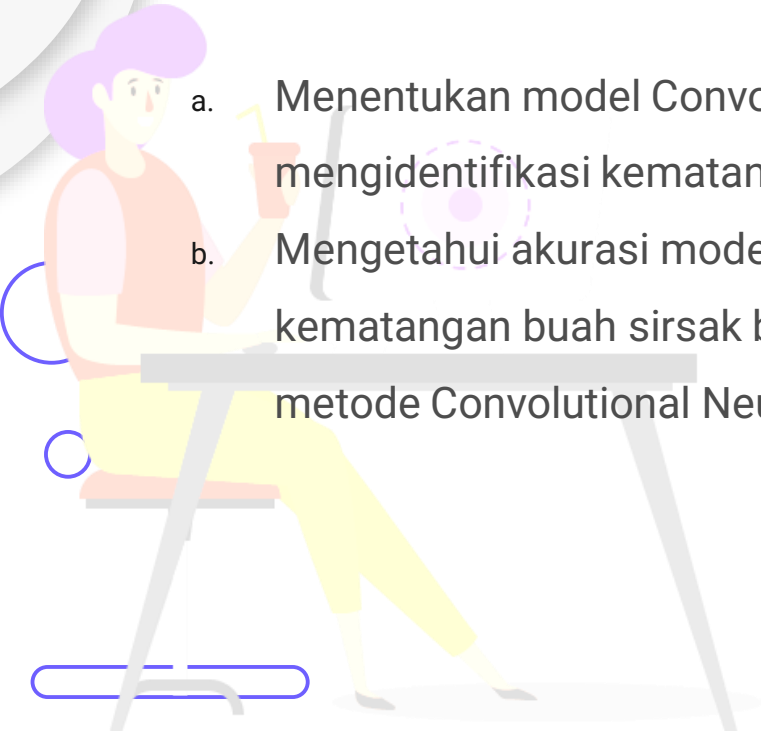


RUMUSAN MASALAH

- 
- a. Bagaimana model Convolutional Neural Network (CNN) untuk mengidentifikasi kematangan buah sirsak berdasarkan tekstur dan warna?
 - b. Berapa akurasi model yang didapatkan dalam mengidentifikasi kematangan buah sirsak berdasarkan tekstur dan warna menggunakan metode Convolutional Neural Network (CNN)?



TUJUAN PENELITIAN

- 
- a. Menentukan model Convolutional Neural Network (CNN) untuk mengidentifikasi kematangan buah sirsak.
 - b. Mengetahui akurasi model yang didapatkan dalam mengidentifikasi kematangan buah sirsak berdasarkan tekstur dan warna menggunakan metode Convolutional Neural Network (CNN).



KAJIAN TEORI

01

Citra

Istilah lain untuk gambar

02

Pengolahan Citra Digital

Masukan berupa citra yang menghasilkan keluaran citra seperti yang diinginkan

03

Convolutional Neural Network

Didesain untuk mengolah data multi dimensi seperti citra digital berwarna

04

Convusion Matrix

Menganalisis seberapa baik classifier mengenali tuple dari kelas yang berbeda.





METODE PENELITIAN

Deskripsi Data

Data primer, Kamera OPPO A15s (13MP), 3120x4160 piksel, 595 citra, jpg.



Perhitungan Akurasi
Mengetahui nilai akurasi model

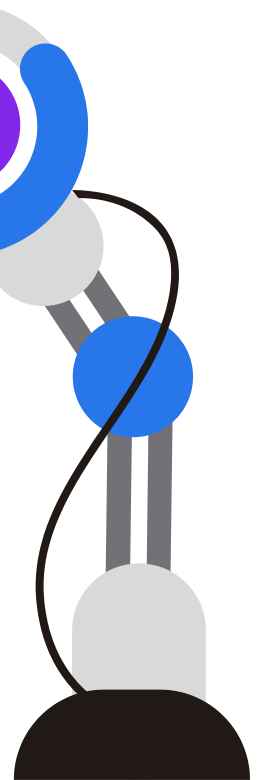
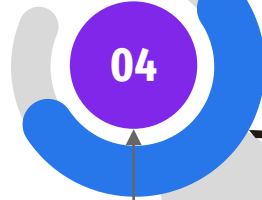
Image Classification
Klasifikasi menggunakan metode CNN

