Sequence Models and Attention

CS280

Spring 2025

Angjoo Kanazawa

Next few lectures

- Today: Transformers / Attention
- Next: Vision Transformers, DINO
- Next Week: Diffusion Models

What is Attention

A Computer Vision perspective

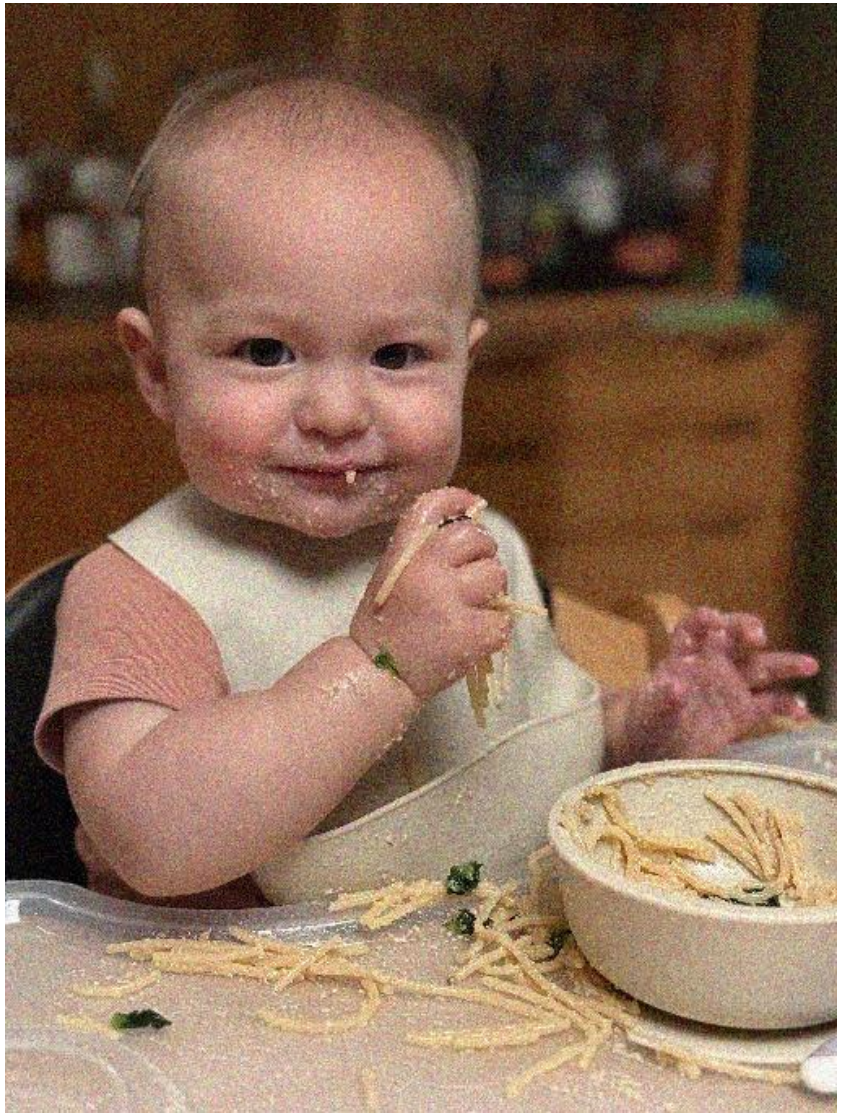
Computer Vision History on filtering

- Gaussian Filter
- Anisotropic Diffusion
- Bilateral Filter
- Non-Local Means

How to filter an image?

Gaussian filter

How to do edge preserving filter?



