What is OpenCV?

OpenCV (Open Source Computer Vision Library) is an open-source library designed for real-time computer vision and image processing tasks. It provides a wide range of tools for processing images, videos, and performing tasks such as object detection, facial recognition, edge detection, and more. It is written in C++, but also provides bindings for Python, Java, and MATLAB.

Features of OpenCV

Image Processing – Filtering, thresholding, edge detection, transformations. Object Detection – Face, eyes, pedestrian, and vehicle detection. Feature Detection and Matching – SIFT, SURF, ORB for identifying key points in images. Machine Learning Integration – Built-in ML algorithms for classification and regression. Video Processing – Capturing, tracking objects, motion analysis. Deep Learning Support – Compatible with frameworks like TensorFlow, PyTorch, and Caffe.

In [1]:
```
#pip install opencv-python
```

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
from PIL import Image
```

In [2]:
```
img=Image.open(r"C:\Users\hi\Downloads\aniq edit image.png")
```

In [3]:
```
img
```

Out[3]:

In [4]:
```
type(img)
```

Out[4]:
```
PIL.PngImagePlugin.PngImageFile
```

In [5]:
```
img.mode
```

Out[5]:
```
'RGBA'
```

In [6]:
```
img_arr=np.shape(img)
```

In [7]:
```
img_arr
```

Out[7]:
```
(637, 392, 4)
```

In [ ]:
```

```

In [8]:
```
import cv2

# Load an image
image=cv2.imread(r"C:\Users\hi\Downloads\aniq edit image.png")
```

```
# Display the image
cv2.imshow("Displayed Image", image)

# Wait for a key press and close the window
cv2.waitKey(0)
cv2.destroyAllWindows()
```

OpenCV Basic Operations

In [9]:
```
import cv2
```

In [10]:
```
#read an image
image=cv2.imread(r"C:\Users\hi\Downloads\aniq edit image.png")
```

In [11]:
```
image
```

Out[11]:
```
array([[[222, 220, 220],
        [222, 220, 220],
        [222, 220, 220],
        ...,
        [247, 243, 242],
        [247, 243, 242],
        [246, 242, 241]],

       [[222, 220, 220],
        [222, 220, 220],
        [222, 220, 220],
        ...,
        [247, 243, 242],
        [247, 243, 242],
        [246, 242, 241]],

       [[222, 220, 220],
        [222, 220, 221],
        [222, 220, 221],
        ...,
        [247, 244, 242],
        [247, 244, 242],
        [247, 243, 242]],

       ...,

       [[ 78,  80,  80],
        [ 76,  78,  78],
        [ 78,  80,  80],
        ...,
        [108, 110, 110],
        [111, 113, 112],
        [109, 111, 110]],

       [[ 75,  77,  76],
        [ 73,  76,  75],
        [ 79,  81,  81],
        ...,
        [113, 115, 115],
        [112, 114, 114],
        [101, 103, 103]],

       [[ 76,  78,  78],
        [ 80,  82,  81],
        [ 79,  81,  80],
        ...,
        [107, 109, 109],
        [ 99, 101, 100],
        [111, 113, 113]]], dtype=uint8)
```

In [13]:
```
cv2.waitKey(0)  #wait for key press
```

Out[13]:
```
-1
```

In [14]:
```
cv2.destroyAllWindows() #close all window
```

In [15]:
```
#cunvert image to grayscale
```

In [16]:
```
import cv2

# Read the image
image=cv2.imread(r"C:\Users\hi\Downloads\aniq edit image.png")  # Ensure "sample

if image is None:
    print("Error: Image not found. Check the file path!")
else:
    # Convert image to Grayscale (✅ Corrected)
    gray_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

    # Display Grayscale Image
    cv2.imshow("Grayscale Image", gray_image)
    cv2.waitKey(0)
    cv2.destroyAllWindows()
```

In [17]:
```
#resize the image
resize_image=cv2.resize(image,(300,300))  # resize image 300*300 pixels
cv2.imshow("resized  image ", resize_image)
```

In [18]:
```
# draw a rectangle on image
img_with_rectangle=image.copy()
```

In [19]:
```
cv2.rectangle(img_with_rectangle,(50,50),(200,200),(0,255,0),3) #green rectangle
cv2.imshow("img_with_rectangle", img_with_rectangle)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [20]:
```
#draw a circle
image_with_circle=image.copy()
cv2.circle(image_with_circle,(150,150),50,(255,0,0),-1) #blue field circle
cv2.imshow("image_with_circle",image_with_circle)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [21]:
```
#Add text
image_with_text = image.copy()
cv2.putText(image_with_text, "Hello Anique zzama!", (10, 30), cv2.FONT_HERSHEY_S
cv2.imshow("Text on Image", image_with_text)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [22]:
```
# 8️⃣ Save the Processed Image
cv2.imwrite("output.jpg", image_with_text)  # Save the final image

print("All operations completed successfully!")
```

```
All operations completed successfully!
```

In [23]:
```
image=cv2.imread(r"C:\Users\hi\Downloads\aniq edit image.png")
```

In [24]:
```
#convert  to grayscale
gray_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
cv2.imshow("Grayscale Image", gray_image)
```

In [ ]:
```

```

In [25]:
```
# Step 4: Image Thresholding
_, binary_thresh = cv2.threshold(gray_image, 127, 255, cv2.THRESH_BINARY)
adaptive_thresh = cv2.adaptiveThreshold(gray_image, 255, cv2.ADAPTIVE_THRESH_GAU
cv2.imshow("Binary Threshold", binary_thresh)
cv2.imshow("Adaptive Threshold", adaptive_thresh)
```