Image Preprocessing

Membuat data mentah menjadi data yang berkualitas

Akuisisi Citra

Pengubahan citra analog menjadi citra digital

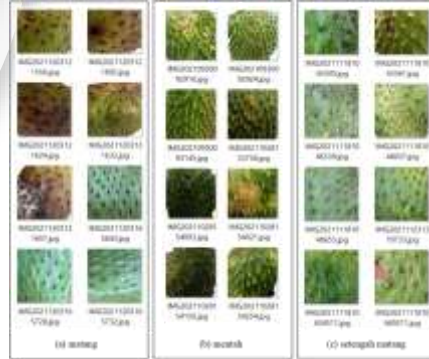




HASIL PENELITIAN

01

AKUISISI CITRA



1 Pengambilan Citra

595 citra :
205 mentah,
125 setengah matang,
265 matang

2 Resize & Cropping

Ukuran citra dari
3120x4160 piksel
menjadi
224x224 piksel



3 ImageFilter

Menggunakan
library Pillow,
ImageFilter.SHARPEN

4 Pembagian Data

476 citra data training,
119 citra data testing

02

IMAGE PREPROCESSING

01

02

#PREPROCESSING

```
train_datagen = ImageDataGenerator(  
    rescale = 1./255,  
    horizontal_flip=True,  
    vertical_flip=True,  
    rotation_range=90)  
  
test_datagen = ImageDataGenerator(rescale = 1./255)
```

ImageDataGenerator

Perintah untuk augmentasi
dan normalisasi citra

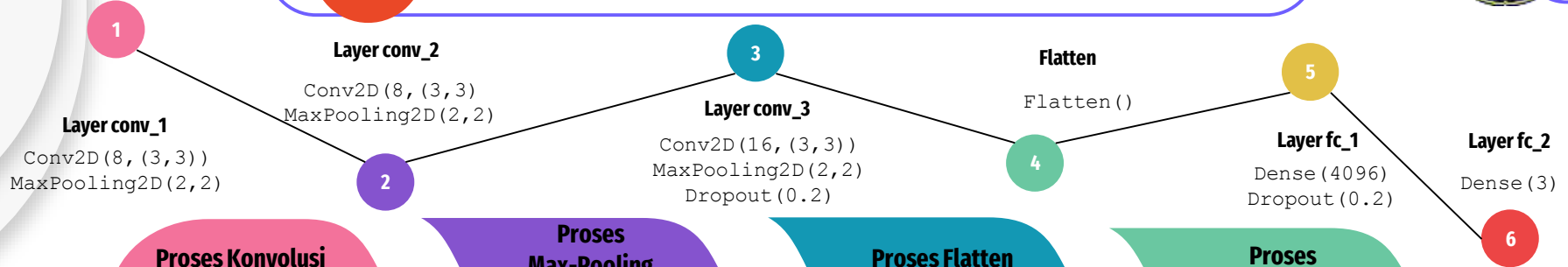
```
#target  
train_generator = train_datagen.flow_from_directory(  
    train_dir,  
    target_size = (150,150),  
    batch_size = 32,  
    color_mode = 'rgb',  
    class_mode = 'categorical',  
    classes=class_name,  
    shuffle=False,  
    seed=123,  
    save_format='png'  
)  
  
test_generator = test_datagen.flow_from_directory(  
    test_dir,  
    target_size = (150,150),  
    batch_size = 32,  
    color_mode = 'rgb',  
    class_mode = 'categorical',  
    classes=class_name,  
    shuffle=False,  
    seed=123,  
    save_format='png')
```

