University of Applied Sciences
Department of
Design, Computer Science, Media
Prof. Dr. D. Richter

Transilvania University Brasov
Faculty of
Mathematics and Informatics
Dr. I. Plajer

ISIP 2019
International School on Image Processing
July 15th - 26th, 2019

detlef.richter@HS-RM.de

University of Applied Sciences
Department of
Design, Computer Science, Media
Prof. Dr. D. Richter

Lectures in Image Processing

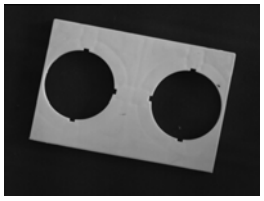
Chapter 7
Hough Transform
(Part 1 : HT for Lines)

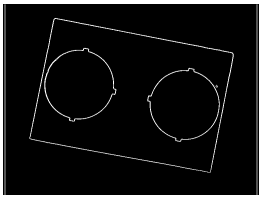
7. Hough Transform

- Graphical data processing : Data → Image
- Digital image manipulation : Image → Image
- Digital image processing : Image → Data

7.0 Hough Transform

7.0 Hough Transform, Goal of Application (1)





The following equations are to be found:
line 1: $y = m_0 * x + n_0$ or $x * \cos \alpha_0 * y * \sin \alpha_0 = r_0$
lines 2 to 4 as well with afterwards known m_0, n_0 or α_0 and r_0.
circle 1: $(x - x_m)^2 + (y - y_m)^2 = r^2$
circle 2 as well with afterwards known x_m, y_m, and r.

7.0 Hough Transform, Goal of Application (2)

Equations of analytical defined shapes.

- orientation of the parabola (1 parameter),
- the focal length f = p/2 or width w = 2p (1 parameter),
- vertex (2 parameters).

Use a-priori knowledge to reduce the number of parameters

7.0 Hough Transform, Goal of Application (2)

Focal length f = p/2 or width w = 2p (1 parameter):

$y^2 = 2\,p\,x$

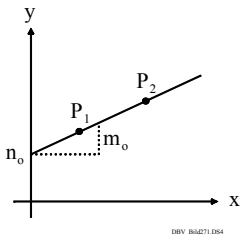
Vertex (2 parameters):

$(y - y_0)^2 = 2\,p\,(x - x_0)$

Orientation of the parabola α (1 parameter):

$$\begin{pmatrix} y' \\ x' \end{pmatrix} = R(\alpha) * \begin{pmatrix} y \\ x \end{pmatrix}$$

7.1 Hough Transform for lines



Equation of straight line :

$y = m_o * x + n_o$

x, y variable; m_o, n_o const.

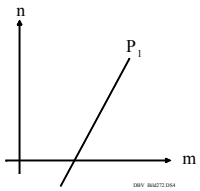
m_o : slope

n_o : interception of y-axis

Consider point P_1 :

$y_1 = m_o * x_1 + n_o$

7.1 Hough Transform for lines



The values of n depend on the slopes m for all lines passing point P_1 of a straight line g_1 :

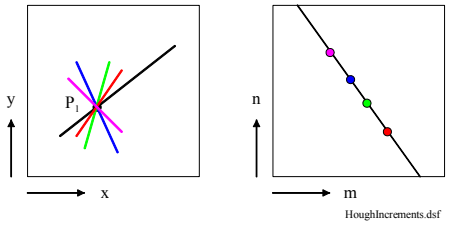
$g_1 : n = - x_1 * m + y_1$,

n, m : variable; x_1, y_1 : const.

$-x_1$: slope;
y_1 : intersection of n-axis

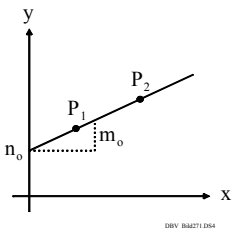
Line g_1 in the m, n-diagram represents point P_1 of x, y-diagram

7.1 Hough Transform for lines



Only one white point P_1 is known. Calculate all potential lines passing P_1. Register all pairs (m,n) in a matrix.

7.1 Hough Transform for lines



Consider point P_2 :

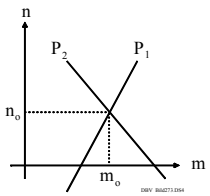
$y_2 = m_o * x_2 + n_o$

Representation of point P_2 within n,m-diagram :

$g_2 : n = - x_2 * m + y_2$,

n, m : variable; x_2, y_2 : const.

