

# Face Recognition

Kelompok Rendang

- Aurellio Fishandy (1806199966)
- Dimas Radhitya (1806199972)
- Mas Rafi Fauzan O. (1806200040)
- Muhamad Fadil Hafiz (1806148763)



# Daftar Isi

GROUP RENDANG

- 1 Pembagian Tugas
- 2 Face Recognition CNN
- 3 Face Recognition BPNN
- 4 Hasil Perbandingan
- 5 Referensi



1

# Pembagian Tugas



**Aurellio Fishandy**

CNN dan PPT



**Dimas Radhitya**

BPNN dan PPT



**Mas Rafi Fauzan**

CNN dan PPT



**M. Fadil Hafiz**

BPNN dan PPT

2

# Face Recognition CNN

GROUP RENDANG



# Langkah dalam CNN

# Mengambil dataset



# Preprocessing



# Modeling (Max Pooling, Convolution, Fully Connected Layer)



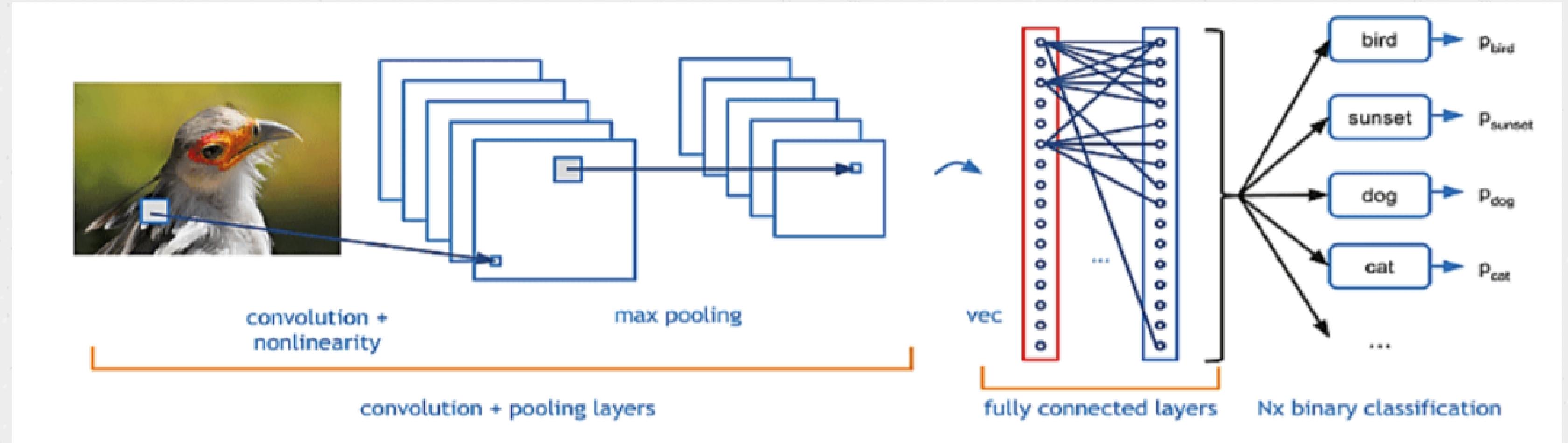
# Fitting/Processing



# Evaluating



# Arsitektur CNN



GROUP RENDANG



# Hyperparameter yang digunakan

GROUP RENDANG

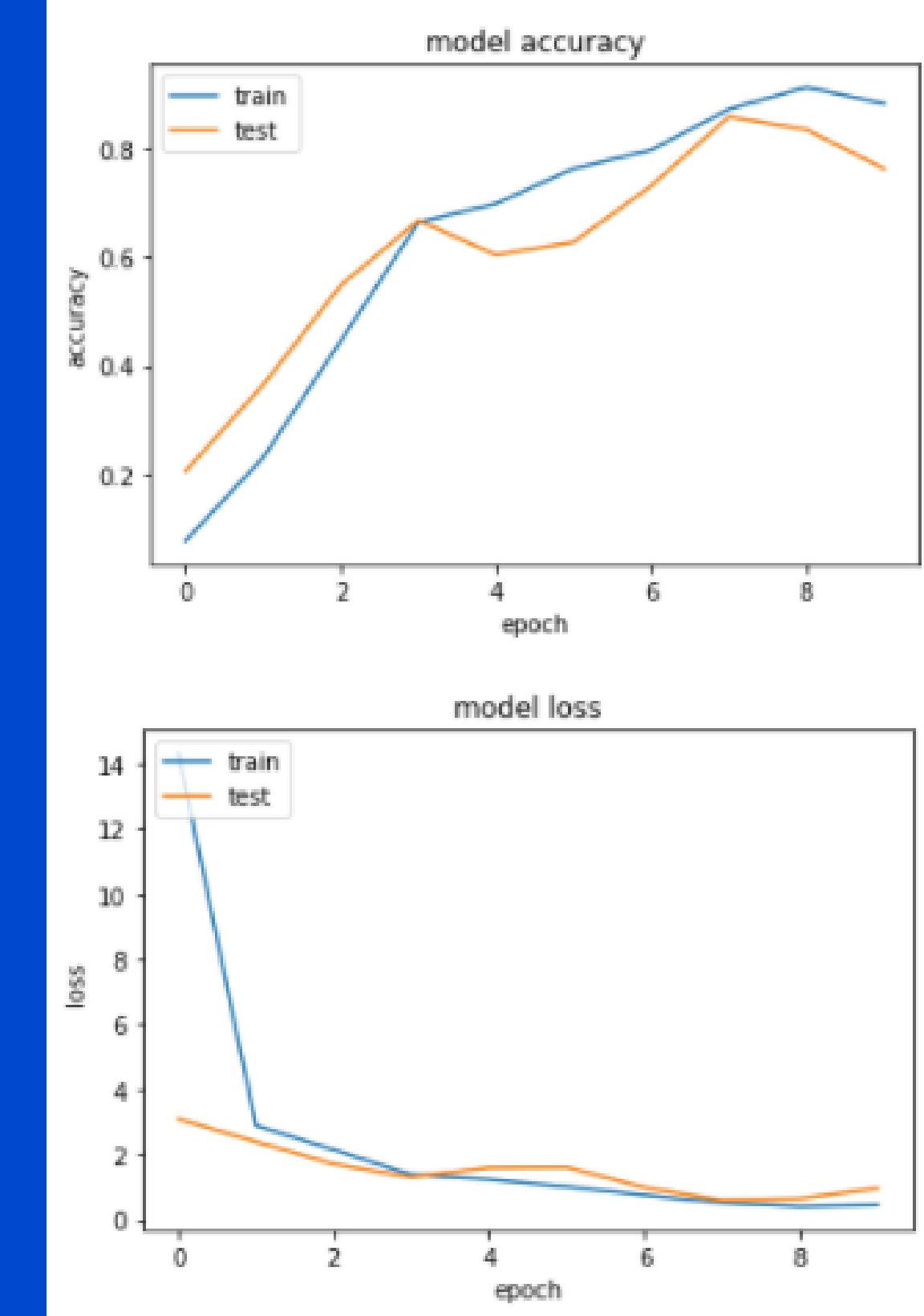
```
train_generator,  
epochs=10,  
verbose=2,  
validation_data=test_generator
```

```
tf.keras.layers.Conv2D(32, (3,3), activation='relu'),  
tf.keras.layers.MaxPooling2D(2, 2),  
tf.keras.layers.Conv2D(64, (3,3), activation='relu'),  
tf.keras.layers.MaxPooling2D(2,2),  
tf.keras.layers.Conv2D(128, (3,3), activation='relu'),  
tf.keras.layers.MaxPooling2D(2,2),  
tf.keras.layers.Conv2D(256, (3,3), activation='relu'),  
tf.keras.layers.MaxPooling2D(2,2),  
tf.keras.layers.Flatten(),  
tf.keras.layers.Dense(512, activation='relu'),  
tf.keras.layers.Dense(42, activation='softmax')
```



# Hasil CNN

GROUP RENDANG



# Hasil CNN

	epoch_1	epoch_2	epoch_3	epoch_4	epoch_5	epoch_6	epoch_7	epoch_8	epoch_9	epoch_10
loss	14.328286	2.888369	2.143970	1.383065	1.235314	1.009574	0.753772	0.513884	0.398030	0.442462
val_loss	3.096312	2.391676	1.706876	1.306509	1.597951	1.611310	0.989413	0.594416	0.650566	0.964000
accuracy	0.078231	0.231293	0.445578	0.663265	0.697279	0.761905	0.795918	0.870748	0.911565	0.880952
val_accuracy	0.206349	0.365079	0.547619	0.666667	0.603175	0.626984	0.730159	0.857143	0.833333	0.761905



