

**ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA ĐIỆN TỬ - VIỄN THÔNG**



**BÁO CÁO BÀI TẬP 5A
TÌM HIỂU VỀ BỘ MÔ TẢ ĐẶC TRƯNG HÌNH DÁNG
HU'S MOMENTS**

NHÓM 10

Sinh viên thực hiện:	Đinh Văn Quang	20DT1
	Hà Phước Phúc	20DT2
	Nguyễn Văn Quý	21DT2
Lớp học phần:	20.38B	
Giảng viên hướng dẫn:	TS. Hoàng Lê Uyên Thực	

Đà Nẵng, 10/2024

1. Giới thiệu về đặc trưng và đặc trưng hình dáng

1.1 Giới thiệu về đặc trưng:

- Đặc trưng là những thuộc tính hoặc thông tin mà chúng ta trích xuất từ hình ảnh để nhận diện, phân loại, hoặc hiểu được nội dung trong hình. Các đặc trưng này thường là những thông tin dễ đo lường và dễ phân biệt, chẳng hạn như màu sắc, hình dáng, kết cấu, và các chi tiết khác trong hình ảnh.

1.2 Giới thiệu về đặc trưng hình dáng:

- Đặc trưng hình dáng moments Hu là một bộ các giá trị toán học được sử dụng trong xử lý ảnh và thị giác máy tính để mô tả hình dạng của đối tượng trong một hình ảnh. Moments Hu được giới thiệu bởi nhà toán học Hu Ming-Kei vào năm 1962, và có một ưu điểm quan trọng là bất biến với các phép biến đổi hình học như tịnh tiến, xoay và co giãn, tức là đối tượng có bị dịch chuyển, xoay hoặc thay đổi kích thước thì các moments này vẫn giữ nguyên giá trị. Điều này làm cho moments Hu trở thành công cụ mạnh mẽ để nhận dạng và phân loại hình dạng trong các ứng dụng thị giác máy tính.

- Cách hoạt động của moments Hu:

Moments Hu dựa trên các moment hình học (geometric moments) của hình ảnh, vốn là những giá trị tính toán được từ sự phân bố điểm ảnh (pixel) trong hình ảnh đó. Những moments này cung cấp thông tin về vị trí, kích thước, và sự phân bố khối lượng của hình dạng trong không gian 2D.

- Hu đã định nghĩa bảy moments bất biến (còn gọi là 7 Hu moments) dưới dạng các biểu thức phi tuyến tính của các moment trung tâm. Các moments này có các tính chất:

+ Bất biến với tịnh tiến (translation invariant): Nếu một đối tượng di chuyển sang một vị trí khác trong ảnh, các moments Hu của nó vẫn không thay đổi.

+ Bất biến với xoay (rotation invariant): Nếu đối tượng bị xoay quanh tâm của nó, các moments Hu cũng không thay đổi.

+ Bất biến với tỷ lệ (scale invariant): Nếu đối tượng bị thay đổi kích thước, các moments Hu vẫn giữ nguyên giá trị.

Moments Hu là một kỹ thuật mạnh mẽ vì tính bất biến với các phép biến đổi hình học, làm cho nó trở thành công cụ lý tưởng trong các bài toán liên quan đến nhận dạng và phân tích hình dáng.

2. Mô tả đặc trưng Hu's moments: cách tính, ứng dụng

2.1 Cách tính đặc trưng moments Hu:

Trích đặc trưng chính là chuyển đổi đối tượng trích được thành một vector đặc trưng đa chiều sao cho vector này chứa đựng các đặc điểm hữu hiệu và riêng

biệt, giúp phân biệt đối tượng này với đối tượng khác mà không cần phải dùng toàn bộ khung hình

Tính toán các giá trị moments Hu làm đặc trưng hình dạng:

Bước 1: Tiền xử lý

Trích đối tượng quan tâm ra khỏi phần còn lại của khung hình, tiếp theo thực hiện chuyển ảnh trên nền đen thành ảnh xám và cuối cùng là ảnh nhị phân với điểm trắng thuộc về vật và điểm đen thuộc về nền.

