**<Autumn 2018>**

**계산영상시스템**

**Assignment #3**

|  |
| --- |
| **심현정 교수님**  **글로벌융합공학부**  **2016314428 김진혁** |

**TOY PROBLEM**

**Code**

im = imread('data/toy\_problem.png');

[height, width] = size(im); % 119\*110

im\_ = im2double(im);

im2var = zeros(height,width);

im2var(1:height\*width) = 1:height\*width;

cnt=0;

A = sparse(((height\*(width-1))+(height-1)\*width)+1, height\*width);

for i = 1:height

for j = 1:width-1

cnt = cnt+1;

A(cnt,im2var(i,j+1)) = 1;

A(cnt,im2var(i,j)) = -1;

b(cnt) = im\_(i,j+1) - im\_(i,j);

end

end

for i = 1:height-1

for j = 1:width

cnt = cnt+1;

A(cnt,im2var(i+1,j)) = 1;

A(cnt,im2var(i,j)) = -1;

b(cnt) = im\_(i+1,j) - im\_(i,j);

end

end

A(cnt+1, im2var(1,1)) = 1;

b(cnt+1) = im\_(1,1);

v = A\b';

toy\_problem = reshape(v, height, width);

imshow(toy\_problem);

imwrite(toy\_problem, 'toy\_problem.png');

**Result**



**POISSON BLENDING**

**Code**

im\_source = im2double(imread('data/penguin.jpg'));

im\_target = im2double(imread('data/hiking.jpg'));

im\_s = imresize(im\_source, 0.25, 'bilinear');

im\_t = imresize(im\_target, 0.25, 'bilinear');

mask = getMask(im\_s);

[im\_s2, mask2] = alignSource(im\_s, mask, im\_t);

[height, width, c] = size(im\_s2);

im2var = zeros(height, width);

im2var(1:height\*width) = 1:height\*width;

cnt=0;

A = sparse(height\*width, height\*width);

b = zeros(height\*width, c);

for i=1:height

for j=1:width

cnt = cnt+1;

if mask2(i,j)

A(cnt,im2var(i,j)) = 4;

A(cnt,im2var(i,j-1)) = -1;

A(cnt,im2var(i,j+1)) = -1;

A(cnt,im2var(i-1,j)) = -1;

A(cnt,im2var(i+1,j)) = -1;

b(cnt,:) = 4\*im\_s2(i,j,:)-im\_s2(i,j-1,:)-im\_s2(i,j+1,:)-im\_s2(i-1,j,:)-im\_s2(i+1,j,:);

else

A(cnt,im2var(i,j)) = 1;

b(cnt,:) = im\_t(i,j,:);

end

end

end

v = A\b;

result\_1 = reshape(v, height, width, c);

imshow(result\_1);

imwrite(result\_1, 'result\_1.png');

**Result**





Laplacian filter를 사용하여 1번 문제와 유사한 방법으로 gradient image를 얻었다. 그 후, 주어진 function인 getMask와 alignSource를 사용해 문제를 해결하였다. 우선은 penguin-chick.jpeg를 먼저 합성한 후, 그 결과에 penguin.jpg을 합성하였다.

**BLENDING WITH MIXED GRADIENTS**

**Code**

im\_source = im2double(imread('data/penguin.jpg'));

im\_target = im2double(imread('data/hiking.jpg'));

im\_s = imresize(im\_source, 0.25, 'bilinear');

im\_t = imresize(im\_target, 0.25, 'bilinear');

mask = getMask(im\_s);

[im\_s2, mask2] = alignSource(im\_s, mask, im\_t);

[height, width, c] = size(im\_s2);

im2var = zeros(height, width);

im2var(1:height\*width) = 1:height\*width;

cnt=0;

A = sparse(height\*width, height\*width);

b = zeros(height\*width, c);

for i=1:height

for j=1:width

cnt = cnt+1;

if mask2(i,j)

A(cnt,im2var(i,j)) = 4;

A(cnt,im2var(i,j-1)) = -1;

A(cnt,im2var(i,j+1)) = -1;

A(cnt,im2var(i-1,j)) = -1;

A(cnt,im2var(i+1,j)) = -1;

grad\_t = 4\*im\_t(i,j,:)-im\_t(i,j-1,:)-im\_t(i,j+1,:)-im\_t(i-1,j,:)-im\_t(i+1,j,:);

grad\_s = 4\*im\_s2(i,j,:)-im\_s2(i,j-1,:)-im\_s2(i,j+1,:)-im\_s2(i-1,j,:)-im\_s2(i+1,j,:);

if abs(grad\_t) > abs(grad\_s)

b(cnt,:) = grad\_t;

else

b(cnt,:) = grad\_s;

end

else

A(cnt,im2var(i,j)) = 1;

b(cnt,:) = im\_t(i,j,:);

end

end

end

v = A\b;

result\_3 = reshape(v, height, width, c);

imshow(result\_3);

imwrite(result\_3, 'result\_3.png');

**Result**





풀이 과정은 2번과 동일하지만, source와 target의 gradient를 비교하여 더 큰 값을 사용했다는 점에서 차이가 있다. 원래부터 배경과 큰 차이가 있어 보이는 penguin-chick.jpeg는 2번과 차이가 거의 없으나, penguin.jpg는 결과가 유의미하게 차이남을 확인할 수 있다. 이 차이는 다음 ‘Your own example’에서 더 확실하게 확인할 수 있다.

**YOUR OWN EXAMPLE**

**Code**

im\_source = im2double(imread('source\_2.png'));

im\_t = im2double(imread('test\_3.jpg'));

im\_s = imresize(im\_source, 0.5, 'bilinear');

mask = getMask(im\_s);

[im\_s2, mask2] = alignSource(im\_s, mask, im\_t);

[height, width, c] = size(im\_s2);

im2var = zeros(height, width);

im2var(1:height\*width) = 1:height\*width;

cnt=0;

A = sparse(height\*width, height\*width);

b = zeros(height\*width, c);

for i=1:height

for j=1:width

cnt = cnt+1;

if mask2(i,j)

A(cnt,im2var(i,j)) = 4;

A(cnt,im2var(i,j-1)) = -1;

A(cnt,im2var(i,j+1)) = -1;

A(cnt,im2var(i-1,j)) = -1;

A(cnt,im2var(i+1,j)) = -1;

b(cnt,:) = 4\*im\_s2(i,j,:)-im\_s2(i,j-1,:)-im\_s2(i,j+1,:)-im\_s2(i-1,j,:)-im\_s2(i+1,j,:);

else

A(cnt,im2var(i,j)) = 1;

b(cnt,:) = im\_t(i,j,:);

end

end

end

v = A\b;

result\_blending = reshape(v, height, width, c);

imshow(result\_blending);

imwrite(result\_blending, 'result\_3\_blending.png');

**Result (Blending vs. Mixing)**

