

Klasifikasi Objek pada Koridor Gedung Pascasarjana PENS

Anggota Kelompok:

1123800010 Ahmada Haiz Zakiyil Ilahi

1123800012 Zulfikar Davbi Mahendra F

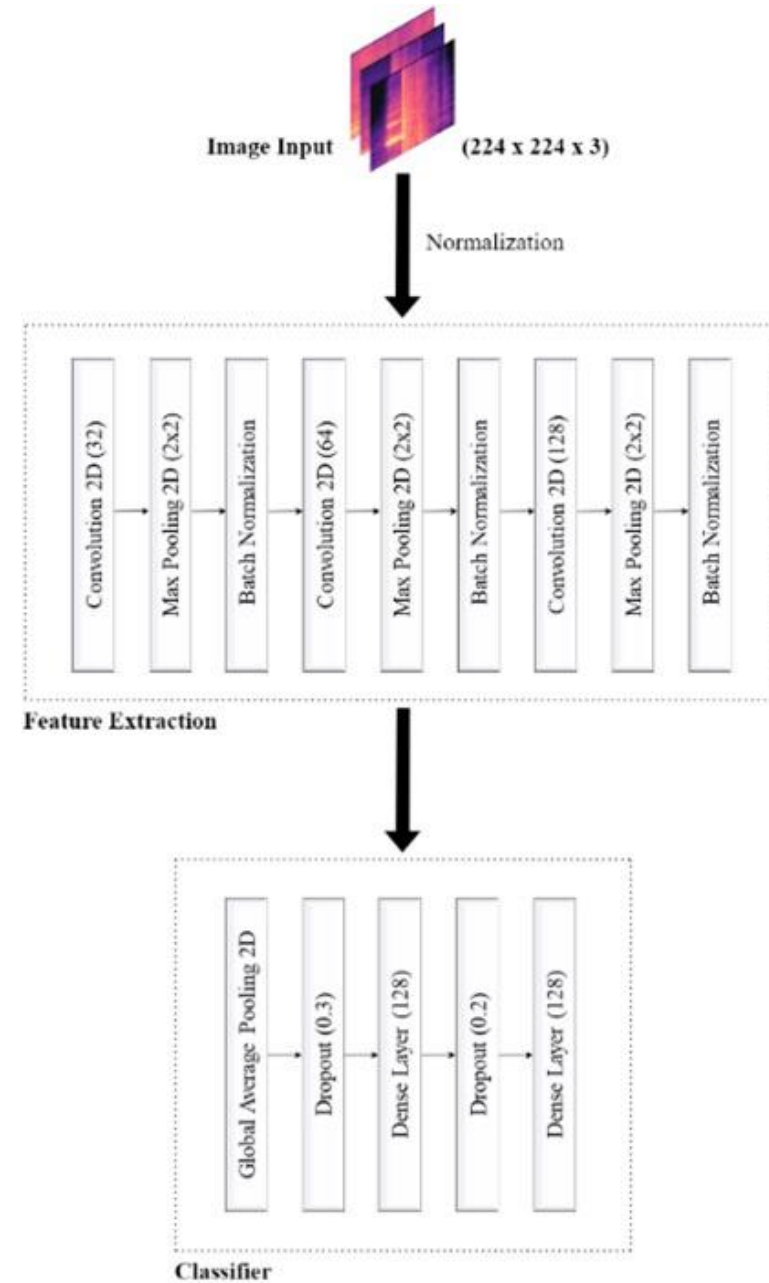
1222800008 Achmad Torikul Huda

1222800012 Rudi Kurniawan

Objek yang diklasifikasi:

- Koridor
- Pintu darurat
- Pintu ruangan
- Lift
- Tangga

Arsitektur



Data Preprocessing

```
# initial parameters
```

```
epochs = 100
```

```
batch_size = 32
```

```
input_size = (224,224)
```

```
# image augmentation part
```

```
train_datagen = ImageDataGenerator(rescale = 1./255,  
                                   rotation_range=40,  
                                   shear_range = 0.2,  
                                   zoom_range = 0.2,  
                                   horizontal_flip = True,  
                                   brightness_range=[0.2,1.0],  
                                   fill_mode='nearest')
```

```
test_datagen = ImageDataGenerator(rescale = 1./255)
```

```
STEP_SIZE_TRAIN=training_set.n//training_set.batch_size
```

```
STEP_SIZE_VALID=val_set.n//val_set.batch_size
```

```
STEP_SIZE_TEST=test_set.n//test_set.batch_size
```

Data Preprocessing

```
model = Sequential([
    Input(shape=(*input_size, 3)),
    Conv2D(32, kernel_size=(3, 3), strides=(1, 1), padding='same'),
    BatchNormalization(),
    Activation('relu'),
    MaxPooling2D(pool_size=(2, 2), strides=(2, 2)),
    Dropout(0.25),
    Conv2D(64, kernel_size=(3, 3), strides=(1, 1), padding='same'),
    BatchNormalization(),
    Activation('relu'),
    Conv2D(64, kernel_size=(3, 3), strides=(1, 1), padding='same'),
    BatchNormalization(),
    Activation('relu'),
    MaxPooling2D(pool_size=(2, 2), strides=(2, 2)),
    Dropout(0.25),
    Conv2D(128, kernel_size=(3, 3), strides=(1, 1), padding='same'),
    BatchNormalization(),
    Activation('relu'),
```

```
    Conv2D(128, kernel_size=(3, 3), strides=(1, 1),
padding='same'),
    BatchNormalization(),
    Activation('relu'),

    Conv2D(128, kernel_size=(3, 3), strides=(1, 1),
padding='same'),
    BatchNormalization(),
    Activation('relu'),
    MaxPooling2D(pool_size=(2, 2), strides=(2, 2)),
    Dropout(0.25),

    GlobalAveragePooling2D(),

    Dense(256, activation='relu'),
    BatchNormalization(),
    Dropout(0.5),

    Dense(128, activation='relu'),
    BatchNormalization(),
    Dropout(0.5),
```

Model Compilation

#Optimizer Adam

```
opt = tf.keras.optimizers.Adam(learning_rate=0.001)
```

#Compiling the model

```
model.compile(optimizer=opt, loss='categorical_crossentropy', metrics=['accuracy'])
```

```
early_stop = EarlyStopping(monitor='val_loss', patience=100, restore_best_weights=True)
```

```
checkpoint =
```

```
ModelCheckpoint("/content/drive/MyDrive/pasca_sarjana/computer_vision/project_koridor/model/custom_model.h5",  
verbose=1, save_best_only=True)
```

Training Process

fit the cnn model to the trainig set and testing it on the test set

```
hist = model.fit_generator(  
    training_set,  
    steps_per_epoch=STEP_SIZE_TRAIN,  
    epochs = epochs,  
    callbacks=[early_stop, checkpoint],  
    validation_data=val_set,  
    validation_steps=STEP_SIZE_VALID)
```

```
76/76 [=====] - ETA: 0s - loss: 0.0458 - accuracy: 0.9850  
Epoch 98: val_loss did not improve from 0.02475  
76/76 [=====] - 72s 956ms/step - loss: 0.0458 - accuracy: 0.9850 - val_loss: 0.0447 - val_accuracy: 0.9861  
Epoch 99/100  
76/76 [=====] - ETA: 0s - loss: 0.0425 - accuracy: 0.9875  
Epoch 99: val_loss did not improve from 0.02475  
76/76 [=====] - 73s 955ms/step - loss: 0.0425 - accuracy: 0.9875 - val_loss: 0.0434 - val_accuracy: 0.9826  
Epoch 100/100  
76/76 [=====] - ETA: 0s - loss: 0.0468 - accuracy: 0.9842  
Epoch 100: val_loss improved from 0.02475 to 0.02392, saving model to /content/drive/MyDrive/pasca_sarjana/computer_vision/project_koridor/model/custom_model.h  
76/76 [=====] - 77s 1s/step - loss: 0.0468 - accuracy: 0.9842 - val_loss: 0.0239 - val_accuracy: 0.9931
```

