Convolution Theorem

Convolution

$a = [2,4,6,7,1]$
$b = [1,3,4]$

$a * b =$

	[2	4	6	7	1]
[4	3	1]			

$= [2, 10, 26, 41, 46, 31, 4]$

Convolution 2-D

$$a = \begin{array}{|c|c|c|c|c|}\hline 3 & 4 & 4 & 6 & 9 \\ \hline 2 & 3 & 7 & 3 & 6 \\ \hline 2 & 5 & 9 & 6 & 3 \\ \hline 5 & 2 & 5 & 4 & 9 \\ \hline 3 & 4 & 2 & 4 & 3 \\ \hline \end{array}$$

$$b = \begin{array}{|c|c|c|}\hline 1 & 2 & 2 \\ \hline 3 & 3 & 4 \\ \hline 3 & 4 & 5 \\ \hline \end{array}$$

$a * b =$

Very important operation in neural networks, computer vision …

Another perspective on Convolution

$$(1,2,3) * (4,5,6) = (4,13,28,27,18)$$

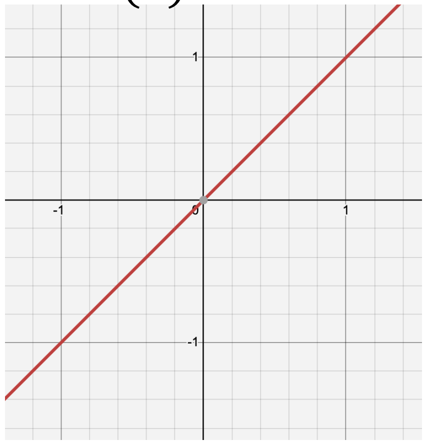
$$(1 + 2x + 3x^2)(4 + 5x + 6x^2) = 4 + 13x + 28x^2 + 27x^3 + 18x^4$$

	1	$2x$	$3x^2$
4	4	$8x$	$12x^2$
$5x$	$5x$	$10x^2$	$15x^3$
$6x^2$	$6x^2$	$12x^3$	$18x^4$

We can find the coefficients of a n-degree polynomial with n points on that polynomial

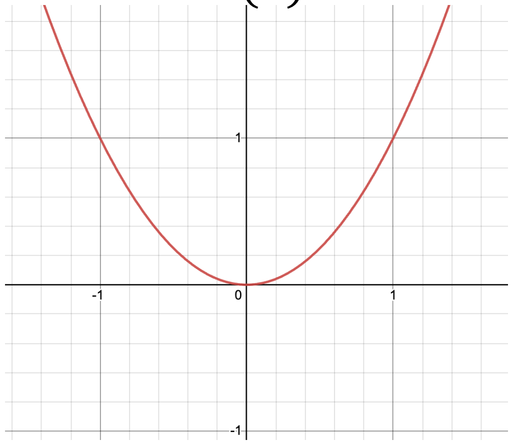
Expanding a polynomial and collecting like terms ≈ convolution

$F(x)$



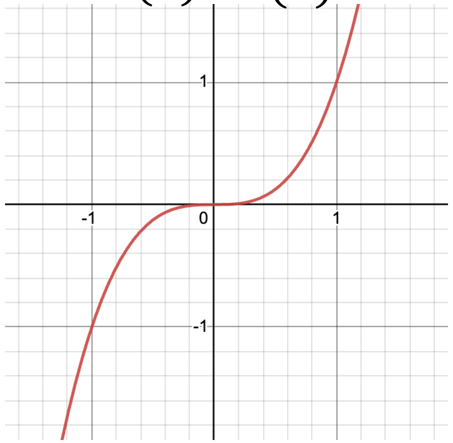
$\times$

$G(x)$



Point-wise multiplication

$F(x) \cdot G(x)$



Coefficients

Coefficients

$(a_0, a_1, a_2, \ldots, a_n)$

$(b_0, b_1, b_2, \ldots, b_n)$

Convolution

$$
\begin{aligned}
c_0 &= a_0 b_0 \\
c_1 &= a_0 b_1 + a_1 b_0 \\
c_2 &= a_0 b_2 + a_1 b_2 + a_2 b_0 \\
&\vdots
\end{aligned}
$$

$= a + bx + cx^2 + dx^3 + ex^4$ has following points $[x_1, x_2, x_3, x_4, x_5]$

This gives us, 5 unknown and 5 points to solve to get the polynomial.

For mathematical simplicity in solving for the pair of equations $[x_1, x_2, x_3, x_4, x_5]$ can be roots of unity $\approx$ Fourier Transformation

Roadmap for Fast Convolution (Convolution Theorem)

$$a = [a_0, a_1, a_2, \dots, a_{n-1}]$$
$$f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{n-1} x^{n-1}$$

$$\xrightarrow{\textit{FFT}}$$

$$\hat{a} = [\hat{a}_0, \hat{a}_1, \hat{a}_2, \hat{a}_3, \dots, \hat{a}_{m+n-1}]$$

$$b = [b_0, b_1, b_2, \dots, b_{n-1}]$$
$$g(x) = b_0 + b_1 x + b_2 x^2 + \cdots + b_{n-1} x^{n-1}$$

$$\xrightarrow{\textit{FFT}}$$

$$\hat{b} = [\hat{b}_0, \hat{b}_1, \hat{b}_2, \hat{b}_3, \dots, \hat{b}_{m+n-1}]$$

$$\downarrow \textit{Multiply (pointwise)}$$

$$\hat{a} \cdot \hat{b} = [\hat{a}_0 \hat{b}_0 + \hat{a}_1 \hat{b}_1 + \hat{a}_2 \hat{b}_2 \dots]$$

$$\xleftarrow{\textit{Inverse FFT}}$$

$$a * b = \begin{bmatrix} a_0 b_0, \\ a_0 b_1 + a_1 b_0, \\ a_0 b_2 + a_1 b_1 + a_2 b_0, \\ \vdots \\ a_{n-1} b_{m-1} \end{bmatrix}$$

Convolution Theorem

$\boxed{\{u * v\}(x) = \mathcal{F}^{-1}\{U \cdot V\} \quad \text{where U and V are the Fourier Transform of } u(x) \text{ and } v(x)}$

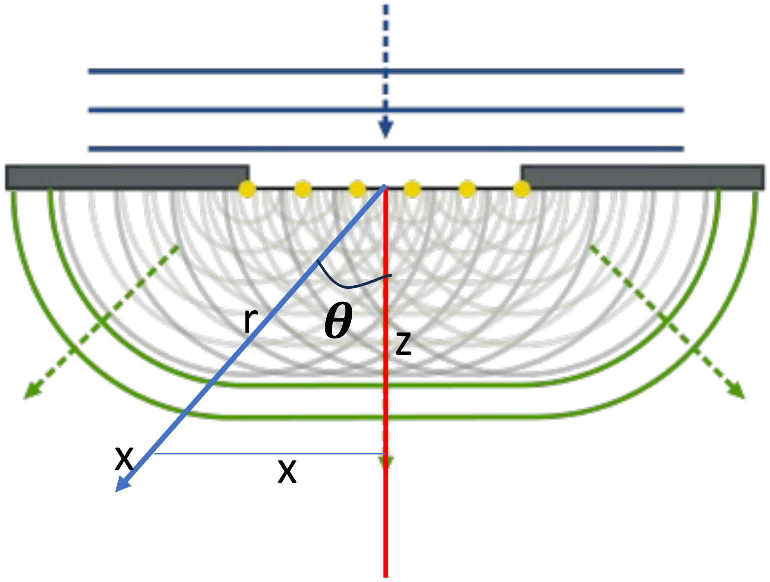
Fresnel Propagation Approximation

Where are we headed ? What are our main questions

- Given the field in one plane how do we find the field in another plane?
- How can we describe the imaging process using wave optics?

Huygen's Principle:

Each point in a field acts as a source emitting a spherical wave

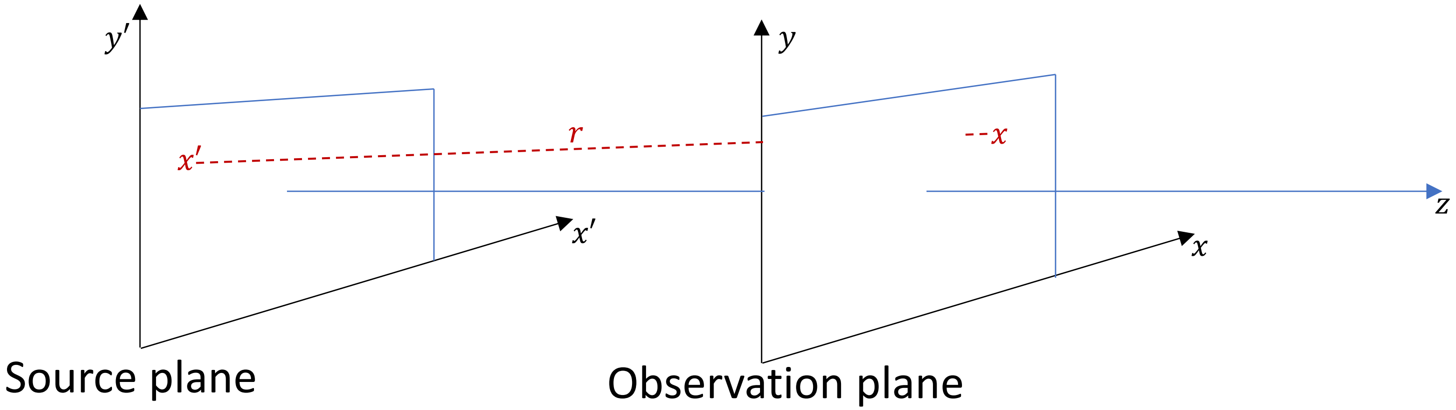


$$\mathcal{U}(x,t) = \cos(kr - wt)$$
$$= \frac{1}{r}\cos(kr - wt)$$
$$= \cos\theta\,\frac{1}{r}\cos(kr - wt)$$
$$= \frac{z}{r}\frac{1}{r}\cos(kr - wt)$$

Impulse response = $\frac{z}{r^2}e^{ikr}$

Rayleigh-Sommerfeld Integral:

add the contributions from all point source



$$r = \sqrt{(x - x')^2 + (y - y')^2 + z^2}$$

Ignored for fields that varies periodically with time (harmonic fields)

$$\boxed{U(x, y, z) = \frac{1}{\lambda i} \iint U(x', y', 0) \frac{z}{r^2}\, e^{ikr} dx' dy'}$$

How do we simplify the integral ? - Fresnel Approximation

- approximate when the propagation distance (z) is large, we can apply some approximation for the expression of r

Using Binomial Expansion:

$$\sqrt{1+b} = 1 + \frac{1}{2}b - \frac{1}{8}b^2 + \cdots$$

$$r = z\sqrt{1 + \frac{(x-x')^2}{z} + \frac{(y-y')^2}{z}}$$

$$r \approx z[1 + \frac{(x-x')^2}{2z} + \frac{(y-y')^2}{2z}]$$

Using only the first 2 terms of the binomial expansion

The integral now becomes:

$$U(x,y) = \frac{e^{ikz}}{i\lambda z} \iint U(x',y') \exp\left\{j\frac{k}{2z}[(x-x')^2 + (y-y')^2]\right\} dx'dy'$$

Starting to look like convolution of 2 functions

Convolution: $(f * g)(x) = \int f(\tau)g(t-\tau)d\tau$

In terms of convolution:

$$U(x,y) = \iint U(x',y')\, h(x-x',y-y')dx'dy'$$

Where the kernel is: $h(x,y) = \frac{e^{ijx}}{i\lambda z} \exp\left\{\frac{jk}{2z}(x^2+y^2)\right\}$

So applying Fresnel approximation we are able to realize that if we study the wave after a certain distance (within the range where Fresnel approximation is valid) we are essentially convoluting a signal with a defined kernel.

What potential could this have in making convolution operations in CNNs ?