

CNN 交通號誌辨識

110034586 熊峯峻

指導教授：邱銘傳 博士

CONTENT



研究背景

- 前言
- 5W1H

研究過程

- 資料來源
- 資料前處理
- 建立模型

研究結果

- 結果分析
- 參數優化

結論

- 未來展望



1

研究背景

1

研究背景-前言



隨著無人車的發展逐漸成熟，其交通號誌辨識的功能也愈發重要。本次報告將以CNN模型辨識車道上種種交通號誌，協助無人車提高辨識交通號誌的能力。

1

研究背景-5W1H

What

辨識交通號誌

why

提高無人車判斷
交通號誌的能力

Where

車道上各種
交通號誌

How

CNN模型

Who

協助無人車辨識

When

無人車行駛時





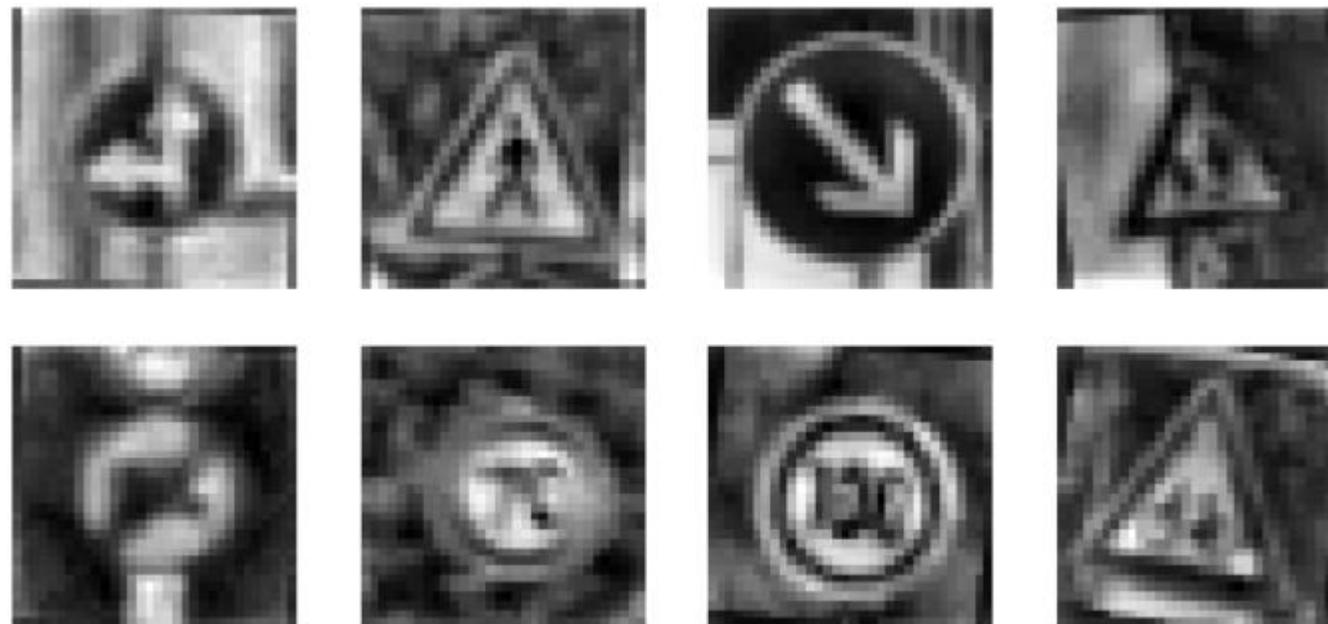
研究過程

2

研究過程-資料來源

Kaggle 公開數據集取得交通號誌圖像資料

訓練集	驗證集
86989	4410



2

研究過程-資料前處理

將原本為32*32的圖像轉為64*64

```
def resize(img):  
    numberOfImage = img.shape[0]  
    new_array = np.zeros((numberOfImage, 64, 64, 1))  
    for i in range(numberOfImage):  
        new_array[i] = tf.image.resize(img[i], (64, 64))  
    return new_array
```

