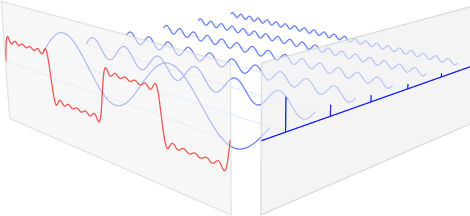
**DIP Summer Seminar HW02 2022**

1. **푸리에 계수, 푸리에 급수, 푸리에 변환에 대해 원리, 목적 설명**

**푸리에 변환** : 시간에 대한 함수를 주파수에 따라 분해 해주는 변환



이 그림에서 적색 신호는 입력 신호, 청색 신호는 푸리에 변환을 통해 얻은 주기함수 성분들이다. 각 주기함수 성분은 고유의 주파수(Frequency)와 강도(Amplitude)를 가지며 이들을 합치면 원본 입력 신호가 된다. 푸리에 변환을 이용하면 입력 신호를 sin, cos 주기함수들의 합으로 분해할 수 있다. 푸리에 변환식은 다음과 같이 나타낼 수 있다. (주의할 점은, 2차원 영상처리의 경우 푸리에 변환식을 다르게 나타낸다.)

* f(x)는 원본 입력신호
* 는 주파수 u인 주기함수 성분
* F(u)는 주기함수 성분의 계수(Coefficient)

시간 축에 존재하던 신호를 주파수로 변환하면 데이터의 양이 줄어든다. 푸리에 급수는 복잡한 신호를 단순하게 분해할 수 있지만, 주기 신호에만 사용할 수 있기 때문에 이를 극복하기 위해 나온 개념이 푸리에 변환이다. 푸리에 변환은 f(x)를 이용해 신호의 주파수 성분을, F(u)를 이용해 지연이 얼마나 되는지에 대한 응답을 알 수 있다.

**푸리에 급수** : 주기 함수를 주파수와 진폭이 다른 사인과 코사인의 합으로 분해한 급수

우선 급수란 수열들의 합을 의미한다. 푸리에는 한가지 중요한 사실을 발견하게 되는데, 같은 형태를 반복하는 주기를 가진 파동은, 아무리 복잡하더라도 단순한 파동의 결합으로 이루어진다는 것이다. 푸리에 급수는 이를 이용해 신호를 기본적인 주기함수인 사인과 코사인의 조합으로 전개한다. 다음은 적분 구간이 (0, 2π)인 푸리에 급수식이다.

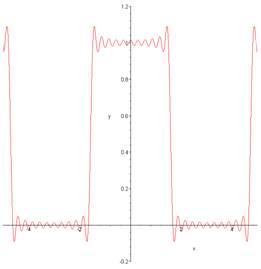
* 와 은 푸리에 계수

**푸리에 계수** : 푸리에 계수의 진폭은 함수를 구성하던 주파수 성분의 크기를 의미한다.

푸리에 계수는 삼각함수의 직교성을 이용해 정적분하여 구할 수 있다.

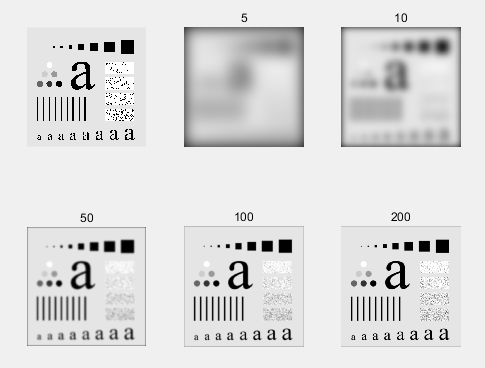
1. **Ideal low-pass filter의 문제점 설명**

Frequency Domain에서 Ideal low-pass filter는 깔끔한 모양을 가지지만, Spatial Domain 에서는 싱크 형태를 갖게 된다. Ideal low-pass filter은 윤곽으로 확인할 수 있는 엣지 부분에서ringing현상이 일어납니다.

 <코사인 급수에서 Ringing이 일어난 모습>

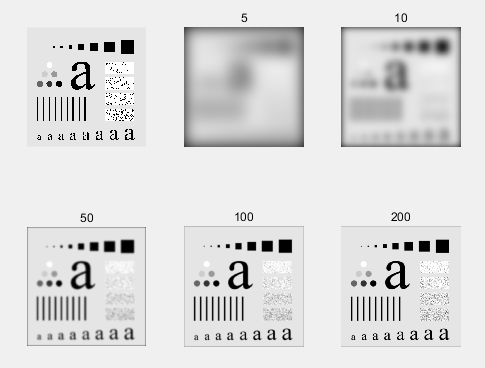
그 이유는 Ideal low-pass filter 가 공간 도메인에서 싱크 함수로 나타나게 되고 이것을 영상과 컨볼루션하게 되면, 0값을 기준으로 진동하는 sin함수 형태를 나타내기 때문입니다.

**3-1. figure 4.35의 주파수 도메인 필터링 과정을 필터 반경을 달리하여 각각 구현, 필터 반경에 따른 차이점 설명, 실험 결과  
(Gaussian low-pass filter 기준) (code 구현 o)**



필터의 반경이 늘어날수록 Blurring이 감소하고 선명한 영상이 나온다. 이는 주파수 필터 반경이 커질수록 통과하는 고주파 신호가 늘어나기 때문이다.

**3-2. 3-1의 구현을 zero-padding 과 replicate padding으로 각각 실험해보고 차이점 비교 (zero-padding, replicate padding 차이 비교) (code 구현 o)**



<zero-padding>

<replica-padding>

1. **ch3 은 공간 도메인 필터링, ch4은 주파수 도메인 필터링에 대한 내용을 다루었습니다. 두 가지 방법에 대한 차이점, 장단점을 비교 설명 하세요.**

공간 도메인 필터링은 주파수로의 변환 과정을 거치지 않기 때문에 계산이 효율적이다. 하지만 주파수 도메인 필터링의 경우 noise를 제거할 때 용이하다. 공간 도메인 필터링은 픽셀을 기준으로 필터링하기 때문에 noise가 있는 영상의 경우 비효율적이다.

제출 파일: 과제 코드 (matlab), 과제 파일(한글, word)

내장함수 fft2, ifft2 사용 가능, padding, 필터설계 직접 코드 구현

**제출 기간: 2022/07/28, 23:59.**