In [ ]:
```
```

In [12]:
```
kernel = np.ones((5,5), np.uint8)
eroded = cv2.erode(binary_thresh, kernel, iterations=1)
dilated = cv2.dilate(binary_thresh, kernel, iterations=1)
cv2.imshow("Eroded Image", eroded)
cv2.imshow("Dilated Image", dilated)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[12], line 2
      1 kernel = np.ones((5,5), np.uint8)
----> 2 eroded = cv2.erode(binary_thresh, kernel, iterations=1)
      3 dilated = cv2.dilate(binary_thresh, kernel, iterations=1)
      4 cv2.imshow("Eroded Image", eroded)

NameError: name 'binary_thresh' is not defined
```

Applications of OpenCV

Facial Recognition – Used in security systems and attendance tracking. Autonomous Vehicles – Object detection and lane tracking. Medical Imaging – Used for analyzing X-rays, MRIs, and CT scans. Augmented Reality (AR) – Used in apps to overlay virtual elements on real-world objects. Robotics – Enables robots to see and recognize objects.

In [13]:
```
!pip install opencv-python opencv-contrib-python numpy dlib face-recognition
```

```
Requirement already satisfied: opencv-python in c:\users\hi\anaconda3\lib\site-pa
ckages (4.11.0.86)
Requirement already satisfied: opencv-contrib-python in c:\users\hi\anaconda3\lib
\site-packages (4.11.0.86)
Requirement already satisfied: numpy in c:\users\hi\anaconda3\lib\site-packages
(1.26.4)
Collecting dlib
  Using cached dlib-19.24.6.tar.gz (3.4 MB)
  Preparing metadata (setup.py): started
  Preparing metadata (setup.py): finished with status 'done'
Collecting face-recognition
  Downloading face_recognition-1.3.0-py2.py3-none-any.whl.metadata (21 kB)
Collecting face-recognition-models>=0.3.0 (from face-recognition)
  Downloading face_recognition_models-0.3.0.tar.gz (100.1 MB)
     ---------------------------------------- 0.0/100.1 MB ? eta -:--:--
     ---------------------------------------- 0.3/100.1 MB ? eta -:--:--
      --------------------------------------- 1.3/100.1 MB 5.2 MB/s eta 0:00:20
      --------------------------------------- 2.4/100.1 MB 5.2 MB/s eta 0:00:19
     - -------------------------------------- 3.4/100.1 MB 4.9 MB/s eta 0:00:20
     - -------------------------------------- 4.5/100.1 MB 5.1 MB/s eta 0:00:19
     -- ------------------------------------- 5.8/100.1 MB 5.1 MB/s eta 0:00:19
     -- ------------------------------------- 7.1/100.1 MB 5.3 MB/s eta 0:00:18
     --- ------------------------------------ 8.4/100.1 MB 5.4 MB/s eta 0:00:17
     --- ------------------------------------ 9.7/100.1 MB 5.5 MB/s eta 0:00:17
     ---- ----------------------------------- 10.7/100.1 MB 5.5 MB/s eta 0:00:17
     ---- ----------------------------------- 12.1/100.1 MB 5.6 MB/s eta 0:00:16
     ----- ---------------------------------- 13.4/100.1 MB 5.7 MB/s eta 0:00:16
     ----- ---------------------------------- 14.7/100.1 MB 5.7 MB/s eta 0:00:16
     ------ --------------------------------- 16.0/100.1 MB 5.7 MB/s eta 0:00:15
     ------ --------------------------------- 17.3/100.1 MB 5.8 MB/s eta 0:00:15
     ------- -------------------------------- 18.4/100.1 MB 5.7 MB/s eta 0:00:15
     ------- -------------------------------- 19.9/100.1 MB 5.8 MB/s eta 0:00:14
     -------- ------------------------------- 21.2/100.1 MB 5.8 MB/s eta 0:00:14
     -------- ------------------------------- 22.5/100.1 MB 5.8 MB/s eta 0:00:14
     --------- ------------------------------ 23.9/100.1 MB 5.9 MB/s eta 0:00:13
     --------- ------------------------------ 25.2/100.1 MB 5.9 MB/s eta 0:00:13
     ---------- ----------------------------- 26.5/100.1 MB 5.9 MB/s eta 0:00:13
     ---------- ----------------------------- 27.8/100.1 MB 5.9 MB/s eta 0:00:13
     ----------- ---------------------------- 29.1/100.1 MB 5.9 MB/s eta 0:00:12
     ----------- ---------------------------- 30.1/100.1 MB 5.9 MB/s eta 0:00:12
     ------------ --------------------------- 31.7/100.1 MB 6.0 MB/s eta 0:00:12
     ------------ --------------------------- 33.0/100.1 MB 6.0 MB/s eta 0:00:12
     ------------- -------------------------- 34.3/100.1 MB 6.0 MB/s eta 0:00:11
     ------------- -------------------------- 35.4/100.1 MB 5.9 MB/s eta 0:00:11
     ------------- -------------------------- 35.7/100.1 MB 5.9 MB/s eta 0:00:11
     ------------- -------------------------- 35.9/100.1 MB 5.7 MB/s eta 0:00:12
     -------------- ------------------------- 36.4/100.1 MB 5.5 MB/s eta 0:00:12
     -------------- ------------------------- 36.4/100.1 MB 5.5 MB/s eta 0:00:12
     -------------- ------------------------- 36.7/100.1 MB 5.2 MB/s eta 0:00:13
     -------------- ------------------------- 37.0/100.1 MB 5.2 MB/s eta 0:00:13
     -------------- ------------------------- 37.5/100.1 MB 5.1 MB/s eta 0:00:13
     -------------- ------------------------- 37.7/100.1 MB 5.0 MB/s eta 0:00:13
