Numpy and OpenCV

Dr Frazer Noble

Introduction

In this presentation, I will describe:

- Basic operations using NumPy and OpenCV.

Requirements

To follow along with this tutorial, you will need the following tools:

- Python 3.8.6.
- Visual Studio Code 1.53.1.

You will also need to install the following Python packages:

- OpenCV.
- NumPy.

It is assumed that you are using Windows; however, these instructions should be easily adapted to Linux.

Getting Started

Open Visual Studio Code. To open the app: Open the Start menu, type `Visual Studio Code`, and then select the app.

Open the Explorer tab. To display the tab: Left click `View > Explorer` or press `Ctrl` + `Shift` + `E`. This will display the Explorer tab.

Left click on the `Open Folder` button. This will display the Open Folder prompt. Browse to the following directory:

```
C:/Users/%USER%/Documents
```

Note: Replace `%USER%` with your own username. My username is fknoble; hence, the path is `C:/Users/fknoble/Documents`.

In `C:/Users/%USER%/Documents` create a new folder named `opencv_01`. To create a new folder: Right click in the Explorer tab, left click `New Folder`, and rename it.

In `C:/Users/%USER%/Documents/opencv_01` create a new file named `mat.py`. To create a new file: Right click on `/opencv_01` in the Explorer tab, left click `New File`, and rename it. The file will open automatically.

`/opencv_01` should contain the following files and folders:

```
/opencv_01
    mat.py
```

`mat.py`

Type the following code into `mat.py`:

```
import cv2 as cv
import numpy as np
```

This snippet will import OpenCV's python module as `cv` and NumPy's python module as `np`.

Type the following code into `mat.py`:

```
def main():

    img = np.zeros((480, 640, 1), dtype=np.uint8)

    cv.rectangle(img, (100, 100), (400, 400), 255, -1)

    cv.imshow("img", img)
    cv.waitKey(1)
```

This snippet begins `main()`'s definition; defines an array named `img`; draws a filled rectangle on it; and then displays the array in the `img` window.

