R.U.R. 2010

Visual Systems of robots

Andrej Lúčny

Department of Applied Informatics

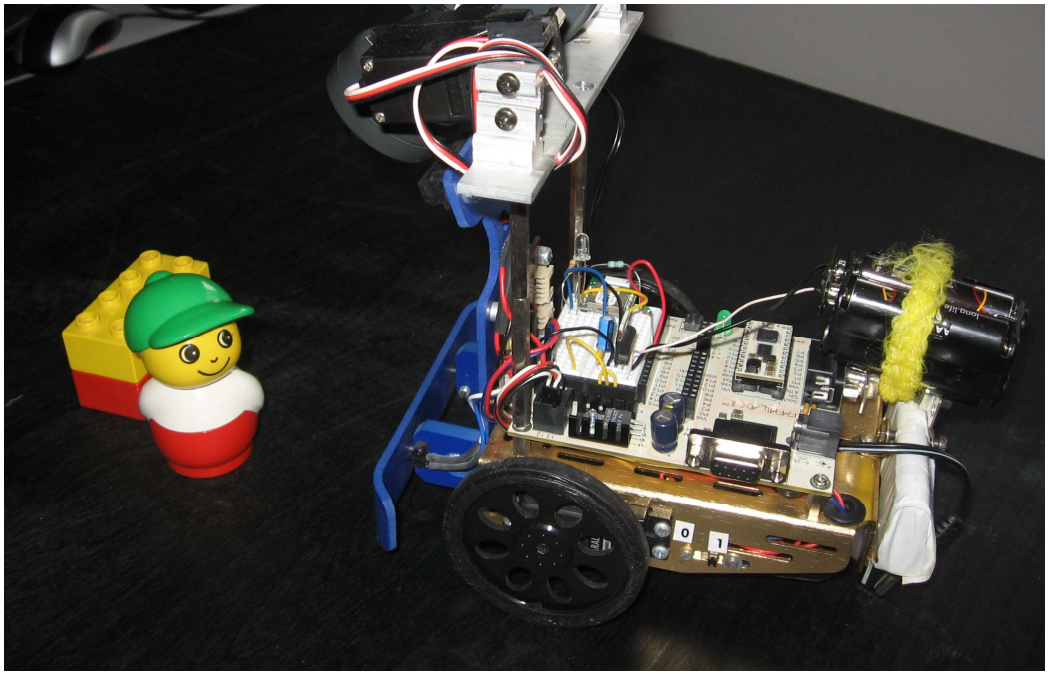
FMFI UK Bratislava

andy@microstep-mis.com

www.microstep-mis.com/~andy

Robot vision

Robot needs to generate proper actions upon what is seen



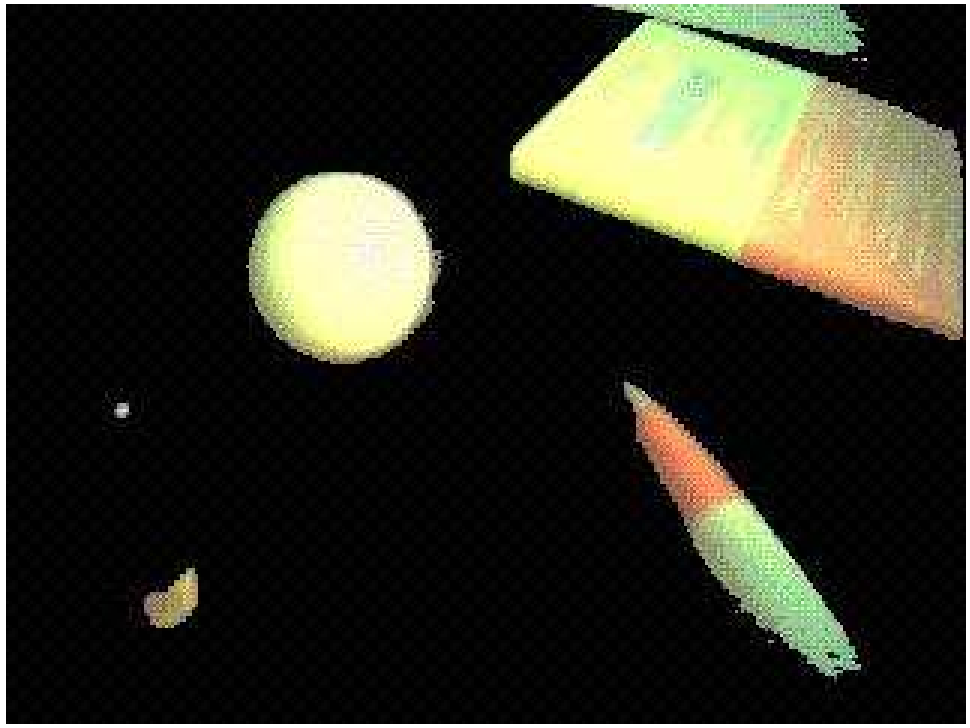
Robot Vision

- Input images
- 2D processing
- 3D processing
- Object recognition

Input images

Input image

(1,1) (1,w)