     -------------- ------------------------- 38.0/100.1 MB 4.9 MB/s eta 0:00:13
     --------------- ------------------------ 38.5/100.1 MB 4.8 MB/s eta 0:00:13
     --------------- ------------------------ 39.3/100.1 MB 4.8 MB/s eta 0:00:13
     --------------- ------------------------ 40.1/100.1 MB 4.8 MB/s eta 0:00:13
     ---------------- ----------------------- 41.2/100.1 MB 4.8 MB/s eta 0:00:13
     ---------------- ----------------------- 42.2/100.1 MB 4.8 MB/s eta 0:00:13
     ---------------- ----------------------- 43.3/100.1 MB 4.8 MB/s eta 0:00:12
     ----------------- ---------------------- 44.3/100.1 MB 4.8 MB/s eta 0:00:12
```

```
     ---------------- ---------------------- 45.6/100.1 MB 4.8 MB/s eta 0:00:12
     ----------------- ---------------------- 46.7/100.1 MB 4.8 MB/s eta 0:00:12
     ----------------- ---------------------- 47.7/100.1 MB 4.8 MB/s eta 0:00:11
     ------------------ --------------------- 48.5/100.1 MB 4.8 MB/s eta 0:00:11
     ------------------ --------------------- 49.5/100.1 MB 4.8 MB/s eta 0:00:11
     ------------------- -------------------- 50.6/100.1 MB 4.8 MB/s eta 0:00:11
     ------------------- -------------------- 51.6/100.1 MB 4.8 MB/s eta 0:00:11
     -------------------- ------------------- 53.0/100.1 MB 4.8 MB/s eta 0:00:10
     -------------------- ------------------- 54.3/100.1 MB 4.9 MB/s eta 0:00:10
     -------------------- ------------------- 55.6/100.1 MB 4.9 MB/s eta 0:00:10
     --------------------- ------------------ 56.6/100.1 MB 4.9 MB/s eta 0:00:09
     --------------------- ------------------ 57.7/100.1 MB 4.9 MB/s eta 0:00:09
     --------------------- ------------------ 58.7/100.1 MB 4.9 MB/s eta 0:00:09
     ---------------------- ----------------- 60.0/100.1 MB 4.9 MB/s eta 0:00:09
     ---------------------- ----------------- 61.1/100.1 MB 4.9 MB/s eta 0:00:08
     ----------------------- ---------------- 62.4/100.1 MB 4.9 MB/s eta 0:00:08
     ----------------------- ---------------- 63.7/100.1 MB 5.0 MB/s eta 0:00:08
     ------------------------ --------------- 65.0/100.1 MB 5.0 MB/s eta 0:00:08
     ------------------------ --------------- 66.1/100.1 MB 5.0 MB/s eta 0:00:07
     ------------------------- -------------- 67.4/100.1 MB 5.0 MB/s eta 0:00:07
     ------------------------- -------------- 68.7/100.1 MB 5.0 MB/s eta 0:00:07
     -------------------------- ------------- 69.7/100.1 MB 5.0 MB/s eta 0:00:07
     -------------------------- ------------- 70.8/100.1 MB 5.0 MB/s eta 0:00:06
     -------------------------- ------------- 71.8/100.1 MB 5.0 MB/s eta 0:00:06
     --------------------------- ------------ 72.6/100.1 MB 5.0 MB/s eta 0:00:06
     --------------------------- ------------ 73.7/100.1 MB 5.0 MB/s eta 0:00:06
     --------------------------- ------------ 74.7/100.1 MB 5.0 MB/s eta 0:00:06
     ---------------------------- ----------- 76.0/100.1 MB 5.0 MB/s eta 0:00:05
     ---------------------------- ----------- 77.3/100.1 MB 5.0 MB/s eta 0:00:05
     ---------------------------- ----------- 78.6/100.1 MB 5.0 MB/s eta 0:00:05
     ----------------------------- ---------- 79.4/100.1 MB 5.0 MB/s eta 0:00:05
     ----------------------------- ---------- 80.7/100.1 MB 5.0 MB/s eta 0:00:04
     ----------------------------- ---------- 82.1/100.1 MB 5.1 MB/s eta 0:00:04
     ------------------------------ --------- 83.4/100.1 MB 5.1 MB/s eta 0:00:04
     ------------------------------ --------- 84.4/100.1 MB 5.1 MB/s eta 0:00:04
     ------------------------------- -------- 85.5/100.1 MB 5.1 MB/s eta 0:00:03
     ------------------------------- -------- 86.5/100.1 MB 5.1 MB/s eta 0:00:03
     ------------------------------- -------- 87.8/100.1 MB 5.1 MB/s eta 0:00:03
     -------------------------------- ------- 88.6/100.1 MB 5.1 MB/s eta 0:00:03
     -------------------------------- ------- 89.4/100.1 MB 5.1 MB/s eta 0:00:03
     -------------------------------- ------- 89.9/100.1 MB 5.0 MB/s eta 0:00:03
     --------------------------------- ------ 90.7/100.1 MB 5.0 MB/s eta 0:00:02
     --------------------------------- ------ 91.5/100.1 MB 5.0 MB/s eta 0:00:02
     --------------------------------- ------ 92.3/100.1 MB 5.0 MB/s eta 0:00:02
     ---------------------------------- ----- 93.3/100.1 MB 5.0 MB/s eta 0:00:02
     ---------------------------------- ----- 94.4/100.1 MB 5.0 MB/s eta 0:00:02
     ---------------------------------- ----- 95.7/100.1 MB 5.0 MB/s eta 0:00:01
     ---------------------------------- ----- 96.7/100.1 MB 5.0 MB/s eta 0:00:01
     ----------------------------------- ---- 98.0/100.1 MB 5.0 MB/s eta 0:00:01
     ----------------------------------- ---- 99.4/100.1 MB 5.0 MB/s eta 0:00:01
     ----------------------------------- ---- 100.1/100.1 MB 5.0 MB/s eta 0:00:01
