Simple Image Analysis By Moments
Version 0.2*

Maintained by: Johannes Kilian

March 15, 2001

Abstract

This document is a preliminary version of a manual on using moments to analyze the form of objects in an image. Only moments up to second order are used to get simple features of the object, like center of gravity, semi-axes, orientation and eccentricity of the object ellipsis.
If you have comments or can contribute to this document, please mail J. Kilian

Contents

1 General remarks on moments

1.1 Definition

The definition of moments of the gray value-function $f(x, y)$ of an object is the following:

*This document is for public distribution

$$m_{p,q} = \int\int x^p y^q f(x,y) dx dy \tag{1}$$

The integration is calculated over the area of the object. Generally each other pixel based feature instead of the gray value could be used to calculate the moments of the object.

Using binary images the gray value function $f(x,y)$ becomes

$$f(x,y) = b(x,y) = \{ \begin{array}{cc} 1 & \text{Object} \\ 0 & \text{Background} \end{array} \tag{2}$$

and can be neglected in the subsequent formulas.

1.2 Order of moments

Moments are generally classified by the order of the moments. The order of a moment depends on the indices p and q of the moment $m_{p,q}$ and vice versa. The sum $p+q$ of the indices is the order of the moment $m_{p,q}$.

Considering this, the following moments are defined:

- zero order moment $((p,q) = (0,0))$

$$m_{0,0} = \int\int dx dy b(x,y) \tag{3}$$

  The zero order moment describes the area A of the object.

- first order moments $((p,q) = (1,0)$ or $(0,1))$

$$\begin{aligned} m_{1,0} &= \int\int dx dy x f(x,y) \\ m_{0,1} &= \int\int dx dy y f(x,y) \end{aligned} \tag{4}$$

  The first order moments contain information about the center of gravity of the object;

$$\begin{aligned} x_c &= \frac{m_{1,0}}{m_{0,0}} \\ y_c &= \frac{m_{0,1}}{m_{0,0}} \end{aligned} \tag{5}$$

- second order moments $((p,q) = (2,0)$ or $(0,2)$ or $(1,1))$

$$\begin{aligned} m_{2,0} &= \int\int dx dy x^2 f(x,y) \\ m_{0,2} &= \int\int dx dy y^2 f(x,y) \\ m_{1,1} &= \int\int dx dy x y f(x,y) \end{aligned} \tag{6}$$

- ...

1.3 Spatial, central and central normalized moments

Formula (1) describes general *spatial moments* of the object.

From the spatial moments the *central moments* can be derived by reducing the spatial moments with the center of gravity (x_c, y_c) of the object, so all the the central moments refer to the center of gravity of the object. Expressed as formula the central moments are calculated as follows

$$\mu_{p,q} = \int\int (x - x_c)^p (y - y_c)^q f(x, y) dxdy \tag{7}$$

From this the following facts result:

$$\begin{aligned} \mu_{0,0} &= m_{0,0} \\ \mu_{1,0} &= \mu_{0,1} = 0 \end{aligned} \tag{8}$$

The central moments of first or higher order can directly be derived from the spatial moments by

$$\mu_{p,q} = \frac{m_{p,q}}{m_{0,0}} - \left(\frac{m_{1,0}}{m_{0,0}}\right)^p * \left(\frac{m_{0,1}}{m_{0,0}}\right)^q \tag{9}$$

Using formula (9) the central moments of first and second order can be derived from spatial moments as follows:

$$\begin{aligned} \mu_{1,0} &= \frac{m_{1,0}}{m_{0,0}} - \left(\frac{m_{1,0}}{m_{0,0}}\right) = 0 \\ \mu_{0,1} &= \frac{m_{0,1}}{m_{0,0}} - \left(\frac{m_{0,1}}{m_{0,0}}\right) = 0 \\ \mu_{2,0} &= \frac{m_{2,0}}{m_{0,0}} - \left(\frac{m_{1,0}}{m_{0,0}}\right)^2 = \frac{m_{2,0}}{m_{0,0}} - x_c^2 \\ \mu_{0,2} &= \frac{m_{0,2}}{m_{0,0}} - \left(\frac{m_{0,1}}{m_{0,0}}\right)^2 = \frac{m_{0,2}}{m_{0,0}} - y_c^2 \\ \mu_{1,1} &= \frac{m_{1,1}}{m_{0,0}} - \left(\frac{m_{1,0}}{m_{0,0}}\right) * \left(\frac{m_{0,1}}{m_{0,0}}\right) = \frac{m_{1,1}}{m_{0,0}} - x_c * y_c \end{aligned} \tag{10}$$

where x_c, y_c is the center of gravity of the object (see formula (5)).

The main advantage of central moments is their invariancy to translations of the object. Therefore they are suited well to describe the form of the object.

A disadvantage of the spatial and central moments is their dependency on the size of the object. This is disturbing, when trying to compare objects, which are looked at from different distances. To allow a comparison of those objects, scaling of the moments is necessary. Normally the area A of the object is used as a scaling factor. Dividing the central moments $\mu_{p,q}$ with powers of A we get *central normalized moments* $\nu_{p,q}$.

$$\nu_{p,q} = \frac{\mu_{p,q}}{m_{0,0}^{\frac{p+q}{2}+1}} \tag{11}$$

The main advantage of normalized moments is their invariancy to the size of the object.

2 Moments in OpenCV

This document might be outdated , please check OpenCV document on internet.

As a data structure to maintain the moments the moment state structure `CvMoments` is defined in file `CvTypes.h` as follows:

```
typedef struct CvMoments {
    double  m00, m10, m01, m20, m11, m02, m30, m21, m12, m03;
    double  mu20, mu11, mu02, mu30, mu21, mu12, mu03;
    double  inv_sqrt_m00;
} CvMoments;
```

Description of the members of the `CvMoments`-structure:

`m00 ...m03` Spatial moments $m_{p,q}$ of zero to third order

`mu20 ...mu03` Central moments $\mu_{p,q}$ of second and third order. By definition $\mu_{0,0} = m_{0,0}$ and $\mu_{1,0} = \mu_{0,1} = 0$, therefore only second and third order central moments are stored in the structure.

`inv_sqrt_m00` is the inverse square root of the spatial moment `m00`. This value can be used to calculate the normalized moments $\nu_{p,q}$ from the central moments $\mu_{p,q}$

There are several functions in OpenCV, dealing with moments:

- `cvMoments()` - calculates moments up to third order of image plane and fills moment state structure. (see OpenCV Reference Manual, Chapter 6 - Image Statistics).
- `cvGetSpatialMoment()` - retrieves spatial moment from the moment state structure. (see OpenCV Reference Manual, Chapter 6 - Image Statistics).
- `cvGetCentralMoment()` - retrieves central moment from the moment state structure. (see OpenCV Reference Manual, Chapter 6 - Image Statistics).
- `cvGetNormalizedCentralMoment()` - retrieves normalized central moment from the moment state structure. (see OpenCV Reference Manual, Chapter 6 - Image Statistics).
- `cvGetHuMoments()` - calculates seven moment invariants from the moment state structure. (see OpenCV Reference Manual, Chapter 6 - Image Statistics).
- `cvContoursMoments()` - calculates contour moments up to order 3.(see OpenCV Reference Manual, Chapter 3 - Contour Processing).

Information about moments are found in Chapter 3 (Contour Processing) and Chapter 6 (Image Statistics) of the OpenCV Reference Manual.

As the calculation of moments in OpenCV with `cvContourMoments()`is based on the contour processing of OpenCV, the moments can only be determined on binary images. To use the contour moments to perform a simple object analysis, the object has to be separated in the image: the considered object has to be separated from the rest of the image with binarization.

It has to be considered that the moments calculated in OpenCV do have negative values, because `cvContourMoments()` is affected by the contour orientation of OpenCV. Because contours are retrieved in mirrored coordinate system (origin is top-left, y-axis is oriented downward), all the contours do have clockwise orientation and thus do have negative area according to Green's formula. From this fact result negative moments with clockwise oriented contours.

Errors in `cvContourMoments()`: *Up to version* `Nov 08 2000` *there are severe errors in calculating the central moments, but the spatial moments are calculated correctly. For this reason, use formula (9) to determine the central moments correctly! Also the sign of the spatial moment is determined wrongly (it is positive in each case and should be negative in some cases (with clockwise oriented contours)). This errors should be corrected in version later than version* `Nov 08 2000`

3 Object Features based on moments

The moments are features of the object, which allow a geometrical reconstruction of the object. They do not have a direct understandable geometrical meaning, but usual geometrical parameters can be derived from them.

In the following sections only moments up to second order are used to derive fairly simple object features. These features exactly describe simple objects up to the complexity of an ellipsis.

To get more precise description of complex objects, you have to use moments of higher order and more complex moments like Zernike- or Legendre-moments. The more complex your object is, the higher the order of your describing moments should be to get a minimal error reconstructing your object by your moments.

3.1 Area A of the object

Considering the gray value function $f(x, y)$ as the density of the object $\rho(x, y)$, the spatial moment of zero order $m_{0,0}$ is the area A of the object.

$$A = m_{0,0} \tag{12}$$

OpenCV offers a separate function `cvContourArea()` to calculate the area of the object bounded by the contour, but it could also retrieved from member m_{0_0} of the `struct CVMoments`.

3.2 Center of Gravity x_c, y_c

As mentioned in formula (5) the coordinates x_c and y_c of the center of gravity are simply described by the spatial moments of first order $m_{1,0}$ and $m_{0,1}$ divided by the zero order moment $m_{0,0}$ (i.e. the area A of the object).

$$\begin{aligned} x_c &= \frac{m_{1,0}}{A} = \frac{m_{1,0}}{m_{0,0}} \\ y_c &= \frac{m_{0,1}}{A} = \frac{m_{0,1}}{m_{0,0}} \end{aligned} \tag{13}$$

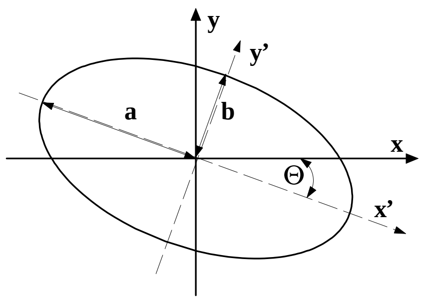

Figure 1: Image Ellipse

3.3 Tensor of moments

The analogy of the moments to mechanical moments allows a deeper understanding of the central moments of second order $\mu_{2,0}$, $\mu_{0,2}$ and $\mu_{1,1}$. They contain terms, in which the gray value function $f(x, y)$, i.e. the density $\rho(x, y)$ of the object is multiplied with the square of the distance from the center of gravity (x_c, y_c). Exactly the same terms are available in the inertial tensor, known from physical mechanics. The three central moments of second order build the components of the inertial tensor of the rotation of the object about its center of gravity:

$$J = \begin{bmatrix} \mu_{2,0} & -\mu_{1,1} \\ -\mu_{1,1} & \mu_{0,2} \end{bmatrix} \tag{14}$$

Using the inertial tensor analogy several further parameters could be derived from the central moments of second order.

3.3.1 Semi-major and Semi-major axes a and b

The main inertial axis could be derived by calculating the eigenvalues of the inertial tensor:

$$\lambda_{1,2} = \sqrt{\frac{1}{2} * (\mu_{2,0} + \mu_{0,2}) \pm \sqrt{4 * \mu_{1,1}^2 - (\mu_{2,0} - \mu_{0,2})^2}} \tag{15}$$

The main inertial axes of the object correspond to the semi-major and semi-minor axes a and b of the image ellipse which can be used as a approximation of the considered object. The main inertial axes are those axes, around which the object can be rotated with minimal (major semi-axis a) or maximal (minor semi-axis b) inertia.

