Lab 2.

|  |  |  |  |
| --- | --- | --- | --- |
| 제출일 | 11/27 | 학번 |  |
| 전공 | 전자공학과 | 이름 |  |

|  |  |
| --- | --- |
| 문제 | 1번: 일 때 신호원을 고려하여 fs를 설정하여 X[k]의 Magnitude, Phase를 구하라  2번: 원본 이미지와 노이즈 이미지를 DFT하여 주파수 영역의 사진을 보고 필터를 적용한 후 IDFT 과정을 통해서 깨끗한 이미지를 생성하라 |
| 주요 변수 | 문제 1번  N, k0, k1: N은 샘플링 수에 해당하는 변수이며, k0와 k1은 샘플링 인덱스에 해당하는 변수이다. 입력 신호가 주파수가 다른 두 신호의 합이므로 샘플링 인덱스를 두가지로 하여 접근해야 한다. 또한 Sampling Rate가 이 세 변수에 의해 정해지므로 에일리어싱이 발생하지 않도록 잘 조율해야만 한다.  x1, x2, x: 주파수 성분에 따라 x1, x2에 discrete의 결과 데이터를 담아 x로 최종 처리한다.  X1, X2, X: 역시 주파수 성분에 따라 X1, X2에 각각 x1, x2을 이산 푸리에 변환한 결과 데이터를 담아 X로 최종 처리한다.  문제 2번  fft\_result: DFT 및 필터 적용 결과를 담은 데이터다.  crow, ccol: 각각 이미지 중앙 위치에 해당하는 데이터를 의미한다.  low\_pass: 로우 패스 필터를 적용하기 위해 이미지 중앙 기준 반지름 30의 원을 기준으로 바깥쪽을 제거하였다. (저주파 성분을 중앙으로 고주파 성분을 가장자리로 이동시킨 후에 적용하기에 문제없다.)  gaussian: 가우시안 필터를 적용하기 위해 주파수 도메인 가우시안 공식을 구현하였다.  magnitude\_spectrum: DFT 데이터의 Magnitude에 로그 스케일을 적용한 결과를 담은 데이터다.  normalized\_spectrum:  magnitude\_spectrum의 최댓값과 최솟값을 기반으로 정규화한 결과를 담은 데이터다.  ifft\_result:  fft\_result의 IDFT 적용 결과를 담은 데이터다. |
| 알고리즘 | 문제 1번 코드의 알고리즘은 다음과 같은 단계로 이루어져 있다: 샘플링(CTD), 이산 푸리에 변환(DFT), Magnitude & Phase 데이터 저장.  샘플링의 경우 샘플링의 수 N을 100개, Sampling Rate f\_s를 10Hz으로 하여 진행하였다. f\_max가 8Hz이므로 Nyquist Frequency보다 크기 때문에 에일리어싱을 방지할 수 있다. 또한 이 코드에서는 Sampling Rate가 아니라 Sampling Index를 변수로 설정했기 때문에 다음과 같은 공식에 맞추어 변수들을 설정해야 한다. PPT에 제시된 공식을 통해 구현 가능하다.  이산 푸리에 변환의 경우 PPT에 제시된 공식을 그대로 코드로 구현하여 구할 수 있다. 푸리에 변환 값의 절댓값이 Magnitude, 아크 탄젠트 값이 Phase이므로 이것을 코드로 구현하여 데이터로 저장한다. 전자의 경우 abs() 함수를 통해, 후자의 경우 atan2() 함수를 통해 구현한다. 그러나 아크 탄젠트 자체가 부동 소수점 문제에 굉장히 취약해서, 의미 있는 값이 나오는 경우로 한정해서 데이터를 저장한다.  Sampling Index 공식 (N은 샘플링 수, f\_s는 Contiunous에서의 주파수, f\_c는 Sampling Rate)  입력 신호에 따른 DFT 결과 (에일리어싱이 나타나지 않음을 가정, DFT의 주기성 고려)  문제 2번 코드의 알고리즘은 다음과 같은 단계로 이루어져 있다: 이미지 데이터 로드, 이산 푸리에 변환, 주파수 범위 쉬프트, 로우 패스 필터, 가우시안 필터, 주파수 범위 역쉬프트, 스펙트럼 생성 (로그 스케일 & 정규화), 이산 푸리에 역변환, 데이터 기반 이미지 생성.  Python의 패키지 Scipy에서 제공하는 푸리에 변환 함수는 시간 복잡도가 O(NlogN)이므로 중첩 반복문 구현의 시간 복잡도가 O(N^4)인 것에 비해 매우 효율적이다.  푸리에 변환 함수를 거친 데이터는 주파수 범위가 [0, 2pi]로 맞추어져 있어 이미지를 통한 스펙트럼 분석 및 필터 적용이 어려울 수 있으므로 주파수 범위를 [-pi, pi]로 이동시켜주는 것이 좋다. (이것을 Shifting이라고 칭한다.) Numpy에서 해당 함수를 제공한다. Shifting을 거치면 중앙부는 저주파 성분, 가장자리부는 고주파 성분인 스펙트럼 이미지를 얻을 수 있다.  로우 패스 필터의 경우 Shifting을 거친 스펙트럼 이미지에서 가장자리 부분에 고주파 성분이 위치한다는 것을 이용한다. 두 점 사이 거리 공식을 기반으로 이미지 중앙을 원점으로 하는 원을 가정하여, 그 원의 외각 영역 데이터를 제거하는 방식으로 로우 패스 필터를 구현한다.  가우시안 필터의 경우 Shifting와 로우 패스 필터를 거친 스펙트럼 이미지 데이터에 정규 분포를 곱한다. 정규 분포가 중점에서 멀어질수록 그 값이 작아진다는 특징이 로우 패스 필터의 역할을 할 수 있게 한다. 또한 중점에 위치한 값이 1이 되게 하기 위해 정규 분포의 지수 부분만 이용한다.  스펙트럼의 경우 최솟값과 최댓값의 차가 너무 클 것을 우려하여 DFT 값의 절댓값에 1을 더한 값에 로그를 취한 값을 채택하였다. (1을 더하는 이유는 로그 값이 음수로 도출되는 것을 방지하기 위해서다.) 또한 로그 값의 최댓값과 최솟값을 기반으로 정규화를 진행하여 시각적으로 분석하기 쉽도록 데이터를 구성하였다.  이제 역Shifting과 역변환을 거친 데이터, 스펙트럼 데이터를 기반으로 이미지를 생성하고, 그 이미지를 기반으로 노이즈를 제거하는 방법을 분석한다.  가우시안 필터 (u\_0, v\_0는 각 주파수 영역의 중심 좌표)  로그 스케일 & 정규화 |
| 결과 분석 | 문제 1번:  그림 1.  주어진 신호를 샘플링 개수를 100개, Sampling Rate를 10Hz로 하여 만들어진 이산 푸리에 변환의 Magnitude 그래프이다. Sampling Rate가 Nyquist Frequency 8Hz보다 크기 때문에 에일리어싱은 발생하지 않는다. 가로축은 몇 번째 샘플에 해당하는지(Sampling Index)를 의미하며 5개의 간격을 기준으로 하였다. 우선 컴퓨터의 부동 소수점 한계로 모든 샘플에서 매우 작은 값들을 가지고 있어 엑셀에서 y축의 단위가 괴상하게 잡히지만, 각 막대의 값은 각각 100, 200에 해당한다. 이것은 이산 푸리에 변환의 Magnitude가 일반 푸리에 변환의 Magnitude에 샘플링 수를 곱한 결과와 같다는 것을 의미한다. Sin 성분의 계수가 2였고, Cos 성분의 계수가 4였으므로, 일반 푸리에 변환의 계수는 각각 -j, 2으로 나오게 되고 이 둘의 Magnitude가 1, 2라는 것과 여기에 샘플링 개수 100을 곱하면 100, 200이라는 것이 이것을 증명한다. 또한 값을 가지는 샘플의 위치는 25, 40, 60, 75인데, 주파수에 대응시켜보면 5pi/10, 8pi/10, 2pi - 8pi/10 = 12pi/10, 2pi - 5pi/10 = 15pi/10이다. 이것은 이산 푸리에 변환의 주파수가 일반 푸리에 변환의 주파수에 샘플링 수를 나눈 결과와 같다는 것을 의미하며, 이산 푸리에 변환의 2pi 주기성에 의해 주파수 성분이 Shift되어 발현되었다는 것도 의미한다.    그림 2.  Magnitude와 동일한 조건으로 만들어진 Phase 그래프이다. Phase는 arctan(원래 신호의 허수부 / 원래 신호의 실수부)로 구현되기 때문에, 부동 소수점 오차에 매우 취약하다. 따라서 사전에 코드를 통해 정보가 25, 40, 60, 75에서만 유효하기에 Phase도 그 부분에서만 유효하다는 점을 이용하여 사전적으로 그 외 영역의 데이터를 거르도록 한다. 왜 저 4부분에서만 유효한지는 Magnitude 부분에서 설명했으므로 생략한다. 이론적으로 25에서는 j가 정보의 계수이므로 pi/2가, 40과 60에서는 1이 정보의 계수이므로 0이, 75에서는 -j가 정보의 계수이므로 -pi/2가 나타나게 되는 것이 이론적으로 맞다. 그래프 역시 거의 동일한 결과를 나타내며 이론을 증명하고 있다.  문제 2번:    그림 1, 2.  원본 이미지에 비해 노이즈 이미지는 대각선의 노이즈가 반복적으로 나타나고 있다. 이 노이즈로 인해 이미지의 색(밝기) 변화가 두드러지게 크기 때문에 고주파 성분이 큰 규모를 가지게 될 것을 예측할 수 있다.    그림 3, 4.  원본 이미지와 노이즈 이미지의 주파수 스펙트럼은 다음과 같이 나타난다. 노이즈의 원인이 되는 고주파 성분들이 이미지 정중앙에 위치한 것을 확인할 수 있다. 해당 스펙트럼은 원점이 좌측 상단으로 설정된 결과물이다. DFT는 주파수 2pi 기준으로 반복되는 주기성을 띄기 때문에, 가장자리 부분에 저주파 성분이, 중앙 부분에 고주파 성분이 위치하게 된다.    그림 5.  보통 주파수 스펙트럼을 분석할 땐 저주파 성분을 중앙으로, 고주파 성분을 가장자리로 위치를 바꾼다. 이렇게 변환하면 원점이 정중앙에 위치하게 되어 스펙트럼을 분석하기 편해진다. (그래프로 나타내는 주파수의 범위가 [0, 2pi]에서 [-pi, pi]로 이동시켰다고 볼 수도 있다.) 중앙에서 멀어질수록 약해지는 가우시안 필터를 적용하기도 편해지고, 로우 패스 필터의 경우 중앙을 기준 만들어진 원을 경계로 외부 영역을 제거하는 것으로 구현할 수 있다.    그림 6, 7, 8.  중앙을 원점으로 하는 원을 기준으로 외각 성분을 제거함으로써 고주파 성분을 제거하는 로우 패스 필터를 구현할 수 있다. 여기서는 반지름을 30으로 하여 필터를 구현하였다. 이 필터를 거쳐서 나오게 된 이미지를 보면 대각선의 노이즈가 사라진 것을 눈으로도 직접 확인할 수 있다. 그러나 아직 이미지 전체에 전반적으로 노이즈가 남아있는 것도 확인할 수 있다.    그림 9, 10, 11.  전반적인 노이즈를 개선하기 위해 가우시안 필터를 적용한다. 여기서는 표준편차를 15라고 가정하고 구현하였다. 표준편차가 작을수록 정규 분포 값이 중앙에 밀집된다는 특징이 있다. 가우시안 필터가 적용된 스펙트럼을 보니 원의 가장자리 부분이 필터 적용 전과 비교해 약간 어두워진 것을 확인할 수 있었다. 최종 출력된 이미지를 보더라도 노이즈가 어느 정도 개선된 것을 확인할 수 있다. 그러나 고주파 성분을 약화시킨 만큼 이미지의 윤곽들이 흐려지는 Blur 현상도 일어난 것을 볼 수 있다. |
| 느낀점 | 주파수 영역에서 이미지나 음성을 처리하는 것이 얼마나 편리한지 직접 경험해볼 수 있는 기회였다. 개인적으로 생성형 이미지에 관심이 많아 집에서 Stable Diffusion을 직접 돌려보는 것이 취미인데, 거기서 가우시안 필터나 Canny와 같은 엣지를 얻는 필터들을 정확히 어떤 것인지 모르고 사용했던 기억이 있다. 그러나 이번 과제를 통해 라플라시안 필터, 가우시안 필터 등을 찾아보면서 이것이 어떻게 적용되고 어떤 역할을 해서 그러한 결과물이 도출되었는지 명확히 알 수 있게 되었다. 생성형 이미지 AI에 접근하기에 앞서 이미지가 어떻게 처리되는지를 다루는 영상 처리를 이해하는 것은 매우 중요하다. 사실 영상 처리는 4학년 과목에 해당하는지라, 2학년인 나에게는 이러한 경험을 해볼 수 있는 기회 자체가 적은데, 3학년 과목인 디지털 신호 처리에서 해볼 수 있었던 것은 큰 행운이다. 한편 부동 소수점 오차 문제를 직접적으로 느껴본 경험은 이번이 처음이었다. Phase를 arctan으로 구현하는 과정에서 전혀 예상하지 못한 그래프가 도출되어 당황했었다. 이러한 불미스러운 일을 방지하기 위해서 부동 소수점 오차는 해결하기 어렵지만 컴퓨터과학자들이 꼭 해결되어야 하는 큰 난제임을 느꼈다. |