     ----------------------------------- ---- 100.1/100.1 MB 5.0 MB/s eta 0:00:00
  Preparing metadata (setup.py): started
  Preparing metadata (setup.py): finished with status 'done'
Requirement already satisfied: Click>=6.0 in c:\users\hi\anaconda3\lib\site-packa
ges (from face-recognition) (8.1.7)
Requirement already satisfied: Pillow in c:\users\hi\anaconda3\lib\site-packages
(from face-recognition) (10.4.0)
Requirement already satisfied: colorama in c:\users\hi\anaconda3\lib\site-package
s (from Click>=6.0->face-recognition) (0.4.6)
```

```
Downloading face_recognition-1.3.0-py2.py3-none-any.whl (15 kB)
Building wheels for collected packages: dlib, face-recognition-models
  Building wheel for dlib (setup.py): started
  Building wheel for dlib (setup.py): finished with status 'error'
  Running setup.py clean for dlib
  Building wheel for face-recognition-models (setup.py): started
  Building wheel for face-recognition-models (setup.py): finished with status 'do
ne'
  Created wheel for face-recognition-models: filename=face_recognition_models-0.
3.0-py2.py3-none-any.whl size=100566178 sha256=f03c997bd0e766f36bc06a8e683888c183
74a471f0c03ea52ffa126b7fb9cc35
  Stored in directory: c:\users\hi\appdata\local\pip\cache\wheels\8f\47\c8\f44c5a
ebb7507f7c8a2c0bd23151d732d0f0bd6884ad4ac635
Successfully built face-recognition-models
Failed to build dlib
```

```
  error: subprocess-exited-with-error

  python setup.py bdist_wheel did not run successfully.
  exit code: 1

  [47 lines of output]
  C:\Users\hi\AppData\Local\Temp\pip-install-xqxioikr\dlib_75cba4d93e0346f28b1ea1
f0f52cc761\setup.py:234: SyntaxWarning: invalid escape sequence '\('
    major = re.findall("set\(CPACK_PACKAGE_VERSION_MAJOR.*\"(.*)\"", open(cmake_f
ile).read())[0]
  C:\Users\hi\AppData\Local\Temp\pip-install-xqxioikr\dlib_75cba4d93e0346f28b1ea1
f0f52cc761\setup.py:235: SyntaxWarning: invalid escape sequence '\('
    minor = re.findall("set\(CPACK_PACKAGE_VERSION_MINOR.*\"(.*)\"", open(cmake_f
ile).read())[0]
  C:\Users\hi\AppData\Local\Temp\pip-install-xqxioikr\dlib_75cba4d93e0346f28b1ea1
f0f52cc761\setup.py:236: SyntaxWarning: invalid escape sequence '\('
    patch = re.findall("set\(CPACK_PACKAGE_VERSION_PATCH.*\"(.*)\"", open(cmake_f
ile).read())[0]
  running bdist_wheel
  running build
  running build_ext

  ================================================================================
=
  ================================================================================
=
  ================================================================================
=

                    CMake is not installed on your system!

      Or it is possible some broken copy of cmake is installed on your system.
      It is unfortunately very common for python package managers to include
      broken copies of cmake.  So if the error above this refers to some file
      path to a cmake file inside a python or anaconda or miniconda path then you
      should delete that broken copy of cmake from your computer.

      Instead, please get an official copy of cmake from one of these known good
      sources of an official cmake:
          - cmake.org (this is how windows users should get cmake)
          - apt install cmake (for Ubuntu or Debian based systems)
          - yum install cmake (for Redhat or CenOS based systems)

      On a linux machine you can run `which cmake` to see what cmake you are
      actually using.  If it tells you it's some cmake from any kind of python
      packager delete it and install an official cmake.

      More generally, cmake is not installed if when you open a terminal window
      and type
         cmake --version
      you get an error.  So you can use that as a very basic test to see if you
      have cmake installed.  That is, if cmake --version doesn't run from the
      same terminal window from which you are reading this error message, then
      you have not installed cmake.  Windows users should take note that they
      need to tell the cmake installer to add cmake to their PATH.  Since you
      can't run commands that are not in your PATH.  This is how the PATH works
      on Linux as well, but failing to add cmake to the PATH is a particularly
      common problem on windows and rarely a problem on Linux.

  ================================================================================
```

```
=
  ============================================================================
=
  ============================================================================
=
  [end of output]

  note: This error originates from a subprocess, and is likely not a problem with
pip.
  ERROR: Failed building wheel for dlib
ERROR: ERROR: Failed to build installable wheels for some pyproject.toml based pr
ojects (dlib)
```