Flow_from_directory

Perintah untuk menghasilkan kumpulan
data dari citra dengan syarat tertentu

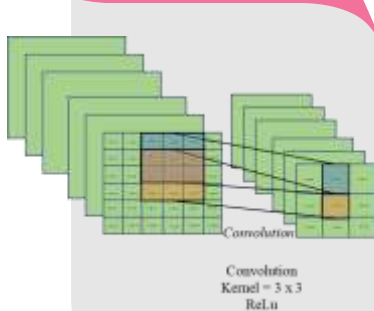
03

IMAGE CLASSIFICATION



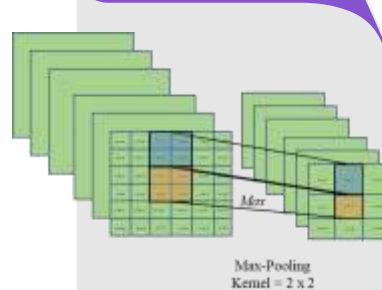
Proses Konvolusi

$$\frac{W - N + 2P}{S} + 1$$



Proses Max-Pooling

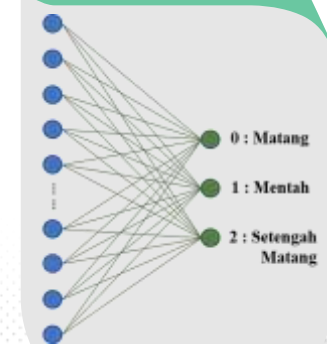
$$\frac{W - N + 2P}{S} + 1$$



Proses Flatten



Proses Fully-connected



Model Penelitian

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 148, 148, 8)	224
max_pooling2d (MaxPooling2D)	(None, 74, 74, 8)	0
conv2d_1 (Conv2D)	(None, 72, 72, 8)	584
max_pooling2d_1 (MaxPooling2D)	(None, 36, 36, 8)	0
conv2d_2 (Conv2D)	(None, 34, 34, 16)	1168
max_pooling2d_2 (MaxPooling2D)	(None, 17, 17, 16)	0
dropout (Dropout)	(None, 17, 17, 16)	0
Flatten (Flatten)	(None, 4624)	0
dense (Dense)	(None, 4096)	18944000
dropout_1 (Dropout)	(None, 4096)	0
dense_1 (Dense)	(None, 3)	12291
Total params: 18,958,267		
Trainable params: 18,958,267		
Non-trainable params: 0		

```
model = tf.keras.models.Sequential([
    tf.keras.layers.Conv2D(8, (3,3), activation='relu', input_shape = (158,158,3)),
    tf.keras.layers.MaxPooling2D(2,2),
    tf.keras.layers.Conv2D(8, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2,2),
    tf.keras.layers.Conv2D(16, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2,2),
    tf.keras.layers.Dropout(0.2, seed=123),
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(4096, activation='relu'),
    tf.keras.layers.Dropout(0.2, seed=123),
    tf.keras.layers.Dense(3, activation='softmax')
])

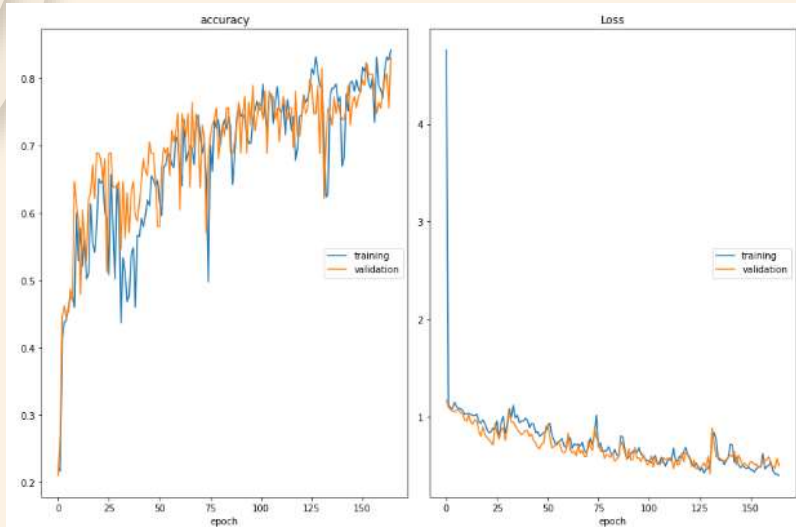
model.summary()
```



04

PERHITUNGAN AKURASI MODEL

01



```
accuracy
  training      (min:  0.216, max:  0.842, cur:  0.842)
 validation     (min:  0.210, max:  0.832, cur:  0.832)
Loss
  training      (min:  0.398, max:  4.756, cur:  0.398)
 validation     (min:  0.415, max:  1.171, cur:  0.500)
15/15 [=====] - 10s 639ms/step - loss: 0.3984 - accuracy: 0.8424 - val_loss: 0.4997 - val_accuracy: 0.8319
```

```
history = model.fit(
    train_generator,
    epochs = 165,
    validation_data = test_generator,
    callbacks = [callbacks, PlotLossesKeras()])
```

