Convolution as Matrix Multiplication

\[ f(x) \rightarrow h(x) \rightarrow g(x) \]

\[ g(x) = h(x) \ast f(x) \]

\[
\begin{bmatrix}
1 & 0 & 0 & 1 \\
1 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 \\
0 & 0 & 1 & 1 \\
0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
1 \\
3 \\
-2 \\
1 \\
-1
\end{bmatrix}
\begin{bmatrix}
a_0 & a_1 & a_2 & a_3 \\
a_1 & a_0 & a_1 & a_2 \\
a_2 & a_1 & a_0 & a_3 \\
a_3 & a_2 & a_1 & a_0
\end{bmatrix}
\]

Toeplitz matrix – constant along every diagonal

Circulant matrix – each row is a circular shift of the previous row

Another Example

\[ H = \begin{bmatrix}
3 & 0 & 0 & 1 & 2 \\
2 & 3 & 0 & 0 & 1 \\
1 & 2 & 3 & 0 & 0 \\
0 & 1 & 2 & 3 & 0 \\
0 & 0 & 1 & 2 & 3
\end{bmatrix} \]

\[ H^T = \begin{bmatrix}
3 & 2 & 1 & 0 & 0 \\
0 & 3 & 2 & 1 & 0 \\
0 & 0 & 3 & 2 & 1 \\
1 & 0 & 0 & 3 & 2 \\
2 & 1 & 0 & 0 & 3
\end{bmatrix} \]

2-d Convolution as Matrix Multiplication

\[ H = \begin{bmatrix}
H_0 & H_1 & H_2 & H_3 \\
H_2 & H_0 & H_2 & H_3 \\
H_3 & H_2 & H_0 & H_2 \\
H_3 & H_2 & H_3 & H_0
\end{bmatrix} \]

Each block is a circulant matrix.

H is a doubly block circulant matrix.
2-d Convolution as Matrix Multiplication

- For a 512 by 512 image:
  - \( f \) and \( g \) are 262144 by 1
  - \( H \) is 262144 by 262144
- Computing \((H^TH)^{-1}H^T\) directly is not feasible
- Solution is subject to numerical instability
- Looking ahead: we will convert a “bad” problem to a “good” problem by looking for a “well behaved” or “smooth” solution

A Little Matrix Calculus

\[
\frac{\partial}{\partial x} f(x) = \begin{bmatrix}
\frac{\partial}{\partial x_1} f(x) \\
\frac{\partial}{\partial x_2} f(x) \\
\vdots \\
\frac{\partial}{\partial x_N} f(x)
\end{bmatrix}
\]

\[
\frac{\partial}{\partial \lambda} (u^T \lambda) = u
\]

\[
\frac{\partial}{\partial \lambda} (u^T P \lambda) = P \lambda + P^T \lambda
\]

\[
\frac{\partial}{\partial \lambda} (u^T P^T P \lambda) = 2P^T P \lambda
\]

Lagrange Multiplier Method

Minimize \( f(y) \) subject to \( g(x) = 0 \).

Define: \( \phi(x, \lambda) = f(x) + \lambda^T g(x) \)

Then optimization problem can be solved by solving

\[
\frac{\partial}{\partial x} \phi(x, \lambda) = 0
\]

\[
g(x) = 0
\]

for \( x \) and \( \lambda \)

Constrained Least Squares Algorithm

1. Choose an initial guess for \( y \)
2. Calculate the reconstruction

\[
\tilde{F}(u, v) = \left[ \frac{H^T(u, v)}{[H(u, v)]^T + y\gamma(u, v)]} \right] G(u, v)
\]
3. Calculate the residual

\[
R(u, v) = G(u, v) - H(u, v) \tilde{F}(u, v)
\]

\[
\phi(y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} |R(u, v)|^2
\]
4. Compare residual to target value, and either
   a. Choose a better value for \( y \) and go to 2 above, or
   b. We are done
Blind Deconvolution

- Reconstruct image with imperfect knowledge of degradation function $H(u,v)$
- If there is no noise, then blind deconvolution in 2 or more dimensions can be solved using polynomial factorization
- In general, problem is ill-conditioned, solution may not be unique
- Need to use a priori assumptions about the undegraded image, the degradation function, or both:
  - Assume a form for the degradation function such as motion blur or lens defocus, then estimate model parameters by examining zeros in the frequency plane
  - Assume undegraded image is an autoregressive (AR) process and that the degradation function is a moving average (MA) process, then fit data to a ARMA model

Iterative Blind Deconvolution

$$F_k(u,v) = \frac{\hat{H}_k^*(u,v)G(u,v)}{|\hat{H}_k(u,v)|^2 + \alpha |\hat{F}_k(u,v)|^2}$$

$$H_k(u,v) = \frac{\hat{F}_k^*(u,v)G(u,v)}{|\hat{F}_k(u,v)|^2 + \alpha |\hat{H}_k(u,v)|^2}$$

Blind Deconvolution

- Constraints for the image:
  - The image data is a non-negative
  - The image has a limited region of support
  - The image contains sharp edges or point features
- Constraints for the degradation function:
  - The degradation function is circularly symmetric
  - The degradation function is non-negative
  - The degradation function has a limited region of support