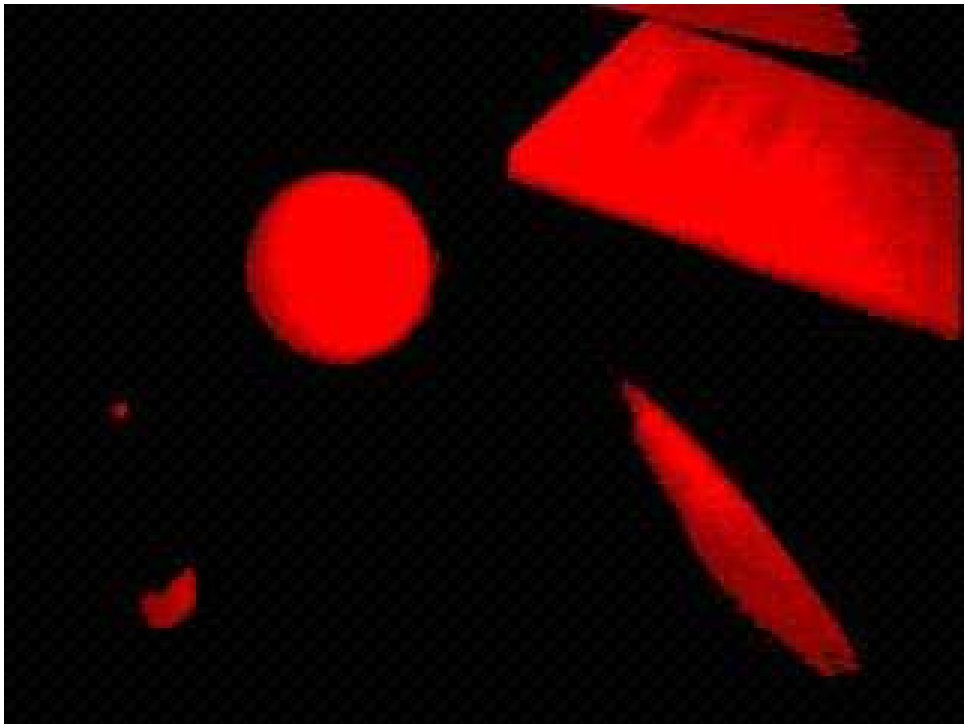


(h,1) (h,w)

Field c[h,w], each item c(x,y) is a vector containing three numbers from range 0..255, representing red, green and blue primary colors

Red ingredient

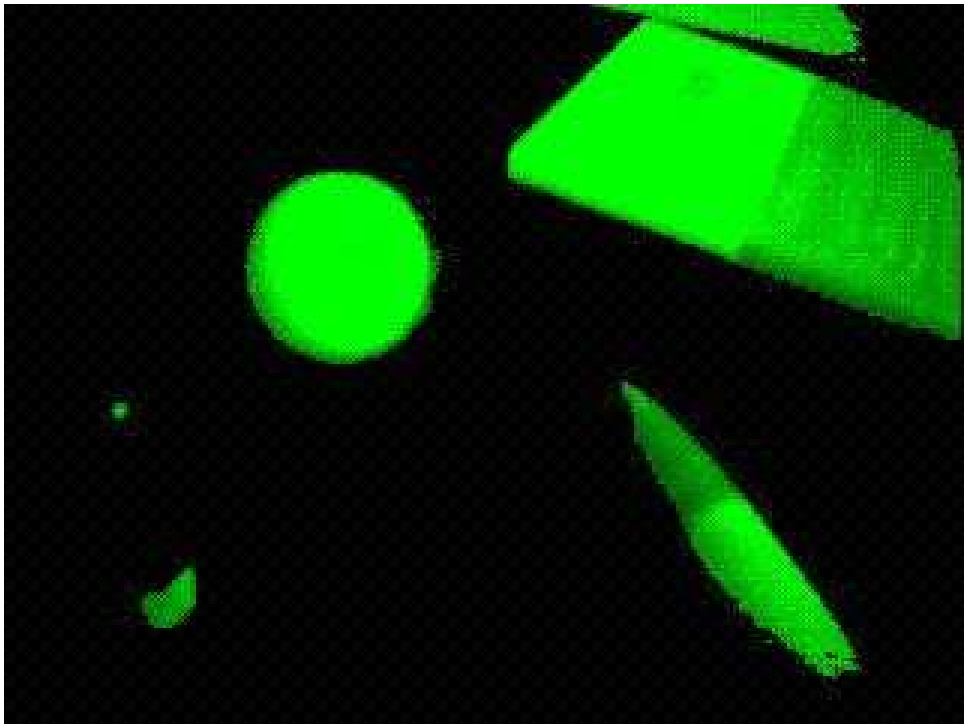
(1,1,1) (1,w,1)



(h,1,1) (h,w,1)

