

Signals and Systems HW#5

전기정보공학부

2017-17088 박찬정

1. Hands-on Audio Sampling: With Your Voice!

(a)

야구팀 한화 이글스의 김성근 감독 응원가를 녹음하여 loveSeongGeun.mp3 파일로 저장하였다.

(b)

audioread 함수를 사용하여 음성 파일을 가져왔다. Sampling Rate는 44100Hz 이다.

(c)

원본과 큰 차이점은 찾지 못했다.

(d)

음질이 매우 떨어진 점을 확인할 수 있었다. 특히 고음 부분이 제대로 표현되지 않고 울리는 듯한 느낌이 든다.

(e)

Sampling의 영향으로 주파수 도메인에서 Aliasing이 생긴 것이 이유이다. (b)의 경우 1/2 Sampling으로 인해 생기는 Aliasing의 영향이 거의 없었던 것으로 보이지만, 1/8 Sampling의 경우 주파수 도메인의 형태가 정수 주파수 지점을 기준으로 8배 양쪽으로 늘어진 형태를 갖게 되므로 Aliasing의 영향이 커질 수밖에 없다. 그 때문에 음원이 많이 달라졌고, 특히 고주파 영역에서 위화감을 많이 느끼게 된다.

2. Hands-on Image Sampling: With Your Own Image!

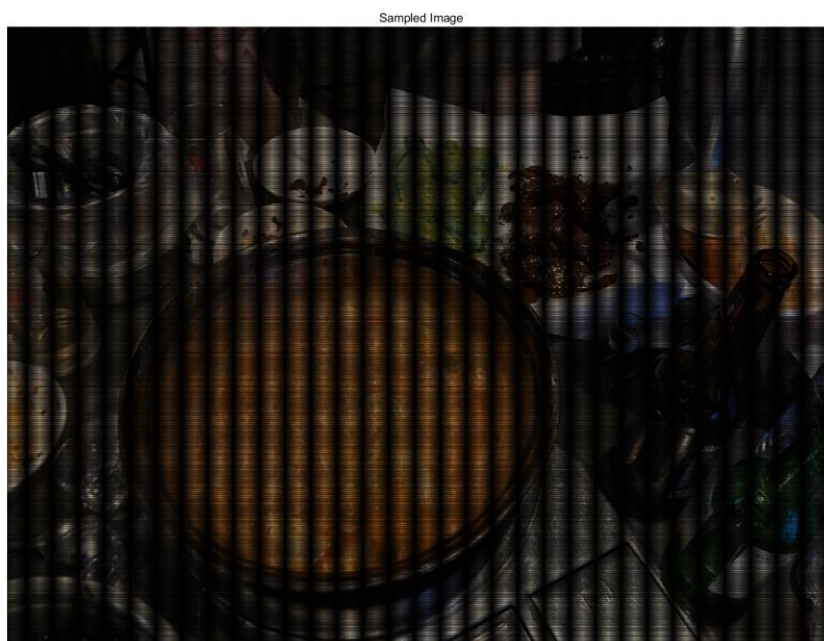
(a)

낙성대역 부근 동아전에서 먹었던 김치찌개와 민트초코치킨, 초코치킨의 사진을 가져왔다.

(b)

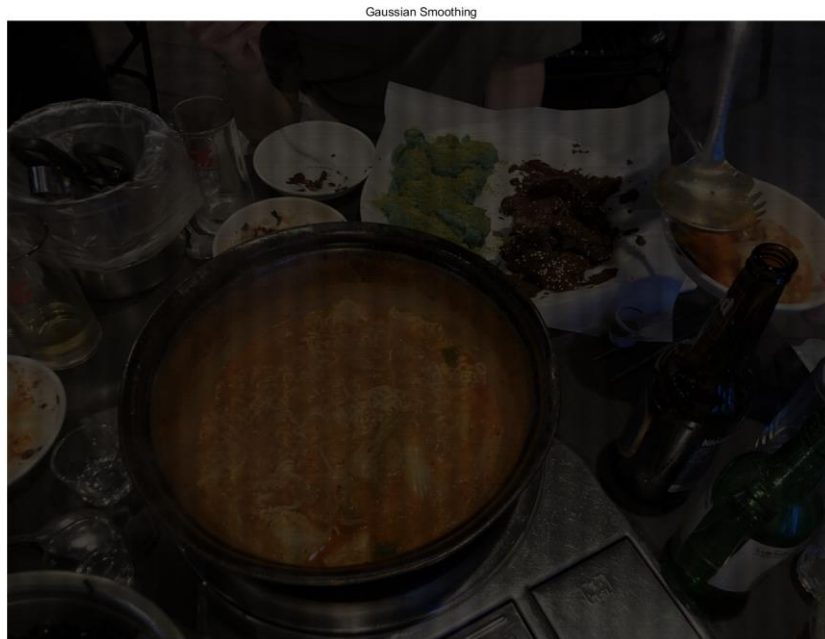


(c)



(d)

fspecial 함수를 사용하여 표준편차가 1인 3x3 크기 가우시안 필터를 생성하고 샘플링된 이미지와 합성곱 하였다.



(e)

Gaussian Filter로 합성곱을 하는 과정에서 비어 있던 hole에 주변 3x3 픽셀의 값들이 더해지기 때문이다.

(f)

비어있던 hole의 RGB값이 증가하는 것과 반대로, 원래 채워져있던 픽셀의 경우 Gaussian Filter 행렬의 중앙에 있는 값에 따라 줄어들게 된다. 예를 들어, 이번에 사용한 표준편차가 1인 3x3 크기 Gaussian Filter는 행렬 중앙에 있는 값이 0.2042이고, 이로 인해 합성곱 과정에서 원래 이미지의 RGB값이 0.2042배로 줄어든다는 뜻이다. 게다가 샘플링의 결과로 값을 갖고 있는 픽셀의 주변 3x3 범위에는 자신 이외에 모두 hole이므로 합성곱 과정에서 다른 픽셀로부터 값이 더해지지도 않는다.

Gaussian Filter의 모든 원소의 합은 1이므로 전체 이미지의 RGB값의 합은 합성곱 이전과 이후에 달라지지 않는다. 즉 샘플링 과정에서 이미 이미지의 RGB값의 합이 대략 1/4배가 되었으므로 Gaussian Filter를 거친 이후에도 이미지의 RGB값의 합은 원본의 약 1/4이고, 그만큼 이미지도 어

두어진다.

이 때문에 이 단계에서 얻은 이미지는 매우 어두운 상태가 된다.

(g)

$K=4$ 일 때 원본에 가장 가깝게 보였다. 위에서 설명한 바와 같이 이미지의 전체 RGB값의 합은 샘플링 과정에서 약 $1/4$ 이 되었고, Gaussian Filter는 전체 RGB값을 유지하므로, 이후 전체적으로 RGB값을 4배 해 주면 원본과 거의 비슷한 합계 RGB값을 갖게 된다.

Final Image



3. Natural Sampling

(a)

경우에 따라 가능하다.

(b)

$$y(t) = x(t) \cdot s(t) \longleftrightarrow X(f) * S(f),$$

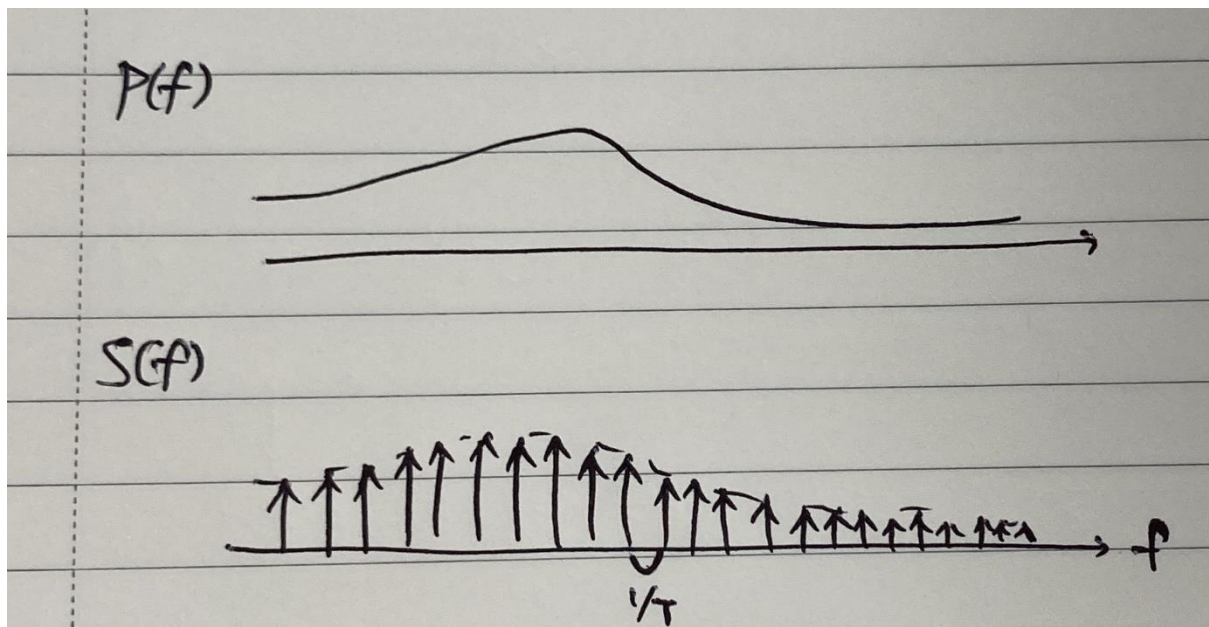
$$s(t) = \sum_{k=-\infty}^{\infty} T p(t-kT) = \sum_{k=-\infty}^{\infty} T S(t-kT) * p(t)$$

$$= T u_T(t) * p(t)$$

$$S(f) = U_T(f) \cdot P(f)$$

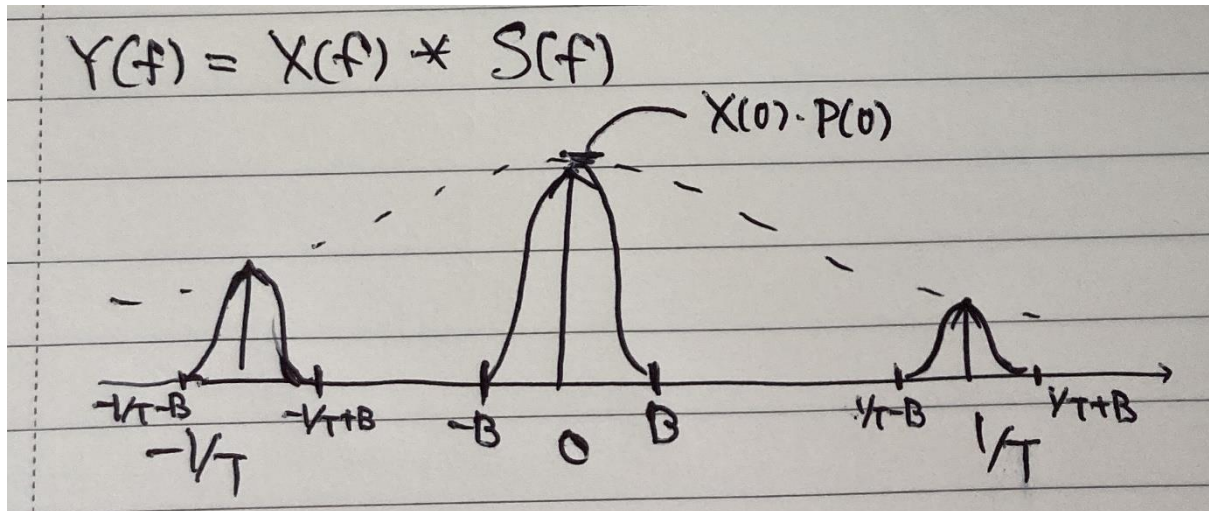
$s(t)$ 를 train of pulse로 두고 위와 같이 표현할 수 있다.

이를 주파수 도메인에서 보면 아래와 같다.

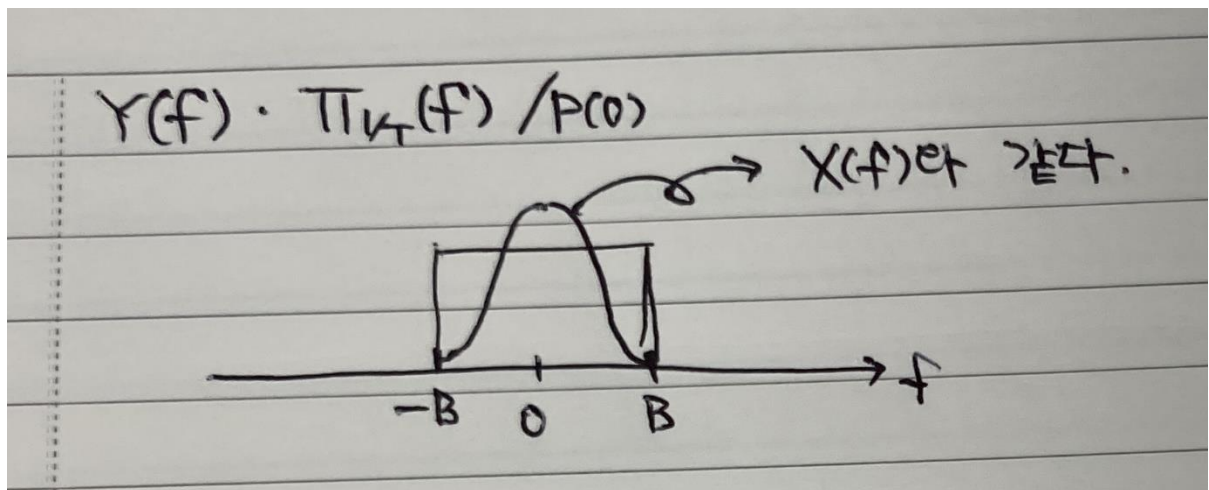


따라서 이 경우에도 Aliasing이 존재하지 않기 위한 조건은 동일하다.

$$\text{조건} : 2B \leq 1/T$$



이 때 아래와 같이 Low Pass Filter를 사용하여 원래의 $X(f)$, $x(t)$ 를 복원할 수 있다.



4. Applications of Signals and Systems: What Interests You?

작년 10월 경, 군 복무를 하고 있었을 때, 우연히 '국방오픈소스아카데미' 라는 곳에서 주최하는 온라인 해커톤을 알게 되어 참가하게 되었습니다. 핸드폰 반납 인증 시스템을 주제로 팀을 꾸려서 열심히 개발을 했었는데 그 과정에서 QR코드를 사용하게 되었습니다. 핸드폰에 저장된 키 값으로 TOTP를 생성하여 QR코드로 표현하고, 반납기에서 그 QR코드를 읽어 확인하는 방식으로 시스템을 구상하였는데, 그 과정에서 어떻게든 QR코드가 노출되고 일반 사용자가 마음만 먹으면 TOTP 값을 알아내는 것이 마음에 들지 않았습니다. 물론 TOTP값으로 키 값을 알아낼 수는 없겠지만 아무튼 마음에 들지 않았습니다. 그래서 QR코드를 변조시켜 볼 수 없을지 고민하던 중에, RGB값을 활용하는 방안을 생각하게 되었습니다.

정말 단순한 방식으로 QR코드에서 검은 부분을 R, G, B 중 한 가지에 대응시키고 나머지는 노이즈를 집어넣는 방식이었습니다. 복호화 과정에서는 R, G, B 중 알맞은 색상을 골라 나머지는 버림으로써 원래의 QR코드를 얻을 수 있었습니다. 여기서, 다 까먹은 선형대수학을 간신히 떠올

려가며 암호화/복호화 방식을 발전시켜 나갔습니다. 아래 사진이 대략적인 예시이고, 당시 생각을 정리한 페이지도 있습니다(<https://kookmoban.gitbook.io/osam/undefined-2/qr>).



그래서 신호 및 시스템을 공부하고 매트랩으로 이미지 처리를 살짝 맛보면서 이 때 생각이 많이 났습니다. 조잡한 방식이 굉장히 부끄러우면서도 만약 이 때 신호 및 시스템에서 배운 내용들을 알고 있었다면 더 좋고 세련된 암호화, 복호화 과정을 만들어볼 수 있었을 거라는 생각이 났습니다. 물론 아직 학부도 마치지 못한 작은 존재이지만, 배운 것들을 바탕으로 실제로 활용할 수 있는 무언가를 만들어내는 것은 정말 재미있는 일 같습니다. 제 미래를 아직 결정하지는 못했지만 대학원을 졸업한 뒤에 많은 기술을 개발하며 각종 이미지들을 다루고 가공하는 저의 모습을 상상하면 나름 괜찮은 것도 같습니다.

신호 및 시스템을 배우면서 굉장히 재미있었고 어떻게 쓰일지 상상하는 것도 좋았습니다. 교수님과 조교님들께서도 좋은 수업을 해 주시고 도움도 많이 주셔서 감사합니다. 이제 제가 시험을 잘 봄으로써 유종의 미를 거둔다면 더할 나위 없을 것 같은데... 열심히 공부해 보겠습니다. 다시 한번 감사드립니다.