

โมเดลประมาณค่าตำแหน่งกึ่งกลางของตาจากรูปภาพของดวงตา

Best Model

```
Epoch 281/350
142/143 ----- 0s 48ms/step - loss: 0.0321
Epoch 281: val_loss improved from 0.95983 to 0.95557, saving model to /content/drive/MyDrive/Strabismus Prescreening/Models/350Epochs/Xception_nadam_huber_noCrop_suffix_centered_PupilFromEye_72/best_weights.ke
143/143 ----- 12s 63ms/step - loss: 0.0321 - val_loss: 0.9556
```

Model nadam

Loss: 0.0321

Val_loss: 0.9556

Link download:

<https://drive.google.com/drive/folders/1lyZKENLZFQ6VjpqMXe6TyttRmO7XrYMH?usp=sharing>

การตั้งค่าโฟลเดอร์สำหรับบันทึกผล

```
# Set the folder path for saving model weights and history
str_save_folder = "/content/drive/MyDrive/Strabismus Prescreening/Models/" + str(ep) + 'Epochs/' + str_model + "_" + op + "_" + loss_function
str_save_folder += "_noCrop_suffix_centered_PupilFromEye_72/"
```

บรรทัดนี้สร้างโฟลเดอร์ใหม่สำหรับบันทึกผลการฝึกแต่ละโมเดล โดยใช้ชื่อโมเดล, optimizer และ loss function เป็นส่วนหนึ่งของชื่อโฟลเดอร์ เพื่อให้แยกผลการฝึกโมเดลแต่ละตัวได้ง่าย

```
# Create folder if it doesn't exist
if not os.path.exists(str_save_folder):
    os.makedirs(str_save_folder)
```

หากโฟลเดอร์ที่กำหนดไว้ยังไม่มีอยู่ ระบบจะสร้างโฟลเดอร์ใหม่นี้โดยอัตโนมัติ

การสร้างและคอมไพล์โมเดล

```
# Load base Xception model
base_model = Xception(weights='imagenet', include_top=False, input_shape=(TARGET_SIZE[0], TARGET_SIZE[1], 3))
```

โหลดโมเดลพื้นฐาน Xception โดยเอาเฉพาะส่วนที่เป็น feature extraction (ไม่รวมส่วนบนสุด include_top=False)

การบันทึกผลโมเดล

```
# Define callbacks
checkpoint_path = os.path.join(str_save_folder, 'best_weights.keras') # Change to .keras format
checkpoint = ModelCheckpoint(
    checkpoint_path, monitor='val_loss', save_best_only=True, mode='min', verbose=1
)
```

ตั้งค่า ModelCheckpoint เพื่อบันทึก weights ของโมเดลที่มีค่าการสูญเสีย (val_loss) ต่ำที่สุดในไฟล์ .keras

```
# Save the training history as a CSV file
history_df = pd.DataFrame(history.history)
history_df.to_csv(os.path.join(str_save_folder, 'history.csv'), index=False)
```

หลังจากฝึกเสร็จ จะบันทึกประวัติการฝึก (เช่น loss, val_loss แต่ละ epoch) เป็นไฟล์ .csv

การฝึกโมเดล

```
# Train the model
history = model_final.fit(
    x_train, y_box_train, # Training data
    validation_data=(x_validation, y_box_validation), # Validation data
    epochs=ep, verbose=1,
    callbacks=[checkpoint] # Save best model weights
)
```

ฝึกโมเดลโดยใช้ชุดข้อมูล x_train และ y_box_train พร้อมทั้งชุด validation และกำหนด callback เพื่อบันทึกผลลัพธ์

Member

65125056 นายธีรชัย สังเฑระ 6

65108813 นางสาวสาธิตา ถนอมทรัพย์ 6

65119810 นายศิวกร หลีเหม 6