Green ingredient

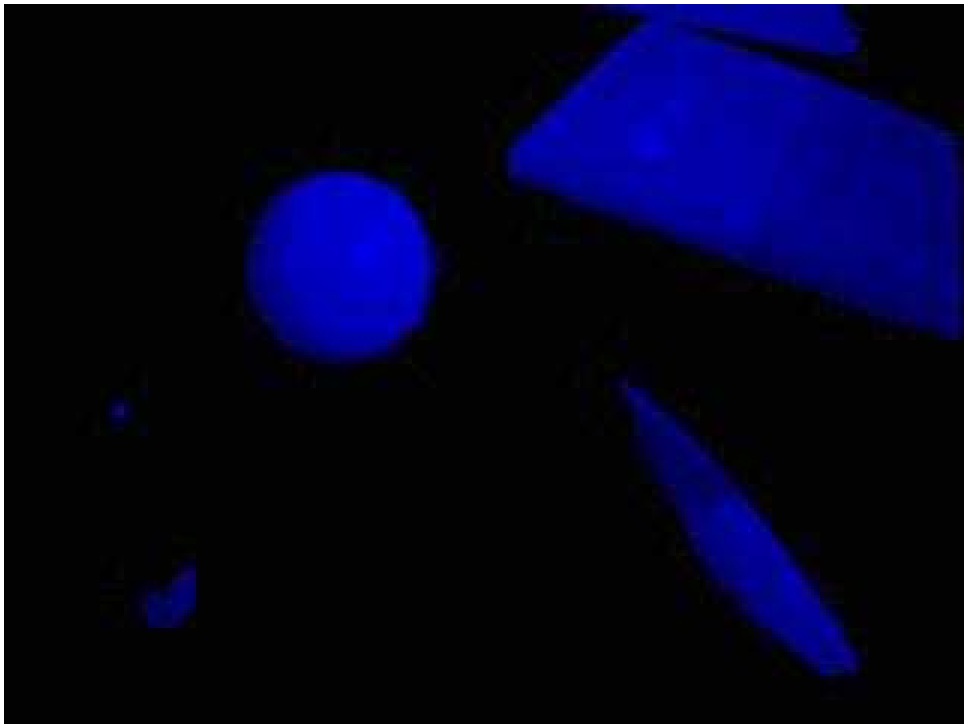
(1,1,2) (1,w,2)



(h,1,2) (h,w,2)

Blue ingredient

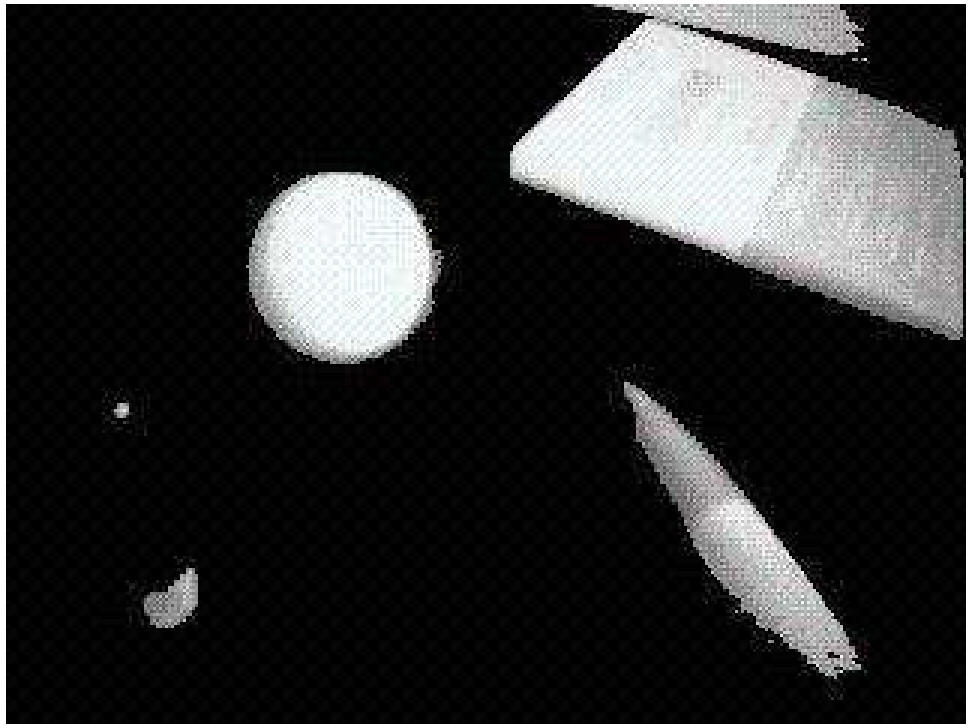
(1,1,3) (1,w,3)



(h,1,3) (h,w,3)

Grayscale image

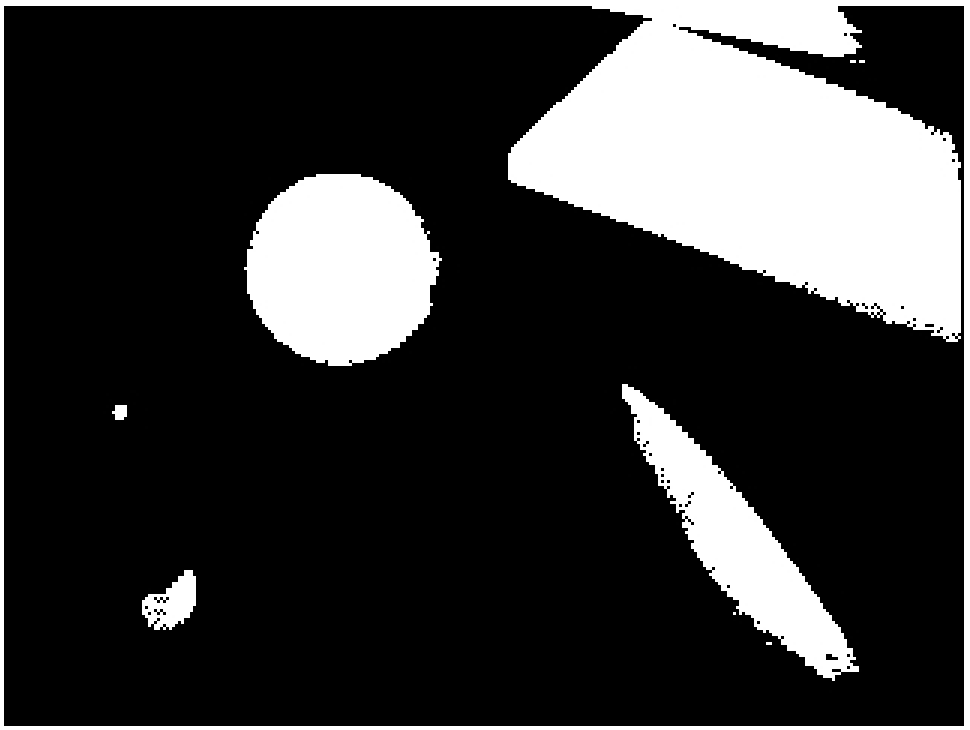
(1,1) (1,w)