02

Data	Jumlah Data	Nilai Loss	Nilai Accuracy
Training	476	0,398	0,842
Testing	119	0,500	0,832



Data Training

476 Citra



```
confusion_matrix(Ytrain,Ytrain_pred)
array([[202,  7,  3],
       [ 12, 149,  3],
       [ 27, 17, 56]], dtype=int64)
```

Prediksi Benar Matang: 202
 Prediksi Salah Matang malah Mentah: 7
 Prediksi Salah Matang malah Setengah Matang: 3
 Total Prediksi Salah: 10

Prediksi Salah Mentah malah Matang: 12
 Prediksi Benar Mentah: 149
 Prediksi Salah Mentah malah Setengah Matang: 3
 Total Prediksi salah: 15

Prediksi Salah Setengah Matang malah Matang: 27
 Prediksi Salah Setengah Matang malah Mentah: 17
 Prediksi Benar Setengah Matang: 56
 Total Prediksi Salah: 44

$$\begin{aligned}
 \text{Akurasi} &= \frac{TP + TN}{TP + TN + FP + FN} \\
 &= \frac{202 + 149 + 56}{202 + 149 + 27 + 7 + 3 + 3 + 15 + 17 + 56} \\
 &= \frac{407}{476} = 85,50\%
 \end{aligned}$$



119 Citra

Data Testing

```
confusion_matrix(Ytest,Ytest_pred)
array([[48,  3,  2],
       [  3, 37,  1],
       [  6,  5, 14]], dtype=int64)
```

Prediksi Benar Matang: 48
 Prediksi Salah Matang malah Mentah: 3
 Prediksi Salah Matang malah Setengah Matang: 2
 Total Prediksi Salah: 5

Prediksi Salah Mentah malah Matang: 3
 Prediksi Benar Mentah: 37
 Prediksi Salah Mentah malah Setengah Matang: 1
 Total Prediksi salah: 4

Prediksi Salah Setengah Matang malah Matang: 6
 Prediksi Salah Setengah Matang malah Mentah: 5
 Prediksi Benar Setengah Matang: 14
 Total Prediksi Salah: 11

$$\begin{aligned}
 \text{Akurasi} &= \frac{TP + TN}{TP + TN + FP + FN} \\
 &= \frac{48 + 37 + 14}{48 + 37 + 14 + 2 + 3 + 1 + 3 + 6 + 14} \\
 &= \frac{99}{119} = 83,19\%
 \end{aligned}$$





PERBANDINGAN BEBERAPA PARAMETER CNN

Augmentasi Citra

Warna	Ukuran Citra	Akurasi	
		Training	Testing
Grayscale	100 x 100	66,60%	63,03%
	150 x 150	68,07%	55,46%
	224 x 224	87,18%	83,19%
	100 x 100	79,20%	75,63%
RGB	150 x 150	85,50%	83,19%
	224 x 224	84,66%	76,47%

Kernel Layer

Kernel Layer	Akurasi	
	Training	Testing
32, 32, 64	76,5%	73,9%
8, 8, 16	89,7%	82,4%
16, 32, 32	81,7%	76,5%

Jumlah Neuron

Jumlah Neuron	Akurasi	
	Training	Testing
1024	91,2%	79,8%
4096	84,50%	83,19%
8192	87,18%	83,19%

