



# 수치해석 HW#11

RGB, YUV 사이의 상관계수 구하기

Python(opencv) 사용

컴퓨터소프트웨어학부 2015005187 최철훈



# 목차

1. 이미지 확보
2. RGB, YUV포맷으로 분리하기
3. 상관계수 구하기
4. 상관계수 비교하기
5. 마무리

# 이미지 확보 이미지 확보

구글링하여 10개의 다양한 colorful한 이미지를 확보하였다. 적당한 계산량을 위하여 사이즈가 너무 큰 이미지는 적절하게 조절하였다.





# RGB, YUV 포맷으로 변경하기

RGB로 분리

각각의 이미지들을 RGB포맷으로 분리하였다.

```
src = cv2.imread(imagePath, cv2.IMREAD_COLOR)  
b, g, r = cv2.split(src)
```



# RGB, YUV 포맷으로 변경하기 YUV로 분리

각각의 이미지들을 RGB포맷으로 분리하였다.

```
yuv_image = cv2.cvtColor(src, cv2.COLOR_BGR2YUV)  
y, u, v = cv2.split(yuv_image)
```



# 상관계수 구하기 공식 적용

각각의 상관계수를 공식을 그대로 적용하여 구하였다. 추가로 G-Y가 비슷하다고 하여 G-Y사이의 상관계수도 구해보았다.

```
b_mean = np.mean(b)
g_mean = np.mean(g)
r_mean = np.mean(r)
b_var = np.var(b)
g_var = np.var(g)
r_var = np.var(r)
b_sig = math.sqrt(b_var)
g_sig = math.sqrt(g_var)
r_sig = math.sqrt(r_var)
```

```
y_mean = np.mean(y)
u_mean = np.mean(u)
v_mean = np.mean(v)
y_var = np.var(y)
u_var = np.var(u)
v_var = np.var(v)
y_sig = math.sqrt(y_var)
u_sig = math.sqrt(u_var)
v_sig = math.sqrt(v_var)
```

```
#BG
exp = 0
for i in range(0, b.shape[0]):
    for j in range(0, b.shape[1]):
        exp = exp + ((b[i, j] - b_mean) * (g[i, j] - g_mean))
cov = exp/(b.shape[0] * b.shape[1])
corr = cov/(b_sig * g_sig)
print(corr, end=" ")
print("BG")

#GR
exp = 0
for i in range(0, b.shape[0]):
    for j in range(0, b.shape[1]):
        exp = exp + ((g[i, j] - g_mean) * (r[i, j] - r_mean))
cov = exp/(b.shape[0] * b.shape[1])
corr = cov/(g_sig * r_sig)
print(corr, end=" ")
print("GR")

#BR
exp = 0
for i in range(0, b.shape[0]):
    for j in range(0, b.shape[1]):
        exp = exp + ((b[i, j] - b_mean) * (r[i, j] - r_mean))
cov = exp/(b.shape[0] * b.shape[1])
corr = cov/(b_sig * r_sig)
print(corr, end=" ")
print("BR")

#GY
exp = 0
for i in range(0, b.shape[0]):
    for j in range(0, b.shape[1]):
        exp = exp + ((g[i, j] - g_mean) * (y[i, j] - y_mean))
cov = exp/(b.shape[0] * b.shape[1])
corr = cov/(g_sig * y_sig)
print(corr, end=" ")
print("GY")
```

```
#YU
exp = 0
for i in range(0, b.shape[0]):
    for j in range(0, b.shape[1]):
        exp = exp + ((y[i, j] - y_mean) * (u[i, j] - u_mean))
cov = exp/(b.shape[0] * b.shape[1])
corr = cov/(y_sig * u_sig)
print(corr, end=" ")
print("YU")

#YV
exp = 0
for i in range(0, b.shape[0]):
    for j in range(0, b.shape[1]):
        exp = exp + ((y[i, j] - y_mean) * (v[i, j] - v_mean))
cov = exp/(b.shape[0] * b.shape[1])
corr = cov/(y_sig * v_sig)
print(corr, end=" ")
print("YV")

#UV
exp = 0
for i in range(0, b.shape[0]):
    for j in range(0, b.shape[1]):
        exp = exp + ((u[i, j] - u_mean) * (v[i, j] - v_mean))
cov = exp/(b.shape[0] * b.shape[1])
corr = cov/(u_sig * v_sig)
print(corr, end=" ")
print("UV")
```



# 상관계수 구하기 라이브러리 적용

각각의 상관계수를 numpy의 라이브러리를 적용하여 구현해보았다. 라이브러리를 적용하기 위해 이미지 행렬을 벡터로 만들어서 적용하였다.

```
b = b.flatten()
g = g.flatten()
r = r.flatten()

y = y.flatten()
u = u.flatten()
v = v.flatten()

bg_cor = np.corrcoef(b, g)
gr_cor = np.corrcoef(g, r)
br_cor = np.corrcoef(b, r)
yg_cor = np.corrcoef(y, g)
yu_cor = np.corrcoef(y, u)
yv_cor = np.corrcoef(y, v)
uv_cor = np.corrcoef(u, v)
```

```
print(bg_cor[0, 1], end=" ")
print("BG")
print(gr_cor[0, 1], end=" ")
print("GR")
print(br_cor[0, 1], end=" ")
print("BR")
print(yg_cor[0, 1], end=" ")
print("YG")
print(yu_cor[0, 1], end=" ")
print("YU")
print(yv_cor[0, 1], end=" ")
print("YV")
print(uv_cor[0, 1], end=" ")
print("UV")
```

# 상관계수 비교하기

## 공식과 라이브러리 비교

공식을 적용하여 코딩한 결과와 라이브러리를 그대로 사용한 결과가 소수점 아래에서 매우 근소한 차이가 날 뿐 동일한 것을 확인할 수 있다.

### 공식 적용 결과

```
0.6456625015457873 BG
0.6590323474296607 GR
0.1365256114908485 BR
0.9786310258799241 GY
-0.3156456221000481 YU
-0.2563808378507057 YV
-0.5744232275359443 UV
-----image01
0.3971996538664258 BG
0.5914227656811681 GR
-0.304914948322089 BR
0.9486400901264317 GY
-0.46254719680518186 YU
0.32999335277372177 YV
-0.8694663473532375 UV
-----image02
0.7258836779954447 BG
0.8998938446810201 GR
0.6082574545332774 BR
0.986126164703054 GY
-0.44261145280901626 YU
0.4171078930933317 YV
-0.6123208752060899 UV
-----image03
```

### 라이브러리 사용 결과

```
0.6456625015457718 BG
0.6590323474296809 GR
0.13652561149084685 BR
0.9786310258798755 YG
-0.3156456221000369 YU
-0.25638083785070126 YV
-0.5744232275359776 UV
-----image01
0.39719965386643463 BG
0.5914227656811869 GR
-0.3049149483219776 BR
0.9486400901264318 YG
-0.4625471968052116 YU
0.3299933527737148 YV
-0.8694663473532488 UV
-----image02
0.7258836779953339 BG
0.8998938446812536 GR
0.6082574545333765 BR
0.9861261647030859 YG
-0.44261145280912045 YU
0.4171078930932881 YV
-0.6123208752060625 UV
-----image03
```



# 상관계수 비교하기

## 각각의 성분 비교

B-G, G-R, B-R, G-Y, Y-U, Y-V, U-V를 각각 비교하였다. G-Y는 거의 1에 가깝게 나왔다. B-R은 B-G와 B-R보다 상대적으로 낮게 나왔다. U-V는 완전 반대방향으로 나왔다. Y-U, Y-V는 상관성이 많이 떨어지는 것으로 나왔다.

```
0.6456625015457873 BG
0.6590323474296607 GR
0.1365256114908485 BR
0.9786310258799241 GY
-0.3156456221000481 YU
-0.2563808378507057 YV
-0.5744232275359443 UV
```

-----image01

```
0.3971996538664258 BG
0.5914227656811681 GR
-0.304914948322089 BR
0.9486400901264317 GY
-0.46254719680518186 YU
0.32999335277372177 YV
-0.8694663473532375 UV
```

-----image02

```
0.7258836779954447 BG
0.8998938446810201 GR
0.6082574545332774 BR
0.986126164703054 GY
-0.44261145280901626 YU
0.4171078930933317 YV
-0.6123208752060899 UV
```

-----image03

```
0.9509117402526444 BG
0.7138025645021122 GR
0.5592702986121364 BR
0.9847887175679774 GY
0.21604354369131215 YU
-0.41554852576992196 YV
-0.8489898159125263 UV
```

-----image04

```
0.883864442091354 BG
0.8674446806103543 GR
0.735956185335171 BR
0.9868531343169769 GY
-0.2163405475521317 YU
-0.0006996836317533403 YV
-0.5321278190550213 UV
```

-----image05

```
0.9581733192066613 BG
0.9338545465971492 GR
0.8751965352322161 BR
0.9922144392541821 GY
-0.5465680595569099 YU
0.459687907622423 YV
-0.7321659813671313 UV
```

-----image06

```
0.9041648274455543 BG
0.7040868932142357 GR
0.5644108196543604 BR
0.9758699109518212 GY
0.15407142368707968 YU
-0.19437555796640268 YV
-0.6977860814765923 UV
```

-----image07

```
0.6094784741919315 BG
0.6379706592190272 GR
0.2236548714694873 BR
0.9557521595195783 GY
-0.3671141506966867 YU
0.12443193509445684 YV
-0.5961022951510336 UV
```

-----image08

```
0.8102866730803278 BG
0.7540154325626022 GR
0.38379566518432223 BR
0.980334098080355 GY
-0.08307270104577409 YU
0.04523138505940942 YV
-0.8259943206056098 UV
```

-----image09

```
0.688526098544582 BG
0.805130643487534 GR
0.3316257902659889 BR
0.9865764325154874 GY
-0.1718552578943185 YU
-0.11152904550367793 YV
-0.7180977581118291 UV
```

-----image10

## 마무리 알게 된 점

1. G와 Y는 매우 비슷한 성분임을 알 수 있다.
2. G는 B와 R과 어느 정도 상관성이 있지만 B와 R 사이의 상관성이 G에 비해 상대적으로 많이 떨어진다.
3. U와 V는 서로 상관성이 정 반대인 성분이다.
4. Y는 U와 V와 상관성이 거의 없다.

# 마무리

감사합니다.

[https://github.com/cheol-hoon/Numerical\\_Analysis](https://github.com/cheol-hoon/Numerical_Analysis)