(h,1) (h,w)

Field g[h,w]: each item g(x,y) is a number from range 0..255 and represents light intensity at a particular point

Binary image



Field t[h,w]: each item t(x,y) is a number 0 (black) or 255 (white) and represents presence of an object on image

Color to grayscale conversion

- For computer any linear combination of color ingredients, e.g.:

  $g(i,j) = (c(i,j,1) + c(i,j,2) + c(i,j,3)) / 3$

- For human eye the optimal coefficients are:

  $g(i,j) = 0.3*c(i,j,1) + 0.59*c(i,j,2) + 0.11*c(i,j,2)$

Grayscale to binary conversion

- Threshold

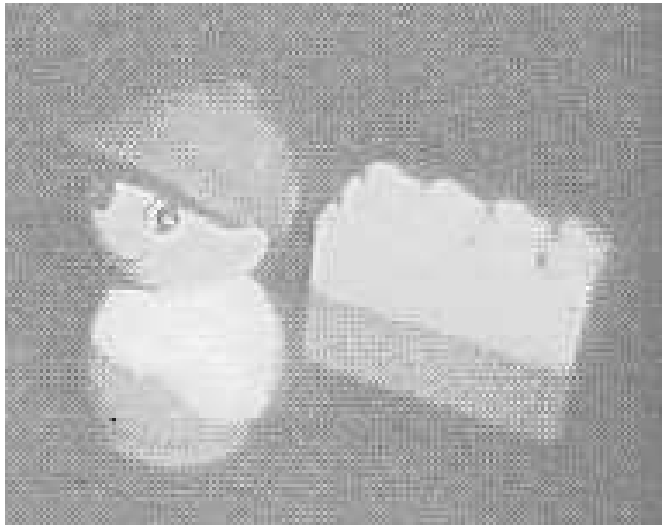
$$b(i,j) = \begin{cases} 255 & g(i,j) \geq \theta \\ 0 & g(i,j) < \theta \end{cases}$$

- Edge detectors
- Object recognizers, …

2D processing

Enhancement of grayscale image

- Blur and other Smoothing (e.g. anisotropic diffusion)



1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

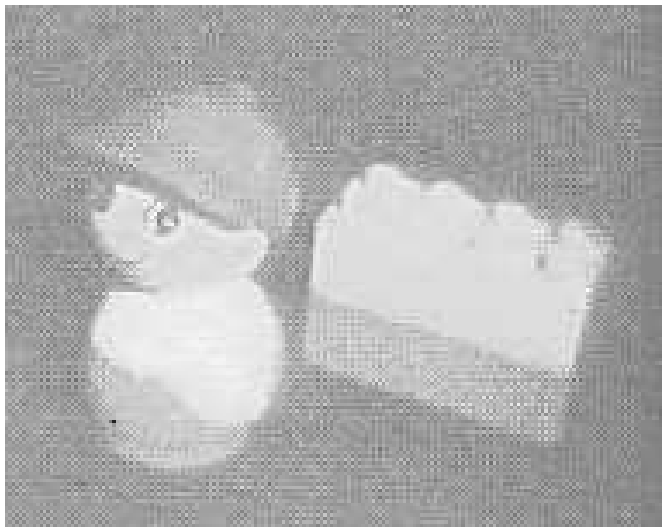
⇨

pixel is replaced by average of close pixels

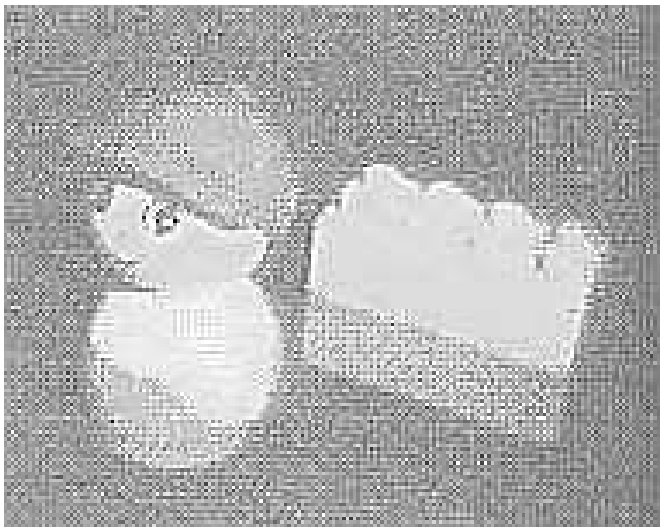
noise is decreased (better), edges are worse

