

기말 프로젝트

디지털 영상처리(3192) 202011376 조현서

1. 문제 정의 및 개발 환경

한 학기 동안 배운 point, area, frame processing을 이용하여 video에서의 객체를 검출하는 과제입니다.

본 과제는 Hand Detection in Video, Vehicle Detection in Video두 가지의 문제로 정의되어 있습니다.

본 과제를 수행함에 있어 이용한 개발 환경은 colab default 환경으로 python 3.7, numpy와 opencv 패키지들입니다.

2. 해결 방안

A. Hand Detection

손은 사람의 행동을 인식하는데 있어 핵심적인 요소입니다.

손의 움직임이나 모양 등의 특징을 이용하여 사람의 행동, 상태 등의 다양한 특징들을 예측할 수 있습니다.

이 문제는 위 과정들의 처음 단계에 해당하는 문제로, 초록색 배경을 가지고 있는 비디오에서 객체인 손을 검출하는 문제이며 다음과 같은 순서로 해결할 수 있습니다.

1. 전처리
2. 피부색 정의와 Thresholding
3. Morphology
4. Connected Component Labeling

이들의 세부적인 내용은 다음과 같습니다.

1. 전처리

노이즈를 제거하기 위한 Gaussian Smoothing을 처리해 줍니다.

실내 환경이기에 빛의 밝기 분포가 일정하며, 밝기가 어둡거나 밝아 잘 안보이는 객체가 존재하지 않는 단순한 상황이기에 대비 개선은 진행하지 않습니다.

2. 피부색 정의와 Thresholding

피부색을 배경색과 구분되는 색으로 정의하되, intensity는 인종별로 차이가 존재하므로 고려하지 않았습니다. 이 색에 해당하는 색을 255, 해당하지 않는 배경을 0으로 처리해 줍니다.

3. Morphology

Close operation을 이용하여 존재할 수도 있는 미세 구멍들을 제거해 줍니다.

4. Connected Component Labeling

검출된 객체를 이어진 구성 요소 별로 묶어주는 단계입니다. 해당 문제에서는 하나의 객체와 배경이 존재하기에 배경을 제외한 구성 요소는 하나가 나오게 됩니다.

5. Boxing

CCL을 통해 얻은 객체들의 좌표를 이용하여 빨간색 상자를 표시합니다.

B. Vehicle Detection

자동차는 환경 오염, 교통 사고 및 교통 체증 등의 다양한 사회 문제를 야기합니다.

CCTV를 통한 도로 상황 분석은 저러한 문제들을 해결하는데 큰 도움이 될 수 있습니다.

두 번째 문제는 차량 인식 문제로 이러한 분석의 시작 단계로 볼 수 있으며 다음과 같은 순서로 해결할 수 있습니다.

1. 전처리
2. 배경 정의
3. 배경과 프레임 사이의 차이를 이용한 차량 인식
4. 이전 프레임과의 차이를 이용한 모션 인식
5. Morphology
6. Connected Component Labeling

이들의 세부적인 내용은 다음과 같습니다.

1. 전처리

Hand detection에서와 마찬가지로 노이즈를 제거하기 위한 Gaussian Smoothing을 처리해 줍니다.

이는 Medoan Smoothing보다 효과는 더 안좋을 수 있지만 cctv video를 realtime으로 처리해 줘야 하는 것을 고려했을 때 더 합리적인 선택이라고 할 수 있습니다.

또한 CCTV는 외부에 있기 때문에 빛의 분포가 매 순간 일정하지 않으며 시간 때마다 다양할 수 있으므로 그 패턴 또한 일정하지 않습니다.

따라서 Histogram Equalization을 이용하여 이러한 문제점을 해결해 줍니다.

2. 배경 정의

배경은 각 preprocessing이 끝난 frame들의 median을 이용합니다. 연산이 다소 오래걸릴 수 있다는 단점이 있지만 한번 정의하면 he과정을 거쳤기에 계속해서 쓸 수 있으니 이 방법을 선택합니다.

이 상태에서 vehicle을 찾되, 마지막으로 boxing이미지로 표현할 때에는 원본이미지에 표현합니다.

3. 배경과 프레임 사이의 차이를 이용한 차량 인식

배경 - 프레임을 빼 줍니다.

이 때, 절댓값을 이용하지 않고 부호까지 고려하여 이 값이 threshold를 넘으면 차량이라고 생각합니다.

이를 통해 절댓값으로 thresholding 한 경우보다 그림자를 더 쉽게 제거할 수 있습니다.

단점으로는 어두운 색의 차량이 잘 검출이 안될 수 있다는 점인데, 이는 이후에 morphing을 통해 해결할 수 있습니다.

4. 이전 프레임과의 차이를 이용한 모션 인식

이전 프레임과 현재 프레임의 절댓값의 차이를 이용하여 이전 프레임과 현재 프레임 사이의 모션을 검출할 수 있습니다.

이 때 모션이라고 인식하는 기준인 threshold는 3에서 이용한 threshold와는 다른 값을 이용하여 표현 가능한 범위를 늘릴 수 있습니다.

여기에 3에서 얻은 결과를 and 연산을 하여 차량을 보다 더 섬세하게 인식할 수 있습니다.

3,4 단계에서의 hyper parameter에 해당하는 threshold 값은 관찰을 통한 경험적인 방법으로 설정합니다.