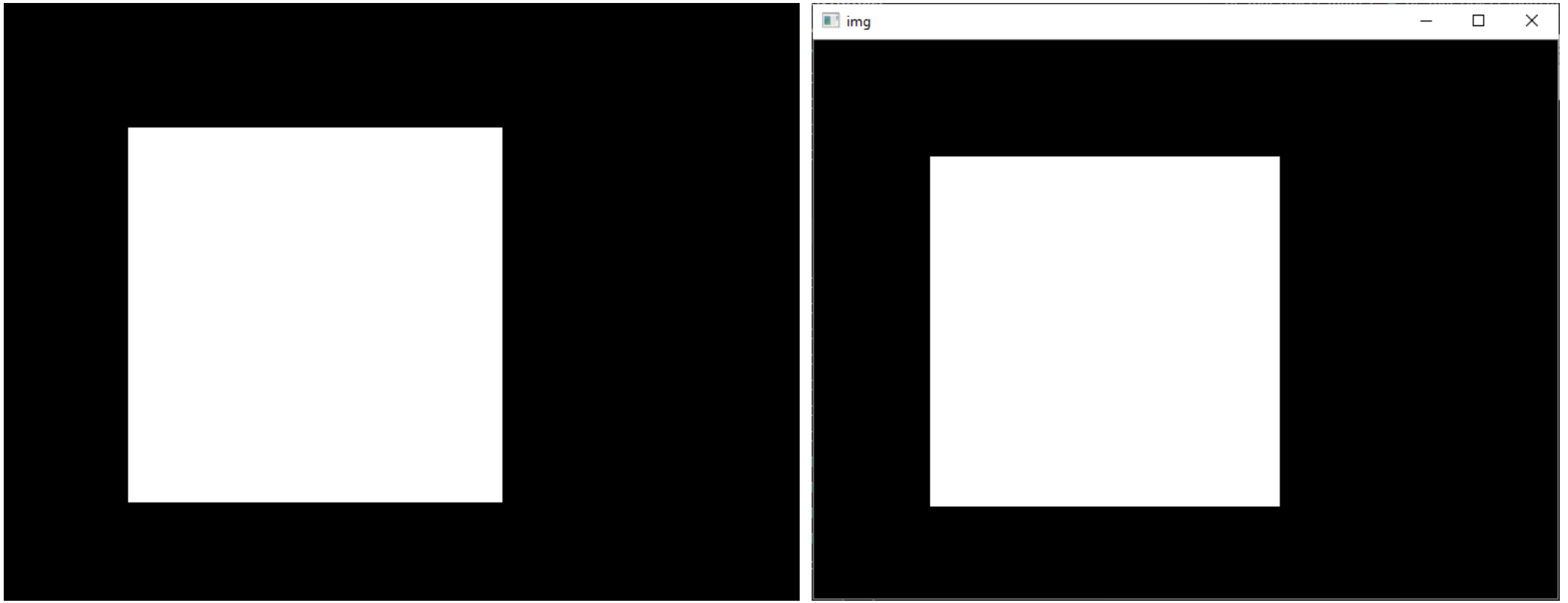

Figure: (Left) The `img` array; and (Right) `img` displayed in the `img` window.

Type the following code into `mat.py`:

```
    background_img_1 = np.zeros((480, 640, 1), dtype=np.uint8)
    background_img_2 = np.zeros((480, 640, 1), dtype=np.uint8)

    cv.rectangle(background_img_1, (150, 150,), (350, 350), 255, -1)
    cv.rectangle(background_img_2, (200, 200,), (450, 450), 255, -1)

    background_img = background_img_1 + background_img_2

    cv.imshow("background_img", background_img)
    cv.waitKey(1)
```

This snippet defines arrays named `background_img_1` and `background_img_2`; draws filled rectangles on them; adds the arrays together; and then displays the result in the `background_img` window.

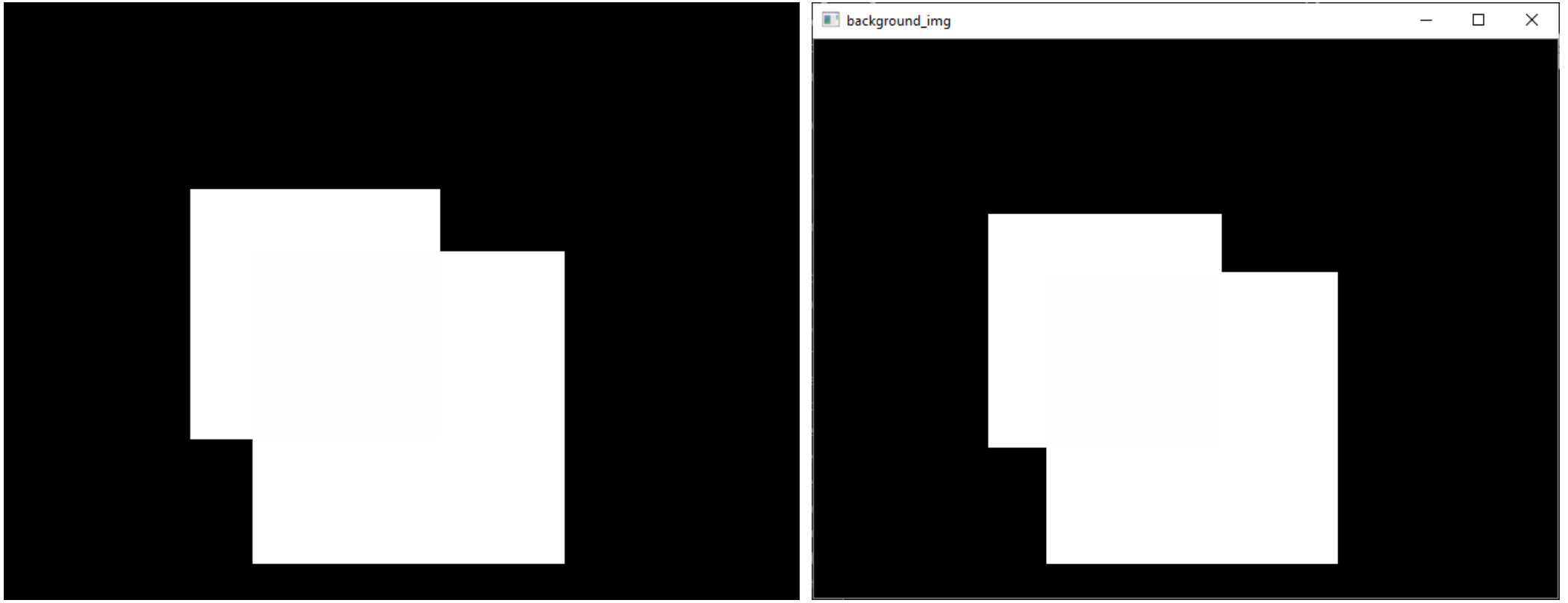

Figure: (Left) The `background_img` array; and (Right) `background_img` displayed in the `background_img` window.

Type the following code into `mat.py`:

```
    sub_img = img - background_img

    cv.imshow("sub_img", sub_img)
    cv.waitKey(0)

    cv.destroyAllWindows()

    return 0
```

This snippet defines an array named `sub_img`, which is assigned the difference between `img` and `background_img`; displays the array in the `sub_img` window; and then waits for user input before destroying all windows.

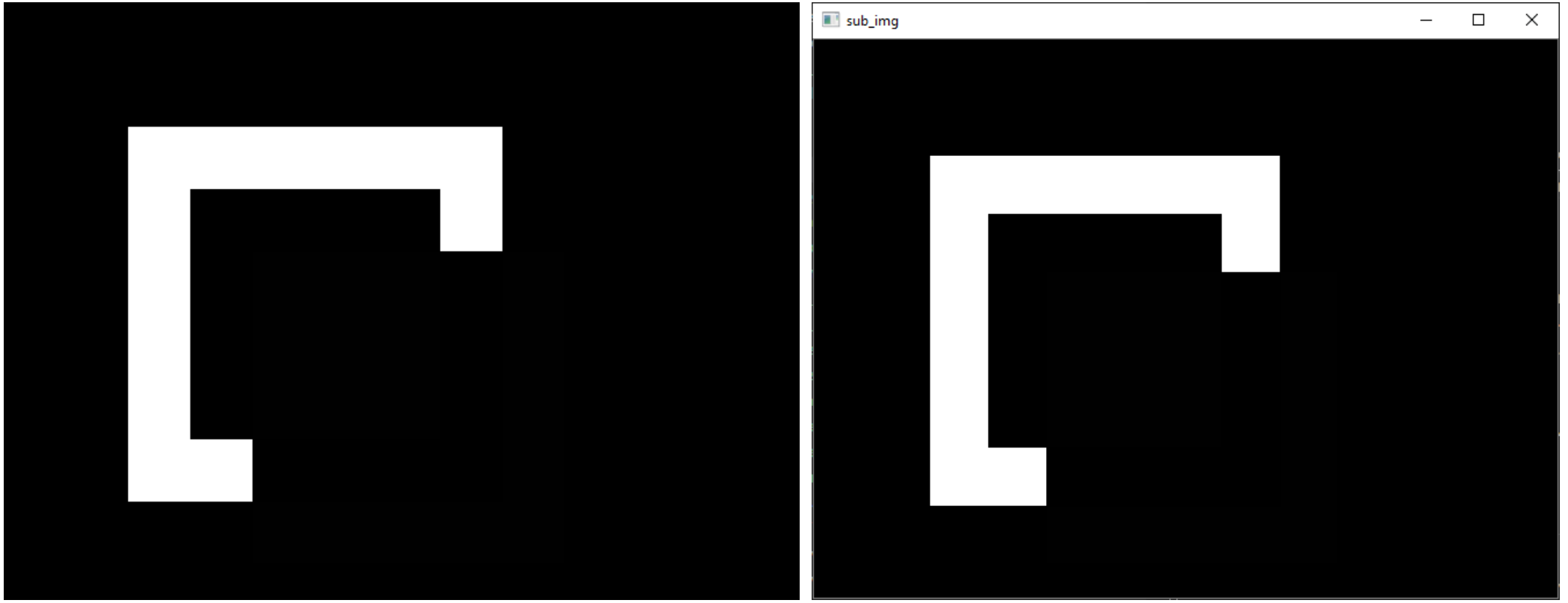

Figure: (Left) The `sub_img` array; and (Right) `sub_img` displayed in the `sub_img` window.

Type the following code into `mat.py`:

```
if __name__ == '__main__':
    main()
```

This snippet will call `main()` when the `mat.py` is run.

Run `mat.py`

Open a new terminal in Visual Studio Code. To open a new terminal: Left click `View > Terminal` or press `Ctrl`+`` ` ``.

Type the following commands into the terminal and then press `Enter` after each one:

```
cd ./opencv_01
python mat.py
```

This will change the current directory to the `/opencv_01` sub-directory and then run `mat.py`.

Press any key to close the windows and stop `mat.py`.

Conclusion

In this presentation, I have described:

- Basic operations using NumPy and OpenCV.

References

1. https://docs.opencv.org/.