**LAPORAN PRAKTIKUM PENGOLAHAN CITRA DIGITAL**

**10. SMOOTHING FILTERS IN THE SPATIAL DOMAIN**



**Disusun oleh :**

**Nama : Andhika Rizky Cahya Putra**

**NPM : 2226250071**

**Kelas : IF4A**

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS ILMU KOMPUTER DAN REKAYASA**

**UNIVERSITAS MULTI DATA PALEMBANG**

**2024**

**TUTORIAL : SMOOTHING FILTERS IN THE SPATIAL DOMAIN**

**Goal**

The goal of this tutorial is to learn how to implement smoothing filters in the spatial

domain.

**Objectives**

* Learn how to use the fspecial function to generate commonly used kernels.
* Explore applying smoothing filters to images using the imfilter function.
* Learn how to implement uniform and nonuniform averaging masks.
* Learn how to implement a Gaussian mask.

**Procedure**

In the first part of this procedure, we will use the imfilter function to implement a 3×3 mean (average) filter. We could easily generate the mask array ourselves (nine values, each equal to 1/9), but the IPT offers a function that will automatically create this and several other commonly used masks.

1. Load the cameraman image and prepare a subplot.



1. Create a mean (averaging) filter automatically through the fspecial function.



**Question 1** Explain what the value of the variable fn represents.

|  |
| --- |
|  |

**Question 2** What other commonly used masks is the fspecial function capable of generating?

|  |
| --- |
|  |

1. Filter the cameraman image with the generated mask.



**Question 3** What was the effect of the averaging filter?

|  |
| --- |
|  |

The mean filter we just implemented was a uniform filter—all coefficients were equivalent. The nonuniform version of the mean filter gives the center of the mask (the pixel in question) a higher weighted value, while all other coefficients are weighted by their distance from the center. This particular mask cannot be generated by the fspecial function, so we must create it ourselves.

1. Create a nonuniform version of the mean filter.

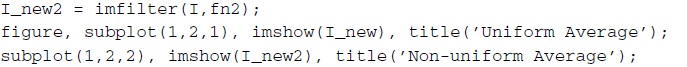




**FIGURE 10.1** Uniform and nonuniform averaging masks.

Recall that the uniform mean filter could be created by generating a 3 × 3 matrix of 1’s, and then multiplying each coefficient by a factor of 1/9. In the nonuniform mean filter implantation above, note that the sum of all the original values in the filter equals 16—this is why we divide each coefficient by 16 in the second step. Figure 10.1 illustrates the previous two masks we created.

1. Filter the original image with the new, nonuniform averaging mask.

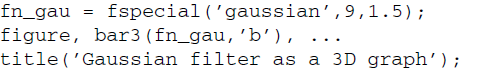


**Question 4** Comment on the subjective differences between using the uniform averaging filter and the nonuniform averaging filter.

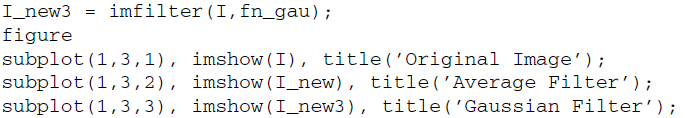
|  |
| --- |
|  |

The Gaussian filter is similar to the nonuniform averaging filter in that the coefficients are not equivalent. The coefficient values, however, are not a function of their distance from the center pixel, but instead are modeled from the Gaussian curve.

1. Create a Gaussian filter and display the kernel as a 3D plot.



1. Filter the cameraman image using the Gaussian mask.

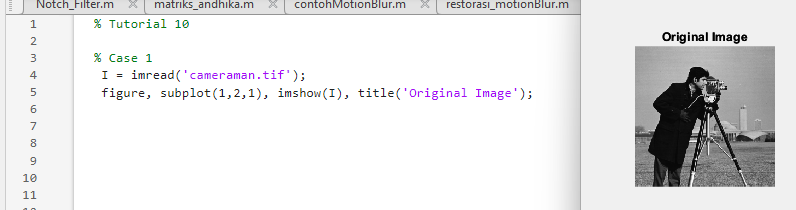


**Question 5** Experiment with the size of the Gaussian filter and the value of σ. How can you change the amount of blur that results from the filter?

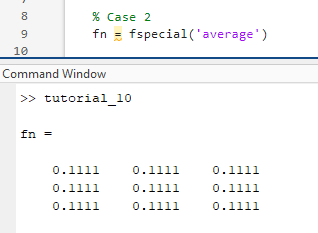
|  |
| --- |
|  |

**Screenshot Kode**

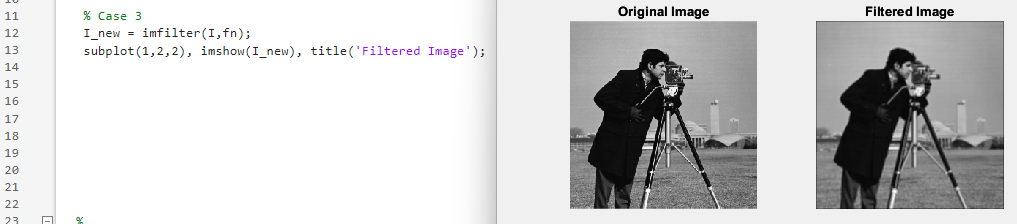
**Case 1**

****

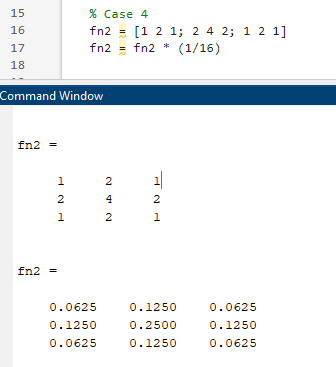
**Case 2**

****

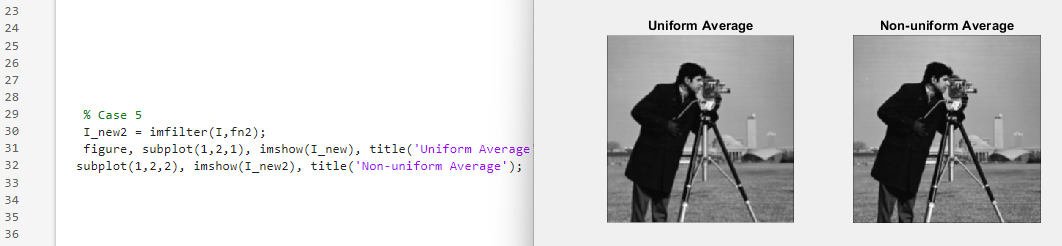
**Case 3**

****

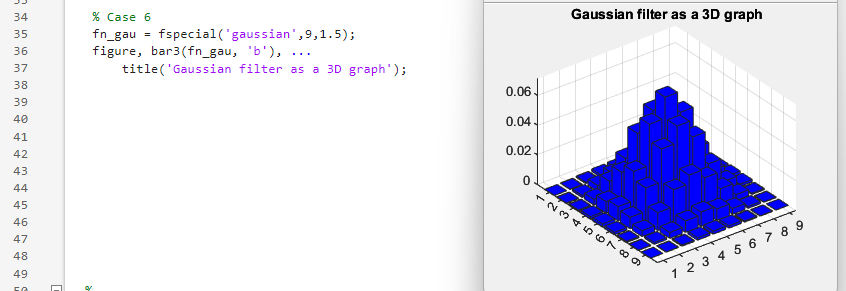
**Case 4**

****

**Case 5**

****

**Case 6**

****

**Case 7**

****