안개 제거 유무에 따른 도로 인스턴스 분할 성능의 차이

컴퓨터소프트웨어학과3학년 윤건용

목차

1.프로젝트 개요

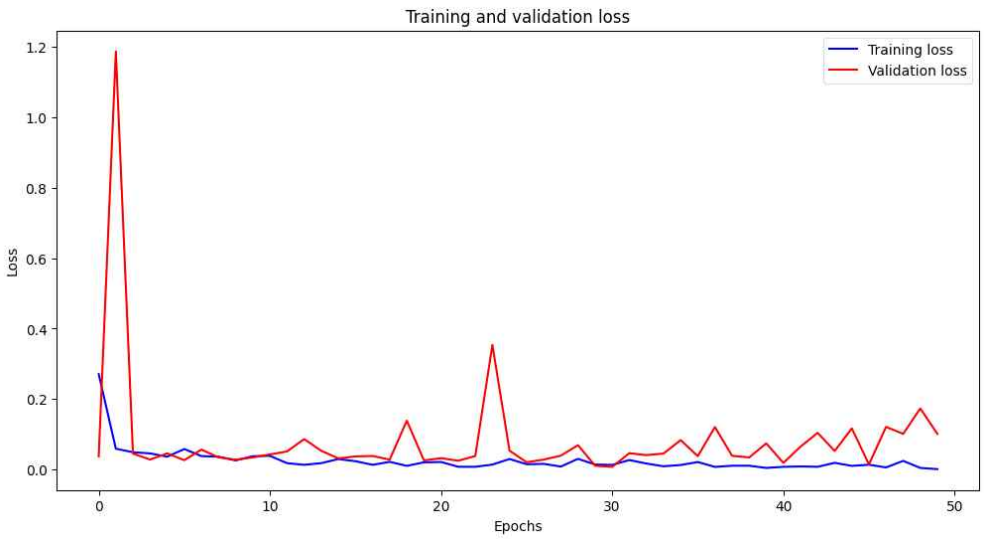
- 이미지 내 안개 검출 모델
- 도로 인스턴스 분리 모델
- 이미지의 안개 제거
- 도로 인스턴스 추출
- 결과 비교

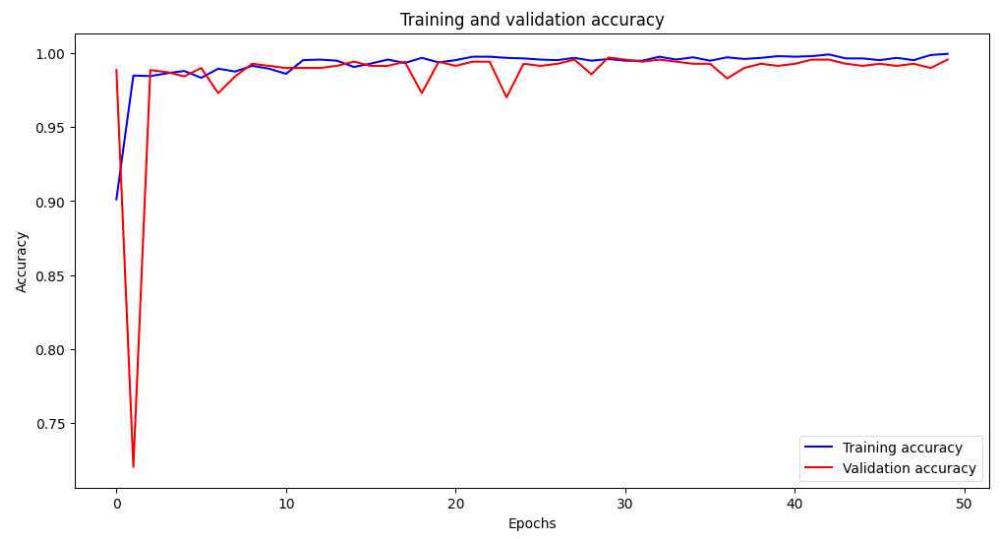
2.안개 유무 확인 모델 생성

모델 구조	Model: "sequential" _____ Layer (type) Output Shape Param # ========== conv2d (Conv2D) (None, 148, 148, 32) 896 activation (Activation) (None, 148, 148, 32) 0 max_pooling2d (MaxPooling2 (None, 74, 74, 32) 0 D) conv2d_1 (Conv2D) (None, 72, 72, 32) 9248 activation_1 (Activation) (None, 72, 72, 32) 0 max_pooling2d_1 (MaxPoolin (None, 36, 36, 32) 0 g2D) conv2d_2 (Conv2D) (None, 34, 34, 64) 18496 activation_2 (Activation) (None, 34, 34, 64) 0 max_pooling2d_2 (MaxPoolin (None, 17, 17, 64) 0 g2D) flatten (Flatten) (None, 18496) 0 dense (Dense) (None, 64) 1183808 activation_3 (Activation) (None, 64) 0 dropout (Dropout) (None, 64) 0 dense_1 (Dense) (None, 1) 65 activation_4 (Activation) (None, 1) 0 ========== Total params: 1212513 (4.63 MB) Trainable params: 1212513 (4.63 MB) Non-trainable params: 0 (0.00 Byte) _____

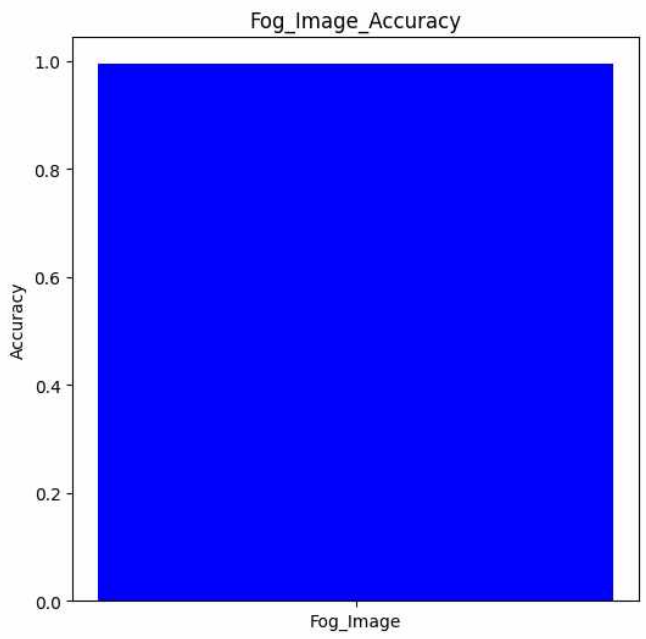
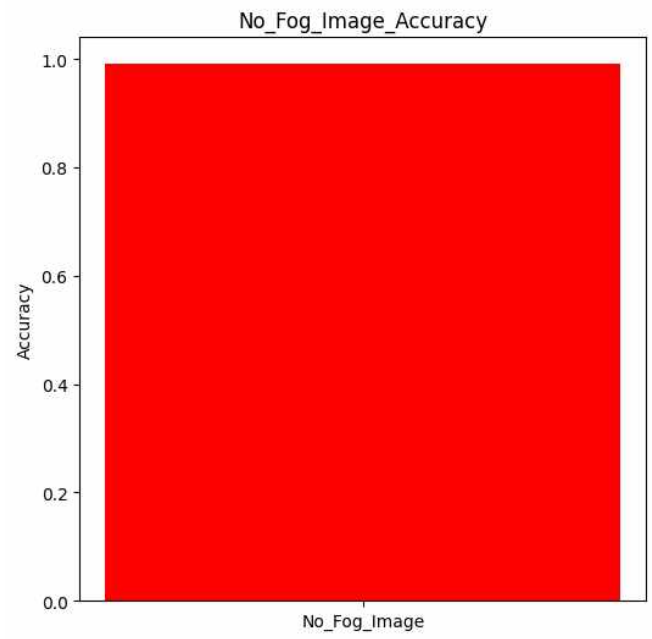
모델 학습 방법	history = model.fit(train_generator, steps_per_epoch=2622// 32, epochs=50, validation_data=validation_generator, validation_steps=726// 32)

안개 유무 확인 모델의 학습 과정의 손실, 정확도 그래프





- loss: 0.0012 - accuracy: 0.9996 - val_loss: 0.1012 - val_accuracy: 0.9957

재검증한 모델의 정확도		
	정확도: 99.48849104859335%	정확도: 99.1044776119403%

3.도로 인스턴스 분리 모델 생성

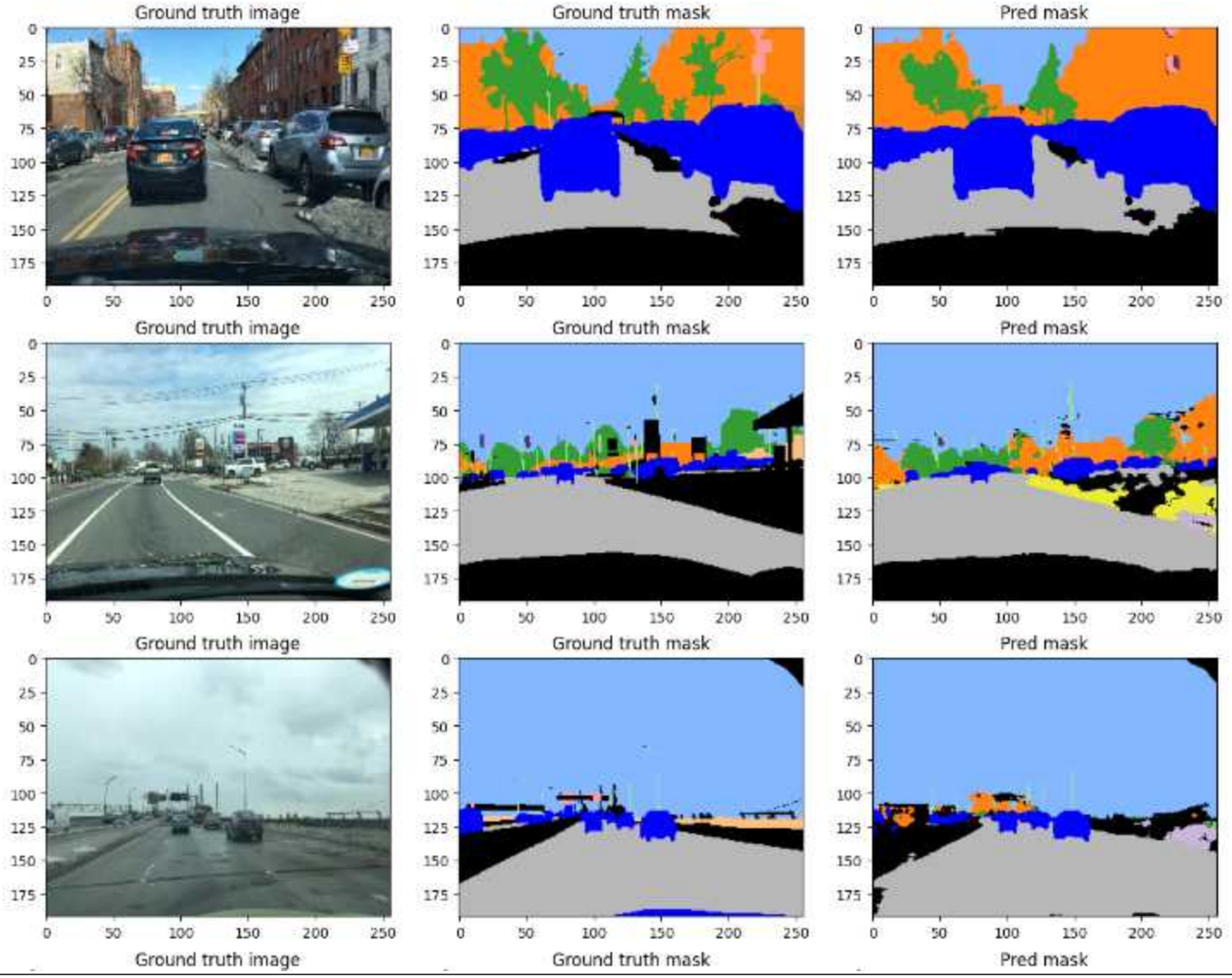
설정한 인스턴스 색상

도로	회색	사람	빨강
보도	연한 노랑	차	파랑
울타리	주황	트럭	파랑
식물	녹색	버스	파랑
하늘	하늘색	알 수 없음	검정

모델 구조	Model: "model_1" _____ _____ Layer (type) Output Shape Param # Connected to ========== ========== input_2 (InputLayer) [(None, 192, 256, 3)] 0 [] conv2d_19 (Conv2D) (None, 192, 256, 64) 1792 ['input_2[0][0]'] conv2d_20 (Conv2D) (None, 192, 256, 64) 36928 ['conv2d_19[0][0]'] max_pooling2d_4 (MaxPoolin(None, 96, 128, 64) 0 ['conv2d_20[0][0]'] g2D) ,,, concatenate_7 (Concatenate (None, 192, 256, 128) 0 ['conv2d_20[0][0]',) 'conv2d_transpose_7[0][0]'] conv2d_35 (Conv2D) (None, 192, 256, 64) 73792 ['concatenate_7[0][0]'] conv2d_36 (Conv2D) (None, 192, 256, 64) 36928 ['conv2d_35[0][0]'] conv2d_37 (Conv2D) (None, 192, 256, 20) 1300 ['conv2d_36[0][0]'] ========== ========== Total params: 34514580 (131.66 MB) Trainable params: 34514580 (131.66 MB) Non-trainable params: 0 (0.00 Byte)

모델 학습 방법

```
epochs = 21
batch_size = 16
model.fit(
     X_train,
     Y_train,
     initial_epoch=initial_epoch,
     steps_per_epoch=7000// 32,
     epochs=epochs,
     validation_data=(X_val, Y_val),
     validation_steps=1000// batch_size
)
```


Ground truth image
Ground truth mask
Pred mask
Ground truth image
Ground truth mask
Pred mask
Ground truth image
Ground truth mask
Pred mask
Ground truth image
Ground truth mask
Pred mask

