

Kernel (image processing)

In image processing, a **kernel**, **convolution matrix**, or **mask** is a small **matrix** useful for blurring, sharpening, embossing, edge-detection, and more. This is accomplished by means of **convolution** between a kernel and an image.

1 Details

Depending on the element values, a kernel can cause a wide range of effects.

The above are just a few examples of effects achievable by convolving kernels and images.

1.1 Origin

The origin is the position of the kernel which is above (conceptually) the current output pixel. This could be outside of the actual kernel, though usually it corresponds to one of the kernel elements. For a symmetric kernel, the origin is usually the center element.

2 Convolution

This is loosely related to a form of **mathematical convolution**. It should be noted that the matrix operation, convolution, is not traditional matrix multiplication, even though it is, typically, denoted by *. In words convolution is: given two three-by-three matrices, one a kernel, and the other an image piece, convolution is the process of multiplying locationally similar entries and summing:

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} * \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} = (1 * a) + (2 * b) + (3 * c) + (4 * d) + (5 * e) + (6 * f) + (7 * g) + (8 * h) + (9 * i)$$

The values of a given pixel in the output image are calculated by multiplying each kernel value by the corresponding input image pixel values. This can be described algorithmically with the following pseudo-code:

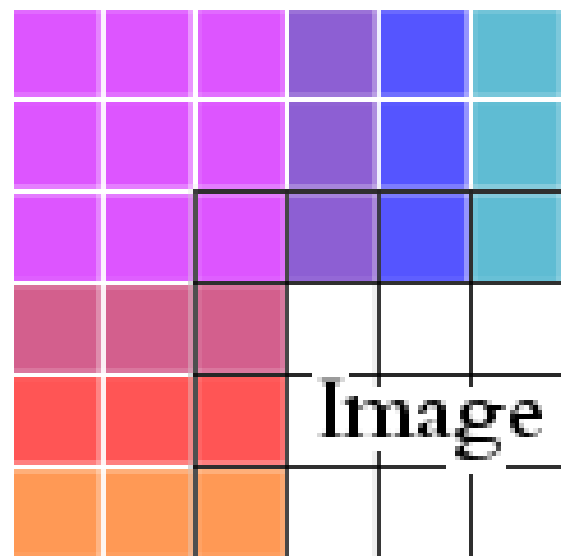
```
for each image row in input image: for each pixel in image row: set accumulator to zero for each kernel row in kernel: for each element in kernel row: if element position corresponding* to pixel position then multiply element value corresponding* to pixel value add result to accumulator endif set output image pixel to accumulator
```

*corresponding input image pixels are found relative to the kernel's origin.

If kernel is symmetric then place the centre(origin) of kernel on the current pixel. Then kernel will be overlapped with neighboring pixels too. Now multiply each kernel element with the pixel value it overlapped with and add all the obtained values. Resultant value will be the value for the current pixel that is overlapped with the center of the kernel.

If the kernel is not symmetric, it has to be flipped both around its horizontal and vertical axis before calculating the convolution as above.^[1]

2.1 Edge Handling



Extend Edge-Handling

Kernel convolution usually requires values from pixels outside of the image boundaries. There are a variety of methods for handling image edges.

Extend The nearest border pixels are conceptually extended as far as necessary to provide values for the convolution. Corner pixels are extended in 90° wedges. Other edge pixels are extended in lines.

Wrap The image is conceptually wrapped (or tiled) and values are taken from the opposite edge or corner.

Crop Any pixel in the output image which would require values from beyond the edge is skipped. This method can result in the output image being slightly smaller, with the edges having been cropped.

2.2 Normalization

Dividing each element in the kernel by the sum of all the absolute values of the elements in the kernel. Normalization ensures that the pixel values in the output image are of the same relative magnitude as those in the input image.

3 External links

- [Implementing 2d convolution on FPGA](#)
- [vImage Programming Guide: Performing Convolution Operations](#)
- [Image Processing using 2D-Convolution](#)
- [GNU Image Manipulation Program - User Manual - 8.2. Convolution Matrix](#)
- [Interactive Demonstration of 3x3 Convolution Kernels](#)

4 References

- [1] http://www.songho.ca/dsp/convolution/convolution2d_example.html
- Ludwig, Jamie (n.d.), “Image Convolution”. Portland State University. http://web.pdx.edu/~jjduh/courses/Archive/geog481w07/Students/Ludwig_ImageConvolution.pdf
 - Lecarme, Olivier; Delvare, Karine (January 2013). *The Book of GIMP: A Complete Guide to Nearly Everything*. No Starch Press, p. 429, ISBN 978-1593273835.
 - Gumster, Jason van; Shimonski, Robert (March 2012). *GIMP Bible*. Wiley, pp. 438–442, ISBN 978-0470523971.
 - Stockman, George C.; Shapiro, Linda G. (February 2001). *Computer Vision*. Prentice Hall, pp. 53–54, ISBN 978-0130307965.

5 Text and image sources, contributors, and licenses

5.1 Text

- **Kernel (image processing)** *Source:* [https://en.wikipedia.org/wiki/Kernel_\(image_processing\)?oldid=680825047](https://en.wikipedia.org/wiki/Kernel_(image_processing)?oldid=680825047) *Contributors:* Bearcat, Cydebot, Marcuscalabresus, Shirt58, Dekart, Alvin Seville, Thrustin, Skamecrazy123, KLBot2, Gnidlih, Frze, ChrisGualtieri, Maecken2, Plotke, Puggsoy, SherbalTea, Michael in wiki, Gblikas, Vamsidhar muthireddy, Nt-occ, Leonardoaraujo.santos and Anonymous: 6

5.2 Images

- **File:Extend_Edge-Handling.png** *Source:* https://upload.wikimedia.org/wikipedia/commons/4/4f/Extend_Edge-Handling.png *License:* CC BY-SA 3.0 *Contributors:* Own work *Original artist:* Michael Plotke
- **File:Vd-Blur1.png** *Source:* <https://upload.wikimedia.org/wikipedia/commons/2/28/Vd-Blur1.png> *License:* CC BY-SA 3.0 *Contributors:* Own work *Original artist:* Michael Plotke
- **File:Vd-Blur2.png** *Source:* <https://upload.wikimedia.org/wikipedia/commons/0/04/Vd-Blur2.png> *License:* CC BY-SA 3.0 *Contributors:* Own work *Original artist:* Michael Plotke
- **File:Vd-Edge1.png** *Source:* <https://upload.wikimedia.org/wikipedia/commons/8/8d/Vd-Edge1.png> *License:* CC BY-SA 3.0 *Contributors:* Own work *Original artist:* Michael Plotke
- **File:Vd-Edge2.png** *Source:* <https://upload.wikimedia.org/wikipedia/commons/8/83/Vd-Edge2.png> *License:* CC BY-SA 3.0 *Contributors:* Own work *Original artist:* Michael Plotke
- **File:Vd-Edge3.png** *Source:* <https://upload.wikimedia.org/wikipedia/commons/6/6d/Vd-Edge3.png> *License:* CC BY-SA 3.0 *Contributors:* Own work *Original artist:* Michael Plotke
- **File:Vd-Orig.png** *Source:* <https://upload.wikimedia.org/wikipedia/commons/5/50/Vd-Orig.png> *License:* CC BY-SA 3.0 *Contributors:* Own work *Original artist:* Michael Plotke
- **File:Vd-Sharp.png** *Source:* <https://upload.wikimedia.org/wikipedia/commons/4/4e/Vd-Sharp.png> *License:* CC BY-SA 3.0 *Contributors:* Own work *Original artist:* Michael Plotke

5.3 Content license

- Creative Commons Attribution-Share Alike 3.0