In [16]:
```python
import cv2

# Load the Haar Cascade for face detection
face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade_fronta

# Capture video from webcam
cap = cv2.VideoCapture(0)

while True:
    ret, frame = cap.read()
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)  # Convert to grayscale

    # Detect faces
    faces = face_cascade.detectMultiScale(gray, 1.3, 5)

    # Draw rectangles around detected faces
    for (x, y, w, h) in faces:
        cv2.rectangle(frame, (x, y), (x + w, y + h), (255, 0, 0), 2)

    cv2.imshow('Face Detection', frame)

    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```

In [18]:
```
!pip install face-recognition dlib opencv-python numpy
```

```
Collecting face-recognition
  Using cached face_recognition-1.3.0-py2.py3-none-any.whl.metadata (21 kB)
Collecting dlib
  Using cached dlib-19.24.6.tar.gz (3.4 MB)
  Preparing metadata (setup.py): started
  Preparing metadata (setup.py): finished with status 'done'
Requirement already satisfied: opencv-python in c:\users\hi\anaconda3\lib\site-pa
ckages (4.11.0.86)
Requirement already satisfied: numpy in c:\users\hi\anaconda3\lib\site-packages
(1.26.4)
Collecting face-recognition-models>=0.3.0 (from face-recognition)
  Using cached face_recognition_models-0.3.0-py2.py3-none-any.whl
Requirement already satisfied: Click>=6.0 in c:\users\hi\anaconda3\lib\site-packa
ges (from face-recognition) (8.1.7)
Requirement already satisfied: Pillow in c:\users\hi\anaconda3\lib\site-packages
(from face-recognition) (10.4.0)
Requirement already satisfied: colorama in c:\users\hi\anaconda3\lib\site-package
s (from Click>=6.0->face-recognition) (0.4.6)
Using cached face_recognition-1.3.0-py2.py3-none-any.whl (15 kB)
Building wheels for collected packages: dlib
  Building wheel for dlib (setup.py): started
  Building wheel for dlib (setup.py): finished with status 'error'
  Running setup.py clean for dlib
Failed to build dlib
```

```
  error: subprocess-exited-with-error

  python setup.py bdist_wheel did not run successfully.
  exit code: 1

  [47 lines of output]
  C:\Users\hi\AppData\Local\Temp\pip-install-c5wg1__y\dlib_abfbf85e6e7b46a09495a9
d8e3cce8c4\setup.py:234: SyntaxWarning: invalid escape sequence '\('
    major = re.findall("set\(CPACK_PACKAGE_VERSION_MAJOR.*\"(.*)\"", open(cmake_f
ile).read())[0]
  C:\Users\hi\AppData\Local\Temp\pip-install-c5wg1__y\dlib_abfbf85e6e7b46a09495a9
d8e3cce8c4\setup.py:235: SyntaxWarning: invalid escape sequence '\('
    minor = re.findall("set\(CPACK_PACKAGE_VERSION_MINOR.*\"(.*)\"", open(cmake_f
ile).read())[0]
  C:\Users\hi\AppData\Local\Temp\pip-install-c5wg1__y\dlib_abfbf85e6e7b46a09495a9
d8e3cce8c4\setup.py:236: SyntaxWarning: invalid escape sequence '\('
    patch = re.findall("set\(CPACK_PACKAGE_VERSION_PATCH.*\"(.*)\"", open(cmake_f
ile).read())[0]
  running bdist_wheel
  running build
  running build_ext

  ================================================================================
=
  ================================================================================
=
  ================================================================================
=

                    CMake is not installed on your system!

      Or it is possible some broken copy of cmake is installed on your system.
      It is unfortunately very common for python package managers to include
      broken copies of cmake.  So if the error above this refers to some file
      path to a cmake file inside a python or anaconda or miniconda path then you
      should delete that broken copy of cmake from your computer.

      Instead, please get an official copy of cmake from one of these known good
      sources of an official cmake:
          - cmake.org (this is how windows users should get cmake)
          - apt install cmake (for Ubuntu or Debian based systems)
          - yum install cmake (for Redhat or CenOS based systems)

      On a linux machine you can run `which cmake` to see what cmake you are
      actually using.  If it tells you it's some cmake from any kind of python
      packager delete it and install an official cmake.

      More generally, cmake is not installed if when you open a terminal window
      and type
         cmake --version
      you get an error.  So you can use that as a very basic test to see if you
      have cmake installed.  That is, if cmake --version doesn't run from the
      same terminal window from which you are reading this error message, then
      you have not installed cmake.  Windows users should take note that they
      need to tell the cmake installer to add cmake to their PATH.  Since you
      can't run commands that are not in your PATH.  This is how the PATH works
      on Linux as well, but failing to add cmake to the PATH is a particularly
      common problem on windows and rarely a problem on Linux.

