

산업 컴퓨터 비전 실제

중간 프로젝트

2023254006 이선경

CONTENTS

01. 주제 선정

02. 알고리즘

03. 결과

04. 향후 연구 방향

01. 주제 선정

1. 주제

- 비행기 이미지에서 전경과 배경의 세분화를 통한 객체 식별

2. 주제 선정 이유

현업

Prepar3D 엔진을 사용하여
지상통제장치, 조종석 등 개발



이유

공항 보안 및 항공 교통
관리의 효율성을 높이기
위한 기술적 도전 과제로,
비행기와 그 배경을 정확
하게 분리하고 식별하는
것은 매우 중요함

1. GrabCut

이미지에서 객체의 정확한 실루엣을 추출할 수 있어 복잡한 배경 속에서도 비행기를 효과적으로 분리할 수 있으므로 GrabCut 사용

2. K-means 클러스터링

색상 데이터를 기반으로 이미지를 세그먼트로 나누어 각 구성 요소를 더욱 명확하게 구분할 수 있으므로 K-means 클러스터링 사용

02. 알고리즘

3. 코드

```
def apply_kmeans_and_return_labels(image, n_clusters=2):
    data = image.reshape((-1, 3))
    data = np.float32(data)
    kmeans = KMeans(n_clusters=n_clusters, random_state=0).fit(data)
    labels = kmeans.labels_
    return labels.reshape(image.shape[:2])

def visualize_with_original_colors(image, labels):
    unique_labels = np.unique(labels)
    segmented_image = np.zeros_like(image)
    for label in unique_labels:
        mask = labels == label
        segmented_image[mask] = image[mask]
    return segmented_image

def grabcut_and_cluster(image):
    mask = np.zeros(image.shape[:2], np.uint8)
    bgdModel = np.zeros((1, 65), np.float64)
    fgdModel = np.zeros((1, 65), np.float64)
    rect = (5, 5, 620, 410)
    cv2.grabCut(image, mask, rect, bgdModel, fgdModel, 5, cv2.GC_INIT_WITH_RECT)
    mask2 = np.where((mask == 2) | (mask == 0), 0, 1).astype('uint8')
    foreground = image * mask2[:, :, np.newaxis]
    background = image * (1 - mask2)[:, :, np.newaxis]

    foreground_labels = apply_kmeans_and_return_labels(foreground)
    background_labels = apply_kmeans_and_return_labels(background)
    foreground_clustered = visualize_with_original_colors(foreground, foreground_labels)
    background_clustered = visualize_with_original_colors(background, background_labels)

    plt.figure(figsize=(10, 5))
    plt.subplot(1, 3, 1)
    plt.imshow(cv2.cvtColor(foreground_clustered, cv2.COLOR_BGR2RGB))
    plt.title('Foreground (Original Colors)')

    plt.subplot(1, 3, 2)
    plt.imshow(cv2.cvtColor(background_clustered, cv2.COLOR_BGR2RGB))
    plt.title('Background (Original Colors)')

    plt.subplot(1, 3, 3)
    plt.imshow(cv2.cvtColor(image, cv2.COLOR_BGR2RGB))
    plt.title('Original Image')

    plt.show()

image_path = '../data/airplane1.jpg'
image = cv2.imread(image_path, cv2.IMREAD_COLOR)
if image is None:
    print("이미지를 로드할 수 없습니다. 경로를 확인하세요.")
else:
    image = cv2.resize(image, (660, 440))
    grabcut_and_cluster(image)
```

K-means 클러스터링 적용



원본 색상 유지

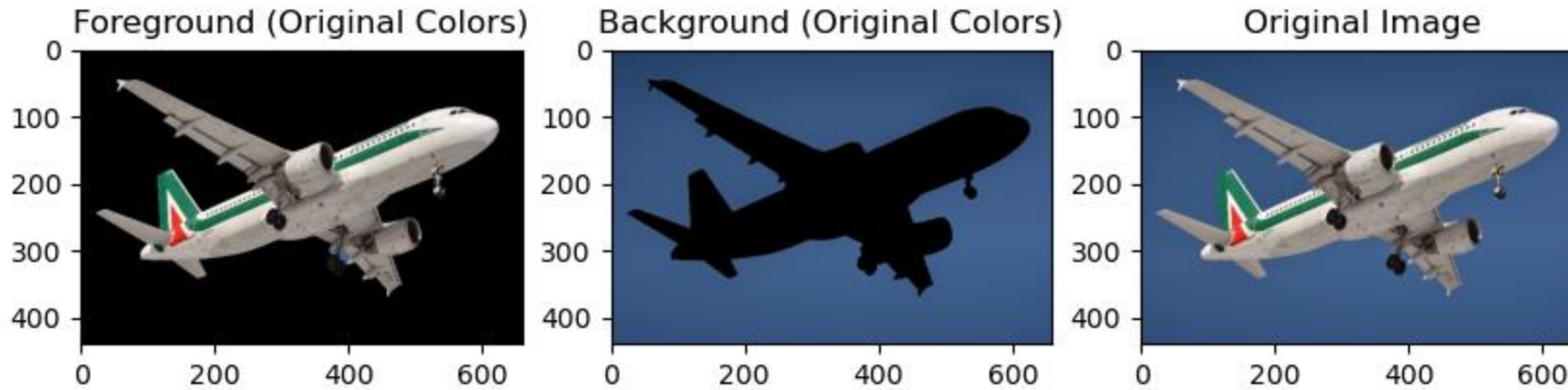


GrabCut 사용하여 전경, 배경 분리



시각화

1. 결과 시각화



전경 – GrabCut 알고리즘 이미지 비행기 분리 및 K-means 클러스터링 적용
배경 – 전경에서 분리된 후 동일한 클러스터링 과정 적용

2. 분석

전경 처리 결과

- 비행기 이미지의 전경 성공적으로 추출, K-means 클러스터링을 통해 세부적인 색상 구조 유지, 명확하게 구분되는 색상 영역으로 나눔

배경 처리 결과

- 상대적으로 단순화되어 전경의 비행기와 대조적으로 잘 나타남

3. 고찰

성능 평가

- GrabCut은 복잡한 배경 속에서도 전경 객체를 효과적으로 분리할 수 있으나, 초기 사각형 위치와 크기에 따라 결과가 달라질 수 있음. 최적의 결과를 얻기 위해서는 사각형 영역을 적절히 설정하는 것이 중요함

K-means 클러스터링 효과, 한계

- 이미지의 다양한 요소를 강조할 수 있지만, 클러스터 수의 선택이 결과에 큰 영향을 미침. 클러스터 수가 적절하지 않으면 세부적인 특징이 손실될 수 있음

향후 개선 방안

- 사용자가 사각형 영역을 보다 정밀하게 조정할 수 있도록 개선해야 함.
- 클러스터 수를 자동으로 결정하는 기법을 도입하여 더 강력한 도구를 개발

1. 다양한 환경에서의 적용성 향상

다양한 조명 조건, 다중 객체 환경, 다양한 배경을 포함하는 이미지 데이터셋에 대한 알고리즘의 적용성을 테스트하고, 이러한 복잡한 환경에서도 높은 성능을 유지할 수 있는 기법을 개발

2. 상업적 및 산업적 응용 분야 확장

비행기 식별 및 분리 기술을 공항 보안 검사, 항공 교통 관리, UAV 모니터링 시스템에 적용하여, 이 기술의 상업적 및 산업적 가치를 높이는 방안 모색