Performance Analysis

```
#Getting the accuracy
```

```
acc = hist.history['accuracy']
```

```
val_acc = hist.history['val_accuracy']
```

```
#Getting the losses
```

```
loss = hist.history['loss']
```

```
val_loss = hist.history['val_loss']
```

```
epochs = range(1, len(acc) + 1)
```

```
#Plotting Training and Validation accuracy
```

```
plt.figure(figsize=(16, 6))
```

```
plt.subplot(1, 2, 1)
```

```
plt.plot(epochs, acc, '-', label='Training Accuracy')
```

```
plt.plot(epochs, val_acc, ':', label='Validation  
Accuracy')
```

```
plt.legend(loc='lower right')
```

```
plt.title('Training and Validation Accuracy')
```

```
plt.xlabel('Epoch')
```

```
plt.ylabel('Accuracy')
```

```
#Plotting Training and Validation Loss
```

```
plt.subplot(1, 2, 2)
```

```
plt.plot(epochs, loss, '-', label='Training Loss')
```

```
plt.plot(epochs, val_loss, ':', label='Validation Loss')
```

```
plt.legend(loc='upper right')
```

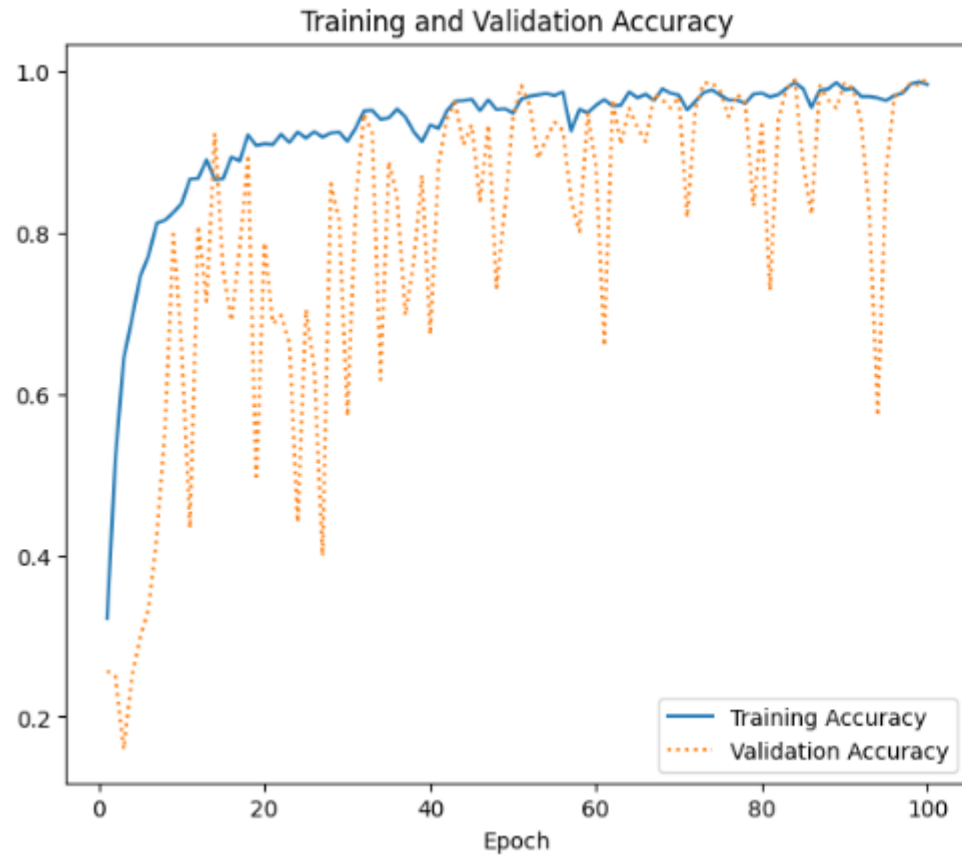
```
plt.title('Training and Validation Loss')
```

```
plt.xlabel('Epoch')
```