  ================================================================================
```

```
=
  ============================================================================
=
  ============================================================================
=
  [end of output]

  note: This error originates from a subprocess, and is likely not a problem with
pip.
  ERROR: Failed building wheel for dlib
ERROR: ERROR: Failed to build installable wheels for some pyproject.toml based pr
ojects (dlib)
```

In [ ]:
```
#!conda install -c conda-forge face-recognition
```

Small Computer Vision Projects for Beginners

Basic Image Processing

In [2]:
```
import cv2
image=cv2.imread(r"C:\Users\hi\Downloads\aniq edit image.png")
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)  # Convert to grayscale
blurred=cv2.GaussianBlur(image,(15,15),0)  #Apply blur
```

In [5]:
```
cv2.imshow("original image",image)
```

In [6]:
```
cv2.imshow("grayscale",gray)
```

In [5]:
```
cv2.imshow("Blured",blurred)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Edge Detection using Canny

In [6]:
```
image=cv2.imread(r"C:\Users\hi\Downloads\aniq edit image.png",0) #Load the grays
```

In [7]:
```
edges=cv2.Canny(image,100,200) #apply canany edge detection
```

In [8]:
```
cv2.imshow("Edges",edges)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Face Detection using OpenCV

In [13]:
```
face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade_fronta
```

In [14]:
```
image=cv2.imread(r"C:\Users\hi\Downloads\aniq edit image.png",0)
```

In [15]:
```
gray=cv2.cvtColor(image,cv2.COLOR_BGR2GRAY)
face=face_cascade.detectMultiScale(gray,1.1,4)
```

```
---------------------------------------------------------------------------
error                                     Traceback (most recent call last)
Cell In[15], line 1
----> 1 gray=cv2.cvtColor(image,cv2.COLOR_BGR2GRAY)
      2 face=face_cascade.detectMultiScale(gray,1.1,4)

error: OpenCV(4.11.0) d:\a\opencv-python\opencv-python\opencv\modules\imgproc\src
\color.simd_helpers.hpp:92: error: (-15:Bad number of channels) in function '__cd
ecl cv::impl::`anonymous-namespace'::CvtHelper<struct cv::impl::`anonymous namesp
ace'::Set<3,4,-1>,struct cv::impl::A0x34dd5b3e::Set<1,-1,-1>,struct cv::impl::A0x
34dd5b3e::Set<0,2,5>,4>::CvtHelper(const class cv::_InputArray &,const class cv::
_OutputArray &,int)'
> Invalid number of channels in input image:
>     'VScn::contains(scn)'
> where
>     'scn' is 1
```

In [16]:
```
for (x, y, w, h) in faces:
    cv2.rectangle(image, (x, y), (x + w, y + h), (255, 0, 0), 2)

cv2.imshow('Face Detection', image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[16], line 1
----> 1 for (x, y, w, h) in faces:
      2     cv2.rectangle(image, (x, y), (x + w, y + h), (255, 0, 0), 2)
      4 cv2.imshow('Face Detection', image)

NameError: name 'faces' is not defined
```

Hand Tracking using OpenCV & Mediapipe

In [19]:
```python
import cv2
import mediapipe as mp

mp_hands = mp.solutions.hands
hands = mp_hands.Hands()
mp_draw = mp.solutions.drawing_utils

cap = cv2.VideoCapture(0)

while cap.isOpened():
    ret, frame = cap.read()
    rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    results = hands.process(rgb)

    if results.multi_hand_landmarks:
        for hand in results.multi_hand_landmarks:
            mp_draw.draw_landmarks(frame, hand, mp_hands.HAND_CONNECTIONS)

    cv2.imshow('Hand Tracking', frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```

