**LAPORAN PRAKTIKUM PENGOLAHAN CITRA DIGITAL**

**3. LOGIC OPERATIONS AND REGION OF INTEREST**

**PROCESSING**



**Disusun oleh :**

**Nama : Garcia Bryan Farrel**

**NPM : 2327250026**

**Kelas : IF41**

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS ILMU KOMPUTER DAN REKAYASA**

**UNIVERSITAS MULTI DATA PALEMBANG**

**2024**

**TUTORIAL 3. LOGIC OPERATIONS AND REGION OF INTEREST**

**PROCESSING**

**Goal**

The goal of this tutorial is to learn how to perform logic operations on images.

**Objectives**

* Explore the roipoly function to generate image masks.
* Learn how to logically AND two images using the bitand function.
* Learn how to logically OR two images using the bitor function.
* Learn how to obtain the negative of an image using the bitcmp function.
* Learn how to logically XOR two images using the bitxor function.

**What You Will Need**

* lindsay.tif
* cameraman2.tif

**Procedure**

Logic operators are often used for image masking. We will use the roipoly function to create the image mask. Once we have a mask, we will use it to perform logic operations on the selected image.

1. Use the MATLAB help system to learn how to use the roipoly function when only an image is supplied as a parameter.

**Question 1** How do we add points to the polygon?

|  |  |
| --- | --- |
|  |  |

**Question 2** How do we delete points from the polygon?

|  |  |
| --- | --- |
| Click kanan titik atau point |  |

**Question 3** How do we end the process of creating a polygon?

|  |
| --- |
| Click kanan 2 kali pada titik polygon |

1. Use the roipoly function to generate a mask for the pout image.



**Question 4** What class is the variable bw?

|  |
| --- |
| hasilnya (bw) adalah sebuah array logika (logical array). Setiap elemen bw hanya memiliki nilai:  1 (true) → di dalam poligon dan 0 (false) → di luar poligon |

**Question 5** What does the variable bw represent?

|  |
| --- |
| bw adalah sebuah *mask biner* (binary mask) |

Logic functions operate at the bit level; that is, the bits of each image pixel are compared individually, and the new bit is calculated based on the operator we are using (AND, OR, or XOR). This means that we can compare only two images that have the same number of bits per pixel as well as equivalent dimensions. In order for us to use the bw image in any logical calculation, we must ensure that it consists of the same number of bits as the original image. Because the bw image already has the correct number of rows and columns, we need to convert only the image to uint8, so that each pixel is represented by 8 bits.

1. Convert the mask image to class uint8.



|  |  |
| --- | --- |
|  |  |

**Question 6** In the above conversion step, what would happen if we used the im2uint8 function to convert the bw image as opposed to just using uint8(bw)? (Hint: after conversion, check what is the maximum value of the image bw2.)

|  |
| --- |
| Tidak bisa karena memiliki ukuran yang berbeda |

1. Use the bitand function to compute the logic AND between the original image and the new mask image.

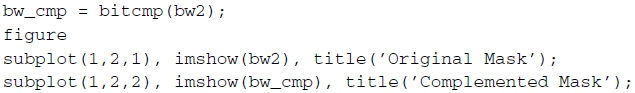


**Question 7** What happens when we logically AND the two images?

|  |  |
| --- | --- |
|  | Karena ukuran beda |

To see how to OR two images, we must first visit the bitcmp function, which is used for complementing image bits (NOT).

1. Use the bitcmp function to generate a complemented version of the bw2 mask.



|  |  |
| --- | --- |
|  |  |

**Question 8** What happened when we complemented the bw2 image?

|  |
| --- |
| Saat complemented gambar bw2 akan terjadi membalikkan semua nilai biner dari 0 menjadi 1 dan sebaliknya |

We can now use the complemented mask in conjunction with bitor.

1. Use bitor to compute the logic OR between the original image and the complemented mask.



|  |  |
| --- | --- |
|  |  |

**Question 9** Why did we need to complement the mask? What would have happened if we used the original mask to perform the OR operation?

|  |
| --- |
| Untuk digunakan ketika kita ingin membalikkan area yang dipilih atau dipengaruhi oleh masker |

The IPT also includes function imcomplement, which performs the same operation as the bitcmp function, complementing the image. The function imcomplement allows input images to be binary, grayscale, or RGB, whereas bitcmp requires that the image be an array of unsigned integers.

1. Complement an image using the imcomplement function.



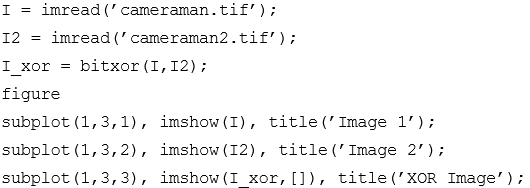
|  |  |
| --- | --- |
|  |  |

**Question 10** How can we check to see that the bw\_cmp2 image is the same as the bw\_cmp image?

|  |  |
| --- | --- |
|  |  |

The XOR operation is commonly used for finding differences between two images.

1. Close all open figures and clear all workspace variables.
2. Use the bitxor function to find the difference between two images.



|  |  |
| --- | --- |
|  |  |

Logic operators are often combined to achieve a particular task. In next steps, we will use all the logic operators discussed previously to darken an image only within a region of interest.

1. Close all open figures and clear all workspace variables.
2. Read in image and calculate an adjusted image that is darker using the imdivide function.



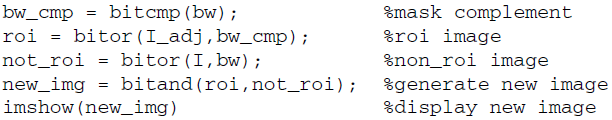
|  |  |
| --- | --- |
| Tidak ada (Lindsay.tif) jadi pakai cameraman.tif |  |

1. Generate a mask by creating a region of interest polygon.



|  |  |
| --- | --- |
|  | Before |
|  | After |

1. Use logic operators to show the darker image only within the region of interest, while displaying the original image elsewhere.



|  |  |
| --- | --- |
|  |  |

**Question 11** How could we modify the above code to display the original image within the region of interest and the darker image elsewhere?

|  |
| --- |
|  |
|  |