3

# Face Recognition BPNN

GROUP RENDANG



# Langkah dalam BPNN

# Mengambil dataset



# Preprocessing



## Melakukan PCA



# Modeling (Polling, Convolution, Fully Connected Layer)



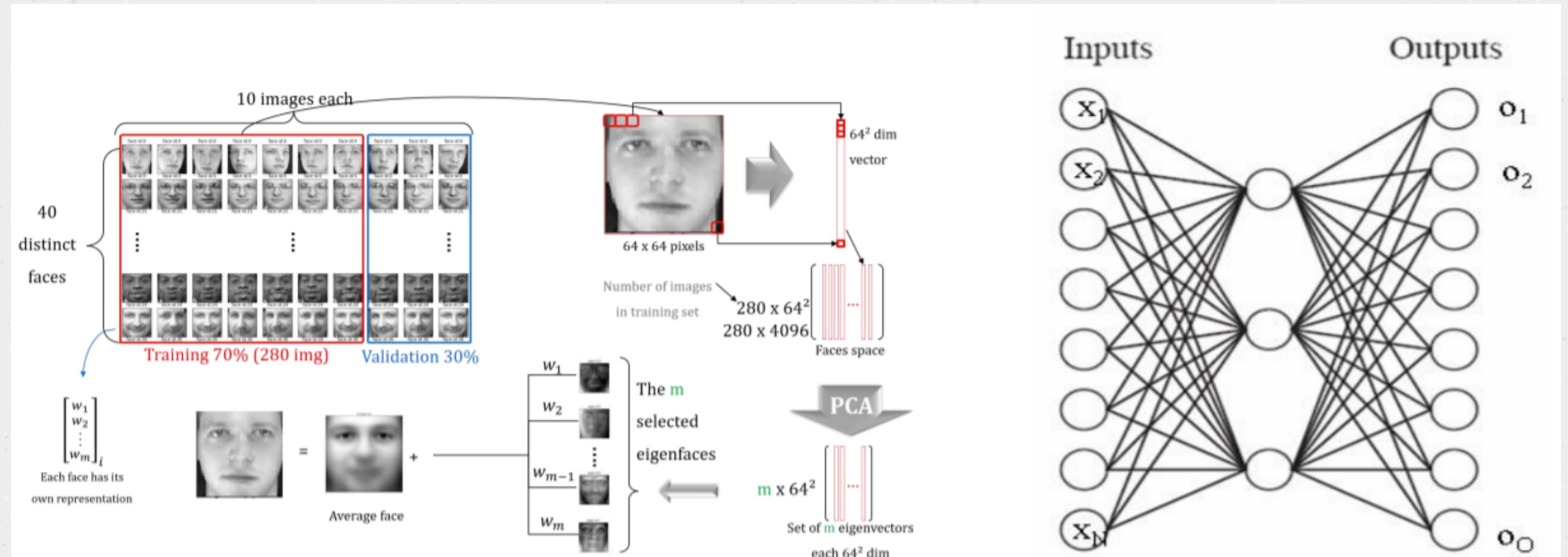
# Fitting/Processing



# Evaluating



# Arsitektur BPNN



GROUP RENDANG