5. Morphology

우선 Open 과정을 통해 작은 점에 해당하는 noise를 제거합니다.

이후 좀 더 큰 kernel을 이용하여 Close과정을 진행하여 작은 구멍들을 제거할 수 있습니다.

5 단계에서의 hyper parameter에 해당하는 iteration, kernel 사이즈 등은 관찰을 통한 경험적인 방법으로 설정합니다.

6. Connected Component Labeling

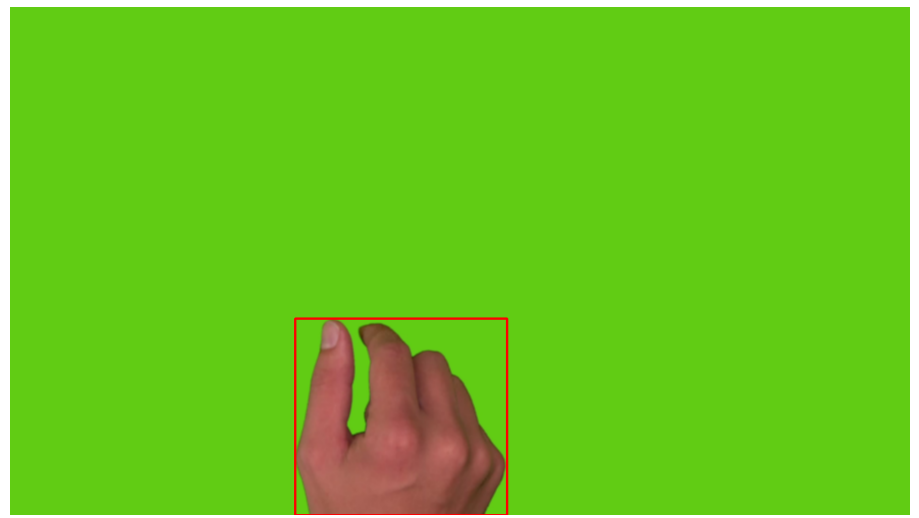
검출된 객체를 이어진 구성 요소 별로 묶어주는 단계입니다. 해당 문제에서는 각각의 차량에 해당하는 객체들이 연결되어 나오게 됩니다.

7. Boxing

CCL을 통해 얻은 객체들의 좌표를 이용하여 전처리를 거치기 이전(gaussian, he 등)의 원본 이미지에 빨간색 상자를 표시합니다.

3. 수행 예 및 결과 분석

A.

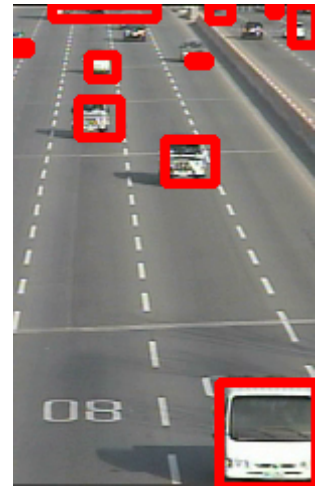
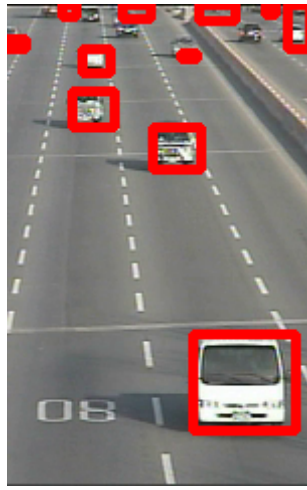
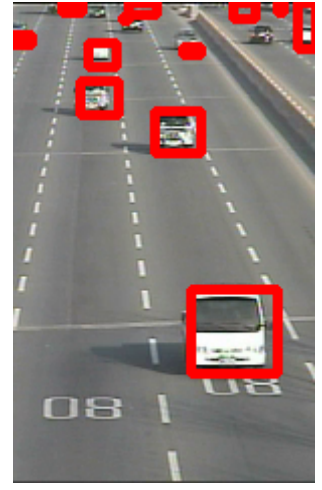
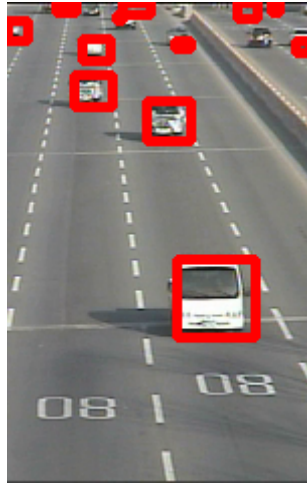
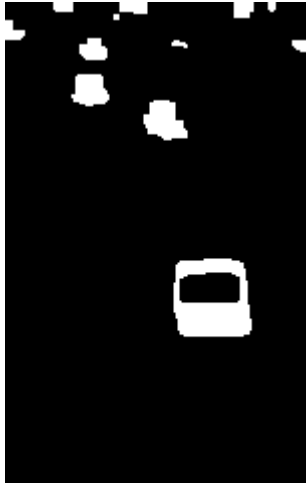


B.

1. Back Ground (Gaussian blurring, Histogram Equalization 수행)



2. 수행 예



두 문제 모두 성공적으로 검출한 것을 확인할 수 있습니다.

붙임. 소스코드 1부 끝.