In [18]:
```
!pip install mediapipe
```

```
Collecting mediapipe
  Downloading mediapipe-0.10.21-cp312-cp312-win_amd64.whl.metadata (10 kB)
Requirement already satisfied: absl-py in c:\users\hi\anaconda3\lib\site-packages
(from mediapipe) (2.1.0)
Requirement already satisfied: attrs>=19.1.0 in c:\users\hi\anaconda3\lib\site-pa
ckages (from mediapipe) (23.1.0)
Requirement already satisfied: flatbuffers>=2.0 in c:\users\hi\anaconda3\lib\site
-packages (from mediapipe) (24.3.25)
Requirement already satisfied: jax in c:\users\hi\anaconda3\lib\site-packages (fr
om mediapipe) (0.4.38)
Requirement already satisfied: jaxlib in c:\users\hi\anaconda3\lib\site-packages
(from mediapipe) (0.4.38)
Requirement already satisfied: matplotlib in c:\users\hi\anaconda3\lib\site-packa
ges (from mediapipe) (3.10.0)
Requirement already satisfied: numpy<2 in c:\users\hi\anaconda3\lib\site-packages
(from mediapipe) (1.26.4)
Requirement already satisfied: opencv-contrib-python in c:\users\hi\anaconda3\lib
\site-packages (from mediapipe) (4.11.0.86)
Collecting protobuf<5,>=4.25.3 (from mediapipe)
  Downloading protobuf-4.25.6-cp310-abi3-win_amd64.whl.metadata (541 bytes)
Collecting sounddevice>=0.4.4 (from mediapipe)
  Using cached sounddevice-0.5.1-py3-none-win_amd64.whl.metadata (1.4 kB)
Requirement already satisfied: sentencepiece in c:\users\hi\anaconda3\lib\site-pa
ckages (from mediapipe) (0.2.0)
Requirement already satisfied: CFFI>=1.0 in c:\users\hi\anaconda3\lib\site-packag
es (from sounddevice>=0.4.4->mediapipe) (1.17.1)
Requirement already satisfied: ml_dtypes>=0.4.0 in c:\users\hi\anaconda3\lib\site
-packages (from jax->mediapipe) (0.4.1)
Requirement already satisfied: opt_einsum in c:\users\hi\anaconda3\lib\site-packa
ges (from jax->mediapipe) (3.4.0)
Requirement already satisfied: scipy>=1.10 in c:\users\hi\anaconda3\lib\site-pack
ages (from jax->mediapipe) (1.14.1)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\hi\anaconda3\lib\site
-packages (from matplotlib->mediapipe) (1.2.0)
Requirement already satisfied: cycler>=0.10 in c:\users\hi\anaconda3\lib\site-pac
kages (from matplotlib->mediapipe) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\hi\anaconda3\lib\sit
e-packages (from matplotlib->mediapipe) (4.51.0)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\users\hi\anaconda3\lib\sit
e-packages (from matplotlib->mediapipe) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\users\hi\anaconda3\lib\site-
packages (from matplotlib->mediapipe) (24.1)
Requirement already satisfied: pillow>=8 in c:\users\hi\anaconda3\lib\site-packag
es (from matplotlib->mediapipe) (10.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\hi\anaconda3\lib\site
-packages (from matplotlib->mediapipe) (3.1.2)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\hi\anaconda3\lib
\site-packages (from matplotlib->mediapipe) (2.9.0.post0)
Requirement already satisfied: pycparser in c:\users\hi\anaconda3\lib\site-packag
es (from CFFI>=1.0->sounddevice>=0.4.4->mediapipe) (2.21)
Requirement already satisfied: six>=1.5 in c:\users\hi\anaconda3\lib\site-package
s (from python-dateutil>=2.7->matplotlib->mediapipe) (1.16.0)
Downloading mediapipe-0.10.21-cp312-cp312-win_amd64.whl (51.0 MB)
   ---------------------------------------- 0.0/51.0 MB ? eta -:--:--
   ---------------------------------------- 0.3/51.0 MB ? eta -:--:--
    --------------------------------------- 0.8/51.0 MB 2.8 MB/s eta 0:00:18
   - -------------------------------------- 2.1/51.0 MB 4.5 MB/s eta 0:00:11
   -- ------------------------------------- 3.4/51.0 MB 4.8 MB/s eta 0:00:10
   --- ------------------------------------ 4.7/51.0 MB 5.2 MB/s eta 0:00:09
   ---- ----------------------------------- 5.5/51.0 MB 5.4 MB/s eta 0:00:09
```

```
   ----- ---------------------------------- 6.8/51.0 MB 5.3 MB/s eta 0:00:09
   ------ ---------------------------------- 8.1/51.0 MB 5.3 MB/s eta 0:00:09
   ------- ---------------------------------- 9.2/51.0 MB 5.4 MB/s eta 0:00:08
   -------- ---------------------------------- 10.5/51.0 MB 5.4 MB/s eta 0:00:08
   --------- ---------------------------------- 11.8/51.0 MB 5.5 MB/s eta 0:00:08
   ---------- ---------------------------------- 13.1/51.0 MB 5.6 MB/s eta 0:00:07
   ---------- ---------------------------------- 13.9/51.0 MB 5.5 MB/s eta 0:00:07
   ----------- ---------------------------------- 15.2/51.0 MB 5.4 MB/s eta 0:00:07
   ------------ ---------------------------------- 15.7/51.0 MB 5.4 MB/s eta 0:00:07
   ------------ ---------------------------------- 16.5/51.0 MB 5.3 MB/s eta 0:00:07
   ------------- ---------------------------------- 17.6/51.0 MB 5.2 MB/s eta 0:00:07
   -------------- ---------------------------------- 18.9/51.0 MB 5.2 MB/s eta 0:00:07
   --------------- ---------------------------------- 20.2/51.0 MB 5.2 MB/s eta 0:00:06
   ---------------- ---------------------------------- 21.5/51.0 MB 5.3 MB/s eta 0:00:06
   ---------------- ---------------------------------- 22.8/51.0 MB 5.4 MB/s eta 0:00:06
   ----------------- ---------------------------------- 24.1/51.0 MB 5.4 MB/s eta 0:00:05
   ------------------ ---------------------------------- 25.2/51.0 MB 5.4 MB/s eta 0:00:05
   ------------------- ---------------------------------- 26.5/51.0 MB 5.4 MB/s eta 0:00:05
   -------------------- ---------------------------------- 27.3/51.0 MB 5.4 MB/s eta 0:00:05
   -------------------- ---------------------------------- 28.6/51.0 MB 5.4 MB/s eta 0:00:05
   --------------------- ---------------------------------- 29.9/51.0 MB 5.4 MB/s eta 0:00:04
   ---------------------- ---------------------------------- 30.7/51.0 MB 5.3 MB/s eta 0:00:04
   ---------------------- ---------------------------------- 31.7/51.0 MB 5.3 MB/s eta 0:00:04
   ----------------------- ---------------------------------- 33.0/51.0 MB 5.4 MB/s eta 0:00:04
   ------------------------ ---------------------------------- 34.3/51.0 MB 5.4 MB/s eta 0:00:04
   ------------------------- ---------------------------------- 35.9/51.0 MB 5.5 MB/s eta 0:00:03
   ------------------------- ---------------------------------- 37.2/51.0 MB 5.5 MB/s eta 0:00:03
   -------------------------- ---------------------------------- 38.3/51.0 MB 5.5 MB/s eta 0:00:03
   -------------------------- ---------------------------------- 39.6/51.0 MB 5.5 MB/s eta 0:00:03
   --------------------------- ---------------------------------- 40.6/51.0 MB 5.5 MB/s eta 0:00:02
   --------------------------- ---------------------------------- 41.9/51.0 MB 5.5 MB/s eta 0:00:02
   ---------------------------- ---------------------------------- 43.3/51.0 MB 5.5 MB/s eta 0:00:02
   ---------------------------- ---------------------------------- 44.3/51.0 MB 5.5 MB/s eta 0:00:02
   ----------------------------- ---------------------------------- 45.1/51.0 MB 5.5 MB/s eta 0:00:02
   ----------------------------- ---------------------------------- 45.9/51.0 MB 5.4 MB/s eta 0:00:01
   ----------------------------- ---------------------------------- 46.1/51.0 MB 5.3 MB/s eta 0:00:01
   ----------------------------- ---------------------------------- 46.7/51.0 MB 5.3 MB/s eta 0:00:01
   ----------------------------- ---------------------------------- 46.9/51.0 MB 5.2 MB/s eta 0:00:01
   ------------------------------ ---------------------------------- 47.7/51.0 MB 5.1 MB/s eta 0:00:01
   ------------------------------ ---------------------------------- 48.5/51.0 MB 5.1 MB/s eta 0:00:01
   ------------------------------ ---------------------------------- 49.5/51.0 MB 5.1 MB/s eta 0:00:01
   ------------------------------  ---------------------------------- 50.9/51.0 MB 5.1 MB/s eta 0:00:01
   ------------------------------  ---------------------------------- 50.9/51.0 MB 5.1 MB/s eta 0:00:01
   ------------------------------ ---------------------------------- 51.0/51.0 MB 5.0 MB/s eta 0:00:00
Downloading protobuf-4.25.6-cp310-abi3-win_amd64.whl (413 kB)
Using cached sounddevice-0.5.1-py3-none-win_amd64.whl (363 kB)
Installing collected packages: protobuf, sounddevice, mediapipe
  Attempting uninstall: protobuf
    Found existing installation: protobuf 5.29.2
    Uninstalling protobuf-5.29.2:
      Successfully uninstalled protobuf-5.29.2
Successfully installed mediapipe-0.10.21 protobuf-4.25.6 sounddevice-0.5.1
ERROR: pip's dependency resolver does not currently take into account all the pac
kages that are installed. This behaviour is the source of the following dependenc
y conflicts.
grpcio-status 1.68.1 requires protobuf<6.0dev,>=5.26.1, but you have protobuf 4.2
5.6 which is incompatible.
```