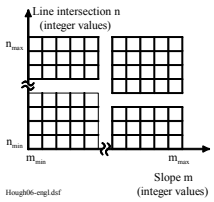
7.1 Hough Transform for lines



The intersecting point (m_o,n_o) of the straight lines g_1 und g_2 represents that line within the x,y-diagram, which passes the points P_1 and P_2 as well.

Same consideration holds for all points P_n on line $y=m_0*x+n_0$ → straight lines in n,m-diagram passing the point m_0,n_0

7.1 Hough Transform for lines



Define m, n-matrix

Find white pixels $P_i(x_i,y_i)$ in the image assumed to lie on lines.

Calculate all values n, $n = - x_i * m + y_i$, for given x_i,y_i with independent variable m [m : integer].

Increment according matrix elements m, n for all combinations m, n of possible lines passing point $P_i(x_i, y_i)$.

m, n : integer values

7.1 Hough Transform for lines

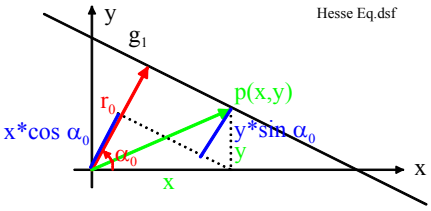
What happens with lines parallel to image columns ?

In this case : $m = \pm\infty$ und $n = \pm\infty$

∞ : not computable in computers.

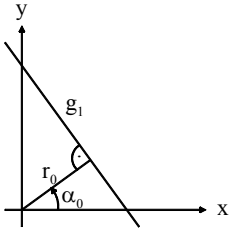
Look for another analytical equation for straight lines in which infinite values don't occur.

7.1 Hough Transform for lines, Hesse Equation



$x * \cos \alpha_0 + y * \sin \alpha_0 = r_o$

7.1.1 Hough Transform for lines, Hesse Equation



Hesse equation of straight lines :

$g_1 : x * \cos \alpha_o + y * \sin \alpha_o = r_o$

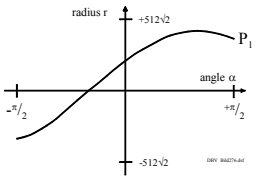
r_o : perpendicular from origin to g_1

α_o : angle between x-axis and perpendicular (math. counterclockwise)

Consider pixel $P_1(x_1,y_1)$ lying upon g_1 :

$x_1 * \cos \alpha_o + y_1 * \sin \alpha_o = r_o$

7.1.1 Hough Transform for lines, Hesse Equation



Consider 512² -image.

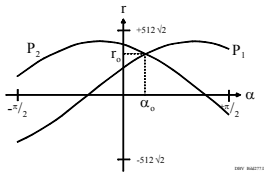
Consider
$x_1 * \cos \alpha + y_1 * \sin \alpha = r$
with α and r as variables

All possible lines passing P_1 are calculated within a range of $-90° \le \alpha < +90°$

Limiting α to an interval of ±90° results in positive and negative radii.

Maximal radius : $\pm 512*\sqrt{2}$

7.1.1 Hough Transform for lines, Hesse Equation



For pixel $P_2(x_2,y_2)$ onto g_1 :

$x_2 * \cos\alpha + y_2 * \sin\alpha = r$

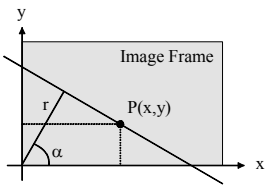
Both functions intersect in point (α_o, r_o).

Proceed with all points onto g_1.

Increment Hough matrix [r][α]

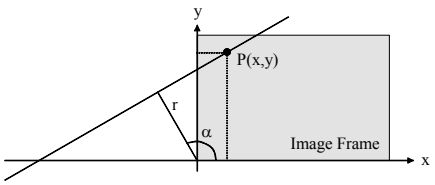
7.1.1 Hough Transform for lines, Hesse Equation

Consider the following situations :



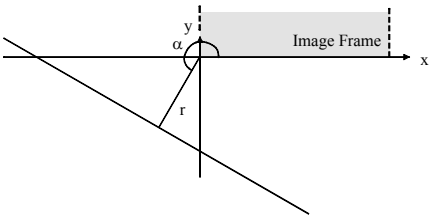
α in 1st quadrant, r positive, line visible in image

7.1.1 Hough Transform for lines, Hesse Equation



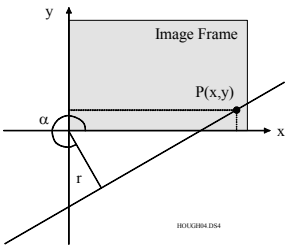
α in 2nd quadrant, r positive , line visible in image

7.1.1 Hough Transform for lines, Hesse Equation



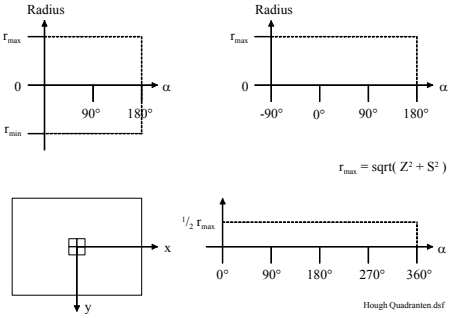
α in 3rd quadrant, r positive, line *not visible* in image

7.1.1 Hough Transform for lines, Hesse Equation



α in 4th quadrant, r positive , line visible in image

7.1.2 Hough Transform, Definition of Matrices



7.1.2 Hough Transform, Definition of Matrices

- Define Hough-Matrix for two quadrants (e.g. angles α : -90° ≤ α < +90° or 0° ≤ α < 180°) and allow positive and negative radii.
- Define Hough-Matrix for three quadrants (in this case for angles α : -90° ≤ α < +180°) and exclude negative radii.
- Define Hough-Matrix for four quadrants and allow positive radii with half of maximum size.

7.1.2 Hough Transform, Definition of Matrices

Memory space needed:

- 2 quadrants : 180 * 960 * 2 = 345 600
- 3 quadrants : 270 * 960 * 1 = 259 200
- 4 quadrants : 360 * 960 * ½ = 172 800

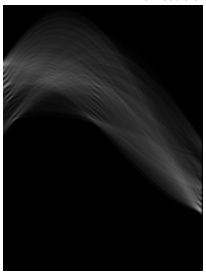
Assumed to have an angle resolution of 1° and radius resolution of 1 Pixel.

7.1.2 Hough Transform, Definition of Matrices

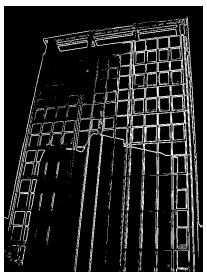
- Resolution of r and α has to be defined according to the specified problem (e.g. α in steps of 1°, 2°, 3° or 5°, radius resolution might be smaller than, equal or larger than one pixel).
- The better the resolution, the more time consuming the calculation.
- Or : Who wants to see details, needs more time to look at.

7.1.5 Hough Transform, Application

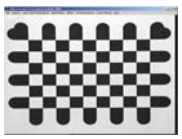
Hough matrix of two quadrants, positive and negative radii, absolute and logarithmic presentation.
How could the original image look like?

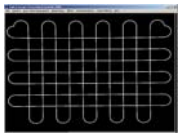


7.1.5 Hough Transform, Application



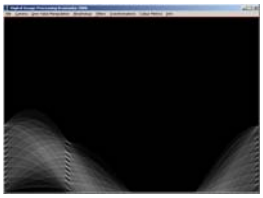
7.1.5 Hough Transform, Application





- Chess board like calibrating pattern (see chapter 6 : Radial symmetric distortion by lenses)
- High-pass filtered image

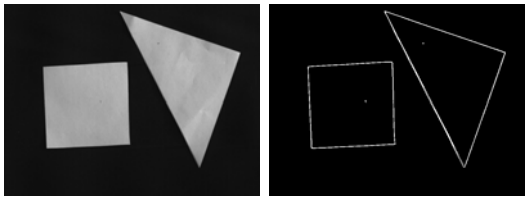
7.1.5 Hough Transform, Application



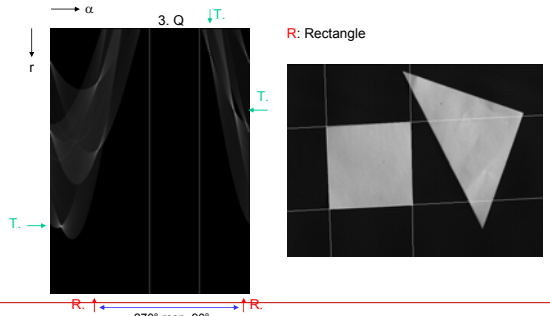
Hough matrix with 4 quadrants
(no results for quadrant 3)

- Representation of Hough-Matrix
- 10 vertical equidistant lines (plus 1 line at image boundary with larger distance) (α = 0° and 360°)
- 7 equidistant horizontal lines (α = 90°)
- Define r, α
- Calculate corresponding intersection points

7.1.5 Hough Transform, Application



7.1.5 Hough Transform, Application



7.1.5 Hough Transform, Application