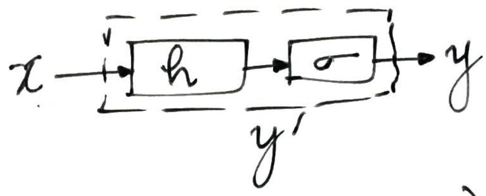
A Frequency Domain Neural Network for Fast Image Super-Resolution

Traditional CNN →



$$y = \sigma(\underbrace{x * h}_{y'})$$

$$\sigma(y') = \begin{cases} y', & y' \geq 0 \\ 0, & y' < 0 \end{cases}$$ for ReLU non-linearity

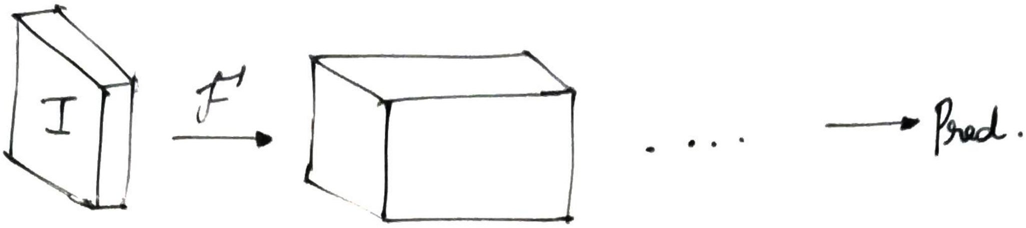
$$\sigma(y') = \max(0, y')$$
$$= y' \odot HS(y')$$

$\odot$ → Hadamard Product

$$HS(y') = \begin{cases} 1, & y' > 0 \\ 0.5, & y' = 0 \\ 0, & y' < 0 \end{cases}$$

$$\mathcal{F}(x * h) = \mathcal{F}(x) \odot \mathcal{F}(h)$$

$$\mathcal{F}((x * h) \odot k) = \Big(\mathcal{F}(x) \odot \underbrace{\mathcal{F}(h)}_{\text{weighing function}}\Big) * \underbrace{\mathcal{F}(k)}_{\text{Smoothing/ Regularization}}$$



Both $F'(h)$ and $F'(k)$ are learnt.

Hartley Transform used instead of Fourier transform to avoid dealing with complex Numbers.

$$\mathcal{H}(y') = \underbrace{\mathcal{R}(y')}_{\text{Real}} - \underbrace{\mathcal{I}(y')}_{\text{Imag.}}$$

$$\mathcal{H}(\mathcal{H}(y')) = y' \Rightarrow \text{Involution}$$

Gradients in the final layers << Gradients in the first few layers.

Use Gradient clipping and normalize parameters of a fixed range $(-\gamma\theta, \gamma\theta)$

10^{-5} / 10^{3}