# Hyperparameter yang digunakan

GROUP RENDANG

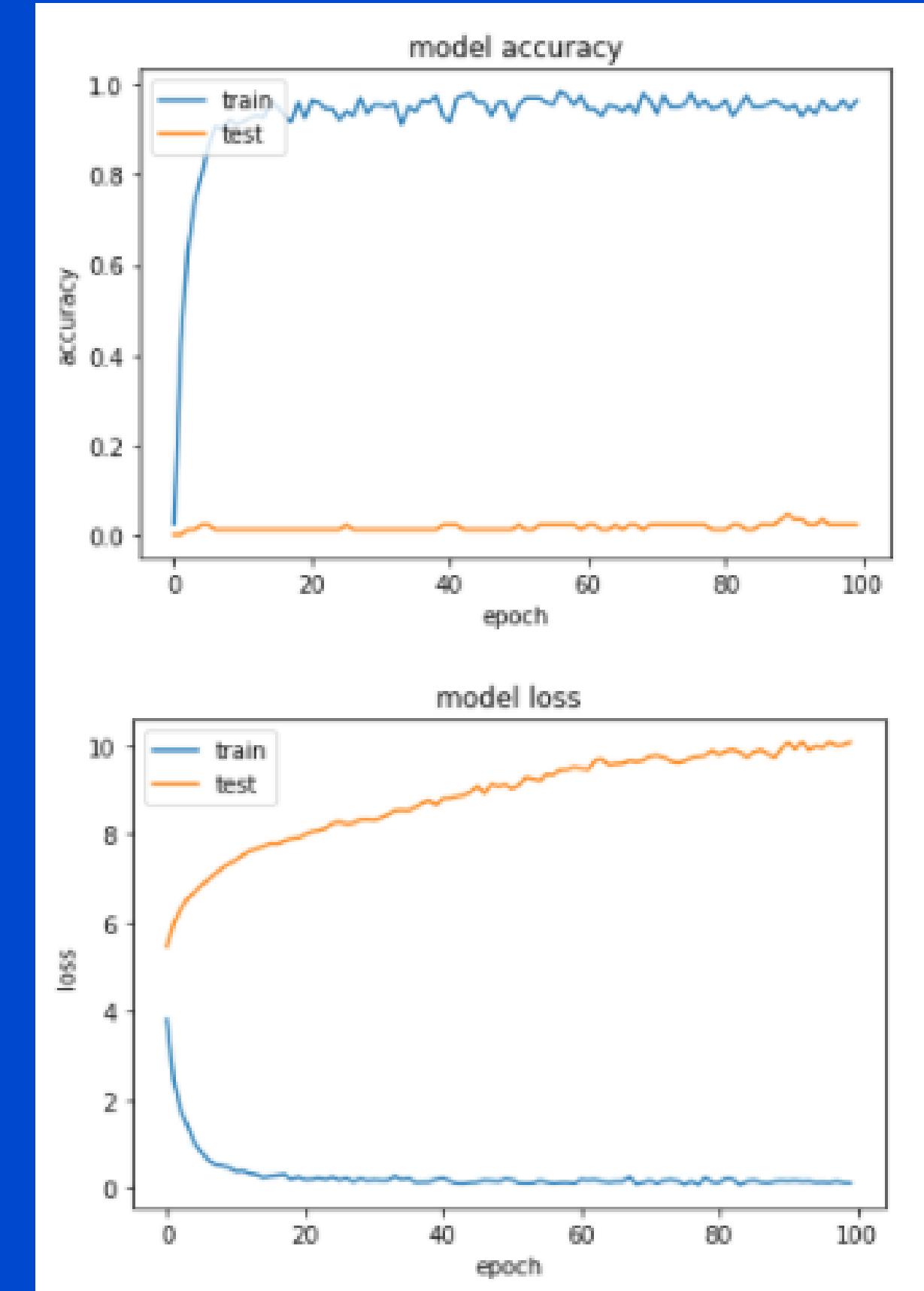
```
epochs=100, batch_size=4, validation_split=0.3, verbose=2)

tf.keras.layers.Flatten(),
GaussianNoise(pca_std),
tf.keras.layers.Dense(512, activation='relu'),
tf.keras.layers.Dense(42, activation='softmax')
```



