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 9로 설정했고, cut off frequency는 그대로 0.25로 설정했습니다.

1. Bilinear 보간 프로그램 실행 결과

텍스트이(가) 표시된 사진

자동 생성된 설명

이번에도 MSE 결과만 놓고 보면 꽤 이상적으로 나왔다고 생각합니다. 업 스케일링 후 다운 스케일링한 MSE가 가장 낮고, 다운 스케일링한 후 업 스케일링한 결과 가운데서 LPF를 사용한 MSE가 더 낮게 나온 점은 예측과 동일합니다.

하지만, 여기서 생각해 본 지점이 있는데 사실 homework 2-1과 똑같은 설정 (탭은 7, cut frequency는 0.25) 으로 MSE를 측정한 결과는 50 언저리가 나왔습니다. 이리저리 LPF 계수들을 건드려보다가 cut off frequency는 0.25가 가장 좋은 MSE가 측정되었고, 따라서 탭 수를 늘려보았습니다. 탭 수를 9개로 늘리자 마자 MSE가 탭 7개를 사용했던 것에 비해 매우 좋은 값을 보여주었습니다.

저는 그 이유가 Bilinear가 더 주변 정보를 많이 활용하기 때문이라고 생각합니다. 어느정도 수준까지는 LPF의 탭 수가 많을수록 LPF의 정확도가 올라간다고 생각했을 때, Bilinear는 X(k)와 X(k+1) 두 개의 원본 데이터로 한 개의 예측 데이터를 생성하기에 LPF에서 정보 손실이 일어났을 때 그로 인한 손해가 S&H보다 Bilinear가 더 클 것으로 예상됩니다. 같은 이유로 3차 보간의 경우 예측을 위해 Bilinear보다 더 많은 주변 정보들을 사용하므로 LPF의 계수를 잘 설정하는 것이 더 중요할 것으로 생각됩니다.

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.

1. 정리

|  |  |  |
| --- | --- | --- |
|  | S&H | Bilinear |
| 512x512->1000x1000->512x512 | MSE : 182.545033 | MSE : 5.982597 |
| 512x512->400x400->512x512 | MSE : 196.817356 | MSE : 19.677761 |
| 512x512->400x400->512x512  (no LPF) | MSE : 231.988354 | MSE : 20.156227 |

|  |  |  |
| --- | --- | --- |
|  | 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) |  |  |