Bước 2: Tính các moments 2 chiều

$$m_{00} = \sum_x \sum_y s(x, y) \quad (1)$$

Trong đó: (x, y) là tọa độ điểm ảnh, $s(x, y)$ là hàm ảnh nhị phân, là 1 hoặc là 0 tùy theo điểm ảnh (x, y) thuộc vùng đối tượng hoặc vùng nền tương ứng

Bước 3: Tính các moments trung tâm nhằm làm cho các moments 2 chiều sử dụng công thức (1) trở nên bất biến đối với sự dịch chuyển của ảnh nhị phân trong khung hình:

$$m_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q s(x, y) \quad (2)$$

Trong đó điểm (\bar{x}, \bar{y}) là trọng tâm của ảnh nhị phân: $\bar{x} = \frac{m_{10}}{m_{00}}$ và $\bar{y} = \frac{m_{01}}{m_{00}}$

Bước 4: Chuẩn hóa các moments trung tâm sử dụng công thức (2) để chúng bất biến với sự co giãn của ảnh nhị phân.

$$M_{pq} = \frac{m_{pq}}{m_{00}^{\frac{p+q}{2}+1}} \quad (3)$$

Bước 5: Tính 7 mô-men Hu dựa vào các mô-men trung tâm chuẩn hoá sử dụng công thức (3)

$$S1 = M_{20} + M_{02}$$

$$S2 = (M_{20} - M_{02}) (M_{20} - M_{02}) + 4M_{11}M_{11}$$

$$S3 = (M_{30} - 3M_{12}) (M_{30} - 3M_{12}) + (M_{30} - 3M_{21}) (M_{30} - 3M_{21})$$

$$S4 = (M_{30} + 3M_{12})^2 + (M_{03} + 3M_{21})^2$$

$$S5 = (M_{30} - 3M_{12})^2 (M_{30} + M_{12}) [(M_{30} + 3M_{12})^2 - 3(M_{03} + 3M_{21})^2] \\ + (3M_{21} + M_{03}) (M_{03} + M_{12}) [3(M_{30} + 3M_{12})^2 - (M_{03} + M_{21})^2]$$

$$S6 = (M_{20} + M_{02}) [(M_{30} + M_{12})^2 - (M_{03} + M_{21})^2] + 4M_{11}(M_{30} + M_{12}) (M_{03} + M_{21})$$

$$S7 = (3M_{21} - M_{03}) (M_{30} + M_{12}) [(M_{30} + M_{12})^2 - 3(M_{21} + M_{03})^2] \\ - (M_{30} - 3M_{12}) (M_{21} + M_{02}) [3(M_{30} + M_{12})^2 - (M_{21} + M_{03})^2]$$

Bước 6: Lấy logHu dựa vào 7 mô-men trung tâm chuẩn hoá

2.2 Ứng dụng của moments Hu:

- Moments Hu thường được sử dụng trong nhiều ứng dụng khác nhau liên quan đến phân tích hình dạng như:

- + Nhận dạng đối tượng: Dùng để phân loại và nhận diện các đối tượng trong ảnh dù chúng có bị xoay, phóng to hoặc di chuyển.
- + Phân đoạn ảnh (Image segmentation): Để phân biệt các vùng khác nhau trong ảnh dựa trên hình dạng.
- + Nhận diện ký tự và dấu hiệu: hệ thống OCR (Optical Character Recognition) hoặc nhận diện biển số xe.
- + Theo dõi vật thể: Dùng để theo dõi các đối tượng trong video, bất kể thay đổi về góc nhìn.

3. Bài tập tính đặc trưng Hu's moments làm thủ công

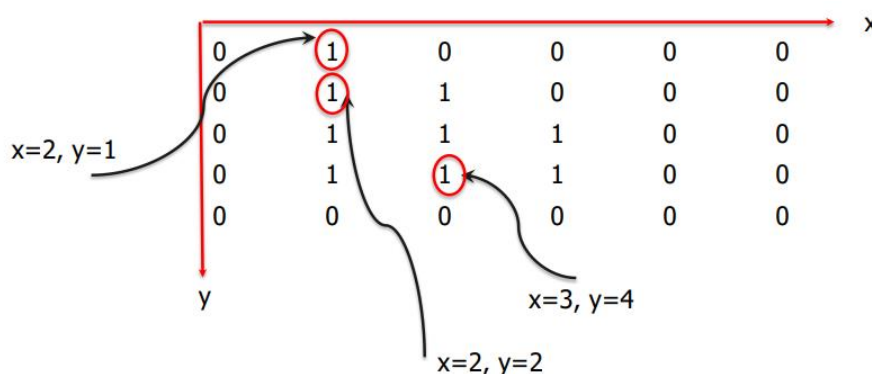
* **Đề bài:** Cho ảnh nhị phân như sau:

0	1	0	0	0	0
0	1	1	0	0	0
0	1	1	1	0	0
0	1	1	1	0	0
0	0	0	0	0	0

Bảng 1. Bảng nhị phân của ảnh

Tính đặc trưng S_1 (m_{00} , \bar{x} , \bar{y} , m_{02} , m_{20} , M_{02} , M_{20} , suy ra S_1) bằng cách thủ công, sau đó kiểm tra lại S_1 bằng code

- Giả sử chọn chiều trục x, y như sau:



- Các moment trung tâm được tính dựa trên tọa độ trọng tâm (\bar{x}, \bar{y}) :

$$m_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q s(x, y) \quad p, q = 0, 1, 2, 3$$

- Các moment trung tâm $m_{00}, m_{10}, m_{01}, m_{20}, m_{02}$

$$m_{00} = \sum_x \sum_y (x - \bar{x})^0 (y - \bar{y})^0 s(x, y) = \sum_x \sum_y s(x, y) \\ = 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 = 9$$

$$m_{10} = \sum_x \sum_y (x - \bar{x})^1 (y - \bar{y})^0 s(x, y) = \sum_x \sum_y x \cdot s(x, y) \\ = 1 \cdot (0) + 2 \cdot (4) + 3 \cdot (3) + 4 \cdot (2) + 5 \cdot (0) + 6 \cdot (0) = 25$$

$$m_{01} = \sum_x \sum_y (x - \bar{x})^0 (y - \bar{y})^1 s(x, y) = \sum_x \sum_y y \cdot s(x, y) \\ = 1 \cdot (1) + 2 \cdot (2) + 3 \cdot (3) + 4 \cdot (3) + 5 \cdot (0) = 26$$

$$m_{20} = \sum_x \sum_y (x - \bar{x})^2 s(x, y) = (2 - 2.78)^2 \cdot (4) + (3 - 2.78)^2 \cdot (3) \\ + (4 - 2.78)^2 \cdot (2) \approx 5,56$$

$$m_{02} = \sum_x \sum_y (y - \bar{y})^2 s(x, y) = (1 - 2.89)^2 \cdot (1) + (2 - 2.89)^2 \cdot (2) \\ + (3 - 2.89)^2 \cdot (3) + (4 - 2.89)^2 \cdot (3) \approx 8,89$$

- Tọa độ trọng tâm của nhị phân:

$$\bar{x} = \frac{\sum_x \sum_y x \cdot s(x, y)}{\sum_x \sum_y s(x, y)} = \frac{m_{10}}{m_{00}} = \frac{25}{9} \approx 2,78$$

$$\bar{y} = \frac{\sum_x \sum_y y \cdot s(x, y)}{\sum_x \sum_y s(x, y)} = \frac{m_{01}}{m_{00}} = \frac{26}{9} \approx 2,89$$

- Các moment chuẩn hóa trung tâm:

$$M_{02} = \frac{m_{02}}{m_{00}^{\frac{0+2}{2}+1}} = \frac{m_{02}}{m_{00}^2} = \frac{8,89}{9^2} \approx 0,11$$

$$M_{20} = \frac{m_{20}}{m_{00}^{\frac{2+0}{2}+1}} = \frac{m_{20}}{m_{00}^2} = \frac{5,56}{9^2} \approx 0,07$$

- Hu's Moment thứ nhất được tính từ các moment chuẩn hóa trung tâm:

$$S_1 = M_{20} + M_{02} = 0,07 + 0,11 \approx 0,18$$

- Kiểm tra S_1 bằng code

```
import numpy as np

I = np.array([
    [0, 1, 0, 0, 0, 0],
    [0, 1, 1, 0, 0, 0],
    [0, 1, 1, 1, 0, 0],
    [0, 1, 1, 1, 0, 0],
    [0, 0, 0, 0, 0, 0]])

x_size, y_size = I.shape
y, x = np.indices(I.shape) + 1

# Tính m00 (tổng cường độ ảnh)
m00 = np.sum(I)
m10 = np.sum(x * I)
m01 = np.sum(y * I)

# Tính trọng tâm x_bar và y_bar
x_bar = m10 / m00
y_bar = m01 / m00

# Tính m02 và m20
m20 = np.sum(((x - x_bar) ** 2) * I)
m02 = np.sum(((y - y_bar) ** 2) * I)

# Tính các moment chuẩn hóa M02 và M20
M20 = m20 / (m00 ** (1 + (2 / 2)))
M02 = m02 / (m00 ** (1 + (2 / 2)))

# Đặc trưng S1
S1 = M20 + M02

# Hiển thị kết quả
print(f'm00 = {m00:.2f}')
print(f'x_bar = {x_bar:.2f}, y_bar = {y_bar:.2f}')
print(f'm20 = {m20:.2f}, m02 = {m02:.2f}')
print(f'M20 = {M20:.2f}, M02 = {M02:.2f}')
print(f'S1 = {S1:.2f}')
```

- Kết quả kiểm tra S_1 bằng code

```
m00 = 9.00  
x_bar = 2.78, y_bar = 2.89  
m20 = 5.56, m02 = 8.89  
M20 = 0.07, M02 = 0.11  
S1 = 0.18
```

4. Code tính đặc trưng Hu's moments và áp dụng để tính Hu's moments cho 1 ảnh lá cây bất kỳ, áp dụng cho CSDL lá cây tự sưu tầm.



Hình 1. Mẫu lá cây

Bước 1: Chuyển ảnh lá RGB sang ảnh xám, rồi chuyển sang ảnh nhị phân, từ ảnh nhị phân tiếp tục chuyển sang mảng 0, 1

```
import numpy as np
from PIL import Image

# Load image and convert to grayscale
filename = "Pic_binary/binary_la.jpg"

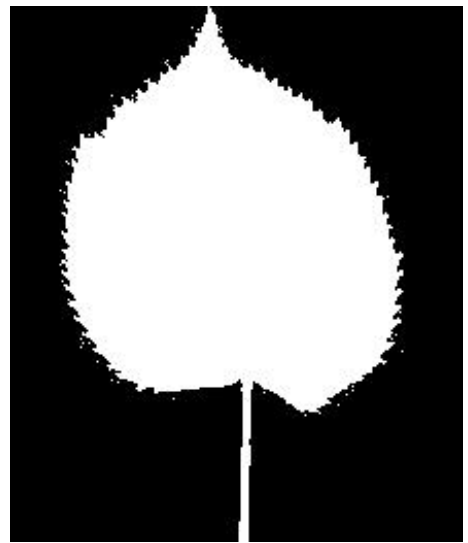
im = Image.open(filename).convert("L")

# Convert to binary image
threshold = 128
im_bin = im.point(lambda p: 255 if p > threshold else 0)

# Convert to NumPy array
data = np.array(im_bin)
```



Hình 2. Ảnh xám của lá



Hình 3. Ảnh nhị phân của lá

Bước 2: Tính moments m_{00} và giá trị \bar{x}, \bar{y} là trọng tâm của ảnh nhị phân


```

# Calculate m00 (total number of white pixels, i.e., pixels with value 255)
m00 = np.sum(data == 255) * 255
print(f"total number of white pixels: {m00}")

# Image dimensions
height, width = data.shape

# Calculate centroid coordinates
x_centroid = np.sum(np.arange(width) * np.sum(data, axis=0)) / m00
y_centroid = np.sum(np.arange(height) * np.sum(data, axis=1)) / m00

print(f'Centroid Coordinates: x = {x_centroid}, y = {y_centroid}')

```

Bước 3: Tính các moments trung tâm, ở đây $p, q = 0,1,2,3$ nên chạy vòng lặp 4 lần để tính m_{pq} tương ứng

```

def central_moment(data, p, q, x_centroid, y_centroid):
    y, x = np.ogrid[:data.shape[0], :data.shape[1]]
    return np.sum(((x - x_centroid) ** p) * ((y - y_centroid) ** q) * data)

# Tính toán các mômen trung tâm lên đến bậc 3
moments = {}
for p in range(4):
    for q in range(4):
        if p + q <= 3:
            moments[f'm_{p}{q}'] = central_moment(data, p, q, x_centroid, y_centroid)

# In kết quả moment trung tâm
print("\nCentral Moments:")
for key, value in moments.items():
    print(f"{key} = {value}")

```

Bước 4: Tính các moments trung tâm chuẩn hóa, ở đây $p, q = 0,1,2,3$ nên ta chạy vòng lặp 4 lần để tính M_{pq} tương ứng

```

def central_normalized_moment(moments, p, q):
    return moments[f'm_{p}{q}'] / (m00 ** ((p + q) / 2) + 1))

# Tính toán các mômen chuẩn hóa trung tâm lên đến bậc 3
normalized_moments = {}
for p in range(4):
    for q in range(4):
        if 0 < p + q <= 3: # Bỏ qua M_00 vì nó luôn bằng 1
            normalized_moments[f'M_{p}{q}'] = central_normalized_moment(moments, p, q)

# In kết quả mômen chuẩn hóa trung tâm
print("\nCentral Normalized Moments:")
for key, value in normalized_moments.items():
    print(f"{key} = {value}")

```

Bước 5: Tính 7 moments Hu theo công thức

```

def calculate_hu_moments(M):
    S = {}
    S[1] = M['M_20'] + M['M_02']
    S[2] = (M['M_20'] - M['M_02']) ** 2 + 4 * (M['M_11'] ** 2)
    S[3] = (M['M_30'] - 3 * M['M_12']) ** 2 + (3 * M['M_21'] - M['M_03']) ** 2
    S[4] = (M['M_30'] + M['M_12']) ** 2 + (M['M_03'] + M['M_21']) ** 2
    S[5] = (M['M_30'] - 3 * M['M_12']) * (M['M_30'] + M['M_12']) * (
        (M['M_30'] + M['M_12']) ** 2 - 3 * (M['M_03'] + M['M_21']) ** 2) + \
        (3 * M['M_21'] - M['M_03']) * (M['M_03'] + M['M_21']) * (
            3 * (M['M_30'] + M['M_12']) ** 2 - (M['M_03'] + M['M_21']) ** 2)
    S[6] = (M['M_20'] - M['M_02']) * ((M['M_30'] + M['M_12']) ** 2 - (M['M_03'] + M['M_21']) ** 2) + \
        4 * M['M_11'] * (M['M_30'] + M['M_12']) * (M['M_03'] + M['M_21'])
    S[7] = (3 * M['M_21'] - M['M_03']) * (M['M_30'] + M['M_12']) * (
        (M['M_30'] + M['M_12']) ** 2 - 3 * (M['M_03'] + M['M_21']) ** 2) - \
        (M['M_30'] - 3 * M['M_12']) * (M['M_03'] + M['M_21']) * (
            3 * (M['M_30'] + M['M_12']) ** 2 - (M['M_03'] + M['M_21']) ** 2)
    return S

hu_moments = calculate_hu_moments(normalized_moments)

# Print Hu's invariant moments
print("\nHu's Invariant Moments:")
for i, moment in hu_moments.items():
    print(f"S{i} = {moment}")

```

Bước 6: Lấy logHu của 7 mô-men vừa tìm được ở bước 5

```
# Optionally, calculate log of absolute values for better numerical stability
log_hu_moments = {f"log|S{i}|": -np.sign(m) * np.log10(abs(m)) for i, m in hu_moments.items()}
print("\nLogarithm of absolute values of Hu's Moments:")
for key, value in log_hu_moments.items():
    print(f"{key} = {value}")

Centroid Coordinates: x = 106.28483606557377, y = 113.72609289617486

Central Moments:
m_00 = 5599800.0
m_01 = 3.725290298461914e-09
m_02 = 11116793199.436476
m_03 = 3445525477.6182404
m_10 = -1.1292286217212677e-08
m_11 = 1237425823.6782787
m_12 = 5728356728.487003
m_20 = 9477020264.35451
m_21 = 45583807674.04189
m_30 = 6057045551.709931

Central Normalized Moments:
M_01 = 2.811262160774604e-19
M_02 = 0.0003545149011044118
M_03 = 4.6432725342500166e-08
M_10 = -8.521638424848019e-19
M_11 = 3.9461550254221494e-05
M_12 = 7.719670522406463e-08
M_20 = 0.0003022224882219141
M_21 = 6.142982936980775e-07
M_30 = 8.162619441257872e-08
```

Hình 4. Kết quả

Hu's Invariant Moments:

S1 = 0.0006567373893263259
S2 = 8.963352238939404e-09
S3 = 3.2497654547277213e-12
S4 = 4.617901929713914e-13
S5 = -3.977770093422328e-25
S6 = 3.807425780933019e-17
S7 = -4.022438225915502e-25

Logarithm of absolute values of Hu's Moments:

log|S1| = 3.182608257786006
log|S2| = 8.047529536476457
log|S3| = 11.488147982219049
log|S4| = 12.33555529442368
log|S5| = -24.400360321775548
log|S6| = 16.419368553716254
log|S7| = -24.395510616800102

Hình 5. Kết quả

TÀI LIỆU THAM KHẢO

- [1]. <https://www.neliti.com/publications/451220/hus-moments-for-visual-pattern-recognition>
- [2]. <https://cvexplained.wordpress.com/2020/07/21/10-4-hu-moments/>
- [3]. <https://github.com/spmallick/learnopencv/blob/master/HuMoments/images/K0.png>
- [4]. <https://www.youtube.com/watch?v=RCodqRopMHs>

*****Source Code:**

https://github.com/haphucc/Module1_AI_24_25/tree/main/Tim_hieu_Hu_moment_feature/Hu_moment_la_cay