

```
from google.colab import drive
drive.mount('/content/drive')
```

Mounted at /content/drive

```
!unzip '/content/drive/MyDrive/Drowning Classification.v1i.folder.zip'
```

```
extracting: train/Not Drowning/92_png.rf.124ea7dd6f51bd2fa5bb3cd186ff89c5.jpg
extracting: train/Not Drowning/92_png.rf.361211e929ea02214f0fdc64d9bf0d6.jpg
extracting: train/Not Drowning/92_png.rf.6b372551acfa61df6b336f1e96fed7a4.jpg
extracting: train/Not Drowning/94_png.rf.4950a8b3b1d06a1d6d6fbcad33fb6c7c.jpg
extracting: train/Not Drowning/94_png.rf.4ead5ab5168fd24e40639f345f868f4c.jpg
extracting: train/Not Drowning/94_png.rf.c3e45f6f640f34c16acc4f7d70967f6a.jpg
extracting: train/Not Drowning/98_png.rf.0c4e483245747da08c94ac60f4be2b10.jpg
extracting: train/Not Drowning/98_png.rf.18af388b03cfb3a0647c34264e7d1f8a.jpg
extracting: train/Not Drowning/98_png.rf.40088ba3317d7ff293265087d10c9b39.jpg
extracting: train/Not Drowning/99_png.rf.74c72154a8ec302b0b043a26f6d21501.jpg
extracting: train/Not Drowning/99_png.rf.ea4ee85953df92b58ce179353f0a8bab.jpg
extracting: train/Not Drowning/99_png.rf.f6555974b80f4da1b353f9ac788ea17c.jpg
extracting: train/Not Drowning/download--1-_jpg.rf.03b93792179d37df944299526ae7b5
extracting: train/Not Drowning/download--1-_jpg.rf.7bc7fb79310b2b7309a90ea8219513
extracting: train/Not Drowning/download--1-_jpg.rf.ebfbf1e818a2a18c0f97c6fb9fa568
extracting: train/Not Drowning/download--2-_jpg.rf.0b979b9c2769e7f45cc4f487d2d555
extracting: train/Not Drowning/download--2-_jpg.rf.7bebf34e56669b5a414260b432c1b3
extracting: train/Not Drowning/download--2-_jpg.rf.f7596fb1d2ce8b1b4f7fa8d3ad6366
extracting: train/Not Drowning/download--3-_jpg.rf.0517316136338cbf0c8e6a0ef22269
extracting: train/Not Drowning/download--3-_jpg.rf.66ee2af2b0f54e810405017f82d4de
extracting: train/Not Drowning/download--3-_jpg.rf.912d421d555d7bd84d0332832297c5
extracting: train/Not Drowning/download--5-_jpg.rf.04f2ea11df0be0e8733b5484f988a9
extracting: train/Not Drowning/download--5-_jpg.rf.1b96641ebbcfc3ab1d89ae5405ad5c
extracting: train/Not Drowning/download--5-_jpg.rf.4479b08d6c3676a97a46c25aa9d70d
extracting: train/Not Drowning/download_jpg.rf.9c14f6ff9c7d25fde0d7b031a6f517cf.j
extracting: train/Not Drowning/download_jpg.rf.b3f3c722eaa17d102127f61f9ff1bee3.j
extracting: train/Not Drowning/download_jpg.rf.f09e05d34bc3250435c21d2044ccf1.j
extracting: train/Not Drowning/images_jpg.rf.3b7c8cb9a80071061a8a9d666d73e201.jpg
extracting: train/Not Drowning/images_jpg.rf.ea05486b84baa8a61518fc0fcbbc2a9c.jpg
extracting: train/Not Drowning/images_jpg.rf.fdd2a9f40a0337682a43dd848ee5cf66.jpg
  creating: valid/
  creating: valid/Drowning/
extracting: valid/Drowning/102_png.rf.70739cd03f12c4ef26b8f5b87491cc85.jpg
extracting: valid/Drowning/104_png.rf.45e1ae5c424b7f92e2565d056028ab46.jpg
extracting: valid/Drowning/115_png.rf.85cea7a546e56109f17a6d8fd966a620.jpg
extracting: valid/Drowning/139_png.rf.825d460a39db2210723aea12205ddbfb9.jpg
extracting: valid/Drowning/149_png.rf.7fb620c8eeae7944ff01ded421a462d2.jpg
extracting: valid/Drowning/164_png.rf.f75b1f228b5d2a8c2f9157cd46059311.jpg
extracting: valid/Drowning/165_png.rf.8c7df87de7e225c0c4a76d73e94ad80d.jpg
extracting: valid/Drowning/185_png.rf.ae17001c5f3c98163a444a33b5fbd271.jpg
extracting: valid/Drowning/187_png.rf.9ae82746270c41ccf252d7faeb47d247.jpg
extracting: valid/Drowning/210_png.rf.e1f0b0908ea315f0326fa5a88a0d5de9.jpg
extracting: valid/Drowning/59_png.rf.13e0ed28b95137ff6909fc7adcb64141.jpg
extracting: valid/Drowning/89_png.rf.22dae80dcb31a734b620ed742ea7be75.jpg
extracting: valid/Drowning/images--2-_jpg.rf.0320a78f839d3f6e3fe266f722880c15.jpg
extracting: valid/Drowning/p12_png.rf.d41feeb1aa36dc761cf0856eb8a04399.jpg
extracting: valid/Drowning/p15_png.rf.5fd93da814a92eeaf6999c91c3b10e91.jpg
extracting: valid/Drowning/p29_png.rf.4cb506810a6524a85014d95914133348.jpg
extracting: valid/Drowning/p37_png.rf.91a9afe4e35e66d3a6432126f6144e4e.jpg
  creating: valid/Not Drowning/
```

```

extracting: valid/Not Drowning/
extracting: valid/Not Drowning/104_png.rf.3d1b77f1716ef4d695bd694220884266.jpg
extracting: valid/Not Drowning/127_png.rf.e59daecb1c08fb01103a4120438b9042.jpg
extracting: valid/Not Drowning/61_png.rf.6fc6c9390110920d12c0f2e93c0f005d.jpg
extracting: valid/Not Drowning/69_png.rf.f5c5b654678b2ed9e0c459c3377e433e.jpg
extracting: valid/Not Drowning/70_png.rf.29e22d8a2966446f659908f6d028ac61.jpg
extracting: valid/Not Drowning/73_png.rf.6cc6af538ee16f70b1e588e9bc363598.jpg
extracting: valid/Not Drowning/77_png.rf.175e1313cbe19de83e38119c06c261d3.jpg

