

테스트 보고서

모바일시스템공학과 : 32157629 한송이

32161620 박산희

소프트웨어학과 : 32157022 고수열

32157204 박상준

Test.1 이미지의 전체 Image Transfer

-이미지 변환코드가 잘 작동되는지 테스트.

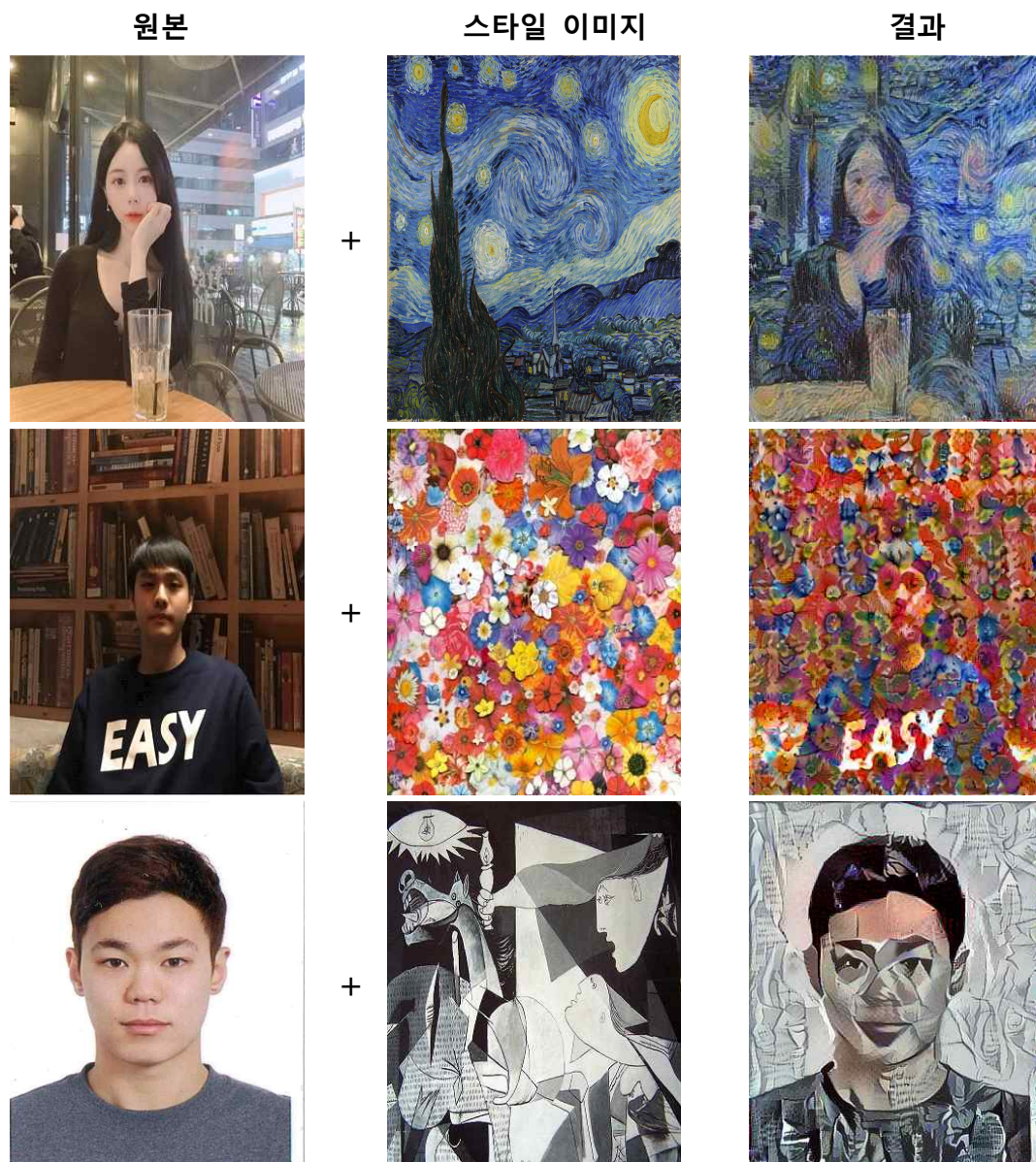
-원본의 이미지가 변환에 영향을 주는지 테스트

-스타일 이미지가 변환에 영향을 주는지 테스트

-알파와 베타 테스트

(완성된 이미지가 원본에 가깝게 합성될지, 스타일 이미지의 가깝게 합성될지)

(1) 여러사진 합성해보기(조원의 사진을 이용)





+



-테스트 결과 : 이미지 변환 과정은 다 잘 작동한다.

그리고 두 번째 사진에서 꽃 형태의 스타일 사진으로 합성 시에는 원본 이미지의 형태가 거의 사라졌다. 알파와 베타를 조정해 테스트하여 원본 이미지의 형태가 잘 유지되도록 기술 보완이 필요하다.

(2) 여러 스타일의 사진을 해보기(스타일 이미지의 영향)



-테스트 결과 : 콘텐츠 이미지가 스타일 이미지에 따라 바뀐다.

이 과정에서 콘텐츠 이미지의 형태나 질감 부분도 다 달라진다. 스타일이 어떤 화풍, 어떤 형태인지에 따라 많은 영향을 주는 것으로 나타났다.

(3)알파와 베타를 조정해

합성이 한쪽에 치우쳐지는 것이 조절 가능한지 테스트

알파:5, 베타 100

알파 20, 베타 100

알파 50, 베타 100



-테스트 결과 : 알파값을 늘리면 스타일 이미지에 더 가까운 형태로 합성이 되어야 한다. 그래서 알파 값을 점차 늘려서 테스트하였다. 완성된 사진의 형태가 바뀌긴 하지만 눈에 띄게 변화하진 않는다.

* 시스템 설계보고서 참고.(알파와 베타관련 내용)

*이미지 정의

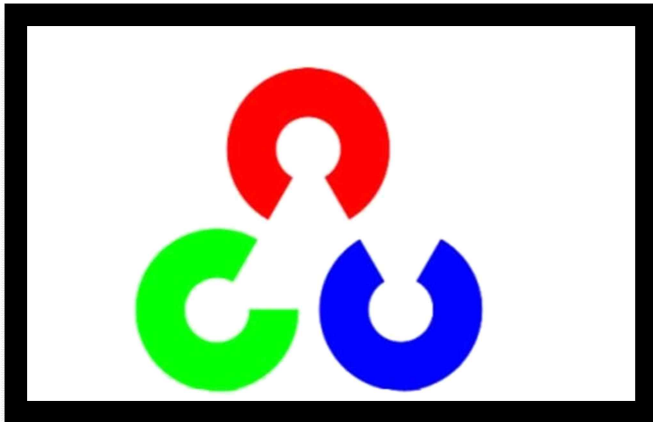
```
#####  
# Algorithm constants  
#####  
# Noise ratio. Percentage of weight of the noise for intermixing with the  
# content image.  
NOISE_RATIO = 0.6  
# Constant to put more emphasis on content loss.  
BETA = 5  
# Constant to put more emphasis on style loss.  
ALPHA = 100  
# Path to the deep learning model. This is more than 50MB so will not be  
# included in the repository, but available to download at the model Zoo:  
# Link: https://github.com/BVLC/caffe/wiki/Model-Zoo  
#  
# Pick the VGG 19-layer model by from the paper "Very Deep Convolutional  
# Networks for Large-Scale Image Recognition".  
VGG_MODEL = 'imagenet-vgg-verydeep-19.mat'  
# The mean to subtract from the input to the VGG model. This is the mean that  
# when the VGG was used to train. Minor changes to this will make a lot of  
# difference to the performance of model.  
MEAN_VALUES = np.array([123.68, 116.779, 103.939]).reshape((1,1,1,3))
```

*위에 α/β 값이 작아질수록 content보다는 style에 더 치중된 결과

Test2. 이미지 내의 부분 객체 따기 (부분합성 관련)

- 이미지 내의 객체의 경계를 잘 인식하는지 테스트
- 이미지 내의 구별된 객체마다 레이블이 잘 붙여지는지 테스트

(1)OpenCV 로고 이미지



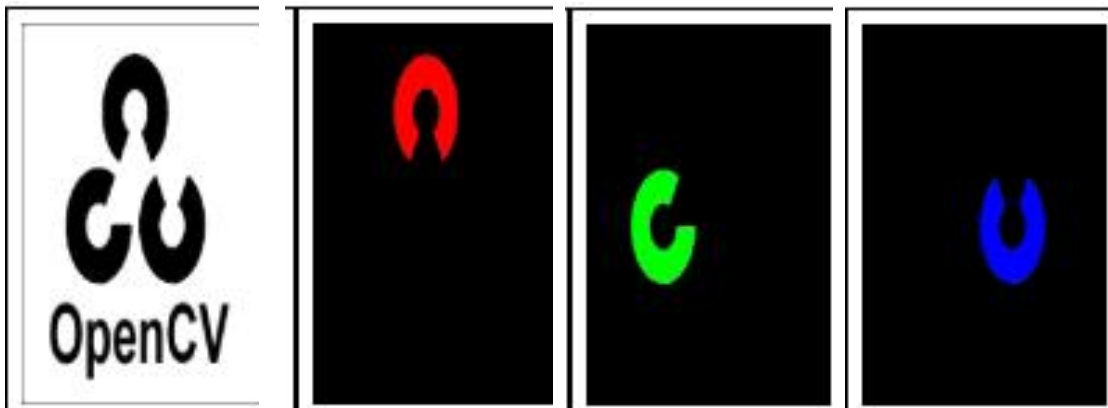
-객체마다 경계를 구분하여 레이블을 매겼다.

1번 레이블(배경)

2번 레이블

3번 레이블

4번 레이블



-폴더 내에 저장 형식

ret(번호).jpg 형태



ret1.jpg



ret2.jpg



ret3.jpg



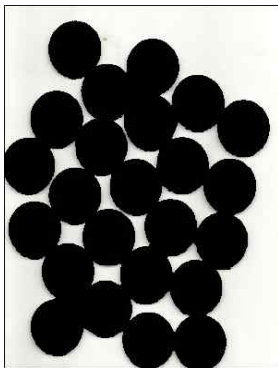
ret4.jpg

(2)Coin 이미지



-객체마다 경계를 구분하여 레이블을 매겼다.

1번 레이블



2번 레이블



3번 레이블



-폴더 내에 저장 형식

ret(번호).jpg 형태



ret4.jpg



ret5.jpg



ret6.jpg



ret7.jpg

(3) 사람 이미지

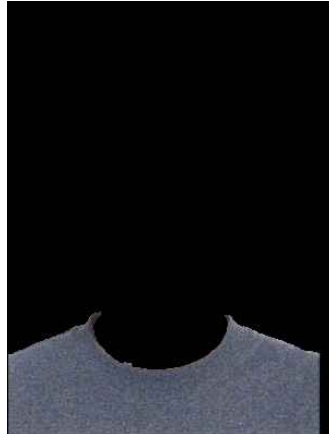


-객체마다 경계를 구분하여 레이블을 매겼다.

1번 레이블



2번 레이블



-폴더 내에 저장 형식

ret(번호).jpg 형태



ret1.jpg



ret2.jpg

-테스트 결과 : 각각의 사진마다 객체를 따는 부분이 엣지(Edge)가 잘 분리되지 않는 부분이 부분적으로 있다. 그러나 합성 시에는 크게 문제가 되진 않았다. 엣지를 기준으로 나눈 객체 마다의 레이블 값들이 적절히 적용되었고 파일저장 부분도 각 레이블 값에 따라 서버에 잘 저장되었다.

하지만, 사람 같은 색상의 경계가 확실하지 않고, 깔끔하지 않은 경우에는 객체의 추출이 어려웠다.

따라서 객체를 더 잘 따는 방법을 보완해야 할 것 같다.

Test3. 부분 객체 이미지 합성

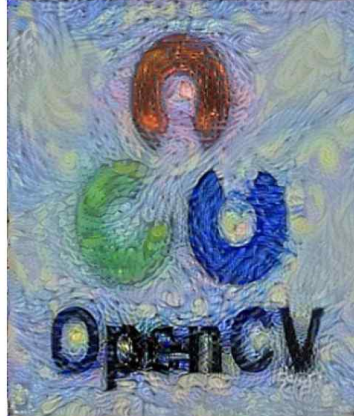
- Test2과정을 통해 구별된 레이블들을 결합 테스트
- 사진의 부분이 적절한 위치에 위치하여 합성되는지 테스트
- 합성 시 색이나 질감부분 테스트

(1). Image Transfer의 완성된 이미지 준비

-woman



-Stary_night



-flower



(2). Test2.과정 진행 (객체 생성)

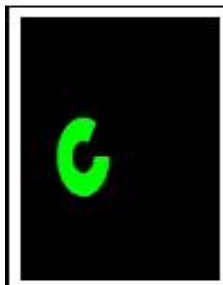
레이블1



레이블2



레이블3



레이블4



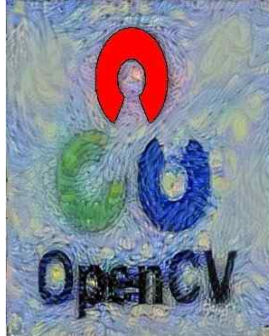
(3). (1)과정의 완성 이미지와 (2) 객체의 합성

Stary_night

레이블 1번과 합성



레이블 2번과 합성



레이블 3번과 합성

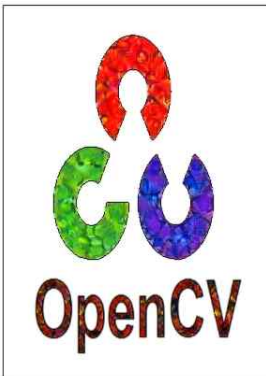


레이블 4번과 합성

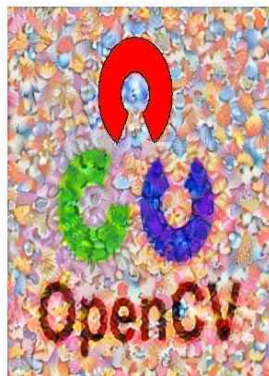


flower

레이블 1번과 합성



레이블 2번과 합성



레이블 3번과 합성

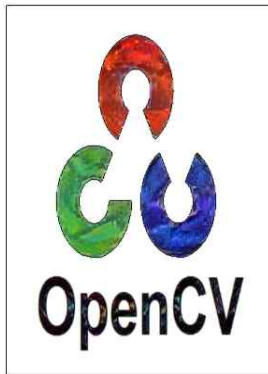


레이블 4번과 합성



woman

레이블 1번과 합성



레이블 2번과 합성



레이블 3번과 합성



레이블 4번과 합성



-테스트 결과 : Image Transfer된 이미지와 각각의 객체의 이미지들을 이미지 합성하였고 두 이미지 합성을 통해 부분합성을 나타냈다.
테스트 결과 이미지 합성 과정에서 알맞은 위치에 적절히 합성이 되었다.
그러나 부분합성 시 이미지 합성 방식을 이용하다 보니 질감 부분에서는 아직 보완점이 필요하다.

Test4.명도 / 채도 / RGB

- 명도가 입력 값에 따라 적절히 바뀌는지 테스트
- 채도가 입력 값에 따라 적절히 바뀌는지 테스트
- RGB가 입력된 이미지에 맞는 색상표가 나오는지 테스트

(1) 원본이미지



(2) 명도, 채도 수정

CHROMA : 45



BRIGHTNESS : 55



CHROMA : -50



BRIGHTNESS : 61



CHROMA : 99



BRIGHTNESS : 99



CHROMA : -90



BRIGHTNESS : -91



(1-3)원본 이미지의 RGB 표



(2-1) 원본이미지



(2-2) 명도, 채도 수정

CHROMA : 51

BRIGHTNESS : -68

CHROMA : 29

BRIGHTNESS : -24



CHROMA : 99

BRIGHTNESS : 99

CHROMA : -99

BRIGHTNESS : -99



(2-3) 원본 이미지의 RGB 표



(3-1) 원본 이미지



(3-2) 명도, 채도 수정

CHROMA : 43



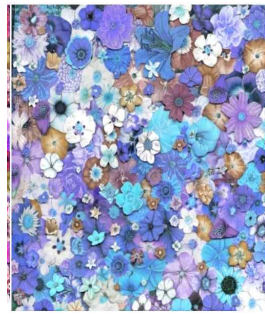
BRIGHTNESS : -34



CHROMA : -35



BRIGHTNESS : 48



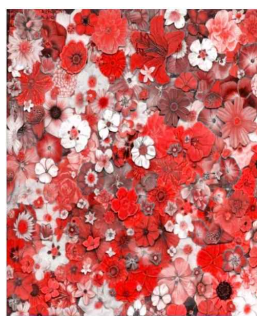
CHROMA : 99



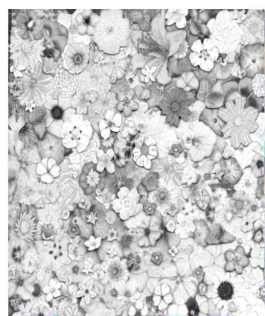
BRIGHTNESS : 99



CHROMA : -99



BRIGHTNESS : -99



(3-3)원본 이미지의 RGB 표



-테스트 결과 : 각각의 이미지마다 RGB 표를 출력할 때에 opencv.jpg같은 단순한 색의 이미지는 표가 잘 나오지만 다양한 색들이 들어있는 이미지들은 표에서 색이 섞여 나오는 특성을 발견했다. 이 부분은 나중에 보완이 필요할 것으로 보인다. 또한, 스타일이미지의 명도와 밝기를 수정하면 새로운 스타일 이미지를 만들 수 있다는 좋은 테스트 결과도 나왔다.

Test 4. 웹 버튼 테스트

1. Image Upload

Select Image.

파일 선택 선택된 파일 없음

Upload Image

(1) ImageUpload

파일선택이 잘 되는지, 서버에 업로드가 잘 되는지 확인

-테스트 결과 : 정상적으로 잘 선택된다. 하지만 사진의 크기가 500KB를 초과하면 선택되지 않고, 사진의 확장자도 JPG, JPEG, PNG 외에는 업로드 되지 않는다. 업로드가 정상적으로 실행되면, 서버의 upload 폴더에 생성된다. (참고로, 사진의 크기의 변경은 윈도우의 기본 프로그램인, 그림판을 이용하면 된다.)

Upload image

Image uploaded successful
File Name: **upload/Upload_test.jpg**
Type: image/jpeg
Size: 223.11 kB
Temp file: C:\Bitnami\wampstack-7.1.29-0\php\tmp\php194A.tmp

*업로드 성공했을 때 나오는 그림

Invalid image format. Allowed formats: JPG, JPEG, PNG.

*선택한 이미지의 확장자가, 지원하는 확장자가 아닐 때

The size of image you are attempting to upload is 4311.74 KB, maximum size allowed is 500 KB

*선택한 이미지의 크기가 500KB를 초과할 때

1. Image Upload

Select Image.

Selected image:



파일 선택 | 파일명: jpg

Upload image

Select Image.

Selected image:



파일 선택 | 파일명: ver2.jpg.jpg

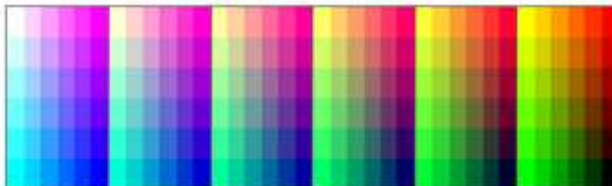
Upload image

(2) Upload image 버튼

선택한 파일이, 서버로 잘 전송되는지 테스트.

-테스트 결과 : Upload image 버튼을 클릭하면 웹 서버로 잘 전송된다.

RGB Color



* RGB Recommand

choose image name:

ex) myFace.jpg ...

쿼리 전송

(3) RGB Recommand 쿼리전송

업로드된 사진에 대해 RGB 추천이 잘 되는지 테스트. 업로드된 사진에 대하여 RGB를 잘 추천해 준다. (파일이름을 입력하고 뒤에 확장자 명까지 붙여줘야 한다.)



RGB Recommand



-테스트 결과 : 업로드된 사진의 RGB의 결과를 추출해서 표로 잘 나타내 준다. 하지만, 사진의 RGB가 5개의 값을 넘어가는 경우에는, 섞여서 나오기 때문에 보색의 영향을 받을 수 있다. 눈에 보이는 색과는 다르게, 보색 때문에 어두운 색깔이 나올 수도 있다.

GAJAH Mixing

(4) Mixing

Contents Image:

ex) myFace.jpg ...

(4-1) Contents image : 파일명을 입력하고 뒤에 확장자까지 입력을 하면, Style image와 합성 할 준비가 된 것이다.

-테스트 결과 : 파일명이 영어 일 경우는 문제없이 잘 선택이 되지만, 한국어 인 경우는 인식하지 못한다.

Style Image:

ex) woman.jpg ...

==(flower.jpg)=-----==(woman.jpg)=-----==(Stary-night.jpg)=



(4-2) Style image : 위 그림들중에서 Contents image 와 합성하고싶은 배경을 Flower.jpg / woman.jpg / Start-night.jpg 중에 하나 선택한다.

-테스트 결과 : 파일명이 영어 일 경우는 문제없이 잘 선택이 되지만, 한국어 인 경우는 인식하지 못한다.

DO YOU WANT TO SKIP?

SKIP

(4-3) **SKIP 버튼** : 부분합성을 안 하고 싶거나, CHROMA & BRIGHTNESS (채도 & 명도)를 바꾸고 싶지 않다면, SKIP 한다.

-테스트 결과 : 정상적으로 작동되며, 버튼을 누르면 다음 페이지로 넘어간다.



(5) **create object** : 위 사진 왼쪽에 create object 버튼을 클릭하면, 오른쪽 사진처럼 각 레이블에 객체별로 따로따로 나뉘어서 나온다.

-테스트 결과 : create object!! 버튼을 누르면 content 이미지에 대한 객체가 출력되어서 나온다. 이미지를 클릭하면 서버에 저장되는 이름이 알림창으로 발생한다.

which one you want to choose?

ex) 1,2,3... 제출

(6) **which one you want to choose?** 아래에, 숫자 입력 하는 칸 : 숫자는 x 번째 레이블을 의미하며, 객체별로 맞는 숫자x를 입력하면 그 입력된 숫자x의 x번째 레이블의 객체와 이미지 합성이 완성된 사진과 합쳐진다.

-테스트 결과 : 버튼을 누르면, x번의 기능이 잘 작동하는지 테스트.

CHROMA & BRIGHTNESS

Chroma :



Brightness :



개요

(7) CHROMA & BRIGHTNESS (채도 & 명도) : Mixing 된 결과물에 대하여 채도와 명도를 조절해 준다.

CHROMA : 51 BRIGHTNESS : -68



CHROMA : 29 BRIGHTNESS : -24



CHROMA : 99 BRIGHTNESS : 99



CHROMA : -99 BRIGHTNESS : -99



-테스트 결과 : 채도와 명도가 정상적으로 잘 조절되며, 채도와 명도에 따라 이미지의 전반적인 느낌이 달라진다 (채도와 명도의 범위는 -100 ~ 100까지다). 명도를 조절했을 때, 가끔 이미지가 깨지는 현상이 발생한다.