Enhancement of grayscale image

- Sharpen



-1	-1	-1
-1	12	-1
-1	-1	-1

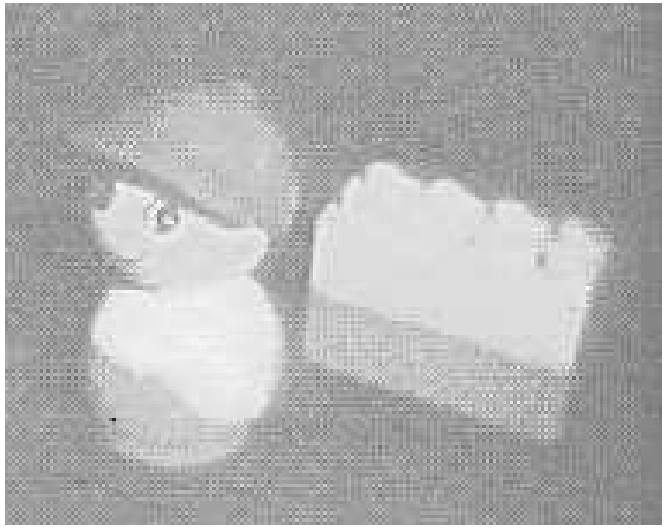


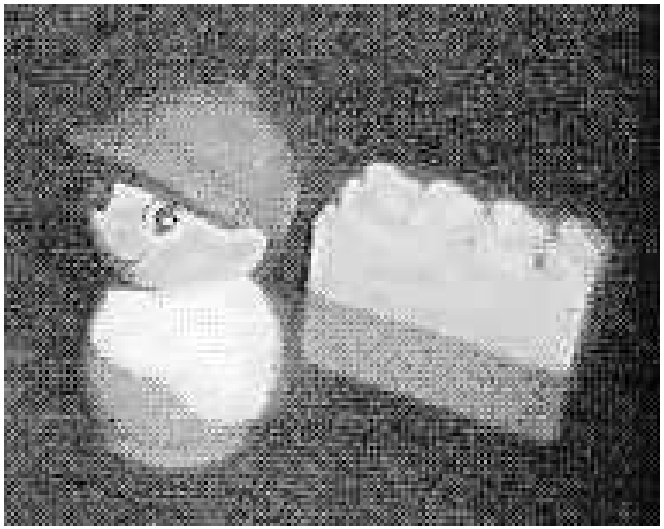
pixel with different close pixels is emphasized

noise is increased (worse), edges are better

Enhancement of grayscale image

- Brightness & Contrast

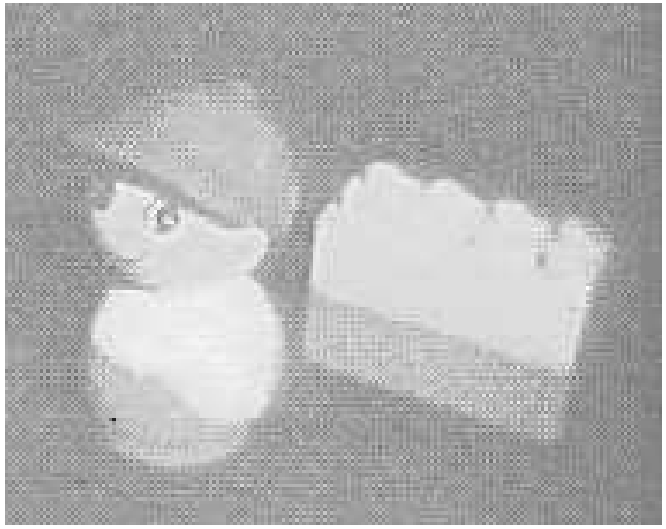




important just for us, not for computer

Enhancement of grayscale image

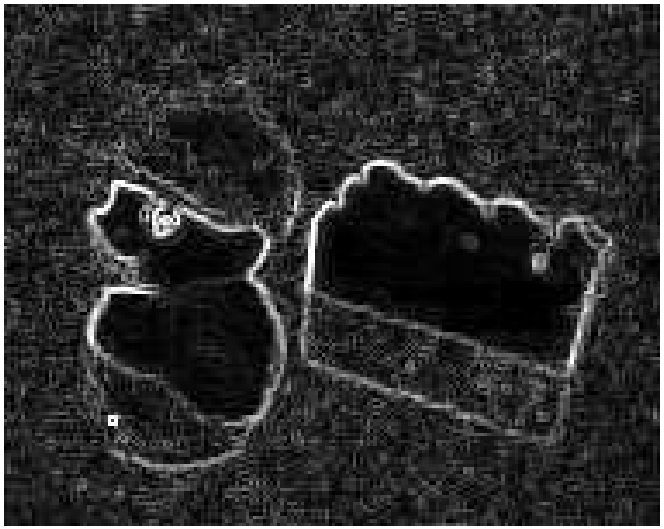
- Edge detection: Sobel operator



-1	-2	-1
0	0	0
1	2	1

⇨

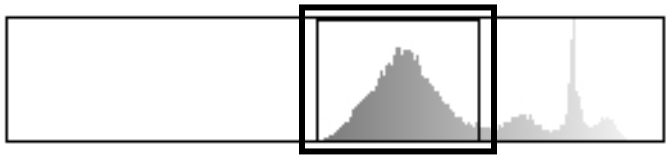
-1	0	1
-2	0	2
-1	0	1

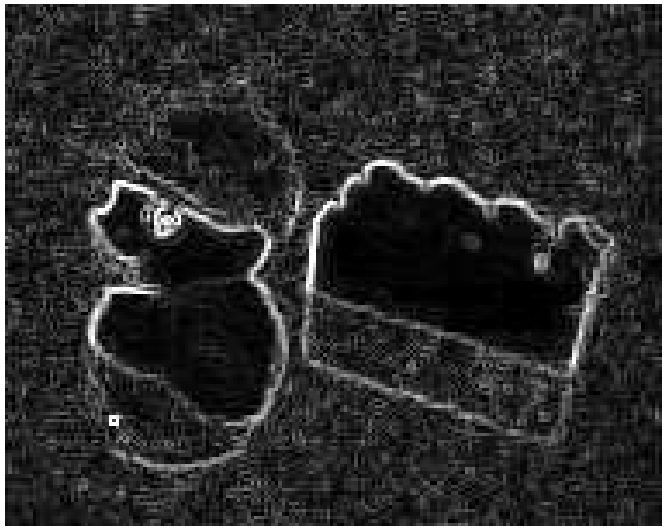


Blur + Sobel = fundament of Canny detector

