ENGI 7854/9804 Image Processing and Applications

Laboratory 4

Hu's invariant moments

Due: July 31,, 2020

Introduction

The objective of this exercise is to understand and calculate some of features used as image descriptors. Hu's seven moment invariants are insensitive to changes in scale, position and rotation. These moment invariants have been extensively applied to image pattern recognition, image registration, and image reconstruction.

Procedure

The 2-D moment of order *(p+q)* of a digital image *f(x,y)* of size $M \times N$ is defined as

$$m_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} x^p y^q f(x,y) \tag{1}$$

where $p = 0, 1, 2, ...$ and $q = 0, 1, 2, ...$ are integers. The corresponding central moment of order $(p+q)$ is defined as:

$$\mu_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x - \bar{x})^p (y - \bar{y})^q f(x,y) \tag{2}$$

where $\bar{x} = \frac{m_{10}}{m_{00}}$ and $\bar{y} = \frac{m_{01}}{m_{00}}$. The normalized central moments are defined as:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}} \tag{3}$$

where $\gamma = \frac{p+q}{2} + 1$ for $p + q = 2, 3, ...$

A set of seven invariant moments can be derived from the second and third moments:

$$\phi_1 = \eta_{20} + \eta_{02} \tag{4}$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + (2\eta_{11})^2 \tag{5}$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \tag{6}$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \tag{7}$$

$$\begin{aligned}\phi_5 = {} & (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ & + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]\end{aligned} \tag{8}$$

$$\phi_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \tag{9}$$

$$\begin{aligned}\phi_7 = {} & (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ & - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]\end{aligned} \tag{10}$$

Calculating the moment invariants

1. Download the grayscale test image (image.png), *calculate_moments.m*, and *Moment_invariants.m* files from D2L under *Lab 4* and save them in the MATLAB working directory. Read the files and understand the implementation of calculating the seven invariant moments.

2. Read the image, obtain the size of the image and pad the image by one-fourth the image size in all directions with zeros. Use this as the basis for next steps (referred to as *Im1*).

3. Create spatial transformation matrix T1 for translation as follows:

   ```
   T1 = maketform('affine', [1 0 0; 0 1 0; Xt Yt 1]);
   % Xt, Yt are one-fourth of your original image
   ```

   Use the above transformation matrix T1 and perform the transformation as follows:

   ```
   Im2 = imtransform(Im1, T, ...
   'XData',[1 size(Im1,1)], 'YData',[1 size(Im1,2)]);
   ```

   Display the translated image Im2;

4. Create spatial transformation matrix T2 for scaling to 0.5 (of original size) as follows:

   ```
   T2 = maketform('affine', [0.5 0 0; 0 0.5 0; 0 0 1]);
   ```

   Use the above transformation matrix T2 and perform the transformation as follows:

   ```
   Im3 = imtransform(Im1, T2, ...
   'XData',[1 size(Im1,1)], 'YData',[1 size(Im1,2)]);
   ```

   Display the translated image Im3;

5. Create spatial transformation matrix T3 to rotate the image by 45 degrees as follows:

   ```
   T3 = maketform('affine', [cos(pi/4) sin(pi/4) 0; -sin(pi/4) cos(pi/4)
      0; 0 0 1]);
   ```

   Use the above transformation matrix T3 and perform the transformation as follows:

```
Im4 = imtransform(Im1, T3, ...
'XData',[-269 size(Im1,1)-270], 'YData',[+111 size(Im1,2)+110]);
```

Display the translated image Im4;

6. Create spatial transformation matrix T4 for rotating the image 90 degrees as follows:

   ```
   T4 = maketform('affine', [cos(pi/2) sin(pi/2) 0; -sin(pi/2) cos(pi/2)
      0; 0 0 1]);
   ```

   Use the above transformation matrix T4 and perform the transformation as follows:

   ```
    Im5 = imtransform(Im1, T4, ...
   'XData',[-539 size(Im1,1)-540], 'YData',[1 size(Im1,2)]);
   ```

   Display the translated image Im5;

7. Flip the original image Im1 from left to right using following command and generate image Im6.

   ```
   Im6=flipdim(Im1,2);
   ```

   Display the mirrored image Im6;

8. The downloaded matlab file *Moment_invariant.m* contains a function *Moment_invariant()*, which calls to another function named as *calculate_moments*. Collectively, these functions calculate the moment invariants.

   Use the function *Moment_invariant()* and pass the images created from step 2 to step 7 (Im1-Im6) from the command line. For example, the following line will display the moment invariants calculated for Im1:

   ```
   Moment_invariants(Im1)
   ```

Discussion

1. You should include the MATLAB code and results in your lab report.

2. Discuss the values of the moment invariants computed for each image, including any similarities or differences. What do these similiarities or differences mean in terms of image features?