Color Detection & Tracking

In [31]:
```
import cv2
import numpy as np

cap = cv2.VideoCapture(0)

while True:
    _, frame = cap.read()
    hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)

    lower_red = np.array([0, 120, 70])
    upper_red = np.array([10, 255, 255])

    mask = cv2.inRange(hsv, lower_red, upper_red)
    result = cv2.bitwise_and(frame, frame, mask=mask)

    cv2.imshow("Original", frame)
    cv2.imshow("Mask", mask)
    cv2.imshow("Detected Color", result)

    if cv2.waitKey(1) & 0xFF == ord("q"):
        break

cap.release()
cv2.destroyAllWindows()
```

Object Counting in an Image

In [30]:
```
import cv2

# Load the image in grayscale mode
image = cv2.imread(r"C:\Users\hi\Downloads\aniq edit image.png", 0)

# Apply thresholding
_, threshold = cv2.threshold(image, 127, 255, cv2.THRESH_BINARY)

# Find contours
contours, _ = cv2.findContours(threshold, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE

# Draw contours on the original image
contour_image = cv2.cvtColor(image, cv2.COLOR_GRAY2BGR)  # Convert back to BGR f
cv2.drawContours(contour_image, contours, -1, (0, 255, 0), 2)

# Display images
cv2.imshow("Original Image", image)
cv2.imshow("Threshold Image", threshold)
cv2.imshow("Contours", contour_image)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

Cartoonify an Image

In [25]:
```
import cv2
```

```
# Load image in grayscale mode
image = cv2.imread(r"C:\Users\hi\Downloads\aniq edit image.png", 0)  # Already g

# Apply Canny edge detection
edges = cv2.Canny(image, 100, 200)

# Load original image in BGR mode for cartoon effect
image_bgr = cv2.imread(r"C:\Users\hi\Downloads\aniq edit image.png")  # Load nor

# Apply cartoon effect
cartoon = cv2.stylization(image_bgr, sigma_s=150, sigma_r=0.25)

# Display results
cv2.imshow("Grayscale Image", image)
cv2.imshow("Edge Detection", edges)
cv2.imshow("Cartoon Effect", cartoon)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [29]:

```
import cv2

face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade_fronta
smile_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade_smile

cap = cv2.VideoCapture(0)

while cap.isOpened():
    ret, frame = cap.read()
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

    faces = face_cascade.detectMultiScale(gray, 1.1, 4)
    for (x, y, w, h) in faces:
        roi_gray = gray[y:y+h, x:x+w]
        smiles = smile_cascade.detectMultiScale(roi_gray, 1.8, 20)

        if len(smiles) > 0:
            cv2.putText(frame, 'Smiling!', (x, y-10), cv2.FONT_HERSHEY_SIMPLEX,

    cv2.imshow('Smile Detector', frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```

In [27]:

```
import cv2

cap = cv2.VideoCapture(0)
_, frame1 = cap.read()
_, frame2 = cap.read()

while cap.isOpened():
    diff = cv2.absdiff(frame1, frame2)
    gray = cv2.cvtColor(diff, cv2.COLOR_BGR2GRAY)
    blur = cv2.GaussianBlur(gray, (5, 5), 0)
    _, thresh = cv2.threshold(blur, 20, 255, cv2.THRESH_BINARY)

    cv2.imshow("Motion Detection", thresh)
```

```
    frame1 = frame2
    _, frame2 = cap.read()

    if cv2.waitKey(10) == ord("q"):
        break

cap.release()
cv2.destroyAllWindows()
```

In [ ]:

In [ ]: