

# Week 2 – Mid Evaluation Assignment (OpenCV Fundamentals)

**Theme:** Understanding OpenCV pipelines, not just function calls

**Tools Allowed:** Python, OpenCV (`cv2`), NumPy, Matplotlib

**Submission Format:** Jupyter Notebook (`.ipynb`)

Use `matplotlib.pyplot.imshow()` to display images. Do not use `cv2.imshow()`.

---

## Part A – Core OpenCV Skills

*(Everyone must attempt)*

---

### Q1. Image I/O and Color Spaces

1. Load any **color image of your choice** using OpenCV.
2. Display the image using **Matplotlib** (not `cv2.imshow`).
3. Convert the image into:
  - Grayscale
  - HSV
4. Display all three images side by side.
5. **Answer in 2–3 lines:**
  - Why does OpenCV read images in **BGR** instead of RGB?

**Expected Learning:**

- Color channel awareness

- OpenCV vs Matplotlib mismatch
  - Conceptual clarity, not syntax
- 

## Q2. Histogram Analysis

1. Convert your image to grayscale.
2. Plot the **grayscale histogram** using OpenCV.
3. Apply **Histogram Equalization**.
4. Plot the histogram **before and after** equalization.
5. Display both images side by side.

**Answer briefly:**

- What visible change do you observe after equalization?
  - In what kind of images is histogram equalization *not* useful?
- 

## Q3. Image Smoothing and Noise Reduction

1. Add **artificial noise** to a grayscale image.
2. Apply the following filters:
  - Average Blur
  - Gaussian Blur
  - Median Blur
3. Display all results in one figure.
4. **Answer:**

- Which filter handled noise best?
  - Which filter preserved edges better?
- 

## Q4. Geometric Transformations

1. Resize the image to half its original size.
2. Rotate the image by **45° about its center**.
3. Translate the image by (**x=50, y=30**) pixels.
4. Display all outputs clearly.

### Answer:

Why do geometric transformations often introduce black regions?

---

## Part B – Thresholding & Segmentation

---

### Q5. Manual vs Automatic Thresholding

1. Convert an image to grayscale.
2. Apply:
  - Global thresholding (manual value)
  - Adaptive thresholding
  - Otsu's thresholding
3. Display all outputs side by side.
4. **Answer:**
  - Which method worked best and why?

- 
- When would global thresholding fail badly?

---

## Q6. Thresholding as Segmentation

1. Choose an image where foreground and background are distinguishable.
2. Use thresholding to isolate the main object.
3. Show:
  - Original image
  - Binary image
  - Masked output

**Explain in 3–4 lines how thresholding acts as segmentation.**

---

## Part C – Thinking & Reflection

*(This section decides who actually understood)*

---

### Q7. Conceptual Reflection

Answer:

1. Why is smoothing often applied **before thresholding**?
  2. Why is HSV sometimes better than RGB for segmentation?
  3. Can histogram equalization negatively affect thresholding? Explain.
  4. Why does OpenCV prefer built-in functions over manual pixel loops?
-

## Bonus

- Implement a **small OpenCV pipeline**:
  - Read image → smooth → threshold → geometric transform → display
- Compare the same thresholding technique on **two very different images** and analyze the result.