Physical analog: Heat diffusion

Heat Equation $$\frac{\partial u}{\partial t} = c\Delta u = \operatorname{div}(c\nabla u)$$

Moving from higher to lower concentration of signal u

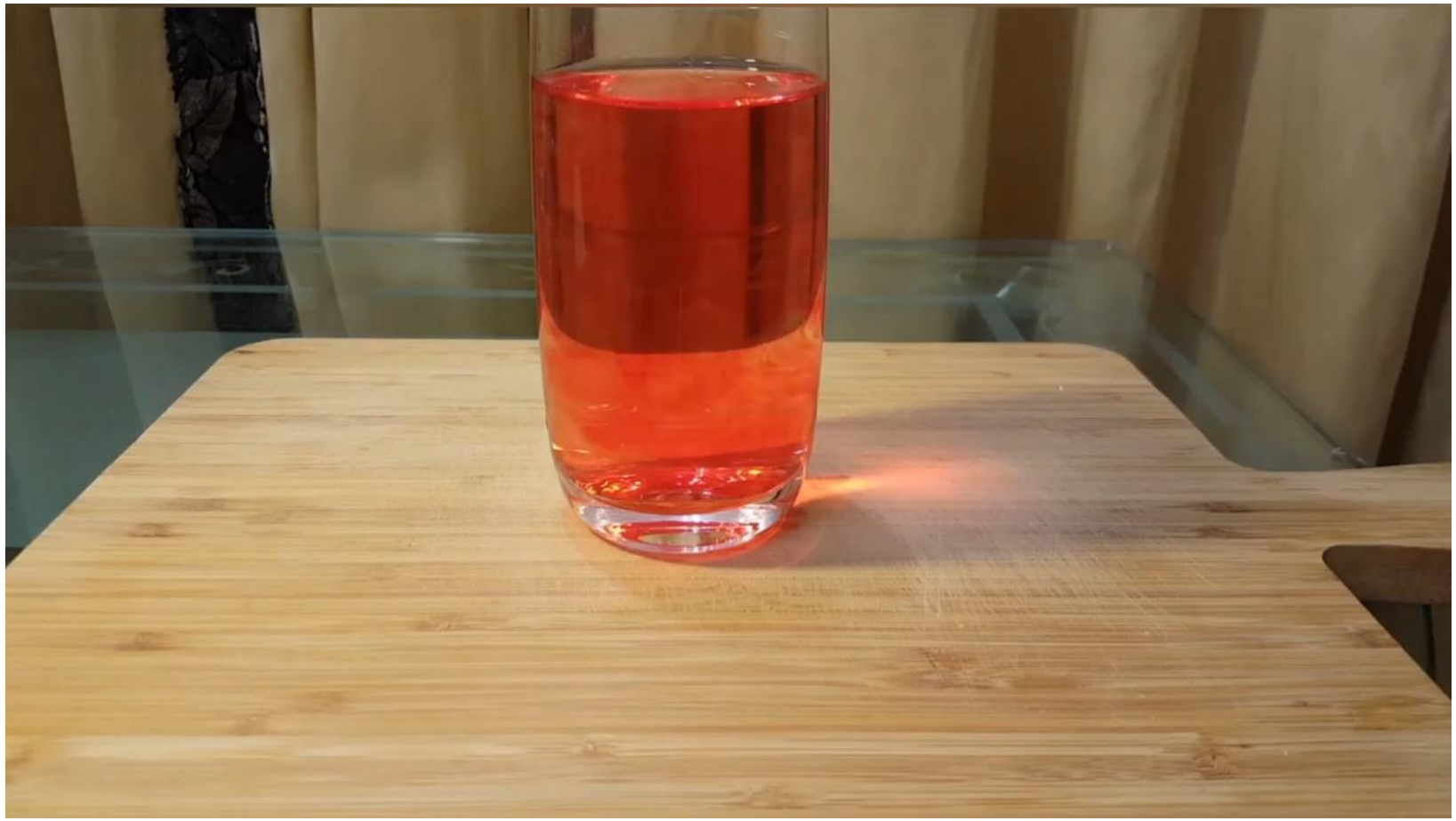
Physical analog: Heat diffusion

Heat Equation $\frac{\partial u}{\partial t} = c\Delta u = \mathrm{div}(c\nabla u)$

Moving from higher to lower concentration of signal u

Solution is exactly convolution with a gaussian kernel!

Useful intuition for later…



Heat Equation $\frac{\partial u}{\partial t} = c\Delta u = \mathrm{div}(c\nabla u)$

SDE formulation (how each particle moves): $dx(t) = cdW(t)$

How to do edge preserving filter?

How do we keep the two soups separate?

Why does gaussian filter destroy the edges?

Anisotropic Diffusion Perona & Malik 1990

- Idea: Look to see if there is a wall, modulate diffusion across the edge!
- Anisotropic Diffusion:
  - Treats the edges in the image like this wall

$$\frac{\partial u}{\partial t} = \mathrm{div}(c(|\nabla u|)\nabla u)$$

  - Makes the diffusion process, edge dependent
  - ➔ **Data dependent** filtering

Anisotropic Diffusion Results

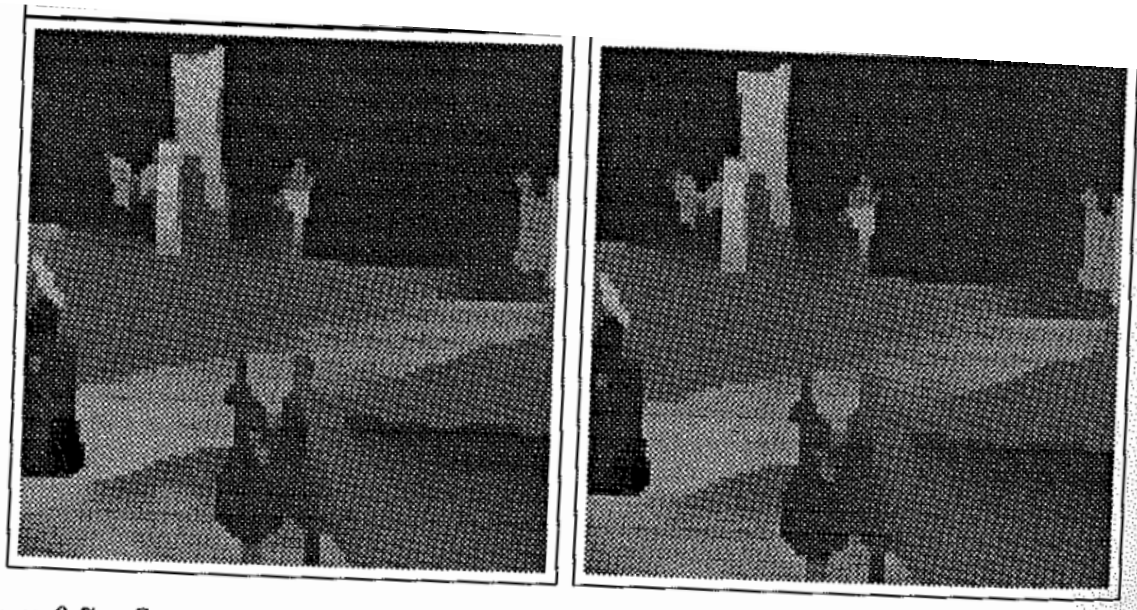


Figure 3.7. Sequence of images produced by anisotropic diffusion. The code presented in figure 3.5 was run on the image at the top-left corner for 10, 20, 30, 60, 100 iterations. The original image has pixel values between 0 (black) and 255 (white) and had a size of 100×100 pixels. The coefficient K was set equal to $K = 10$.

