Open CV Assignment 3

**1. Can image processing be used to recognise faces?**

A. Yes. [Image processing](https://www.easytechjunkie.com/what-is-image-processing.htm) face recognition is a computerized technique that uses an [algorithm](https://www.easytechjunkie.com/what-is-an-algorithm.htm) to locate and recognize a face in an image.

**2. Distinguish between image processing and computer vision.**

A.

|  |  |
| --- | --- |
| **Image Processing** | **Computer Vision** |
| Image processing is mainly focused on processing the raw input images to enhance them or preparing them to do other tasks | Computer vision is focused on extracting information from the input images or videos to have a proper understanding of them to predict the visual input like human brain. |
| Image processing uses methods like Anisotropic diffusion, Hidden Markov models, Independent component analysis, Different Filtering etc. | Image processing is one of the methods that is used for computer vision along with other Machine learning techniques, CNN etc. |
| Examples of some Image Processing applications are- Rescaling image (Digital Zoom), Correcting illumination, Changing tones etc. | Examples of some Computer Vision applications are- Object detection, Face detection, Hand writing recognition etc. |

**3. What is the significance of OpenCV in robotics?**

A. OpenCV is used in robotics, medicine, industrial automation, security, and transportation. For robotics, OpenCV can be used to determine a robot’s location. It can also be used in navigation, [Obstacle avoidance](https://en.wikipedia.org/wiki/Obstacle_avoidance), and [Human-Robot Interaction](https://en.wikipedia.org/wiki/Human%E2%80%93robot_interaction).

It can be used for various applications like Pick and place robot using object detection,etc

**4. What is the meaning of Gaussian Blur in OpenCV?**

A. This is an operation in which the image is convolved with a Gaussian filter instead of the box filter. The Gaussian filter is a low-pass filter that removes the high-frequency components are reduced.

We can perform this operation on an image using the Gaussianblur() method of the imgproc class.

The syntax: GaussianBlur(src, dst, ksize, sigmaX)

This method accepts the following parameters −

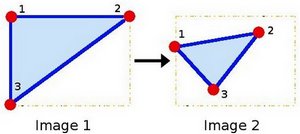
* src − A Mat object representing the source (input image) for this operation.
* dst − A Mat object representing the destination (output image) for this operation.
* ksize − A Size object representing the size of the kernel.
* sigmaX − A variable of the type double representing the Gaussian kernel standard deviation in X direction.

**5. What is Affine Translation, and how does it work?**

1. A. A transformation that can be expressed in the form of a *matrix multiplication* (linear transformation) followed by a *vector addition* (translation).
2. From the above, we can use an Affine Transformation to express:
   1. Rotations (linear transformation)
   2. Translations (vector addition)
   3. Scale operations (linear transformation)

**How do we get an Affine Transformation?**

1. We mentioned that an Affine Transformation is basically a **relation** between two images. The information about this relation can come, roughly, in two ways:
   1. We know both X and T and we also know that they are related. Then our task is to find M
   2. We know M and X. To obtain T we only need to apply T=M⋅X. Our information for M may be explicit (i.e. have the 2-by-3 matrix) or it can come as a geometric relation between points.
2. Since M relates 2 images, we can analyze the simplest case in which it relates three points in both images. Look at the figure below:



the points 1, 2 and 3 (forming a triangle in image 1) are mapped into image 2, still forming a triangle, but now they have changed notoriously. If we find the Affine Transformation with these 3 points (you can choose them as you like), then we can apply this found relation to all the pixels in an image.