

The simplest filter to implement is known as the [mean filter]³. For example if a kernel A was used with $m(\text{size})=3$: $A_{\text{avg}} = 1/3 [1 \ 1 \ 1]$ Let I be an image of size N, m an odd number smaller than N, and A the kernel of a linear filter, that is a mask of size m. Additionally, it is absolutely necessary for all the entries in the kernel to have a sum of one. If the sum is not equal to one, then the kernel must be divided by the sum of the entries (hence the multiplication of the $1/3$). If the requirement is not met, then the filtered image will become brighter than the original image, along with undergoing the specified filtering effect. This limitation on the mean filter fulfills the second portion of the image filtering goal A. This filter is effective at attenuating noise because averaging removes small variations. In a two-dimensional mean filter, the effect of averaging m^2 noisy values around pixel divides the standard deviation of the noise by $\sqrt{m^2} = m(\text{size})$. Each pixel in I (image) is replaced with the mean of the pixels that surround it. Especially, noise is blended into the rest of the picture. A filter that performs average smoothing must use a kernel with all entries being non-negative. The mean filter performs average smoothing on an image. The name perfectly describes the function of this filter. The effect is identical to that of averaging a set of .data to help reduce the effect of outliers