CHAPTER 1 REVIEW NOTES

Computer Vision - Master IPS

EL MAHRAOUI AMAL

I Computer Vision

I.1 What is Computer Vision?

Computer vision is a broader field that enables computers to understand, interpret, and make decisions based on visual data (images or videos). It goes beyond image processing by seeking to extract meaningful information and make sense of the visual content, often in a way similar to how humans perceive the world. Some common tasks in computer vision include:

- **Object recognition**: Identifying and classifying objects within an image.
- **Object tracking**: Following the movement of objects across multiple frames (in video).
- **Scene reconstruction**: Creating 3D models from 2D images.
- **Facial recognition**: Identifying faces in images and videos.
- **Action recognition**: Identifying actions or events happening in a sequence of images.

Goal
The goal of computer vision is to enable a machine to ”see” and understand what is happening in the visual environment and make decisions or predictions based on that understanding.

II Image Processing

II.1 What is Image Processing?

Image processing refers to the manipulation of images to enhance or extract useful information from them. The primary focus is on improving the image quality or preparing it for further analysis. Image processing includes tasks such as:

- **Filtering**: Removing noise, blurring, sharpening, etc.
- **Edge detection**: Identifying boundaries of objects.
- **Transformation**: Rotating, resizing, cropping, or changing color space.
- **Enhancement**: Improving the quality, contrast, or brightness of the image.

Goal
The goal of image processing is to make an image more suitable for human perception or other algorithms to work on. It focuses on altering the image itself.

III Difference Between Image Processing and Computer Vision

Scope: Image processing focuses mainly on the modification or enhancement of images, while computer vision is about understanding the content in images.

End Goal: Image processing aims to make images ready for analysis (like cleaning or extracting features), while computer vision focuses on interpreting and making sense of the visual information, often for decision-making.

IV Which One Comes first and Why?

In the typical workflow of working with images, image processing comes first. Here's why:

- **Preprocessing Step**: In computer vision tasks, images often need to be preprocessed to remove noise, improve clarity, and make them more suitable for analysis. This is where image processing techniques like filtering, resizing, and edge detection are used.
- **Feature Extraction**: Image processing can extract important features from an image (such as edges or color histograms), which are then used in computer vision algorithms for tasks like object recognition or classification.
- **Accuracy**: Image processing helps ensure the quality of the image is high enough for the more complex tasks of computer vision to be successful.

Summary
In summary, image processing prepares the image for further analysis, and computer vision takes the processed image and interprets it to understand its contents.

V Image

An image is a visual representation of information, formed by a grid of pixels, each holding a color or brightness value. Images can be categorized into several types based on their characteristics and representation:

Image Type	Description
Binary Images	Only two colors (e.g., black and white), with each pixel being 0 or 1.
Grayscale Images	Shades of gray from black (0) to white (e.g., 255 in 8-bit).
Color Images	Represented in RGB (Red, Green, Blue) or other color spaces.
Indexed Images	Use a color palette where pixel values correspond to color indices.

Table 1: Types of Images

VI Image Format Conversion

VI.1 RGB to BINARY

The following MATLAB code shows an example of how to convert an RGB image to a binary image :

1. **RGB to Grayscale**: First, the RGB image is converted to grayscale to collapse the three color channels into a single intensity channel. This simplifies the image data, making it easier to process for binary conversion.

2. **Grayscale to Binary**: Once we have a grayscale image, we apply thresholding to decide whether a pixel should be black or white, resulting in a binary image.

   Thresholding: This step is a simple form of image segmentation, where the grayscale image is divided into two classes—pixels that are lighter than the threshold become white (1), and those darker than the threshold become black (0).

```
% Reading the RGB image
rgbImage = imread('image.jpg');  % Replace with your image path

% Converting the RGB image to grayscale
grayImage = rgb2gray(rgbImage);

% Applying a threshold to create a binary image
threshold = 128;  % You can adjust the threshold value as needed (0 to 255)
binaryImage = grayImage > threshold;  % Pixels greater than threshold are white ...
    (1), else black (0)

% Displaying the original image
subplot(1,2,1);
imshow(rgbImage);
title('Original RGB Image');

% Display*ing the grayscale image
subplot(1,2,2);
imshow(binaryImage);
title('Binary Image');

% Saving the binary image
imwrite(binaryImage, 'binary_image.png');  % Save the binary image
```

This two-step process ensures that the conversion from RGB to binary is done in a way that preserves the necessary information (intensity) for segmentation into black and white.

Note
The threshold value can be adjusted depending on the content of the image. A threshold of 128 is a middle value (out of 0 to 255), but in practice, you may need to tune this threshold based on the image's contrast and lighting.

For further assistance with MATLAB functions, you can find a comprehensive MATLAB Basic Functions Cheatsheet in my GitHub repository under the folder **Computer Vision**.