<u>Zernike Moments</u>

Mehr Heer

(0801CS171043)

Contents

Chapter 1

Introduction

1.1 Feature Extraction

Feature extraction involves reducing the number of resources required to describe a large set of data. When performing analysis of complex data one of the major problems stems from the number of variables involved. Feature extraction is a general term for methods of constructing combinations of the variables to get around these problems while still describing the data with sufficient accuracy.

1.2 Hu's Moments

Moments have been used in image processing and classification type problems since Hu introduced them in his groundbreaking publication on moment invariants [2]. Hu used geometric moments and showed that they can be made to be translation and scale invariant.

Hu (1962) introduced seven nonlinear functions which are translation, scale, and rotation invariant. The seven moment invariants are defined as [3], Hu's seven moment invariants have been widely used in pattern recognition, and their performance has been evaluated under various deformation situations.

As Hu's seven moment invariants take every image pixel into account the computation cost will be much higher than boundary-based invariants. Image's spatial resolution decides the total amount of pixels, and to reduce the computation cost. Hu's moment can show the redundant properties due to which reconstruction is very difficult.

Since then more powerful moment techniques have been developed. A notable example is Teague's work on Zernike Moments (ZM); he was the first to use the Zernike polynomials (ZP) as a basis function for the moments.

1.3 Zernike Moments

ZP's were originally used by Fritz Zernike to describe optical aberrations in the 1930s.

Nowadays, they have been adapted for image processing in shape recognition schemes. The orthogonal properties of ZM's suits them better for such applications because unlike geometric moments their invariants can be calculated independently to arbitrary high orders without having to recalculate low order invariants.

These orthogonal properties also allow one to evaluate up to what order to calculate these moments to get a good descriptor for a given database. Teague was the first to acknowledge that an orthogonal basis function set was the solution to the fact that the basis function of geometric moments, x p y q , makes them hard to use for image processing applications because of their information redundancy.

Furthermore, the magnitude of ZM is rotationally invariant, which is crucial for certain image processing applications, such as classifying shapes that are not aligned.

Chapter 2

Mathematical Formulation

2.1 Geometric Moment

A geometric moment is defined to be:

$$m_{p,q}(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^p y^q f(x,y)\,\mathrm{dxdy}$$

2.2 Hu's 7 Invariant Moments

$$\begin{aligned}
I_1 &= \mu_{20} + \mu_{02} \\
I_2 &= (\mu_{20} + \mu_{02})^2 + 4\mu_{11}^2 \\
I_3 &= (\mu_{30} - 3\mu_{12})^2 + (3\mu_{21} - \mu_{03})^2 \\
I_4 &= (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2 \\
I_5 &= (\mu_{30} - 3\mu_{12})(\mu_{30} + \mu_{12}[(\mu_{30} + \mu_{12})^2 \\
&\quad -3(\mu_{21} + \mu_{03})^2] + (3\mu_{21} - \mu_{03})(\mu_{21} \\
&\quad +\mu_{03}[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \\
I_6 &= (\mu_{20} + \mu_{02})[(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \\
&\quad +4mu_{11}(\mu_{30} + \mu_{12})^2(3\mu_{21} + \mu_{03}) \\
I_7 &= (3\mu_{21} - \mu_{03})(\mu_{30} - 3\mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] \\
&\quad -(\mu_{30} - 3\mu_{12})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2]
\end{aligned}$$

2.3 Zernike Polynomial

$$V_{nm}(\rho,\theta) = R_{n,m} exp(jm\theta)$$

The radial polynomial Rn,m is defined as:

$$R_{n,m}(\rho) = \sum_{s=0}^{(n-|m|)/2} (-1)^s . \frac{(n-s)!}{s!(\frac{n+|m|}{2}-s)!(\frac{n-|m|}{2}-s)!}\rho^{n-2s}$$

Rotational invariance is achieved by computing the magnitudes of the ZMs. The rotation of an image is easily expressed in polar coordinates since it is a simple change of angle.

$$A_{nm} = \left[\frac{(n+1)}{\pi}\right] \int_0^{2\pi} \int_0^1 f(\rho,\theta) R_{nm}(\rho) exp(-jm\theta) \rho d\rho d\theta$$

therefore if we plug in the rotated image into our formula we get

$$A'_{nm} = \left[\frac{(n+1)}{\pi}\right] \int_0^{2\pi} \int_0^1 f(\rho,\theta-\alpha) R_{nm}(\rho) exp(-jm\theta) \rho d\rho d\theta$$

by a change of variables $\theta_1 = \theta - \alpha$ we see that

$$\begin{aligned} A'_{nm} &= \left[\frac{(n+1)}{\pi}\right] \int_0^{2\pi} \int_0^1 f(\rho,\theta_1) R_{nm}(\rho) exp(-jm(\theta_1+\alpha)) \rho d\rho d\theta \\ &= \left[\left[\frac{(n+1)}{\pi}\right] \int_0^{2\pi} \int_0^1 f(\rho,\theta_1) R_{nm}(\rho) exp(-jm(\theta_1) \rho d\rho d\theta\right] \cdot exp(-jm(\alpha)) \\ &= A_{nm} exp(-jm(\alpha)) \end{aligned}$$

2.4 Reconstruction of Original Image f(x,y):

$$\hat{f}(x,y) = \sum_{n=0}^{n_{max}} \sum_{m} A_{nm} V_{nm}(\rho,\theta)$$

Chapter 3

Algorithm

1- calculate the Zernike moment (n,m) for an oval shape,
2- rotate the oval shape around its centeroid,
3- calculate the Zernike moment (n,m) again,
4- the amplitude of the moment (A) should be the same for both images
5- the phase (Phi) should be equal to the angle of rotation

Methods:

```
rad = radialpoly(r,n,m)
 where
 r = radius
 n = the order of Zernike polynomial
 m = the repetition of Zernike moment
```

```
Function to find the Zernike moments for an N x N binary ROI
Z, A, Phi = Zernikmoment(src, n, m)
where
src = input image
n = The order of Zernike moment (scalar)
m = The repetition number of Zernike moment (scalar)
and
Z = Complex Zernike moment
A = Amplitude of the moment
Phi = phase (angle) of the moment (in degrees)
```

Chapter 4

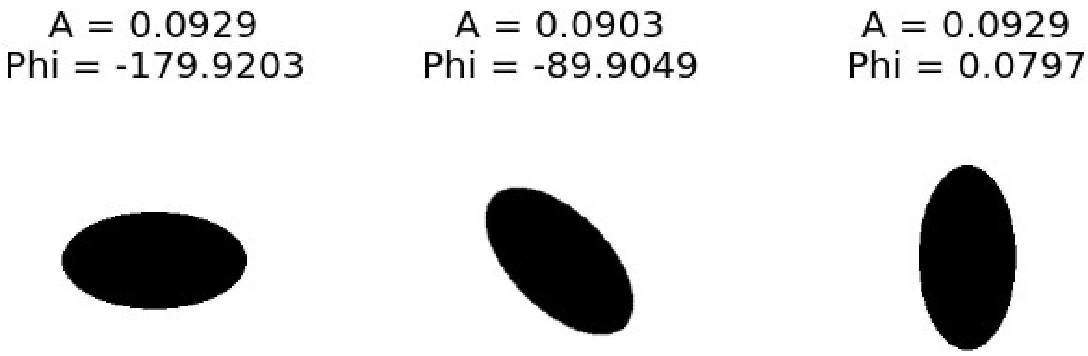
Example

4.1 example.py :

```
import cv2
import matplotlib.pylab as plt
from zm import Zernikemoment

if __name__ == '__main__':
    n = 4
    m = 2
    print('Calculating Zernike moments ...')
    fig, axes = plt.subplots(2, 3)

    imgs = ['Oval_H.png', 'Oval_45.png', 'Oval_V.png']
    for i in range(3):
        src = cv2.imread(imgs[i], cv2.IMREAD_COLOR)
        src = cv2.cvtColor(src, cv2.COLOR_BGR2GRAY)
        Z, A, Phi = Zernikemoment(src, n, m)
        axes[0, i].imshow(plt.imread(imgs[i]))
        axes[0, i].axis('off')
        title = 'A = ' + str(round(A, 4)) + '\nPhi = ' +
str(round(Phi, 4))
        axes[0, i].set_title(title)
    print('Calculation is complete')

    plt.show()
```

```
cd Zernike
python example.py
```

4. 2 Output:



Chapter 5

Learning Outcome

- Zernike Moments are rotation invariant due to orthogonal property which makes them superior to Hu's Moments
- Zernike Moments could be used as a viable shape descriptors
- Zernike Moments can be utilized to extract features from images that describe the shape characteristics of an object. For instance, Zernike moments are utilized as shape descriptors to classify benign and malignant breast masses or the surface of vibrating disks.
- Zernike Moments also have been used to quantify shape of Osteosarcoma cancer cell lines in single cell level.
- Used in IR or visual astronomy and satellite imagery.
- Another application of the Zernike polynomials is found in the Extended Nijboer–Zernike theory of diffraction and aberrations.

A. References

[1] Khotanzad A. and Hong Y. H. Invariant image recognition by zernike moments. IEEE, 12(5):489 – 497, 1990.

[2] Hu M. K. Visual pattern recognition by moment invariants. IRE Transactions on Information Theory, 8(2):179 – 187, 1962.

[3] Teague M. R. Image analysis via the general theory of moments. Optical Society of America, 70(8):920 – 930, 1979.

[4] https://en.wikipedia.org/wiki/Zernike_polynomials