모델 평가

Accuracy: 0.8488716

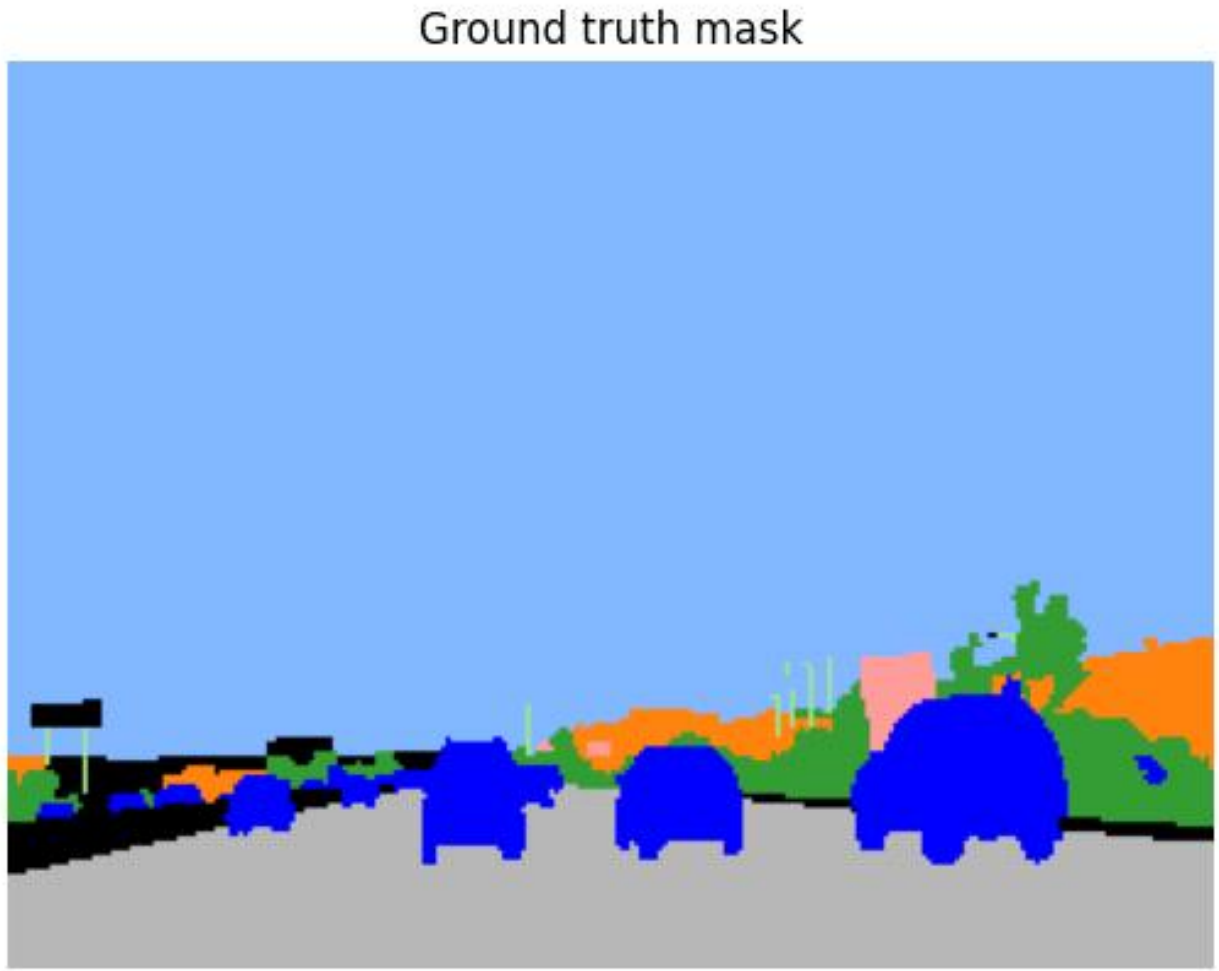
Mean IoU: 0.32686156

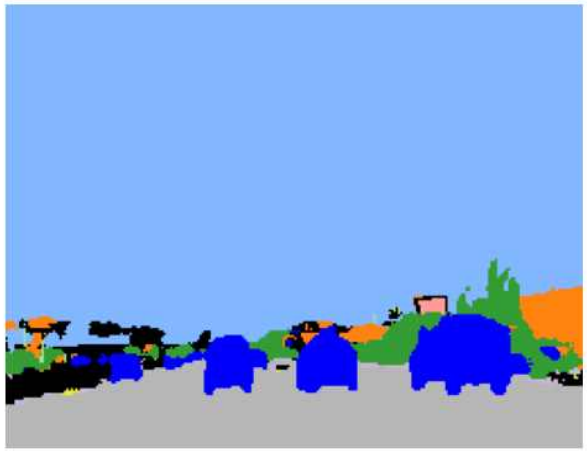
4. 안개제거 유무에 따른 도로 인스턴스 분할 성능의 차이

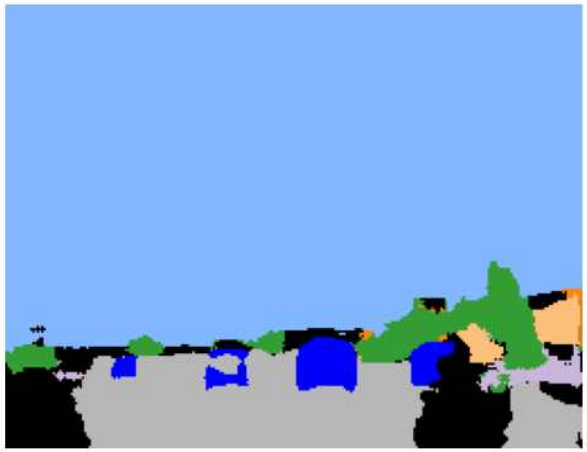
```
fog_image = 'data/make_foggy_image.jpg'# 테스트에 사용할 안개 이미지
test_label = 'data/Image_train_color.png'   # 테스트에 사용할 안개 이미지의 lable이미지
test_Fog_Image = fog_image
# load a single image for testing
test_image = tf.keras.preprocessing.image.load_img(test_Fog_Image, target_size = (150, 150))
# test_image = tf.keras.preprocessing.image.load_img('No_fog_test.jpg', target_size = (150, 150))
test_image = tf.keras.preprocessing.image.img_to_array(test_image)
test_image = np.expand_dims(test_image, axis = 0)
# predict the result
result = model.predict(test_image)
# assuming that the class indices are {0: 'no_fog', 1: 'fog'}
ifresult[0][0] == 1:
prediction = '안개 없음'
else:
prediction = '안개 있음'
print(prediction)
```

Haze image

Dehazed image

실제 마스킹 이미지	





각 이미지에 대한 IoU값

```
1/1 [==============================] - 0s 225ms/step
data\dehazed_image.jpg's IoU_Value: 0.9149979745161461
1/1 [==============================] - 0s 264ms/step
data/make_foggy_image.jpg's IoU_Value: 0.8202321800797318
```

5. 결론