Aurich and Weule 1995
Tomasi and Manduchi 1998…

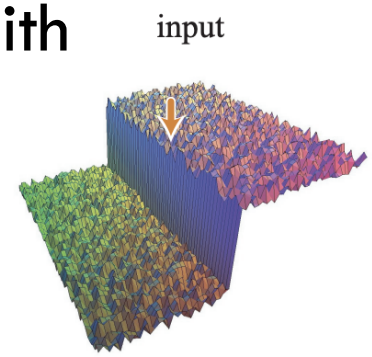
Bilateral Filter

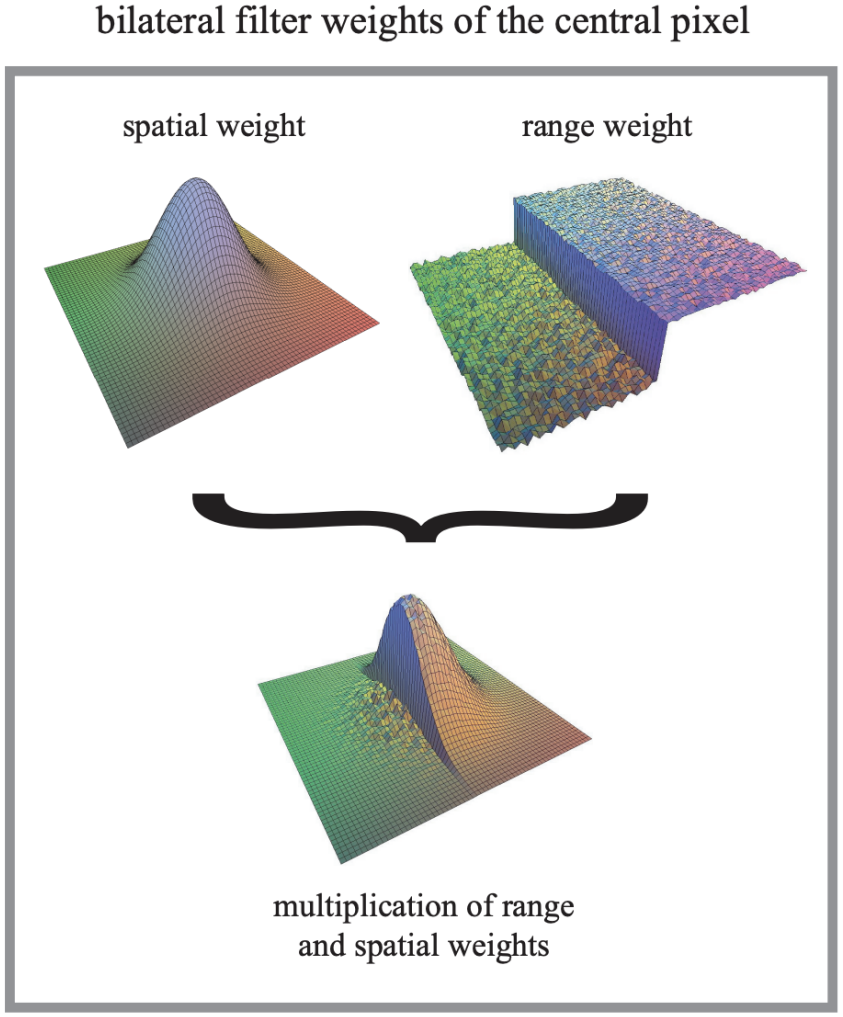
Photo from wikipedia

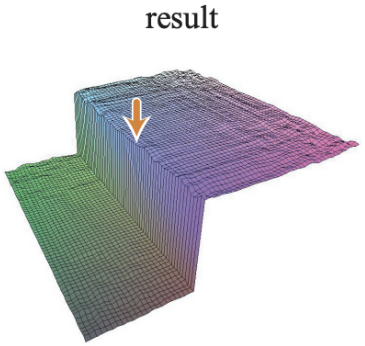
Aurich and Weule 1995
Tomasi and Manduchi 1998…

Bilateral Filter

- Inspired by Anisotropic Diffusion
- Weight gaussian kernel with pixel similarity
- Again data-dependent
- This is core idea of ROPE







