**Source Code:**

import cv2

import mediapipe as mp

import pyautogui

screen\_w, screen\_h = pyautogui.size()

cam = cv2.VideoCapture(0)

face\_mesh = mp.solutions.face\_mesh.FaceMesh(refine\_landmarks=True)

def find\_landmarks\_and\_click(landmarks, frame\_w, frame\_h):

for id, landmark in enumerate(landmarks[474:478]):

x = int(landmark.x \* frame\_w)

y = int(landmark.y \* frame\_h)

cv2.circle(frame, (x, y), 3, (0, 255, 0))

if id == 1:

screen\_x = screen\_w \* landmark.x

screen\_y = screen\_h \* landmark.y

pyautogui.moveTo(screen\_x, screen\_y)

left = [landmarks[145], landmarks[159]]

for landmark in left:

x = int(landmark.x \* frame\_w)

y = int(landmark.y \* frame\_h)

cv2.circle(frame, (x, y), 3, (0, 255, 255))

if (left[0].y - left[1].y) < 0.004:

pyautogui.click()

pyautogui.sleep(1)

while True:

ret, frame = cam.read()

if not ret:

break

frame = cv2.flip(frame, 1)

rgb\_frame = cv2.cvtColor(frame, cv2.COLOR\_BGR2RGB)

output = face\_mesh.process(rgb\_frame)

landmark\_points = output.multi\_face\_landmarks

if landmark\_points:

landmarks = landmark\_points[0].landmark

frame\_h, frame\_w, \_ = frame.shape

find\_landmarks\_and\_click(landmarks, frame\_w, frame\_h)

cv2.imshow('Eye Controlled Mouse', frame)

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

break

cam.release()

cv2.destroyAllWindows()