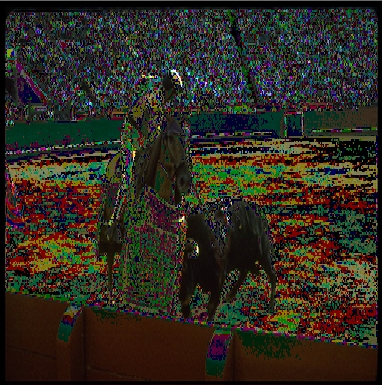
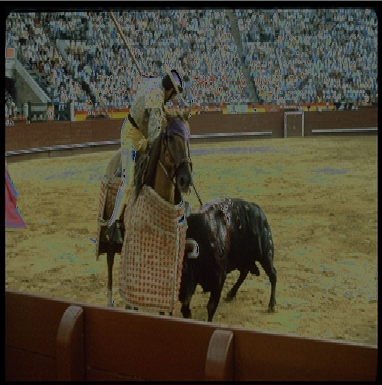
**영상처리 과제 #3**

16010980 이우석

**< 실습 A >**





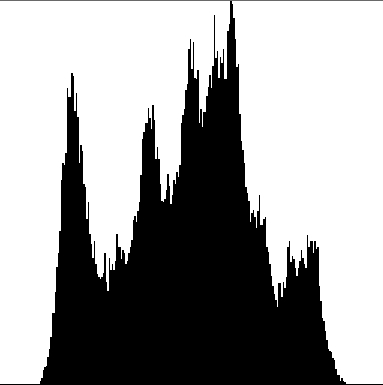




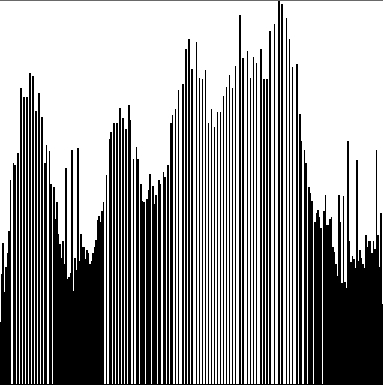
**< 실습 B >**



**< 실습 C >**



**< 실습 D >**



**< 코드 >**

**< List2\_1.c >**

/\* initialize Look-up table \*/

for (i = 0; i < 256; i++)

{

/\* preserve an original image or y is equal to x \*/

temp = i;

/\* Negative image \*/

// temp = 255 - i

/\* Contrast Stretching \*/

// temp = (float)i \* 2 - 128;

/\* Compression \*/

// temp = (float)i / 2 + 128;

/\* Posterized \*/

// temp = (i / 32) \* 32;

/\* Threshold \*/

/\*

if (i > 128) temp = 255;

else temp = 0;

\*/

/\* Bounded Threshold \*/

/\*

if (128 < i && i <= 200) temp = 200;

else if (50 < i && i <= 128) temp = 50;

else temp = i;

\*/

/\* Iso-intensity contouring \*/

/\*

if (60 < i && i < 80) temp = 0;

else if (100 < i && i < 120) temp = 0;

else if (140 < i && i < 160) temp = 0;

else if (180 < i && i < 200) temp = 0;

else temp = i;

\*/

/\* Solarize \*/

/\*

if (i > 128) temp = 255 - i;

else temp = i;

\*/

CLIP(temp, 0, 255); // 맵핑된 후의 밝기 레벨(L) 값이 0 보다 작거나, 255 보다 크면 안되기 때문에.

LUT[i] = temp; // LUT[i] 에 temp 값을 초기화.

}

**< Iplib.c - histogram\_equalize() >**

unsigned long max = 0;

/\* find a max \*/

for (i = 0; i < 256; i++) {

if (histogram[i] > max) {

max = histogram[i];

}

}

/\* normalization \*/

for (i = 0; i < 256; i++) {

histogram[i] = histogram[i] / (float)max \* 255;

}

/\* make a graph \*/

for (i = 0; i < number\_of\_pixels; i++) {

if ((255 - i / 256) > histogram[i % 256])

buffer[i] = 255;

else

buffer[i] = 0;

}