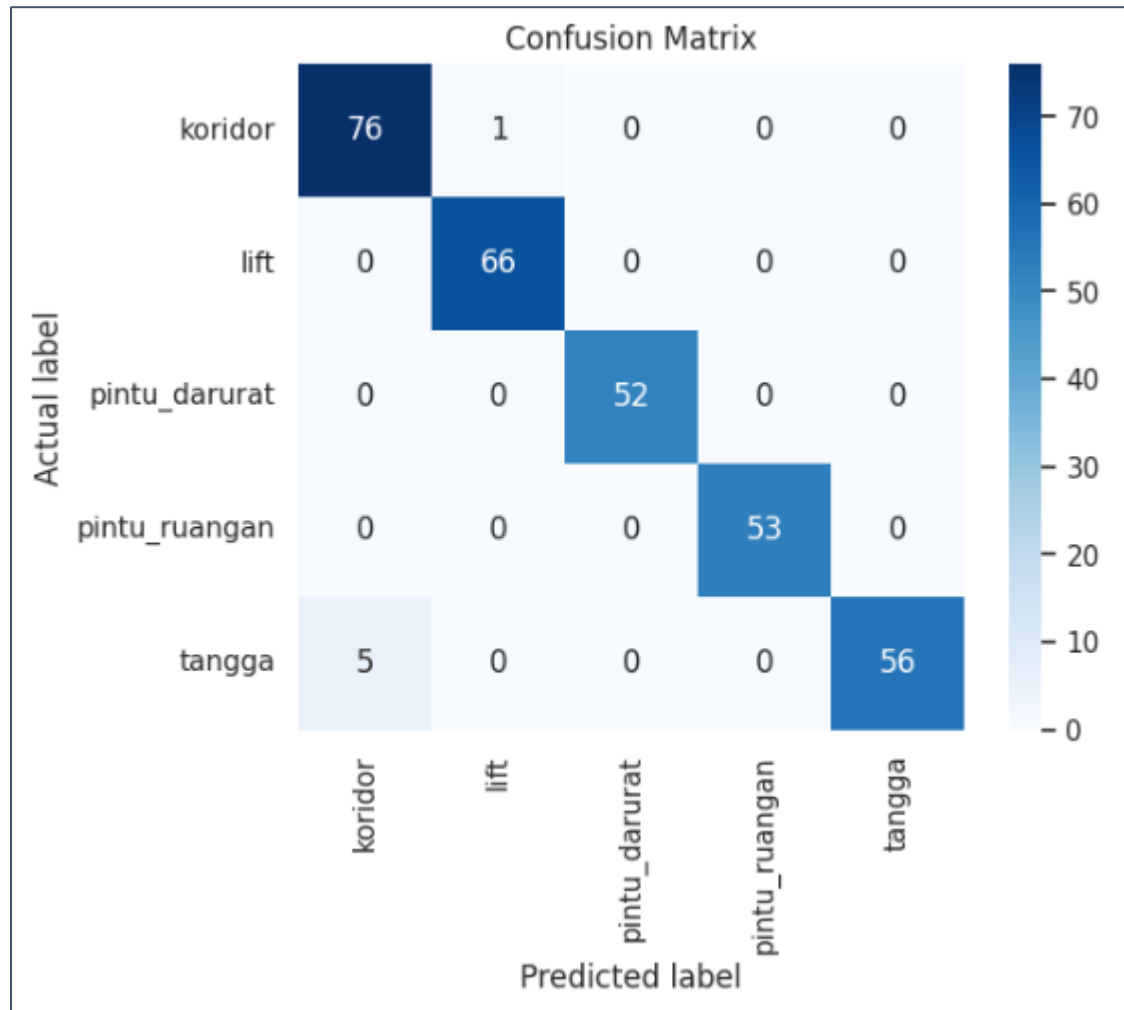
```
plt.ylabel('Loss')
```

```
plt.show()
```


Performance Analysis



Model Evaluation



	precision	recall	f1-score	support
koridor	0.94	0.99	0.96	77
lift	0.99	1.00	0.99	66
pintu_darurat	1.00	1.00	1.00	52
pintu_ruangan	1.00	1.00	1.00	53
tangga	1.00	0.92	0.96	61
accuracy			0.98	309
macro avg	0.98	0.98	0.98	309
weighted avg	0.98	0.98	0.98	309

Result

Predicted: pintu_ruangan
Actual: pintu_ruangan



Predicted: tangga
Actual: tangga



Predicted: pintu_ruangan
Actual: pintu_ruangan



Predicted: tangga
Actual: tangga



Predicted: koridor
Actual: koridor



Predicted: koridor
Actual: koridor



Lampiran

File Program :

https://drive.google.com/drive/folders/1zFRRFU9we_SHYc72MXAxuzHd5yEu1XEJ?usp=drive_link

Link Demo Youtube :

<https://youtu.be/oOj6-5BoGnc?si=rkhfMfOM3LCltDv3>