The semi-axis a and b are illustrated in figure (1).

3.3.2 Orientation θ

The orientation of the object is defined as the tilt angle between the x-axes and the axis, around which the object can be rotated with minimal inertia (i.e. the direction of the major semi-axis a). This corresponds to the eigenvector with minimal eigenvalue. In this direction the object has its biggest extension. It is calculated as follows:

$$\theta = \frac{1}{2} \arctan \frac{2\mu_{1,1}}{\mu_{2,0} - \mu_{0,2}} \tag{16}$$

The tilt angle θ is illustrated in figure (1).

There is an ambiguity in the tilt angle θ of the object which can be resolved by choosing θ always to be the angle between the x-axis and the semimajor axis a, i.e. by definition $a \geq b$

Secondly, we pick the principal value of the arc tangent such that $-\frac{\pi}{2} \leq \arctan x \leq \frac{\pi}{2}$

With this results for the tilt angle θ are arrived that are given in Table 1.

3.4 Roundness κ and Eccentricity ε

Further parameters to describe the form of the object are the roundness κ and eccentricity ε. Both of give a measure for the roundness of the considered object.

The roundness κ can easily be calculated by dividing the square of the perimeter p with the area A:

$$\kappa = \frac{p^2}{A} \tag{17}$$

Because a circle has the maximal Area A within a given perimeter p, a scaling of roundness κ is performed:

$$\kappa = \frac{p^2}{2\pi A} \tag{18}$$

Therefore κ for a circle is equal 1, for other objects > 1.

The eccentricity ε can directly derived from the semi-major and semi-minor axes a and b of the object:

$$\varepsilon = \frac{\sqrt{a^2 - b^2}}{a} \tag{19}$$

ε can be directly calculated from the central moments of second order by

$$\varepsilon = \frac{(\mu_{2,0} - \mu_{0,2})^2 - 4\mu_{1,1}^2}{(\mu_{2,0} + \mu_{0,2})^2} \tag{20}$$

$\mu_{2,0} - \mu_{0,2}$	$\mu_{1,1}$	θ	
Zero	Zero	0^o	
Zero	Positive	$+45^o$	
Zero	Negative	-45^o	
Positive	Zero	0^o	
Negative	Zero	-90^o	
Positive	Positive	$\frac{1}{2} \arctan \frac{2\mu_{1,1}}{\mu_{2,0}-\mu_{0,2}}$	$0^o < \theta < 45^o$
Positive	Negative	$\frac{1}{2} \arctan \frac{2\mu_{1,1}}{\mu_{2,0}-\mu_{0,2}}$	$-45^o < \theta < 0^o$
Negative	Positive	$\frac{1}{2} \arctan \frac{2\mu_{1,1}}{\mu_{2,0}-\mu_{0,2}} + 90^o$	$45^o < \theta < 90^o$
Negative	Negative	$\frac{1}{2} \arctan \frac{2\mu_{1,1}}{\mu_{2,0}-\mu_{0,2}} - 90^o$	$-90^o < \theta < -45^o$

Table 1: Tilt angle θ (orientation) of an image ellipsis

The eccentricity ε can have values from 0 to 1. It is 0 with a perfectly round object and 1 by a line shaped object.

The eccentricity ε is a better measure than the roundness κ of the object, because it has a clearly defined range of values and therefore it can be compared much better.

References

[Jähne, 1997] Jähne, B. (1997). *Digitale Bildverarbeitung*. Springer-Verlag, Berlin-Heidelberg, 4. edition.

[Teague, 1980] Teague, M. R. (1980). Image analysis via the general theory of moments. *Journal of the Optical Society of America*, 70(8):920–930.