University of Idaho
Department of Computer Science
Coeur d'Alene

Lecture #13

Contours, Image Moments, and Hough Transforms

Garrett Wells
revised September 17, 2024

2024-09-17

Lecture #13

Lecture #13
Contours, Image Moments, and Hough Transforms
Garrett Wells
revised September 17, 2024

Table of Contents I

- Table of Contents
- Review
- Contours
- Image Moments
  - Raw Moments
  - Central Image Moments
- Hough Transforms

2024-09-17

Lecture #13
└Table of Contents

└Table of Contents

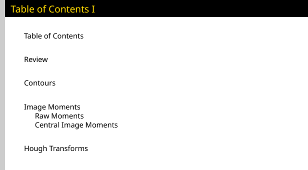


Table of Contents I

Table of Contents

Review

Contours

Image Moments
Raw Moments
Central Image Moments

Hough Transforms

Review

Edge Detection

1. Sobel

Review

Edge Detection

1. Sobel

Review

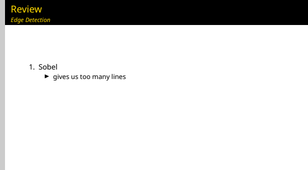
Edge Detection

1. Sobel
   - gives us too many lines

2024-09-17

Lecture #13
└Review

└Review


Review
Edge Detection

1. Sobel
   - gives us too many lines

Review

Edge Detection

1. Sobel
2. Canny

2024-09-17

Lecture #13
└Review

└Review

Review
Edge Detection

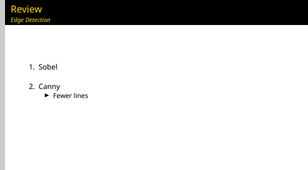
1. Sobel
2. Canny

Review

Edge Detection

1. Sobel

2. Canny
   - Fewer lines



Review

Edge Detection

1. Sobel

2. Canny
   - Fewer lines

Review

Edge Detection

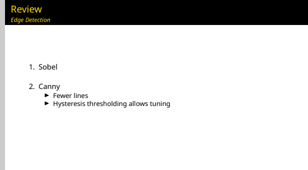
1. Sobel

2. Canny
   - Fewer lines
   - Hysteresis thresholding allows tuning

2024-09-17

Lecture #13
└Review

└Review



Review
Edge Detection

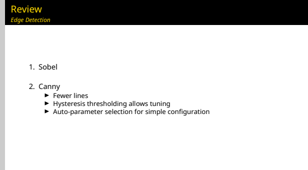
1. Sobel

2. Canny
   - Fewer lines
   - Hysteresis thresholding allows tuning

Review

Edge Detection

1. Sobel
2. Canny
   - Fewer lines
   - Hysteresis thresholding allows tuning
   - Auto-parameter selection for simple configuration



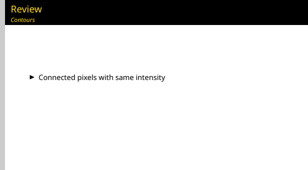
Review
Edge Detection

1. Sobel
2. Canny
   - Fewer lines
   - Hysteresis thresholding allows tuning
   - Auto-parameter selection for simple configuration

Review

Contours

- Connected pixels with same intensity

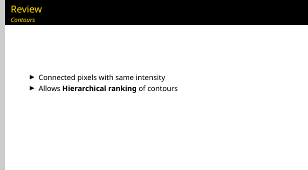


Review
Contours
- Connected pixels with same intensity

Review

Contours

- Connected pixels with same intensity
- Allows **Hierarchical ranking** of contours



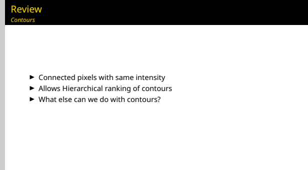
Review
Contours

- Connected pixels with same intensity
- Allows **Hierarchical ranking** of contours

Review

Contours

- Connected pixels with same intensity
- Allows Hierarchical ranking of contours
- What else can we do with contours?



Review
Contours
- Connected pixels with same intensity
- Allows Hierarchical ranking of contours
- What else can we do with contours?

Contours

We can. . .

Contours

We can. . .

Contours

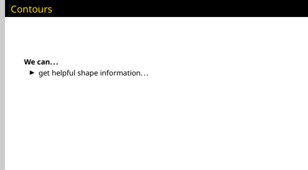
We can. . .

- get helpful shape information. . .

2024-09-17

Lecture #13
└Contours

└Contours



Contours

We can. . .

- get helpful shape information. . .

Contours

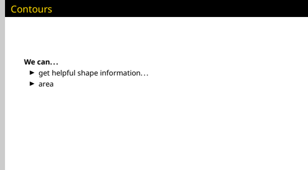
We can…

- get helpful shape information…
- area

2024-09-17

Lecture #13
└Contours

└Contours



Contours

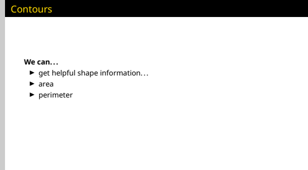
We can…

- get helpful shape information…
- area

Contours

We can…

- get helpful shape information…
- area
- perimeter



Contours

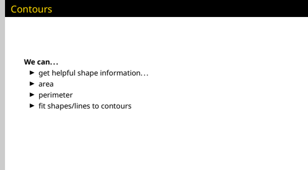
We can…

- get helpful shape information…
- area
- perimeter

Contours

We can...

- get helpful shape information...
- area
- perimeter
- fit shapes/lines to contours



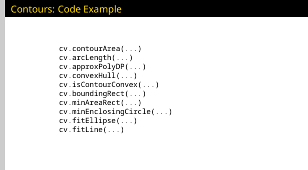
Contours

We can...

- get helpful shape information...
- area
- perimeter
- fit shapes/lines to contours

Contours: Code Example

```
cv.contourArea(...)
cv.arcLength(...)
cv.approxPolyDP(...)
cv.convexHull(...)
cv.isContourConvex(...)
cv.boundingRect(...)
cv.minAreaRect(...)
cv.minEnclosingCircle(...)
cv.fitEllipse(...)
cv.fitLine(...)
```



Contours: Code Example

cv.contourArea(...)
cv.arcLength(...)
cv.approxPolyDP(...)
cv.convexHull(...)
cv.isContourConvex(...)
cv.boundingRect(...)
cv.minAreaRect(...)
cv.minEnclosingCircle(...)
cv.fitEllipse(...)
cv.fitLine(...)

Image Moments

Introduction. . .

Image Moment [1]

A weighted average of pixel intensities, chosen to have some attractive property or interpretation.

Considers:

- considers pixel intensity
- location

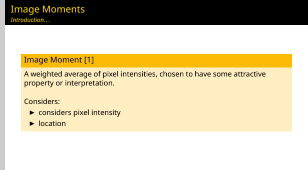


Image Moments
Introduction. . .
Image Moment [1]
A weighted average of pixel intensities, chosen to have some attractive property or interpretation.
Considers:
- considers pixel intensity
- location

1. Considering location & intensity means that we are deriving information that can be used to identify or compare objects.

Image Moments

Introduction...

Image Moment [1]

A weighted average of pixel intensities, chosen to have some attractive property or interpretation.

Considers:

- considers pixel intensity
- location

Types:

- Raw Moments
- Central Moments

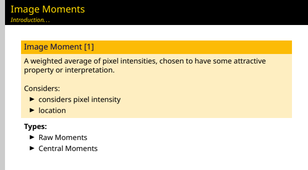


Image Moments
Introduction...

Image Moment [1]

A weighted average of pixel intensities, chosen to have some attractive property or interpretation.

Considers:

- considers pixel intensity
- location

Types:

- Raw Moments
- Central Moments

1. Considering location & intensity means that we are deriving information that can be used to identify or compare objects.

Moments

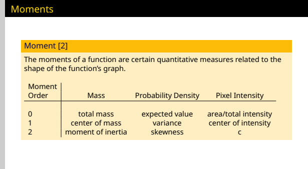
Moment [2]

The moments of a function are certain quantitative measures related to the shape of the function's graph.

Moment Order	Mass	Probability Density	Pixel Intensity
0	total mass	expected value	area/total intensity
1	center of mass	variance	center of intensity
2	moment of inertia	skewness	c

2024-09-17

Lecture #13
└Image Moments
└Moments



Moments

Moment [2]

The moments of a function are certain quantitative measures related to the shape of the function's graph.

Moment Order	Mass	Probability Density	Pixel Intensity
0	total mass	expected value	area/total intensity
1	center of mass	variance	center of intensity
2	moment of inertia	skewness	c

Raw Moments

Raw Moment [1]

For continuous functions (*order* $= p + q$):

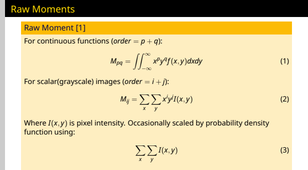
$$M_{pq} = \iint_{-\infty}^{\infty} x^p y^q f(x,y) dxdy \tag{1}$$

For scalar(grayscale) images (*order* $= i + j$):

$$M_{ij} = \sum_x \sum_y x^i y^j I(x,y) \tag{2}$$

Where $I(x,y)$ is pixel intensity. Occasionally scaled by probability density function using:

$$\sum_x \sum_y I(x,y) \tag{3}$$



Raw Moments

Raw Moment [1]

For continuous functions (*order* $= p + q$):

$$M_{pq} = \iint_{-\infty}^{\infty} x^p y^q f(x,y) dxdy \tag{1}$$

For scalar(grayscale) images (*order* $= i + j$):

$$M_{ij} = \sum_x \sum_y x^i y^j I(x,y) \tag{2}$$

Where $I(x,y)$ is pixel intensity. Occasionally scaled by probability density function using:

$$\sum_x \sum_y I(x,y) \tag{3}$$

1. LOTS of information, just know that we are (a) summing values, (b) averaging/scaling values.

Raw Moments

What can we get from a raw moment…

- area
- centroid of shape

Centroid [3]

The geometric center of a figure. It is the arithmetic mean position of all the points in the surface of the figure.

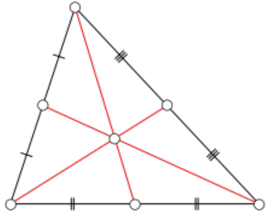
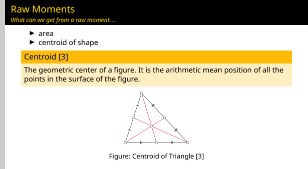


Figure: Centroid of Triangle [3]



Raw Moments

What can we get from a raw moment…

- area
- centroid of shape

Centroid [3]

The geometric center of a figure. It is the arithmetic mean position of all the points in the surface of the figure.

Figure: Centroid of Triangle [3]

Raw Moments

Centroids…

Area

$$M_{00} \tag{4}$$

Centroid

$$\{\bar{x}, \bar{y}\} = \left\{ \frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}} \right\} \tag{5}$$

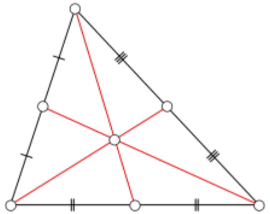
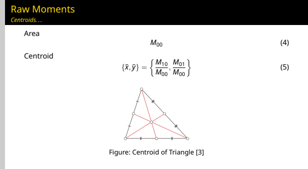


Figure: Centroid of Triangle [3]



Raw Moments
Centroids…
Area
M_{00} (4)
Centroid
$\{\bar{x}, \bar{y}\} = \left\{ \frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}} \right\}$ (5)
Figure: Centroid of Triangle [3]

1. Centroid components, $\bar{x}$ and $\bar{y}$, are used in calculating central image moments.

Central Image Moments

Central Moment [1]

For continuous functions:

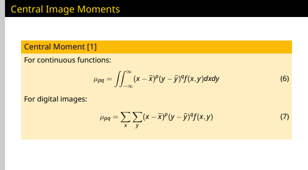
$$\mu_{pq} = \iint_{-\infty}^{\infty} (x - \bar{x})^p (y - \bar{y})^q f(x, y) dxdy \tag{6}$$

For digital images:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y) \tag{7}$$

2024-09-17

Lecture #13
└Image Moments
└Central Image Moments
└Central Image Moments



Central Image Moments

Central Moment [1]

For continuous functions:

$$\mu_{pq} = \iint_{-\infty}^{\infty} (x - \bar{x})^p (y - \bar{y})^q f(x, y) dxdy \tag{6}$$

For digital images:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y) \tag{7}$$

Central Image Moments

Properties

- Translation invariant

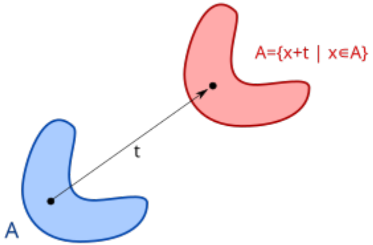
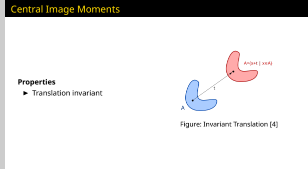


Figure: Invariant Translation [4]


Central Image Moments

Properties

- Translation invariant

Figure: Invariant Translation [4]

1. *"Translation Invariance"* or *"continuous translational symmetry"* meas that translation of the figure does not change the object [4].

Central Image Moments

Properties

- Translation **invariant**

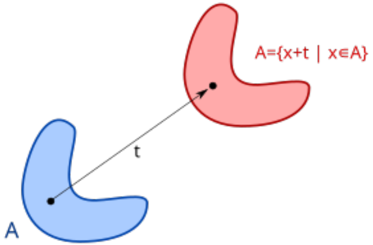
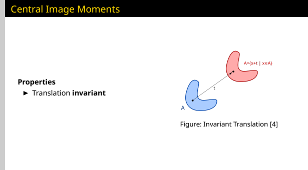


Figure: Invariant Translation [4]


Central Image Moments

Properties
- Translation **invariant**

Figure: Invariant Translation [4]

1. *“Translation Invariance”* or *“continuous translational symmetry”* meas that translation of the figure does not change the object [4].

Central Image Moments

Properties

- ► Translation invariant

Invariance

1. The condition or quality of unchanging; consistency.
2. The property of being mathematically invariant.

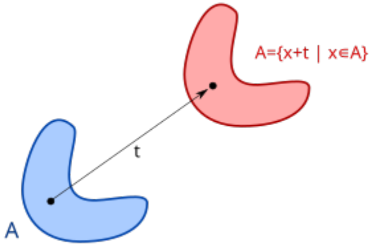
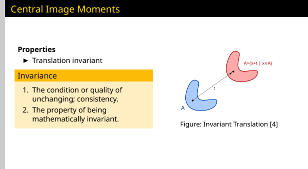


Figure: Invariant Translation [4]


Central Image Moments

Properties
► Translation invariant

Invariance
1. The condition or quality of unchanging; consistency.
2. The property of being mathematically invariant.

Figure: Invariant Translation [4]

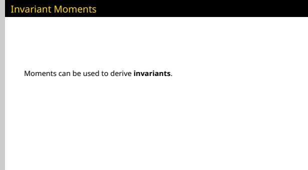
1. *"Translation Invariance"* or *"continuous translational symmetry"* meas that translation of the figure does not change the object [4].

Invariant Moments

Moments can be used to derive **invariants**.

2024-09-17

Lecture #13
└─Image Moments
  └─Central Image Moments
    └─Invariant Moments



Invariant Moments

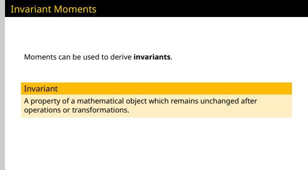
Moments can be used to derive **invariants**.

Invariant Moments

Moments can be used to derive **invariants**.

Invariant

A property of a mathematical object which remains unchanged after operations or transformations.



Invariant Moments

Moments can be used to derive **invariants**.

Invariant

A property of a mathematical object which remains unchanged after operations or transformations.

1. Invariants are appealing because they represent information about an object that doesn’t change under transformation and therefore can be used for comparison.

Invariant Moments

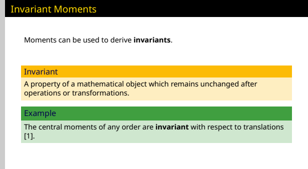
Moments can be used to derive **invariants**.

Invariant

A property of a mathematical object which remains unchanged after operations or transformations.

Example

The central moments of any order are **invariant** with respect to translations [1].



Invariant Moments

Moments can be used to derive **invariants**.

Invariant

A property of a mathematical object which remains unchanged after operations or transformations.

Example

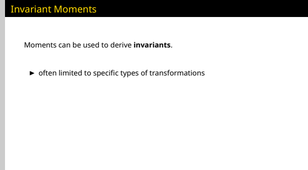
The central moments of any order are **invariant** with respect to translations [1].

1. Invariants are appealing because they represent information about an object that doesn’t change under transformation and therefore can be used for comparison.

Invariant Moments

Moments can be used to derive **invariants**.

- often limited to specific types of transformations



Invariant Moments

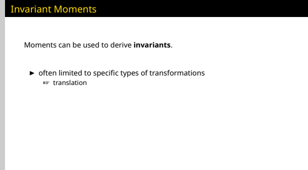
Moments can be used to derive **invariants**.

- often limited to specific types of transformations

Invariant Moments

Moments can be used to derive **invariants**.

- often limited to specific types of transformations
  - ☞ translation



Invariant Moments

Moments can be used to derive **invariants**.

- often limited to specific types of transformations
  - ☞ translation

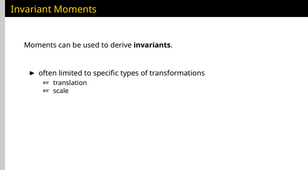
Invariant Moments

Moments can be used to derive **invariants**.

- ► often limited to specific types of transformations
  - ☞ translation
  - ☞ scale

2024-09-17

Lecture #13
└─Image Moments
  └─Central Image Moments
    └─Invariant Moments



Invariant Moments

Moments can be used to derive **invariants**.

- ► often limited to specific types of transformations
  - ☞ translation
  - ☞ scale

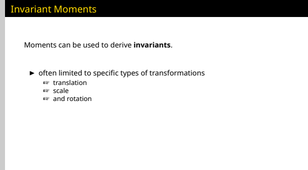
Invariant Moments

Moments can be used to derive **invariants**.

- often limited to specific types of transformations
  - ☞ translation
  - ☞ scale
  - ☞ and rotation

2024-09-17

Lecture #13
└─Image Moments
└─Central Image Moments
└─Invariant Moments



Invariant Moments

Moments can be used to derive **invariants**.

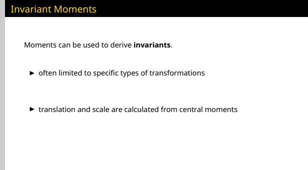
- often limited to specific types of transformations
  - ☞ translation
  - ☞ scale
  - ☞ and rotation

Invariant Moments

Moments can be used to derive **invariants**.

- often limited to specific types of transformations

- translation and scale are calculated from central moments



Invariant Moments

Moments can be used to derive **invariants**.

- often limited to specific types of transformations

- translation and scale are calculated from central moments

Invariant Moments

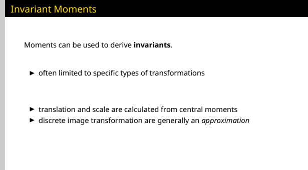
Moments can be used to derive **invariants**.

- often limited to specific types of transformations

- translation and scale are calculated from central moments
- discrete image transformation are generally an *approximation*

2024-09-17

Lecture #13
└─Image Moments
  └─Central Image Moments
    └─Invariant Moments



Invariant Moments

Moments can be used to derive **invariants**.

- often limited to specific types of transformations

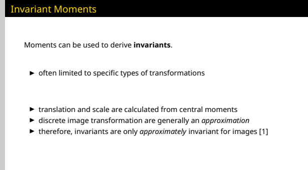
- translation and scale are calculated from central moments
- discrete image transformation are generally an *approximation*

Invariant Moments

Moments can be used to derive **invariants**.

- often limited to specific types of transformations

- translation and scale are calculated from central moments
- discrete image transformation are generally an *approximation*
- therefore, invariants are only *approximately* invariant for images [1]



Invariant Moments

Moments can be used to derive **invariants**.

- often limited to specific types of transformations

- translation and scale are calculated from central moments
- discrete image transformation are generally an *approximation*
- therefore, invariants are only *approximately* invariant for images [1]

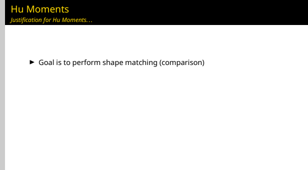
Hu Moments

Justification for Hu Moments. . .

- Goal is to perform shape matching (comparison)

2024-09-17

Lecture #13
└Image Moments
 └Central Image Moments
  └Hu Moments



Hu Moments

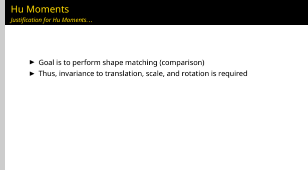
Justification for Hu Moments. . .

- Goal is to perform shape matching (comparison)

Hu Moments

Justification for Hu Moments...

- Goal is to perform shape matching (comparison)
- Thus, invariance to translation, scale, and rotation is required



Hu Moments

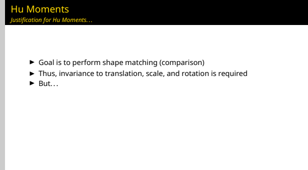
Justification for Hu Moments...

- Goal is to perform shape matching (comparison)
- Thus, invariance to translation, scale, and rotation is required

Hu Moments

Justification for Hu Moments…

- Goal is to perform shape matching (comparison)
- Thus, invariance to translation, scale, and rotation is required
- But…



Hu Moments

Justification for Hu Moments…

- Goal is to perform shape matching (comparison)
- Thus, invariance to translation, scale, and rotation is required
- But…

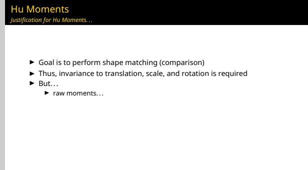
Hu Moments

Justification for Hu Moments…

- Goal is to perform shape matching (comparison)
- Thus, invariance to translation, scale, and rotation is required
- But…
  - raw moments…

2024-09-17

Lecture #13
└─Image Moments
  └─Central Image Moments
    └─Hu Moments


Hu Moments

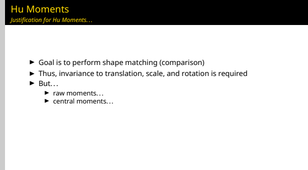
Justification for Hu Moments…

- Goal is to perform shape matching (comparison)
- Thus, invariance to translation, scale, and rotation is required
- But…
  - raw moments…

Hu Moments

Justification for Hu Moments. . .

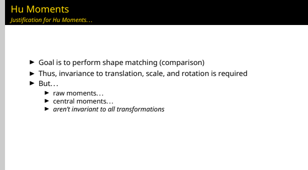
- Goal is to perform shape matching (comparison)
- Thus, invariance to translation, scale, and rotation is required
- But. . .
  - raw moments. . .
  - central moments. . .


Hu Moments
Justification for Hu Moments. . .
- Goal is to perform shape matching (comparison)
- Thus, invariance to translation, scale, and rotation is required
- But. . .
  - raw moments. . .
  - central moments. . .

Hu Moments

Justification for Hu Moments...

- Goal is to perform shape matching (comparison)
- Thus, invariance to translation, scale, and rotation is required
- But...
  - raw moments...
  - central moments...
  - *aren't invariant to all transformations*



Hu Moments

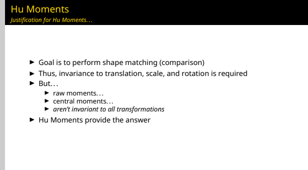
Justification for Hu Moments...

- Goal is to perform shape matching (comparison)
- Thus, invariance to translation, scale, and rotation is required
- But...
  - raw moments...
  - central moments...
  - *aren't invariant to all transformations*

Hu Moments

Justification for Hu Moments…

- Goal is to perform shape matching (comparison)
- Thus, invariance to translation, scale, and rotation is required
- But…
  - raw moments…
  - central moments…
  - *aren't invariant to all transformations*
- Hu Moments provide the answer



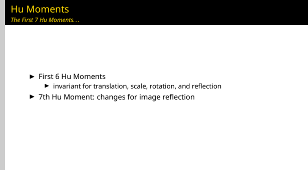
Hu Moments
Justification for Hu Moments…

- Goal is to perform shape matching (comparison)
- Thus, invariance to translation, scale, and rotation is required
- But…
  - raw moments…
  - central moments…
  - *aren't invariant to all transformations*
- Hu Moments provide the answer

Hu Moments

The First 7 Hu Moments. . .

- First 6 Hu Moments
  - invariant for translation, scale, rotation, and reflection
- 7th Hu Moment: changes for image reflection



Hu Moments
The First 7 Hu Moments. . .

- First 6 Hu Moments
  - invariant for translation, scale, rotation, and reflection
- 7th Hu Moment: changes for image reflection

Hu Moments

Calculating invariants from central moments(μ) with:

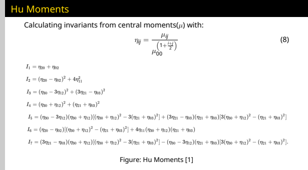
$$\eta_{ij} = \frac{\mu_{ij}}{\mu_{00}^{\left(1+\frac{i+j}{2}\right)}} \tag{8}$$

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

Figure: Hu Moments [1]

2024-09-17

Lecture #13
└Image Moments
  └Central Image Moments
    └Hu Moments


Hu Moments

Calculating invariants from central moments(μ) with:

$$\eta_{ij} = \frac{\mu_{ij}}{\mu_{00}^{\left(1+\frac{i+j}{2}\right)}} \tag{8}$$

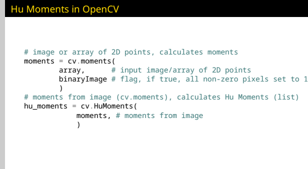
Figure: Hu Moments [1]

Hu Moments in OpenCV

```
# image or array of 2D points, calculates moments
moments = cv.moments(
        array,      # input image/array of 2D points
        binaryImage # flag, if true, all non-zero pixels set to 1
        )
# moments from image (cv.moments), calculates Hu Moments (list)
hu_moments = cv.HuMoments(
            moments, # moments from image
            )
```

2024-09-17

Lecture #13
└Image Moments
  └Central Image Moments
    └Hu Moments in OpenCV

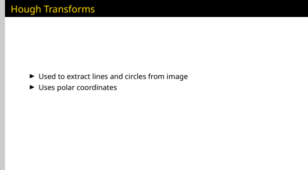


Hu Moments in OpenCV

```
# image or array of 2D points, calculates moments
moments = cv.moments(
        array,      # input image/array of 2D points
        binaryImage # flag, if true, all non-zero pixels set to 1
        )
# moments from image (cv.moments), calculates Hu Moments (list)
hu_moments = cv.HuMoments(
            moments, # moments from image
            )
```

Hough Transforms

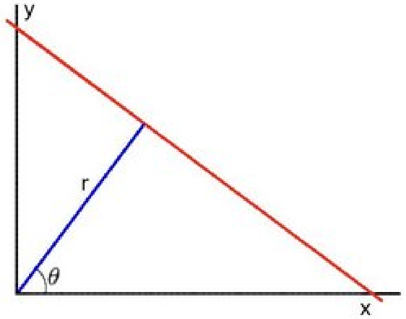
- Used to extract lines and circles from image
- Uses polar coordinates



Hough Transforms

- Used to extract lines and circles from image
- Uses polar coordinates

Hough Transforms



Cartesian Representation:

$$y = mx + b \tag{9}$$

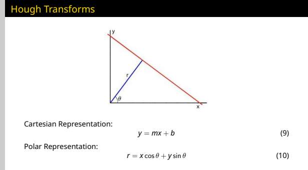
Polar Representation:

$$r = x\cos\theta + y\sin\theta \tag{10}$$

2024-09-17

Lecture #13
└Hough Transforms

└Hough Transforms



Hough Transforms

Cartesian Representation:

$$y = mx + b \tag{9}$$

Polar Representation:

$$r = x\cos\theta + y\sin\theta \tag{10}$$

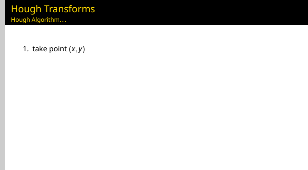
Hough Transforms

Hough Algorithm…

1. take point (x, y)

2024-09-17

Lecture #13
└Hough Transforms
└Hough Transforms



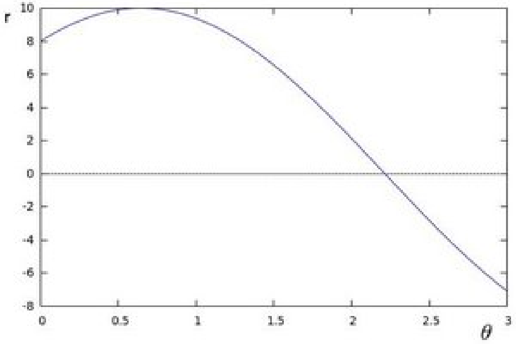
Hough Transforms
Hough Algorithm…

1. take point (x, y)

Hough Transforms

Hough Algorithm…

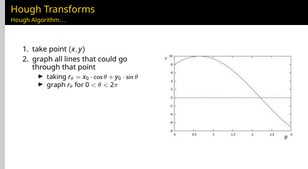
1. take point (x, y)
2. graph all lines that could go through that point
    - taking $r_\theta = x_0 \cdot \cos\theta + y_0 \cdot \sin\theta$
    - graph r_θ for $0 < \theta < 2\pi$



2024-09-17

Lecture #13
└Hough Transforms

└Hough Transforms

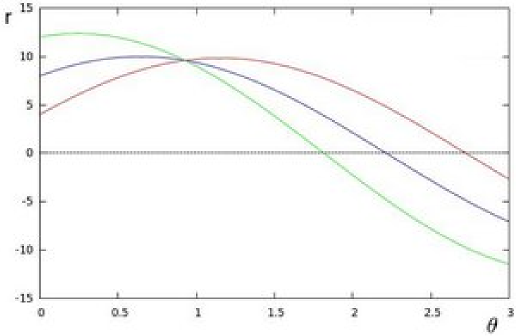

Hough Transforms
Hough Algorithm…

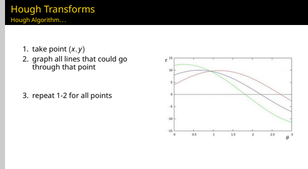
1. take point (x, y)
2. graph all lines that could go through that point
    - taking $r_\theta = x_0 \cdot \cos\theta + y_0 \cdot \sin\theta$
    - graph r_θ for $0 < \theta < 2\pi$

Hough Transforms

Hough Algorithm…

1. take point (x, y)
2. graph all lines that could go through that point
3. repeat 1-2 for all points



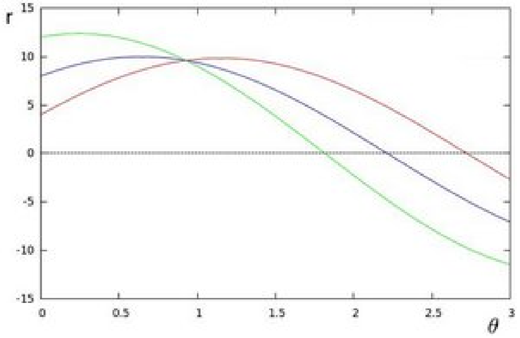

Hough Transforms
Hough Algorithm…

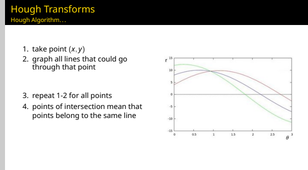
1. take point (x, y)
2. graph all lines that could go through that point
3. repeat 1-2 for all points

Hough Transforms

Hough Algorithm. . .

1. take point (x, y)
2. graph all lines that could go through that point
3. repeat 1-2 for all points
4. points of intersection mean that points belong to the same line

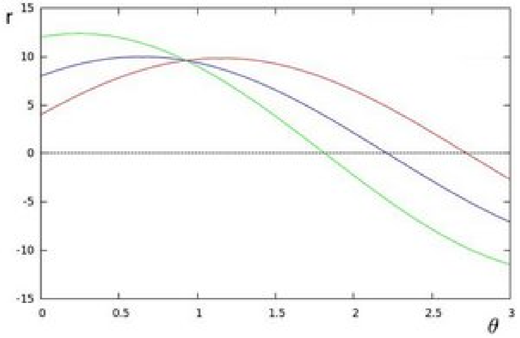


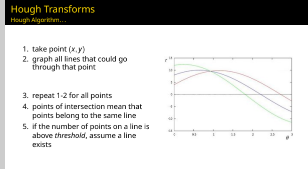

Hough Transforms
Hough Algorithm. . .
1. take point (x, y)
2. graph all lines that could go through that point
3. repeat 1-2 for all points
4. points of intersection mean that points belong to the same line

Hough Transforms

Hough Algorithm…

1. take point (x, y)
2. graph all lines that could go through that point
3. repeat 1-2 for all points
4. points of intersection mean that points belong to the same line
5. if the number of points on a line is above *threshold*, assume a line exists




Hough Transforms
Hough Algorithm…

1. take point (x, y)
2. graph all lines that could go through that point
3. repeat 1-2 for all points
4. points of intersection mean that points belong to the same line
5. if the number of points on a line is above *threshold*, assume a line exists

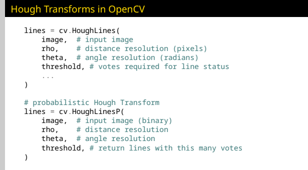
Hough Transforms in OpenCV

```
lines = cv.HoughLines(
    image,  # input image
    rho,    # distance resolution (pixels)
    theta,  # angle resolution (radians)
    threshold, # votes required for line status
    ...
)

# probabilistic Hough Transform
lines = cv.HoughLinesP(
    image,  # input image (binary)
    rho,    # distance resolution
    theta,  # angle resolution
    threshold, # return lines with this many votes
)
```

2024-09-17

Lecture #13
└Hough Transforms
  └Hough Transforms in OpenCV



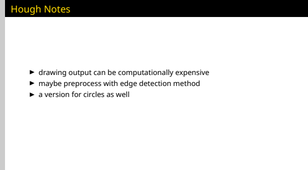
Hough Transforms in OpenCV

```
lines = cv.HoughLines(
    image,  # input image
    rho,    # distance resolution (pixels)
    theta,  # angle resolution (radians)
    threshold, # votes required for line status
    ...
)

# probabilistic Hough Transform
lines = cv.HoughLinesP(
    image,  # input image (binary)
    rho,    # distance resolution
    theta,  # angle resolution
    threshold, # return lines with this many votes
)
```

Hough Notes

- drawing output can be computationally expensive
- maybe preprocess with edge detection method
- a version for circles as well



Hough Notes

- drawing output can be computationally expensive
- maybe preprocess with edge detection method
- a version for circles as well

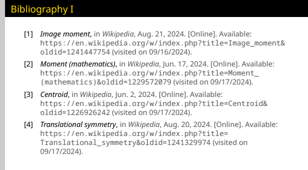
Bibliography I

[1] *Image moment*, in *Wikipedia*, Aug. 21, 2024. [Online]. Available: `https://en.wikipedia.org/w/index.php?title=Image_moment&oldid=1241447754` (visited on 09/16/2024).

[2] *Moment (mathematics)*, in *Wikipedia*, Jun. 17, 2024. [Online]. Available: `https://en.wikipedia.org/w/index.php?title=Moment_(mathematics)&oldid=1229572079` (visited on 09/17/2024).

[3] *Centroid*, in *Wikipedia*, Jun. 2, 2024. [Online]. Available: `https://en.wikipedia.org/w/index.php?title=Centroid&oldid=1226926242` (visited on 09/17/2024).

[4] *Translational symmetry*, in *Wikipedia*, Aug. 20, 2024. [Online]. Available: `https://en.wikipedia.org/w/index.php?title=Translational_symmetry&oldid=1241329974` (visited on 09/17/2024).



Bibliography I

[1] *Image moment*, in *Wikipedia*, Aug. 21, 2024. [Online]. Available: `https://en.wikipedia.org/w/index.php?title=Image_moment&oldid=1241447754` (visited on 09/16/2024).

[2] *Moment (mathematics)*, in *Wikipedia*, Jun. 17, 2024. [Online]. Available: `https://en.wikipedia.org/w/index.php?title=Moment_(mathematics)&oldid=1229572079` (visited on 09/17/2024).

[3] *Centroid*, in *Wikipedia*, Jun. 2, 2024. [Online]. Available: `https://en.wikipedia.org/w/index.php?title=Centroid&oldid=1226926242` (visited on 09/17/2024).

[4] *Translational symmetry*, in *Wikipedia*, Aug. 20, 2024. [Online]. Available: `https://en.wikipedia.org/w/index.php?title=Translational_symmetry&oldid=1241329974` (visited on 09/17/2024).