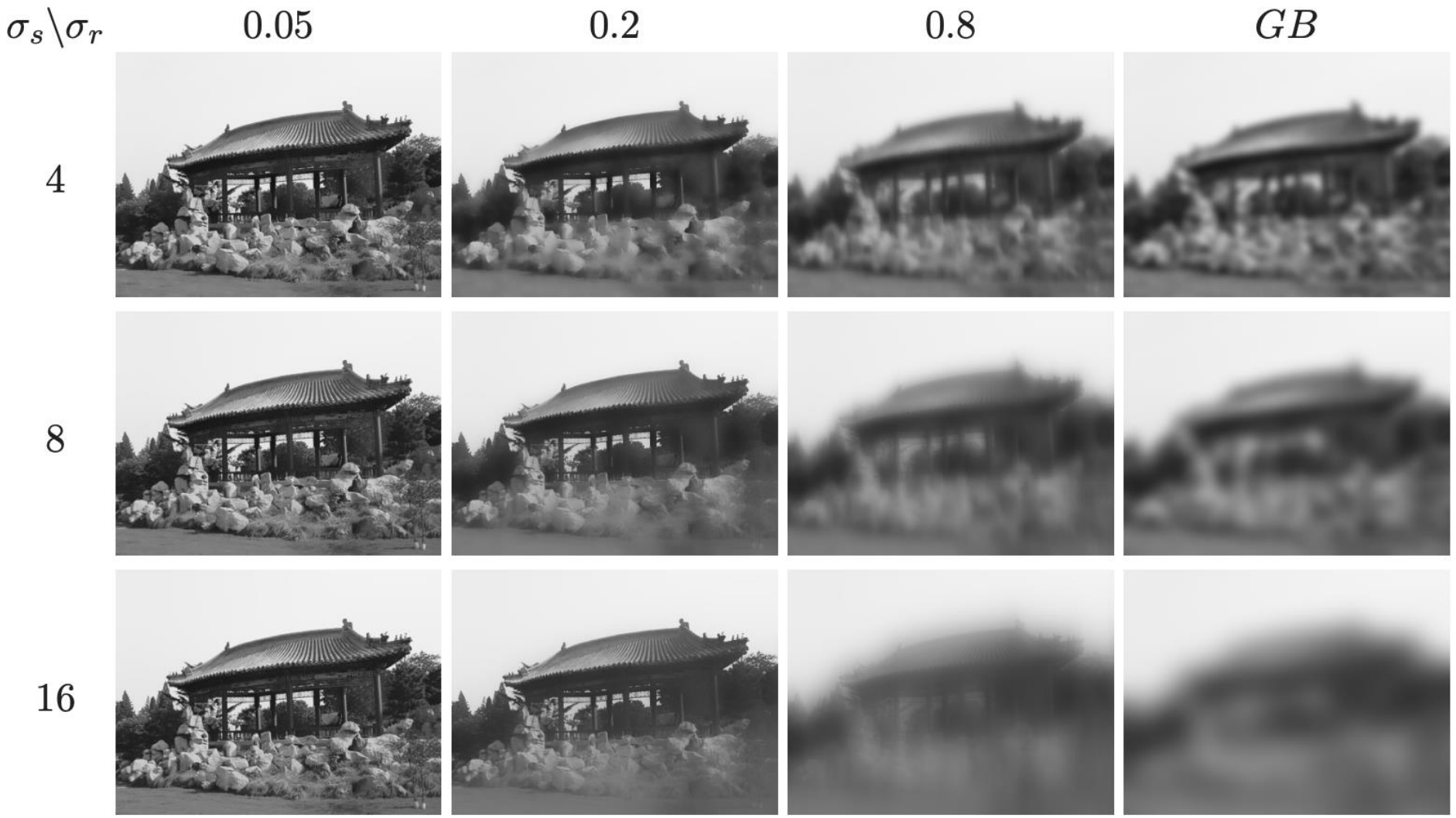
Distance Similarity

$$BF[I]_{\mathbf{p}} = \frac{1}{W_{\mathbf{p}}} \sum_{\mathbf{q} \in \mathcal{S}} G_{\sigma_s}(\|\mathbf{p} - \mathbf{q}\|)\, G_{\sigma_r}(I_{\mathbf{p}} - I_{\mathbf{q}})\, I_{\mathbf{q}}$$

Pixel Similarity

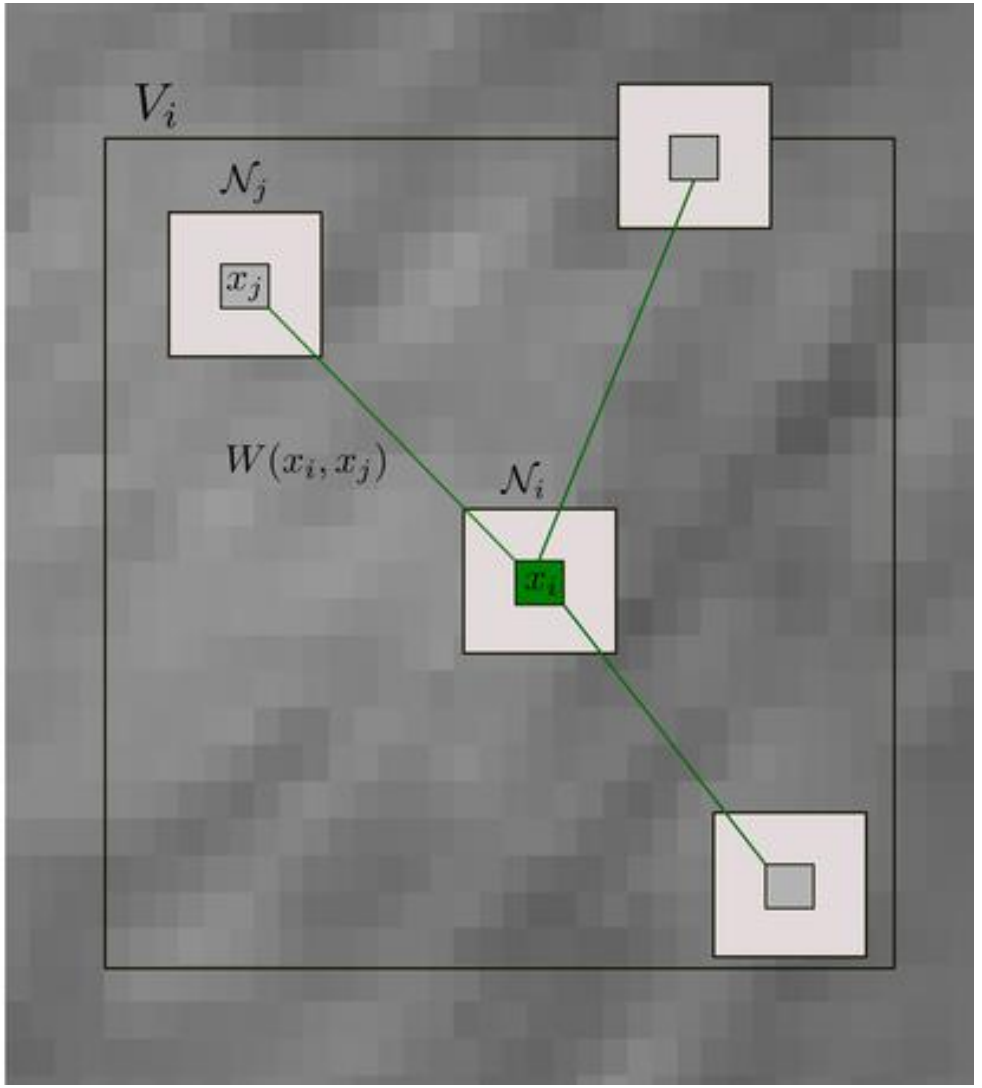
Figure from paris et al.

Bilateral Filter Results



Non-local means Buades, Coll, and Morel 2005

- Idea of a filter is to denoise by averaging similar pixels
- Why look at near by pixels only? The similar pixels **can be anywhere**!
- Idea: filter by average of similar patches, **from everywhere in the image**!
- Data-driven weights (similarity) with large field of view.



Non-local means

$$NL[v](i) = \sum_{j \in I} w(i,j)v(j),$$

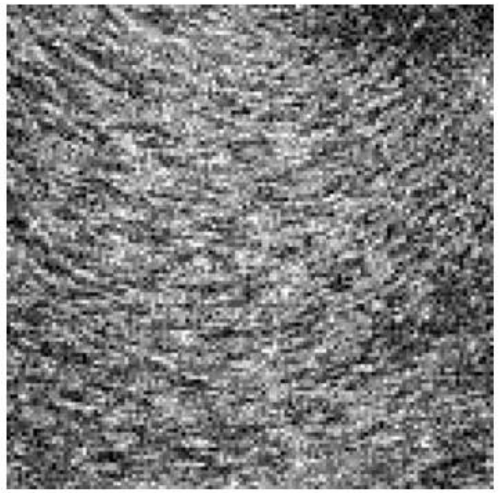
Weight is normalized affinity to all other pixels:

$$w(i,j) = \frac{1}{z_i} e^{-\|v_i - v_j\|/\sigma^2} \qquad z_i = \sum_j e^{-\|v_i - v_j\|/\sigma^2}$$

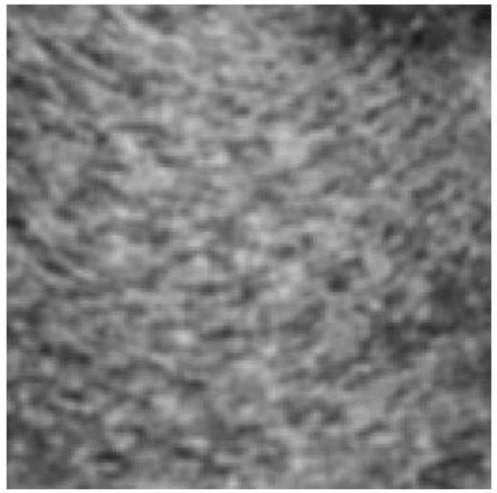
Softmax! In vectorized form, this is:

$$NL = softmax\big(dist(v_i, v_j)\big)V$$

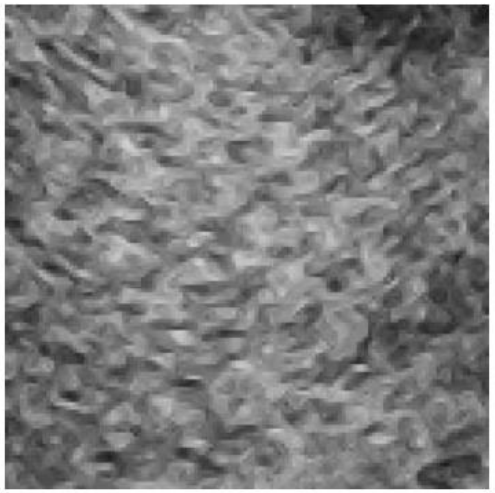
Results



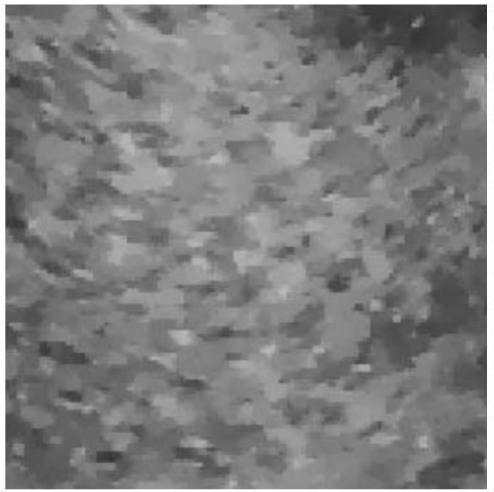
Input



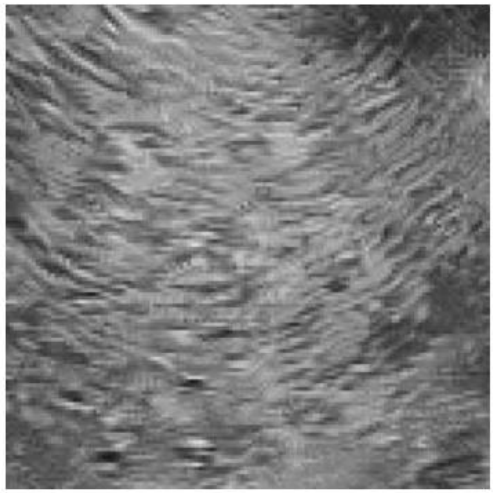
Gaussian Filter



Anisotropic Filter



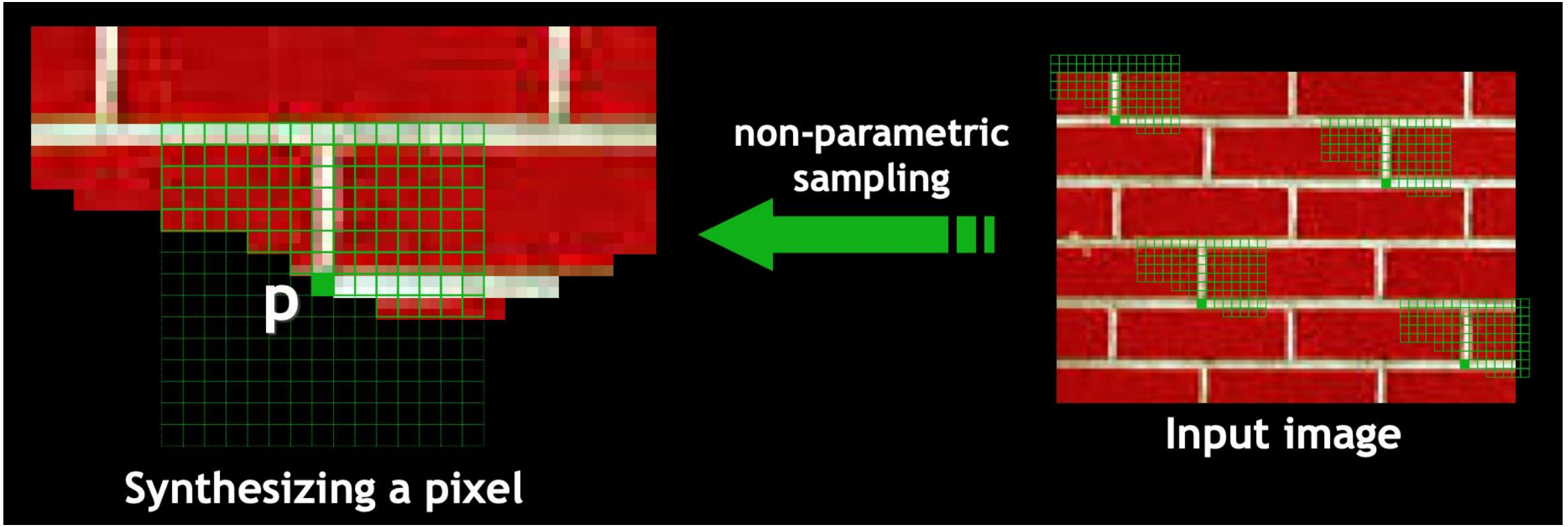
Total Variation



Non-local means

Aside Efros and Leung 1999

- Inspired non-local means



Two key ideas

In contrast to ordinary convlution by a fixed kernel

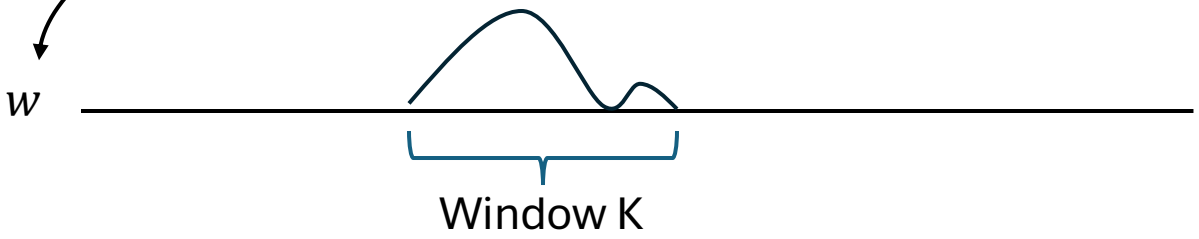
1. **Data-driven kernel** → Weights conditioned on the data point
2. **Full field of view** → non-local connections

These are the key difference between convnets and transformers

Convolution:

$$x' = Wx + b$$

$$x'_i = \sum_j^N w_j x_j$$



Attention:

$$\text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

$$x'_i = \sum_j^N w_j(x_i, x_j)x_j$$