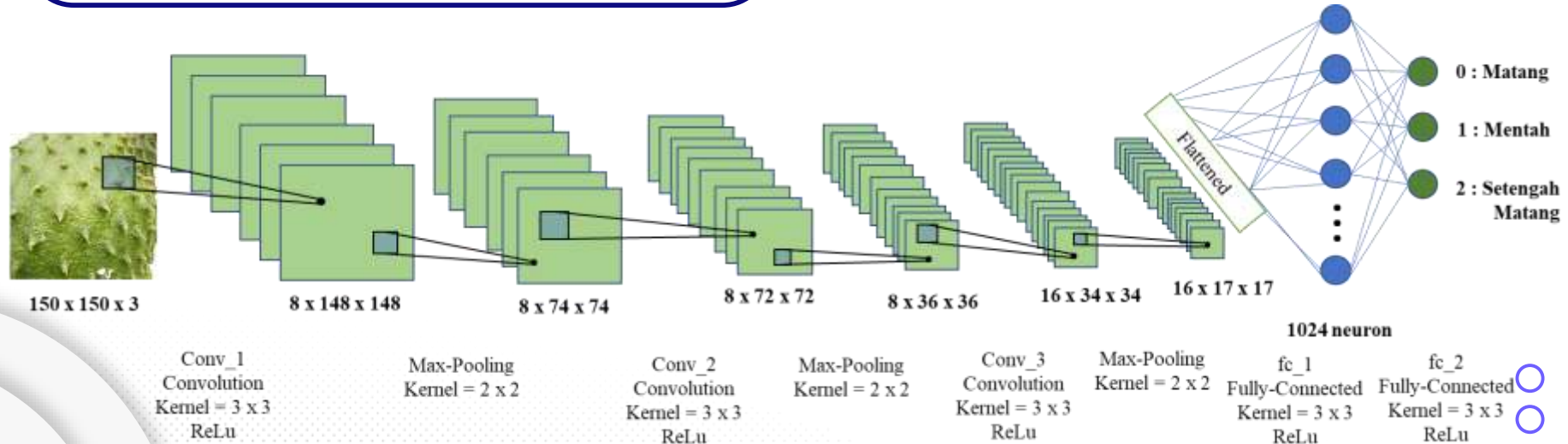
Banyak Epoch

Epoch	Akurasi	
	Training	Testing
205	89,7%	82,4%
170	82,1%	77,3%
167	79,8%	78,2%
166	89,5%	84,03%
165	85,50%	83,19%
153	85,50%	82,35%

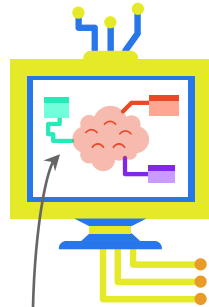
PERBANDINGAN BEBERAPA PARAMETER CNN

PARAMETER

Parameter Model	Nilai
Augmentasi citra	RGB, 150 x 150 piksel
Kernel <i>layer</i>	8, 8, 16
Jumlah Neuron <i>fully-connected layer</i>	4096 neuron
Banyak <i>epoch</i>	165 <i>epoch</i>



PENGUJIAN DATA BARU CITRA



Outputs

```
confusion_matrix(Yuji1, Yuji_pred)  
array([[5, 0, 0],  
       [1, 3, 1],  
       [1, 1, 3]], dtype=int64)
```

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN}$$
$$= \frac{5 + 3 + 3}{5 + 3 + 3 + 1 + 1 + 1 + 1} = \frac{11}{15} = 73,33\%$$

Inputs

15 citra:

- 5 mentah
- 5 setengah matang
- 5 matang



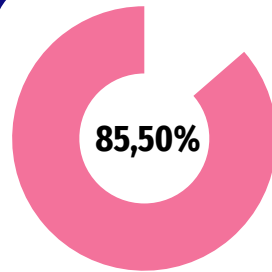


KESIMPULAN

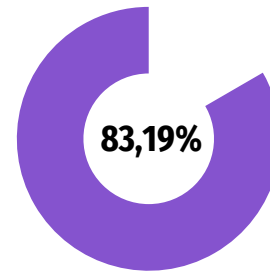
Parameter Model

Parameter Model	Nilai
Augmentasi citra	RGB, 150 x 150 piksel
Kernel layer	8, 8, 16
Jumlah Neuron <i>fully-connected layer</i>	4096 neuron
Banyak <i>epoch</i>	165 <i>epoch</i>

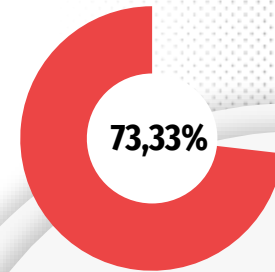
Tingkat Akurasi



Data Training



Data Testing



Data Baru Citra



TERIMA KASIH

