Homework 2-2

1. Homework 2-1 과제 비교

건찬 님과 저의 지난 S&H를 사용해 해상도를 변화시키는 homework 2-1을 비교해보면, 약간의 코드 구성과 512x512->400x400->512x512 의 LPF 필터를 사용한 계수의 값이 다릅니다. 신기하게도 LPF를 사용하지 않은 경우에 대해서는 저의 결과와 건찬 님의 결과가 같았고, 교수님께서 보고서 예시로 보내주신 결과와도 모두 같았습니다. 알고리즘의 단순함 덕분에 모두가 비슷한 결과를 얻은 것으로 보입니다. 다만 LPF의 cut frequency나 탭 수가 달랐던 부분이 LPF를 사용했을 때의 성능을 좌우한 것으로 보입니다. 이 부분에 있어서는 아직 경험이 적기에 어떤 값들을 사용해야 더 성능이 높아지는지에 대한 감각이 없어 누가 더 좋은 필터를 적용하였는지가 무의미하다고 생각합니다.

다만, 저의 경우에는 해상도를 변환해주는 비율에 따라 cut frequency를 다르게 해주면 더 효과적이겠다는 생각을 조금 나중에 하고, 이를 적용하지는 않은 반면 건찬 님의 경우에는 그 비율에 따라 다른 cut frequency를 적용했던 것이 좋은 부분이었다고 생각합니다. 이론적으로 해당 부분과 make filter.m 프로그램을 더 잘 이해했기에 나온 결과라고 생각합니다.

1. 코드 구성

텍스트이(가) 표시된 사진

자동 생성된 설명

이번에도 저번과 같은 구성의 함수들을 사용해 과제를 해결했습니다. Scailing 함수에서 mode==2 부분을 조건문으로 추가해 Bilinear 함수를 작성했습니다.

이번에는 LPF 탭 수를 11로 설정했고, cut off frequency는 0.4와 0.25로 설정했습니다.

(512->400 에서 0.4, 512->298 에서 0.25)

Cut\_off frequency를 LPF 함수에서 입력으로 받을 때 부동 소수점은 == 비교가 힘들어서 소수점 아래 두 자리를 정수로 입력 받았습니다.

1. Bilinear 보간 프로그램 실행 결과

텍스트이(가) 표시된 사진

자동 생성된 설명

S&H와 Bilinear 직전 보고서에서 사용한 LPF 필터가 이상이 있다는 사실을 깨닫고 이를 전면 수정하기위해 노력했습니다. 중간중간 printf를 찍어 내가 원하는 위치에서 컨볼루션 연산이 잘 되는지, 미러링이 잘 적용되는지 확인해가며 고친 결과 제가 원하는 위치에서 미러링까지 잘 되는 것을 확인했습니다.

그러나 MSE 값은 60부근에서 떨어지지 않았습니다. 이에 다시 고민을 시작했고 이 이유에 unsigned char와 float의 차이가 존재하기에 값에 오류가 발생하겠다는 생각을 했습니다. Float 형의 연산 결과물을 아무 처리 없이 unsigned char를 하면 소수점 아래 부분은 그냥 버려지는데 이를 반올림 처리를 해서 연산했습니다.

그 결과, MSE가 대폭 좋아졌음을 볼 수 있습니다. 그러나 LPF를 사용하면 LPF를 사용하지 않을 때보다 결과가 좋아야 함에도 불구하고 그런 결과가 나오지 않아 이 부분에 있어서는 아직 생각 중 입니다.

해당 부분은 Bilinear에서도 float 연산 결과가 나오는 부분에 적용한 결과 bilinear만 사용한 케이스의 mse에서도 개선을 보였습니다.

확실히 자료형 변환에서 손실되는 데이터가 많다는 것을 깨달았습니다.

또한, Bilinear에서 이미지 해상도 확대 시에 마지막 행/열에서 계산을 할 때 이미지 포인터의 영역에 대해 조교 님의 도움으로 더 확실히 이해할 수 있었습니다.

사람, 여자, 모자, 하얀색이(가) 표시된 사진

자동 생성된 설명 사람, 여자, 모자이(가) 표시된 사진

자동 생성된 설명

왼쪽 이미지는 MSE가 가장 낮게 나온 1000x1000으로 키웠다가 다시 원본으로 되돌린 사진이고 두 번째 이미지는 LPF을 사용하지 않고 298x298로 줄인 이미지입니다. 두 이미지 모두 S&H에 비해 객체의 테두리 부분이 많이 자연스러워진 것을 볼 수 있습니다.

개인적으로 제가 얻은 이미지들이 원본에 비해 어떤 점이 다른 지에 대해 아무리 살펴보아도 잘 모르겠는 점이 있지만, 이미지를 조금 더 확대했을 때 298x298 이미지를 원본 이미지와 비교하면 298x298 이미지가 약간 더 흐릿한 느낌이 있고 원본 이미지는 선명해 보이는 느낌이 있습니다.

사람, 여자, 모자, 하얀색이(가) 표시된 사진

자동 생성된 설명

위 이미지는 945x945 이미지입니다. 이미지를 확대했을 때에도 꽤 자연스러운 모습을 볼 수 있습니다. Bilinear는 이미지를 축소했을 때와 확대했을 때 모두 괜찮은 성능을 보이고 있음을 알 수 있습니다.

1. 결론

Bilinear는 S&H보다 확실히 성능이 좋다는 사실이 MSE 지표에서도 나타나고 직접 이미지를 확인해보아도 알 수 있었습니다. S&H만큼은 아니어도 객체의 테두리에 약간의 거친 느낌이 많이 있을 것이라고 생각했는데 그렇지 않아서 놀라웠습니다. Bilinear가 계산량/구현의 어려움 대비 성능이 좋다고 느꼈습니다.

개인적으로 Bilinear는 원본 이미지와 정말 큰 차이가 느껴지지 않는다고 생각하지만, 만약 이를 흑백 이미지가 아닌 컬러 이미지와 같이 더 많은 정보를 포함하거나 정교한 이미지 사진에 적용했을 때에도 Bilinear가 좋은 성능을 보일 지 궁금합니다.

Bilinear는 S&H와 다르게 정수 뿐 아니라 실수 계산이 들어가므로 이 때 형변환에서 생기는 오류를 최소화 하는 것이 중요하다는 것을 느꼈습니다.

1. 정리

|  |  |  |
| --- | --- | --- |
|  | S&H | Bilinear |
| 512x512->1000x1000->512x512 | MSE : 182.545033 | MSE : 2.932796 |
| 512x512->400x400->512x512 | MSE : 196.817356 | MSE : 16.09336 |
| 512x512->400x400->512x512  (no LPF) | MSE : 231.988354 | MSE : 15.339458 |

|  |  |  |
| --- | --- | --- |
|  | S&H | Bilinear |
| 512x512->1000x1000->512x512 |  |  |
| 512x512->400x400->512x512 | 사람, 실외, 여자, 하얀색이(가) 표시된 사진  자동 생성된 설명 |  |
| 512x512->400x400->512x512  (no LPF) |  |  |
| 512x512->945x945 |  |  |
| 512x512->298x298 | 사람, 머리장식, 모자이(가) 표시된 사진  자동 생성된 설명 |  |
| 512x512->298x298  (no LPF) |  |  |