Enhancement of grayscale image

- Threshold

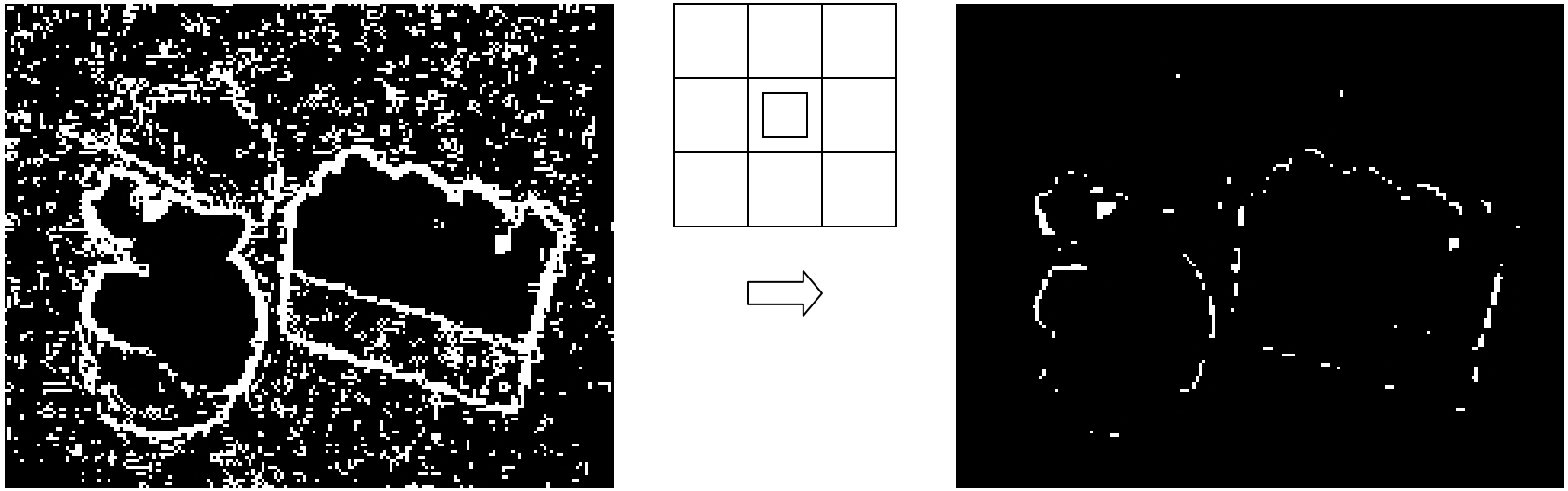




Finally, we need binary image, thus we select proper interval on histogram

Enhancement of binary image

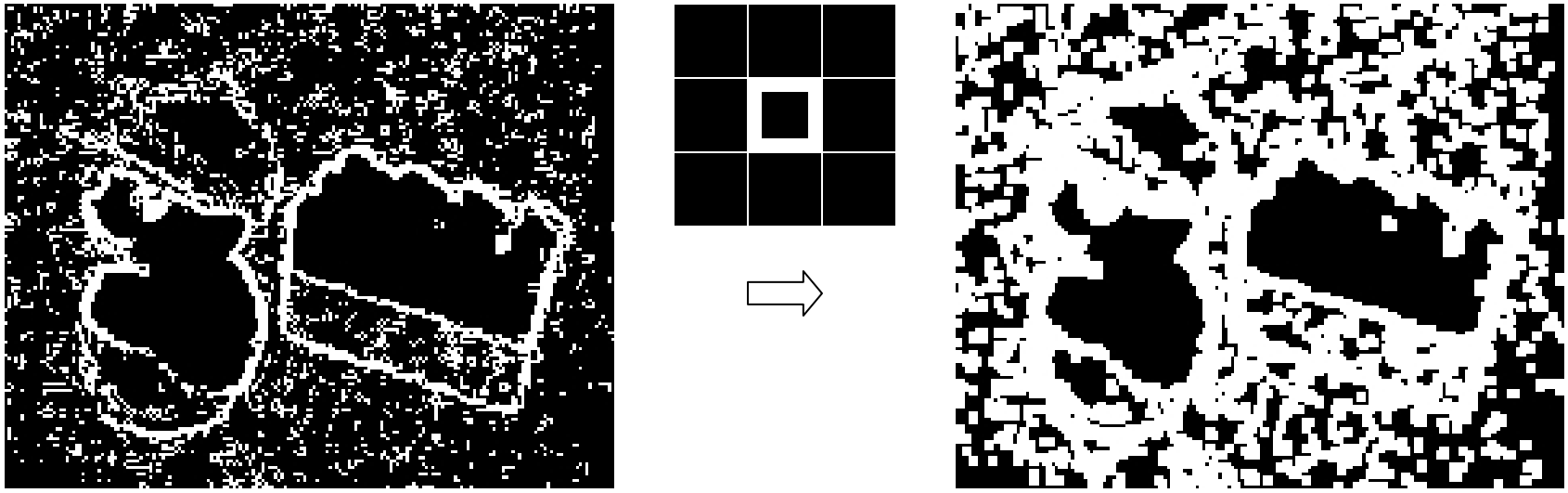
- Erosion



pixel remains white iff all close pixels are white

Enhancement of binary image

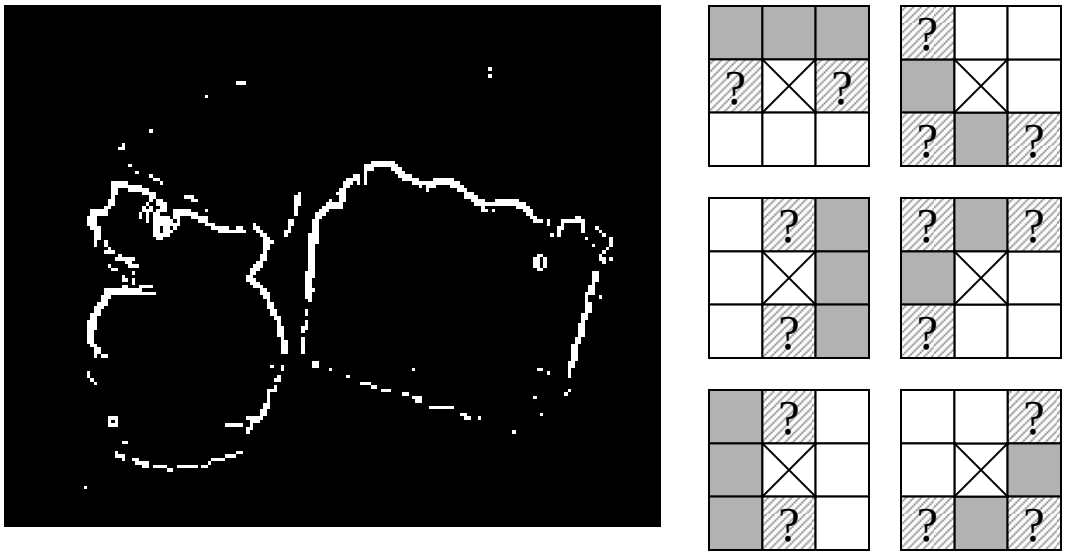
- Dilation



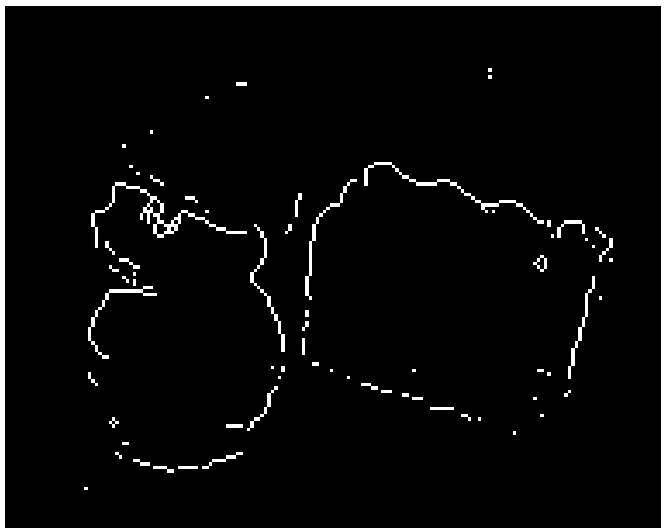
pixel becomes white iff any close pixel is white

