Nguyễn Hoàng Thanh – 2151013087

Code:

```python
import cv2
import numpy as np

# Step 1
roi = cv2.imread("LAB_2/data/pepsi.png")
hsvr = cv2.cvtColor(roi, cv2.COLOR_BGR2HSV)
# hsv_frame_roi = hsvr[200:500, 300:700]  # vùng quan tâm
M = cv2.calcHist([hsvr], [0, 1], None, [180, 256], [0,
180, 0, 256])
cap = cv2.VideoCapture("LAB_2/data/pepsi.mp4")

if not cap.isOpened():
    print("Không thể mở video")
    exit()

while True:
    ret, frame = cap.read()

    if not ret:
        break

    hsvt = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
    I = cv2.calcHist([hsvt], [0, 1], None, [180, 256],
[0, 180, 0, 256])

    # Step 2
    R = M / (I + 1)

    # Step 3: backprojeck
    h, s, v = cv2.split(hsvt)
    B = R[h.ravel(), s.ravel()]
    B = np.minimum(B, 1)
    B = B.reshape(hsvt.shape[:2])
```

```python
    # Step 4: chuẩn hóa
    disc = cv2.getStructuringElement(cv2.MORPH_ELLIPSE,
(5, 5))
    cv2.filter2D(B, -1, disc, B)
    B = np.uint8(B)
    cv2.normalize(B, B, 0, 255, cv2.NORM_MINMAX)  # lệnh
chuẩn hóa

    # Step 5
    ret, thresh = cv2.threshold(B, 15, 255, 0)
    thresh = cv2.merge((thresh, thresh, thresh))
    res = cv2.bitwise_and(frame, thresh)

    # Hiển thị kết quả
    cv2.imshow("Result", res)
    if cv2.waitKey(25) & 0xFF == ord("q"):
        break

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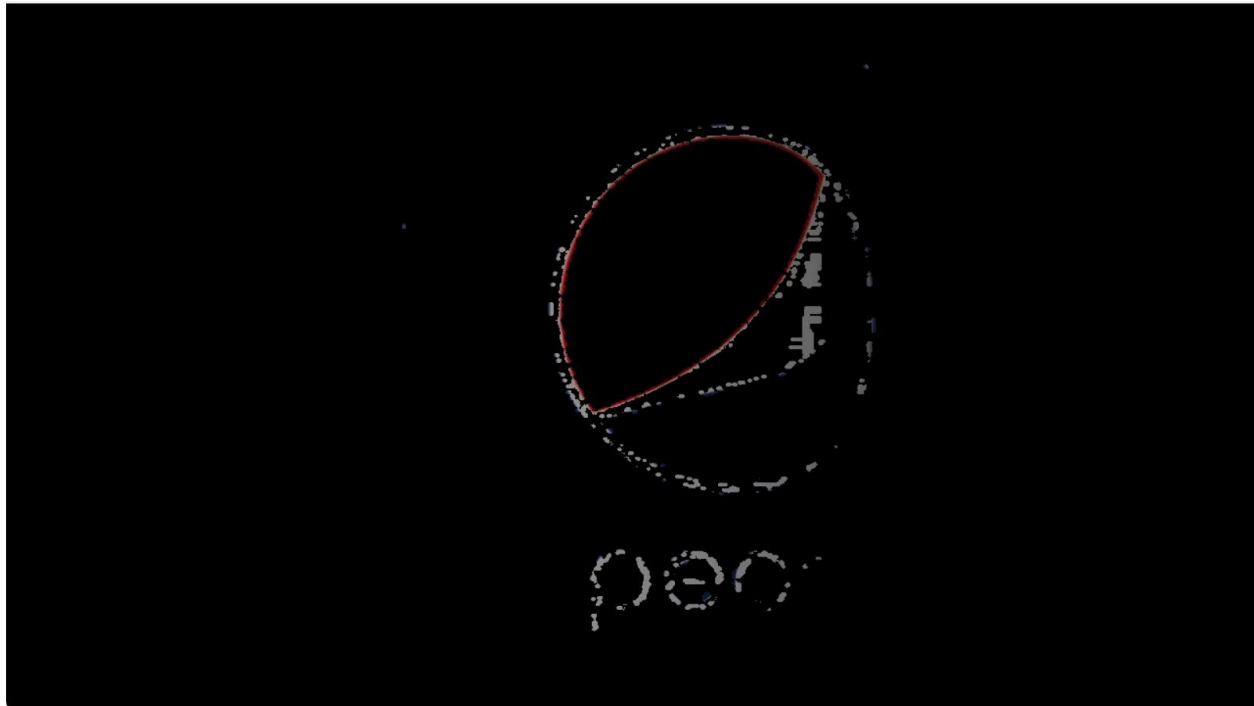
Result:

Using back project function:
code:

```python
import cv2

import numpy as np

# Read logo image and resize it to match video frame size
roi = cv2.imread("LAB_2/data/pepsi.png")
hsv = cv2.cvtColor(roi, cv2.COLOR_RGB2HSV)

# Histogram of logo
M = cv2.calcHist([hsv], [0, 1], None, [180, 256], [0,
180, 0, 256])


cap = cv2.VideoCapture("LAB_2/data/pepsi.mp4")

while True:
    ret, frame = cap.read()
    if not ret:
```

```python
        break

    # Convert frame to HSV color space
    hsvt = cv2.cvtColor(frame, cv2.COLOR_RGB2HSV)

    # Calculate back projection
    B = cv2.calcBackProject([hsvt], [0, 1], M, [0, 180,
0, 256], 1)

    # Thresholding
    ret, thresh = cv2.threshold(B, 20, 255, 0)

    # Create result image
    result = cv2.merge((thresh, thresh, thresh))
    result = cv2.bitwise_and(frame, result)

    # Display result
    cv2.imshow("Result", result)

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

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

result: