

w05a — Bilateral filtering: solution

Q1. The second weight.

A Gaussian weights only by **spatial** distance $\|p - q\|$. The bilateral filter multiplies in a second, **range** weight that depends on the *intensity* difference $|I(p) - I(q)|$.

Edge preserving because: a neighbour on the far side of an edge has a very different intensity, so its range weight is near zero and it contributes almost nothing. Averaging therefore happens *within* a region but not *across* its boundary.

Q2. The 1D signal.

10, 12, 11, 10, **60**, 62, 61, 60

Filtering the first pixel after the edge (value 60) with a ± 3 spatial window and $\sigma_r = 5$:

- **Contribute:** 62, 61, 60 — the neighbours to the right, all within a few intensity levels of 60.
- **Suppressed:** 10, 11, 12 — the neighbours to the left, roughly 50 grey levels away, i.e. $10\sigma_r$. Their range weight is $e^{-50^2/(2 \cdot 5^2)}$, which is zero for practical purposes.

So the output stays near 60; the step is preserved. A plain box or Gaussian filter over the same window would pull it down towards the middle and smear the edge.

Q3. $\sigma_r \rightarrow \infty$.

The range weight $\rightarrow 1$ for every neighbour, so only the spatial term survives:

it degenerates to an ordinary Gaussian filter.

Q4. Why it is not a convolution.

Convolution is **linear** and **shift-invariant**: the same fixed kernel is applied at every pixel, independent of image content. The bilateral filter's weights depend on the intensities in the neighbourhood, so the effective kernel changes from pixel to pixel and depends on the input. It is therefore neither linear nor shift-invariant, and cannot be expressed as convolution with any fixed kernel.

Consequences worth mentioning: no separability, no FFT speed-up, and no simple frequency-domain interpretation.

Either way of putting it captures the point: “it is not linear”, or “the kernel depends on the data”.