# Hasil BPNN

GROUP RENDANG



# Hasil BPNN

GROUP RENDANG

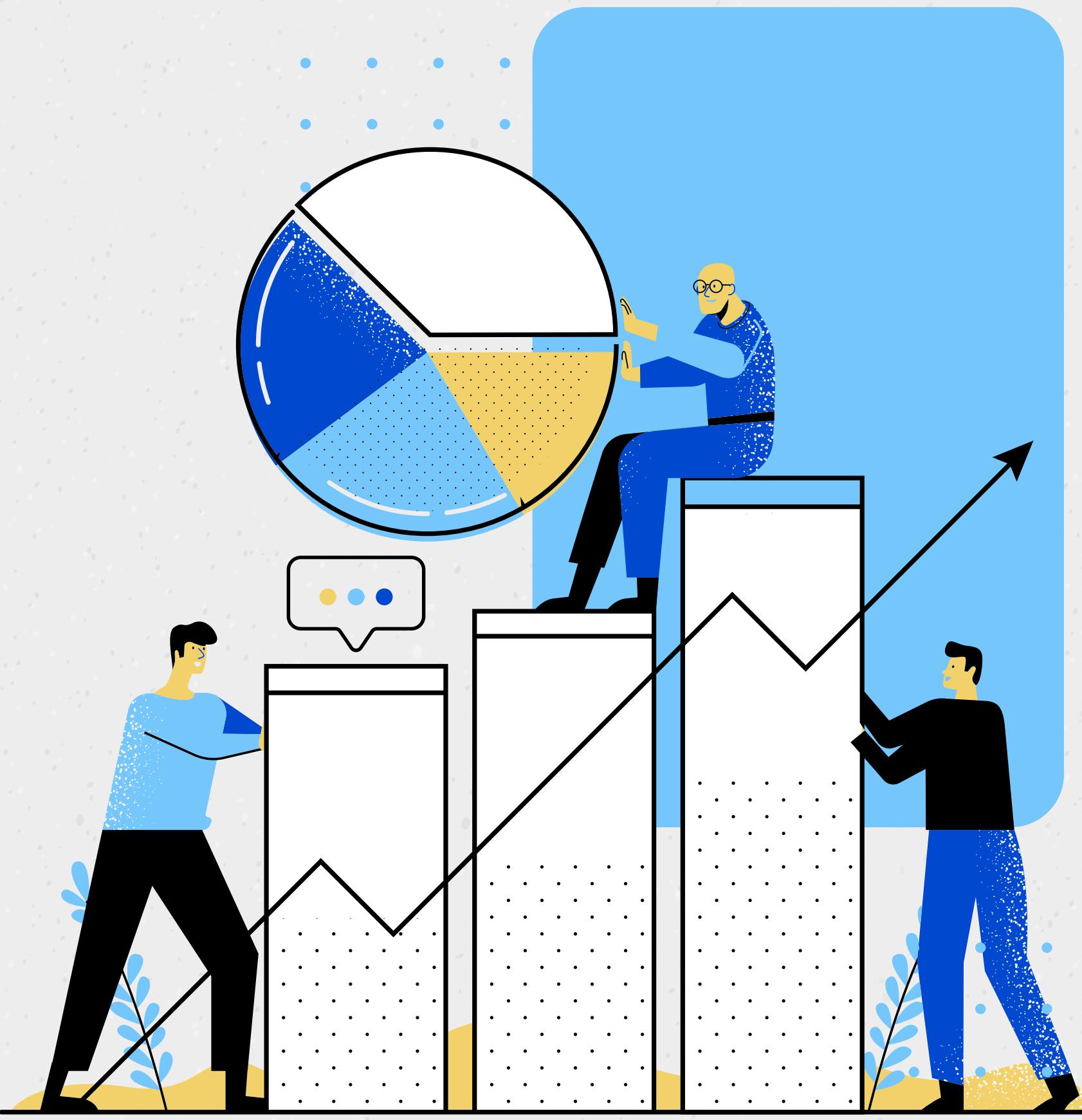
	epoch_91	epoch_92	epoch_93	epoch_94	epoch_95	epoch_96	epoch_97	epoch_98	epoch_99	epoch_100
loss	0.148036	0.168508	0.141810	0.152500	0.123831	0.129403	0.119494	0.149134	0.120644	0.108216
val_loss	10.054066	9.910630	10.071819	9.888276	9.974381	9.941544	10.072932	9.976923	9.998441	10.075235
accuracy	0.956098	0.931707	0.951219	0.936585	0.965854	0.946341	0.946341	0.965854	0.946341	0.965854
val_accuracy	0.033708	0.033708	0.022472	0.022472	0.033708	0.022472	0.022472	0.022472	0.022472	0.022472



4

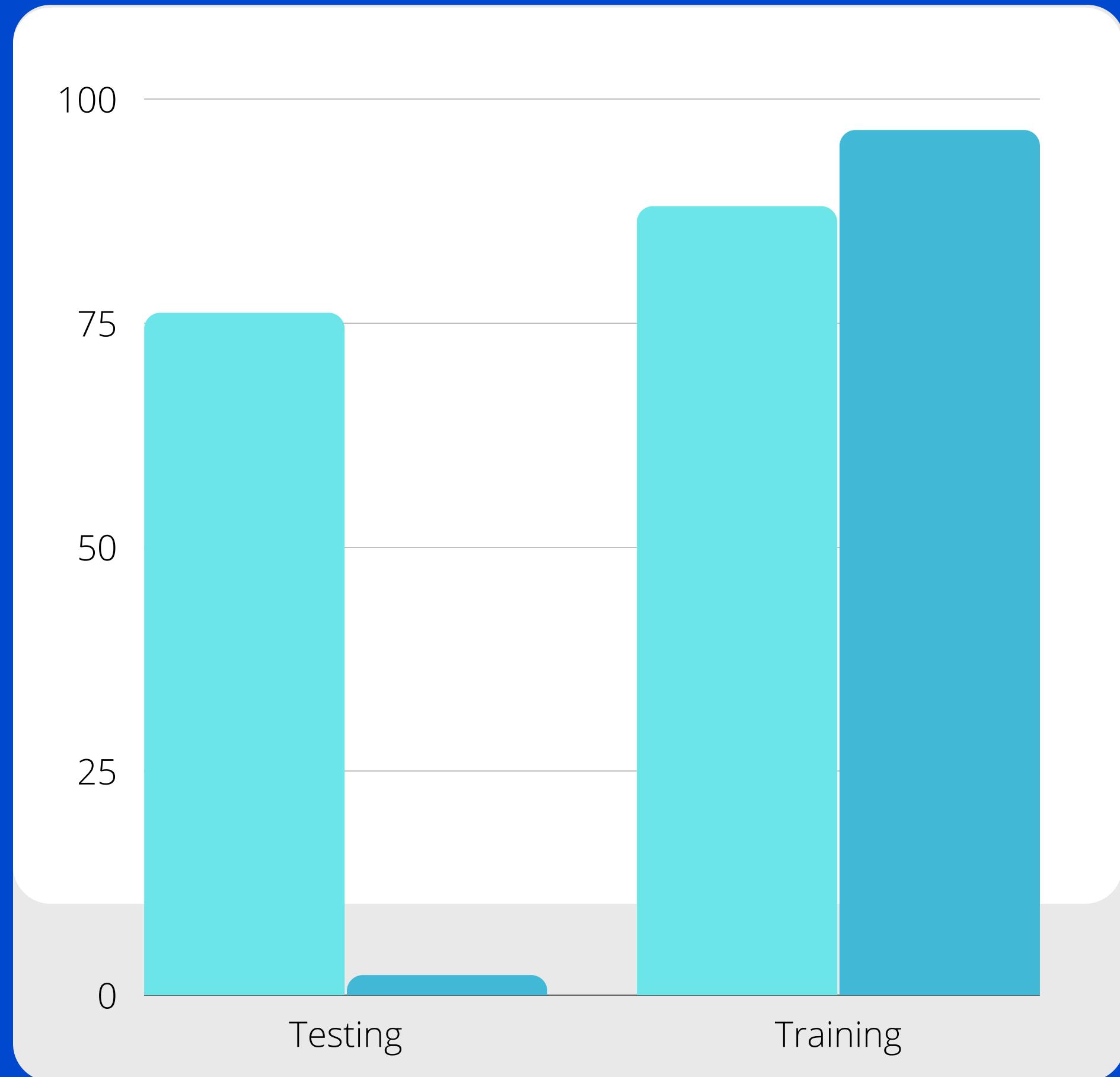
# Hasil Perbandingan

GROUP RENDANG



# Recognition Rate

- CNN
- BPNN



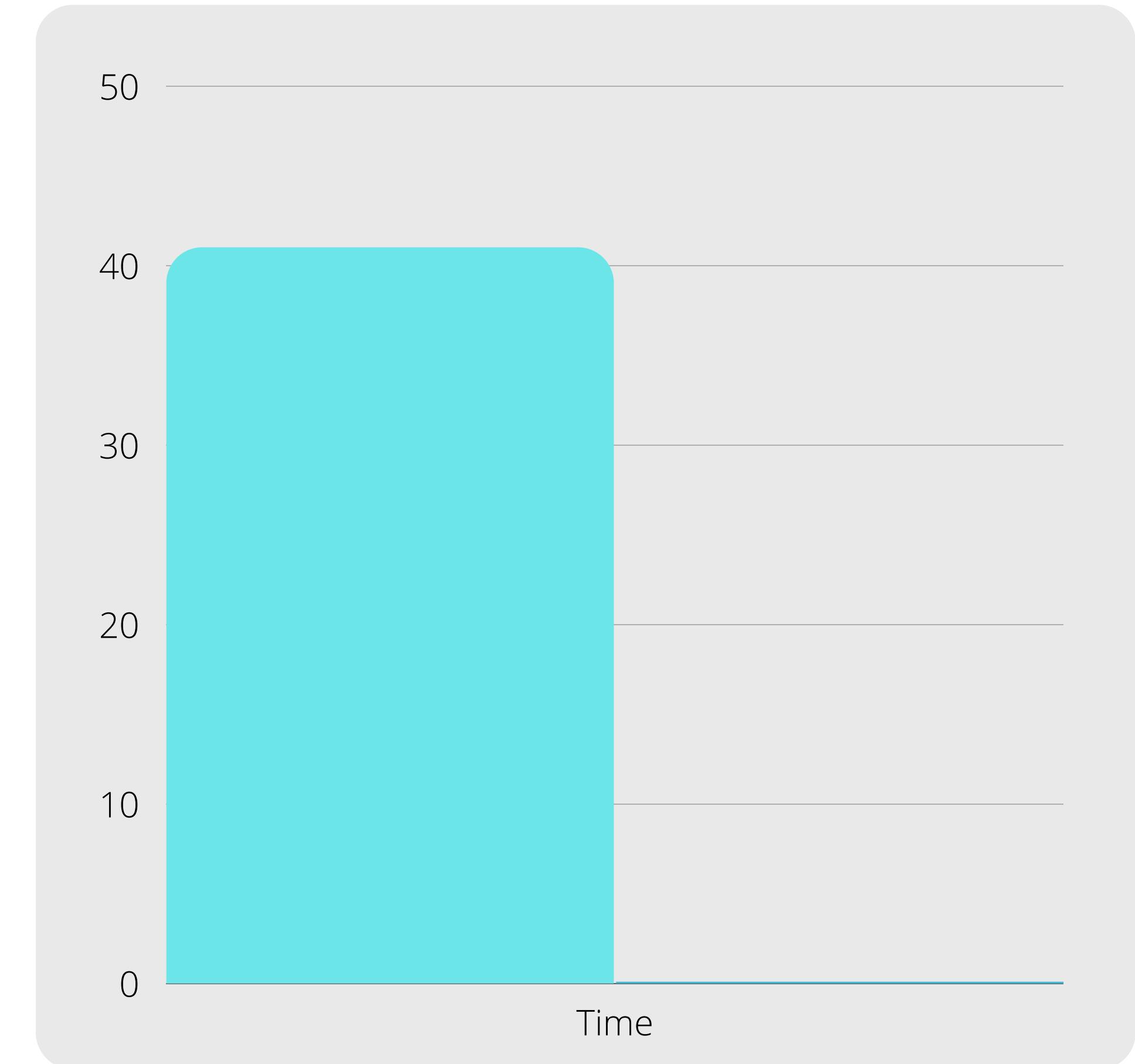
# Waktu Training & Testing



CNN



BPNN



# Kesimpulan

Kesimpulan dari hasil proyek yang telah kami buat, berdasarkan hasilnya dimana CNN memiliki hasil akurasi yang lebih baik dibanding BPNN. Namun hasil akhir dari percobaan ini masih harus diperbaiki lagi dikarenakan BPNN yang kami rancang masih belum sempurna yang disebabkan terdapatnya beberapa kendala dan teknis.



# Referensi

- <https://github.com/Fatemeh-MA/Face-recognition-using-CNN>
- <https://www.dicoding.com/academies/184/tutorials/8522>
- <https://keras.io/api/preprocessing/image/>
- <https://www.kaggle.com/pmmilewski/pca-decomposition-and-keras-neural-network/notebook#PCA-decomposition>
- <https://datascience.stackexchange.com/questions/56712/how-to-use-pca-in-cnn-for-image-recognition-using-keras>