32x32 Image



64x64 Image



2

研究過程-建立模型

Sequential模型

激活函數：relu

加入4層卷積層

4層池化層

1層扁平層

2層全連接層

1層dropout層

```
model = Sequential()

model.add(Conv2D(filters = 128, kernel_size = (4,4), padding = "Same", activation = "relu", input_shape = (64,64,1)))

model.add(MaxPool2D(pool_size = (2,2)))

model.add(Conv2D(filters = 64, kernel_size = (4,4), padding = "Same", activation = "relu" ))

model.add(MaxPool2D(pool_size = (2,2)))

model.add(Conv2D(filters = 32, kernel_size = (4,4), padding = "Same", activation = "relu" ))

model.add(MaxPool2D(pool_size = (2,2)))

model.add(Conv2D(filters = 16, kernel_size = (4,4), padding = "Same", activation = "relu" ))

model.add(MaxPool2D(pool_size = (2,2)))

model.add(Flatten ))

model.add(Dense(units = 512, activation = "relu"))

model.add(Dropout(0.6))

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

2

研究過程-建立模型

優化器(學習率0.001)

```
model.compile(optimizer = "adam", loss = "categorical_crossentropy", metrics = ["accuracy"])  
hist = model.fit(x_train_resized, y_train, batch_size = 512,  
                epochs = 10, validation_data = (x_val_resized, y_val))
```

訓練次數

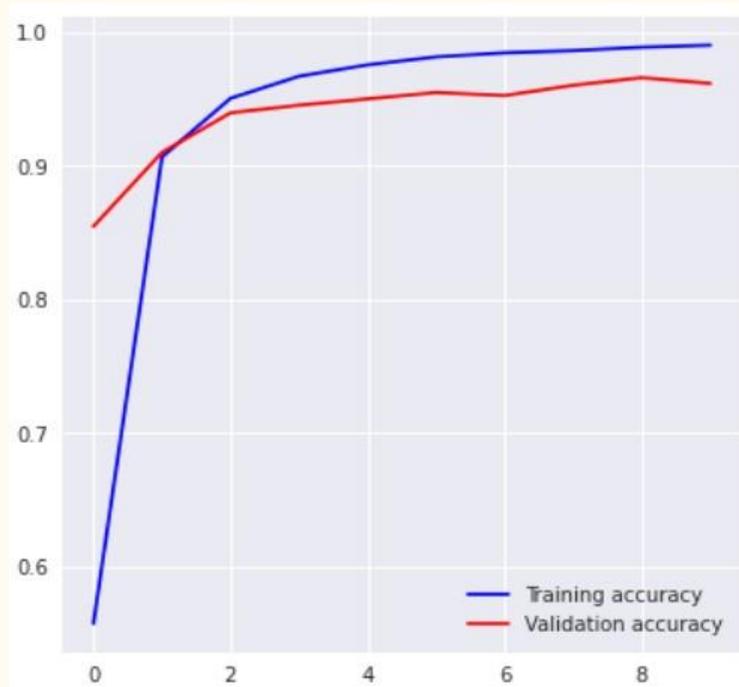
批量大小



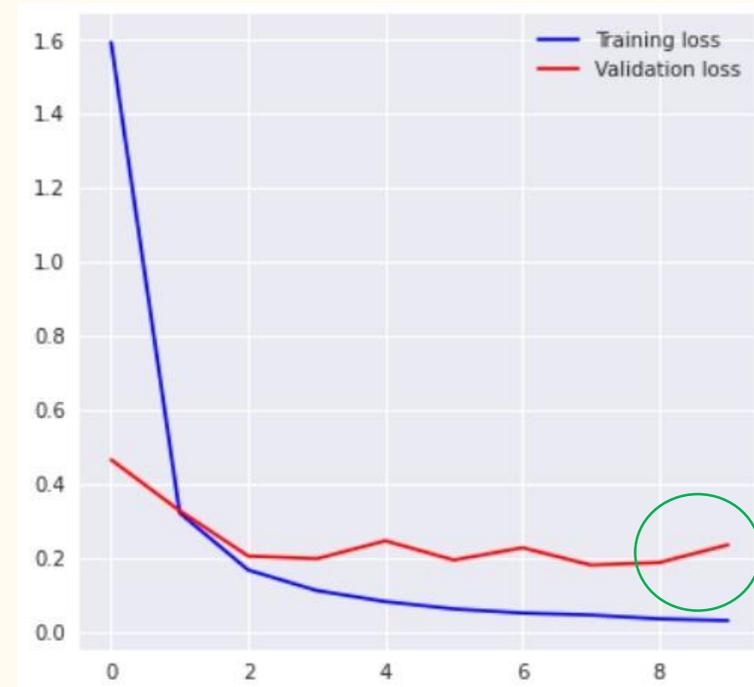
研究結果

3

研究結果-結果分析



Epoch 與Accuracy的關係



稍微上翹

Epoch 與Loss的關係

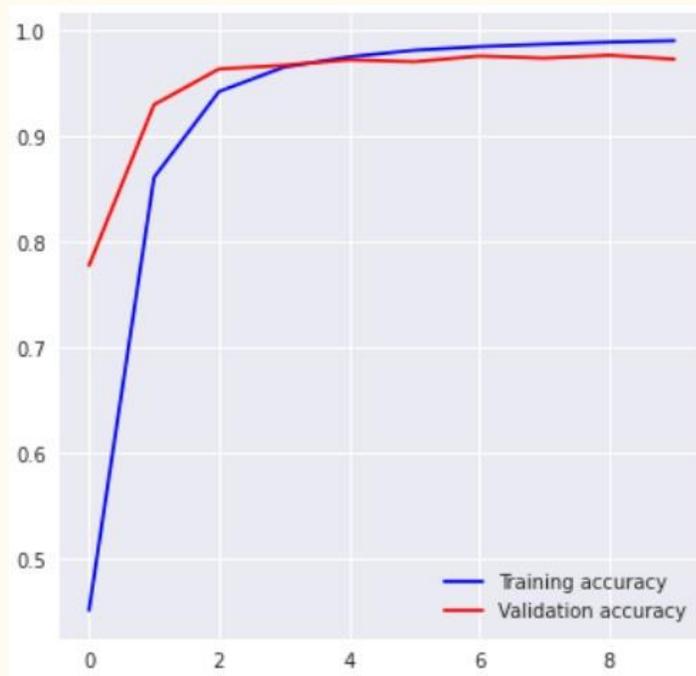
Epoch 10/10
170/170 [=====] - 1606s 9s/step - loss: 0.0299 - accuracy: 0.9901 - val_loss: 0.2350 - val_accuracy: 0.9615

3

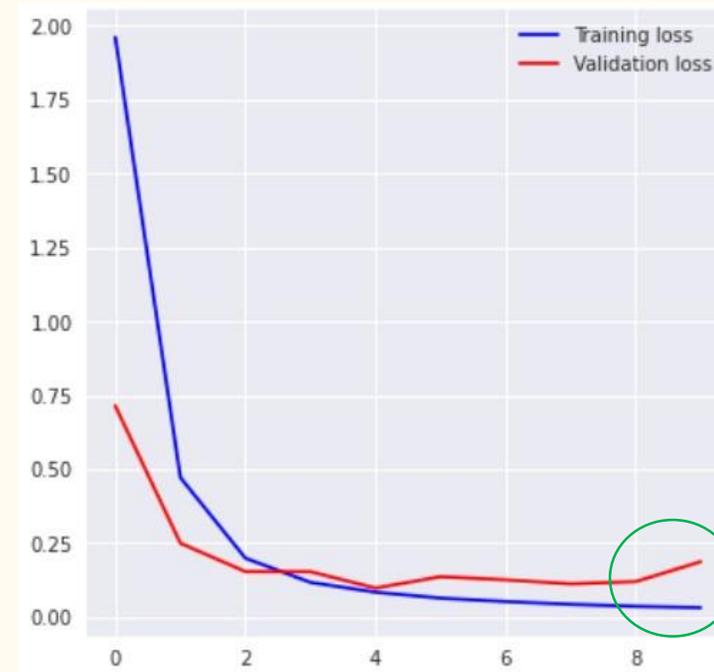
研究結果-參數優化

將optimizer 由adam改為rmsprop

```
model.compile(optimizer = "rmsprop", loss = "categorical_crossentropy", metrics = ["accuracy"])
```



Epoch 與Accuracy的關係



稍微上翹

Epoch 與Loss的關係

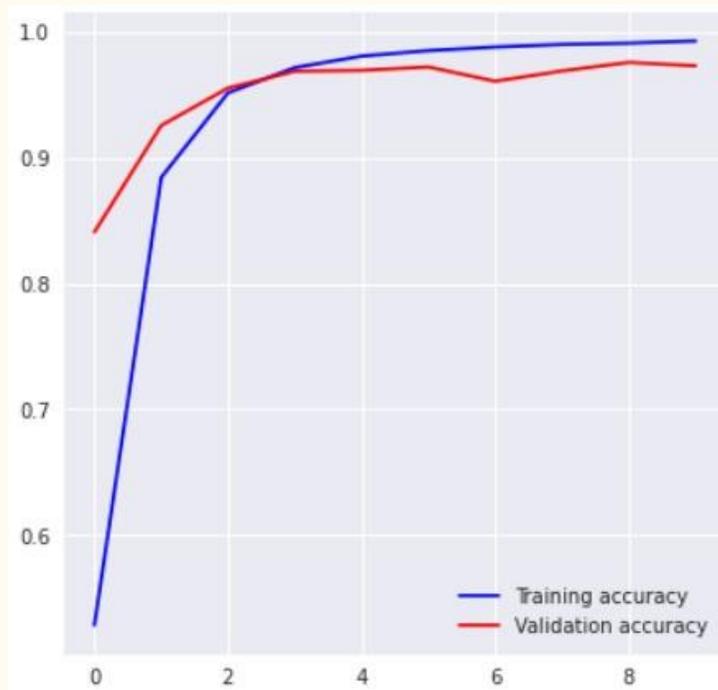
Epoch 10/10
170/170 [=====] - 1464s 9s/step - loss: 0.0289 - accuracy: 0.9906 - val_loss: 0.1865 - val_accuracy: 0.9726

3

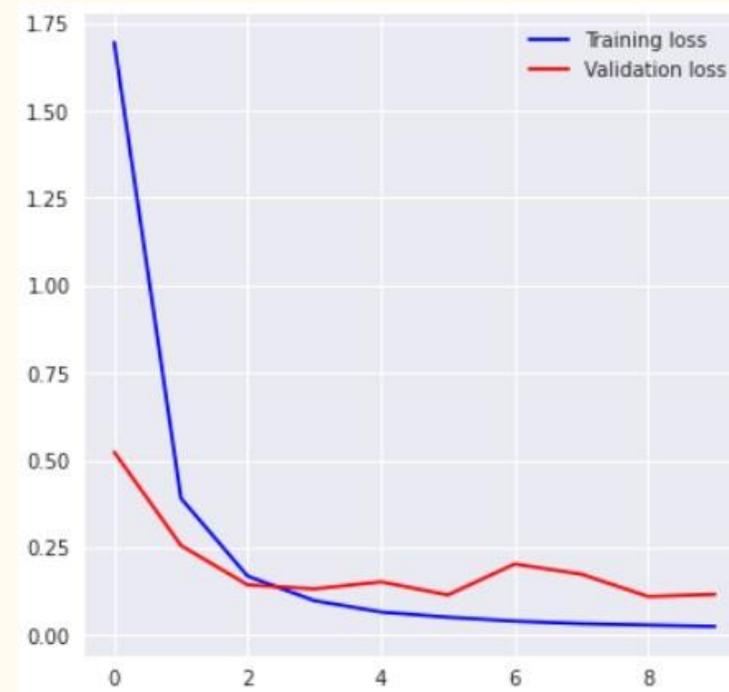
研究結果-結果分析

將dropout層係數 由0.6改為0.5

```
model.add(Dropout(0.5))
```



Epoch 與Accuracy的關係



Epoch 與Loss的關係

Epoch 10/10

170/170 [=====] - 1463s 9s/step - loss: 0.0233 - accuracy: 0.9929 - val_loss: 0.1158 - val_accuracy: 0.9728



結論

- 辨識交通號誌之準確度達到97.28%
- 協助無人車正確地辨別交通號誌
- 降低無人車判斷錯誤

- 增加辨識各國交通號誌的能力
- 加入辨識路面交通號誌的功能

**THANK
YOU**

