In [1]:
```
import os
import random
import numpy as np
from tensorflow.keras.preprocessing.image import img_to_array, load_img, save_img


# 원본 데이터 경로 및 증강 데이터 저장 경로
base_dir = '/Users/User/Desktop/Augmentation/processing/test'  # 원본 이미지가 있는
output_dir = '/Users/User/Desktop/Augmentation/test/sunrise'  # 증강된 이미지를 저장
classes = ['sunrise']  # 클래스 목록
#'rain', 'clear', 'cloudy'
```

In [3]:
```
# 이미지 밝기 및 색상 변환 함수
def adjust_brightness(image, factor):
    """
    이미지의 밝기를 조정하는 함수
    :param image: 입력 이미지 (numpy array)
    :param factor: 밝기 조정 비율
    :return: 밝기 조정된 이미지
    """
    return np.clip(image * factor, 0, 255).astype(np.uint8)

# 좌우 반전 함수
def flip_image(image):
    """
    이미지를 좌우 대칭으로 반전하는 함수
    :param image: 입력 이미지 (numpy array)
    :return: 좌우 반전된 이미지
    """
    return np.fliplr(image)

# 증강 실행
for class_name in classes:
    input_path = os.path.join(base_dir, class_name)
    output_path = os.path.join(output_dir, class_name)
    os.makedirs(output_path, exist_ok=True)

    # 경로 확인
    if not os.path.exists(input_path):
        print(f"경로가 존재하지 않습니다: {input_path}")
        continue

    # 원본 이미지 읽기
    for img_name in os.listdir(input_path):
        img_path = os.path.join(input_path, img_name)
        if img_name.lower().endswith(('.png', '.jpg', '.jpeg')):  # 이미지 파일만 처
            img = load_img(img_path)  # 이미지 로드
            x = img_to_array(img)  # 이미지를 Numpy 배열로 변환

            # 1. 원본 이미지 저장
            save_path_original = os.path.join(output_path, f'aug_original_{img_name
            save_img(save_path_original, x)

            # 2. 좌우 반전 이미지 저장
```

```
            flipped_img = flip_image(x)
            save_path_flipped = os.path.join(output_path, f'aug_flipped_{img_name}'
            save_img(save_path_flipped, flipped_img)

            # 3. 밝기/색상 변경된 원본 이미지 저장
            factor_original = random.uniform(0.8, 1.2)  # 밝기 비율 랜덤 조정
            brightened_original = adjust_brightness(x, factor_original)
            save_path_brightened_original = os.path.join(output_path, f'aug_brighte
            save_img(save_path_brightened_original, brightened_original)

            # 4. 밝기/색상 변경된 좌우 반전 이미지 저장
            factor_flipped = random.uniform(0.8, 1.2)  # 밝기 비율 랜덤 조정
            brightened_flipped = adjust_brightness(flipped_img, factor_flipped)
            save_path_brightened_flipped = os.path.join(output_path, f'aug_brighten
            save_img(save_path_brightened_flipped, brightened_flipped)
```

In [46]:

In [ ]: