

HW 01 – REPORT

소속 : 조선해양공학과

학번 : 201629172

이름 : 차현대

1. 서론

실습 목표 및 이론적 배경 기술 (1~2페이지)

이미지가 픽셀로 이루어져 있다는것을 인식하고, 이미지들의 픽셀값을 변화시키면 이미지도 바뀌는것을 이해하는것, 그리고 이미지를 바꾸는 방법에는 linear한 방법과 nonlinear한 방법이 있다는것을 배경지식으로 두고, 이미지를 바꾸는 함수를 한번 사용함으로써, 어떻게 이미지가 변화하는지 한번 체험해보는 것이 이번 실습의 목표이다. 이러한 실습을 하기위해선 다음과 같은 이론적 배경지식이 필요하다.

이미지는 intensity value들의 matrix이다. 이 이미지들중 grayscale 같은경우는 intensity value를 0 ~ 255 사이값으로 표현하고, Color image 같은 경우는 RGB 또는 HSV로 표현을 하게 된다. 이미지는 함수로 표현이 가능한데, 이를 이용해 이미지에 연산자를 적용해서 이미지를 변경할 수 가 있다. 예를들어, 함수에 따라 밝기를 변화시킬 수 도 있고, 이미지를 좌우반전 또는 상하반전 시킬 수 도 있다. 필터를 사용해 이미지에 다양한 작업을 할 수 있다. 예를들어, edge나 contour를 뽑아낼 수 도있고, 노이즈를 제거하거나, 이미지를 선명하게 만들 수 도 있다. 이런 필터링의 가장 기본적인 연산은 linear filtering의 종류중 하나인 cross-correlation 또다른 하나인 convolution이 있다. 이 linear filtering은 구동 방식이, 한 픽셀의 주위의 값에 가중치를 두어서 합을 구하게 되는데, 이때 가중치 행렬을 kernel이라고 한다.

2. 본론

실습 내용 및 결과 기술 (2페이지 이상)

파이썬 이미지 처리를 위한 라이브러리 PIL과 고성능의 수치계산을 위한 numpy 라이브러리를 불러준다.

PIL을 이용해 사용할 첫번째 메서드는 convert이다.

이 convert 메서드의 인자에 따라, 이미지가 gray scale로 변경될 수 도 있고, 다른 이미지로 변환될 수 도 있다.

다음 메서드는 crop이다.

이 메서드는 4 개의 인자가 필요한데, 차례로 가로 시작점, 세로 시작점, 가로 범위, 세로 범위이다. 현재 chipmunk 의 dimension 이 width:750, height: 599 인데, 인자에 현재 적절한 값을 넣어서 Chipmunk 의 머리를 잘라냈다. 그리고 save 메서드를 이용해 이미지를 chipmunk_head.png 으로 저장한다. 다음 작업은 numpy 라이브러리를 사용하기 위해 img array 를 numpy 형식의 array 로 바꾼후 작업을 한다. 이번 작업은 , gray scale 의 pixel 값들의 intensity 를 올려서, 이미지를 밝게

만들어 줄 것이다. 각 픽셀의 값에 50 을 더해주고, 그 값이 250 이 안된다면, 그 때 픽셀값을 250 으로 맞추도록 할 것이다. 그리고 이 이미지를 저장한것이 chipmunk_head_bright.png 이다. 다음작업은 픽셀값의 intensity 를 절반으로 해서 이미지를 어둡게 만든 chipmunk_head_dark.png 이다.

다음은 chipmunk.png 활동과는 다른 항목인데, image 의 brightness 변화를 서서히 보여주는 것이다. 2 차원 이미지를 만들어주기 위해서 , 256 x 256 짜리 행렬을 만들어, 그것을 이미지화 했다. 물론, 행은 서서히 픽셀 값이 증가하고, 이러한 행들이 열로 256 개 있는것이다. 그결과 , 서서히 밝아지는 gradient image 가 나타난다.

3. 결론

토의 및 결론 (1페이지)

convert the array back to a uint8 array so we can write to a file

im4_array = im4_array.astype('uint8') 이 코드가 잘 이해가 안간다. 왜냐하면, chipmunk_head_bright.png 을 저장할때는 pixel 값들을 uint8로 바꾸지는 않았는데, intensity를 절반값으로 했을때 저장을 하려면, uint8로 변환해야 하기 때문이다. 사실, 두 이미지의 차이는 pixel 값뿐인데, 차이를 인지를 못하겠다. Uint8로 하면, 부호가 없는 정수이기 때문에 이미지 픽셀값이 오버로드가 되어도 양수값을 나타내기 때문에 라고 보는게 합리적인것 같다.

이번 실습을 통해, 픽셀값에 선형적인 연산을 하면서 밝기 조절을 하였다. 다음에는, linear filtering을 더 심화되게 실습을 해서, linear filtering으로 이미지가 어디까지 변화는지 눈으로 직접 확인해보고 싶다.