L-Università ta' Malta

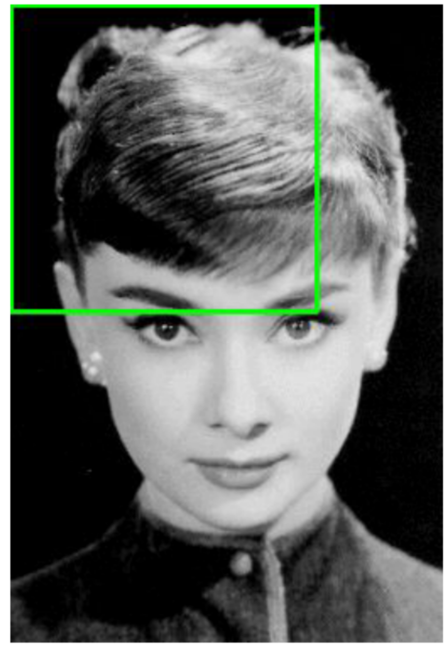
Area Processing - Tutorial

Mark Bugeja, Dylan Seychell

Agenda

- Sliding Window (Striding Window)
- Convolution
- Experiments

Sliding Window



Sliding window (sometimes referred to as a kernel) is simply a rectangular region that moves through an image.

This technique is used extensively in computer vision as an aide to solve a number of tasks such as face recognition, object detection, feature extraction and filter based manipulations.

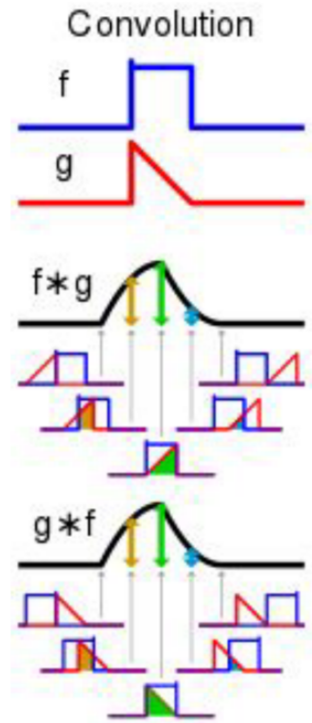
Exercise 1: Sliding Window

Create a function using python that creates an nxn sliding window that strides through an image.

Your function should accept three parameters, an image (to slide through), the `n` dimension of the window and the stride value `s`. *(Stride simply refers to the amount of pixels the window moves through kind of like "speed", for computational purposes in practise we rarely slide the window at a speed of 1 pixel)*

The function does not return anything, but internally it should display a matrix of pixel values that displays the current region the sliding window is in. *(You can also if you want display the sliding window values as an image using `cv2.imshow()` this could potentially help you debug any errors you might have or show the sliding window moving on the image using `cv2.rectange()`)*

Convolution



Convolution in signal theory is simply the mathematical operation of two functions (signals) and the result is a function (signal) that describes how one function affects the shape of the other.

Convolution is also the basis of Convolutional Neural Networks (CNN) which is widely used in a variety of object detection tasks in computer vision.

This “usefulness” of convolution is due to the features that can be extracted from the convolution operation. This includes features such as edges, shapes and textures.

Convolution Kernels

Different kernels result in different effects.

These effects describe the features that can be extracted from the image.

Certain kernels might be better at solving certain tasks. *Can you think of any? And why?*

$$\frac{1}{K^2}\begin{bmatrix}1 & 1 & \cdots & 1\\ 1 & 1 & \cdots & 1\\ \vdots & \vdots & 1 & \vdots\\ 1 & 1 & \cdots & 1\end{bmatrix}\quad \frac{1}{16}\begin{bmatrix}1 & 2 & 1\\ 2 & 4 & 2\\ 1 & 2 & 1\end{bmatrix}\quad \frac{1}{256}\begin{bmatrix}1 & 4 & 6 & 4 & 1\\ 4 & 16 & 24 & 16 & 4\\ 6 & 24 & 36 & 24 & 6\\ 4 & 16 & 24 & 16 & 4\\ 1 & 4 & 6 & 4 & 1\end{bmatrix}\quad \frac{1}{8}\begin{bmatrix}-1 & 0 & 1\\ -2 & 0 & 2\\ -1 & 0 & 1\end{bmatrix}\quad \frac{1}{4}\begin{bmatrix}1 & -2 & 1\\ -2 & 4 & -2\\ 1 & -2 & 1\end{bmatrix}$$

$$\frac{1}{K}\begin{bmatrix}1 & 1 & \cdots & 1\end{bmatrix}\quad \frac{1}{4}\begin{bmatrix}1 & 2 & 1\end{bmatrix}\quad \frac{1}{16}\begin{bmatrix}1 & 4 & 6 & 4 & 1\end{bmatrix}\quad \frac{1}{2}\begin{bmatrix}-1 & 0 & 1\end{bmatrix}\quad \frac{1}{2}\begin{bmatrix}1 & -2 & 1\end{bmatrix}$$

(a) box, $K = 5$ (b) bilinear (c) "Gaussian" (d) Sobel (e) corner

Figure 3.14 Separable linear filters: For each image (a)–(e), we show the 2D filter kernel (top), the corresponding horizontal 1D kernel (middle), and the filtered image (bottom). The filtered Sobel and corner images are signed, scaled up by $2\times$ and $4\times$, respectively, and added to a gray offset before display.

Figure taken from, Szeliski, Richard. Computer vision: algorithms and applications. Springer Science & Business Media, 2010. Page 116.

Exercise 2: Convolution on RoI

Create a function that convolves an `nxn` region of an image with a Sobel kernel.

Your function should accept the region of interest (ROI) of the image of `nxn` dimensions, and a kernel.

Display the result for various images *(You may start the exercise by converting the image to grayscale, ultimately your function should be able to process RGB Images)*

Exercise 3: Convolution on the whole image

Create a script that uses the sliding window and the convolution functions (Sobel) to convolve a whole image.

The script should output an image displaying the original image and the convolved image.

Use histograms to compare how the intensities of the image has changed.

Exercise 4: Different Convolution Kernels

Apply Bilinear and Gaussian kernels to the previous exercise. *(You should have three in total including Sobel)*

Document the results and compare the effect that the convolution of these kernel has on the image.

You may initially use histograms to compare pixel values and intensity, you should also visually compare the results and discuss what potential effect this might have in terms of feature extraction and/or feature understanding. *(We know that at this point this is your opinion. We are looking for insights on what you think no answer is completely right or wrong)*

References

Academic Reference

- Lecture Slides
- Hirschman, Isidore Isaac, and David V. Widder, “The Convolution Transform”. Courier Corporation, 2012.

Insightful Blogs

https://towardsdatascience.com/a-comprehensive-introduction-to-different-types-of-convolutions-in-deep-learning-669281e58215

https://towardsdatascience.com/types-of-convolution-kernels-simplified-f040cb307c37