```

```

import numpy as np
import tensorflow as tf
from tensorflow.keras import layers
from tensorflow.keras.models import Sequential
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense
import matplotlib.pyplot as plt
batch_size = 32
img_height = 180
img_width = 180
data_dir = "/content/drive/MyDrive/Drowning Classification.v1i.folder.zip"

```

```

# Data augmentation on training variable
train_datagen = ImageDataGenerator(rescale=1./255,
zoom_range=0.2,
horizontal_flip=True)

```

```

# Data augmentation on testing variable
test_datagen = ImageDataGenerator(rescale=1./255)
xtrain = train_datagen.flow_from_directory('/content/train',
target_size=(64,64),
class_mode='categorical',
batch_size=100)

```

Found 678 images belonging to 2 classes.

```

xtest = test_datagen.flow_from_directory('/content/test',
target_size=(64,64),
class_mode='categorical',
batch_size=100)

```

Found 28 images belonging to 2 classes.

```
model=Sequential()
```

```

from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense
model=Sequential()

```

```
model.add(Convolution2D(32, (3,3), activation = 'relu', input_shape = (64,64,3) ))
```

```
model.add(MaxPooling2D(pool_size = (2,2)))
```

```
model.add(Flatten())
```

```
model.add(Dense(300, activation = "relu"))
model.add(Dense(150, activation = "relu"))
```

```
model.add(Dense(5, activation = "softmax"))
```

```
model.summary()
```

Model: "sequential_1"

Layer (type)	Output Shape	Param #
=====		
conv2d (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d (MaxPooling2D)	(None, 31, 31, 32)	0
flatten (Flatten)	(None, 30752)	0
dense (Dense)	(None, 300)	9225900
dense_1 (Dense)	(None, 150)	45150
dense_2 (Dense)	(None, 5)	755
=====		
Total params: 9,272,701		
Trainable params: 9,272,701		
Non-trainable params: 0		
=====		

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

```
print(xtrain.class_indices)
```

```
{'Drowning': 0, 'Not Drowning': 1}
```

```
model.fit(xtrain, epochs = 0, steps_per_epoch = len(xtrain))
```

```
<keras.callbacks.History at 0x7fd28aa70310>
```

```
print(xtest.class_indices)
```

```
{'Drowning': 0, 'Not Drowning': 1}
```

```
model.save('Drowning Classification.h5')
```

```
from tensorflow.keras.models import load_model
from keras.preprocessing import image
model=load_model("Drowning Classification.h5")
```

```
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
```

```
drown_img = image.load_img('/content/train/Drowning/100_png.rf.11dd1e96267c3d3925cc078cf41')
x = image.img_to_array(drown_img)
x = np.expand_dims(x,axis=0)
predicted_class=model.predict(x)
```

```
1/1 [=====] - 0s 107ms/step
```

drown_img



```
notdrown_img = image.load_img('/content/train/Not Drowning/100_png.rf.7278a89f7b93062d2daa')
x = image.img_to_array(notdrown_img)
x = np.expand_dims(x,axis=0)
predicted_class=model.predict(x)
```

```
1/1 [=====] - 0s 26ms/step
```

notdrown_img