Enhancement of binary image

- Thinning



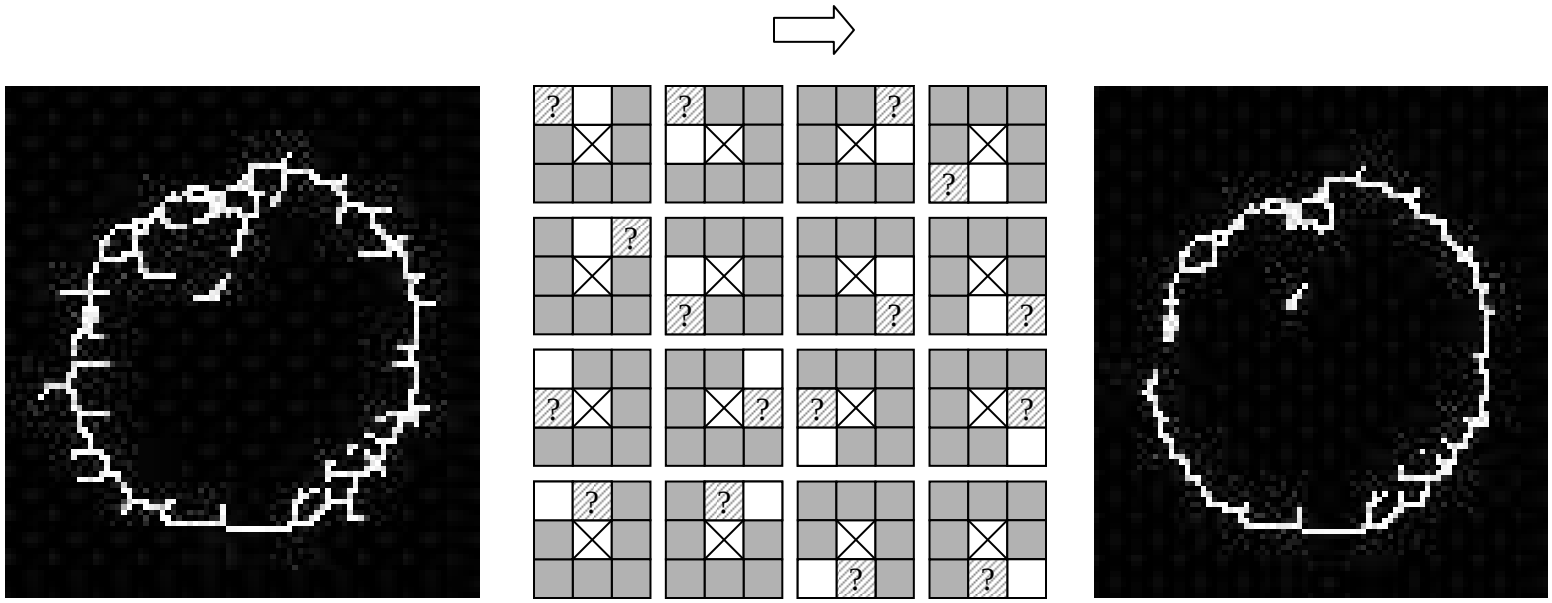
Skeleton



we reduce isolated point, horizontal and vertical lines and horns while it is possible

Enhancement of binary image

- Pruning



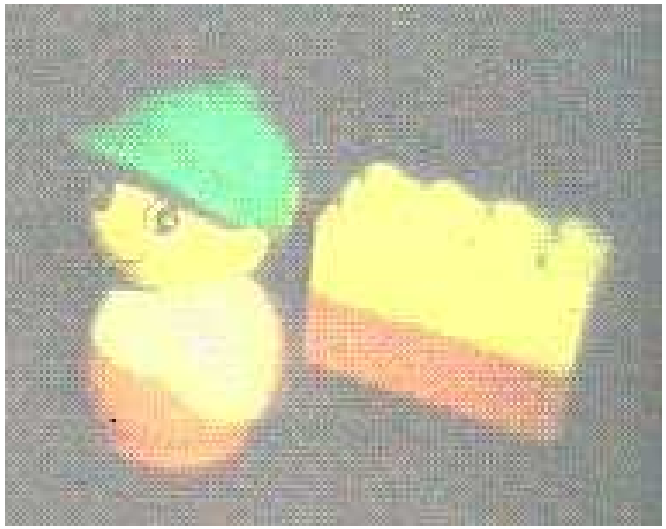
we reduce ledges while it is possible

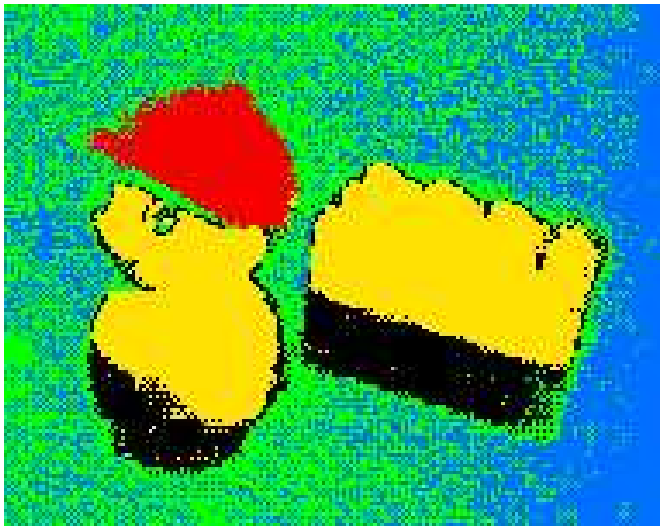
Enhancement of color image

- Segmentation

K-means clustering,

Region growing, …





256 x 256 x 256 colors

5 colors

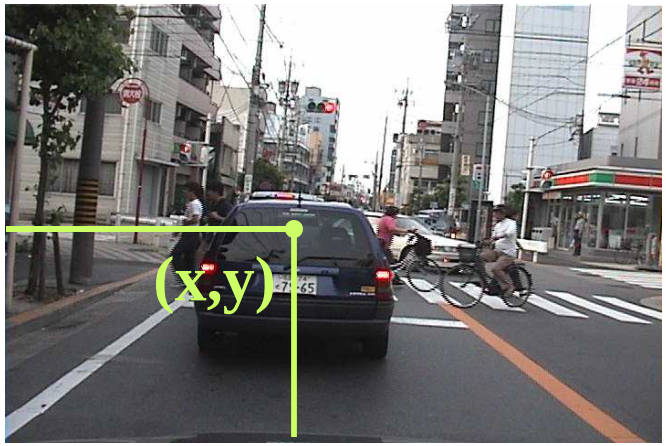
3D processing

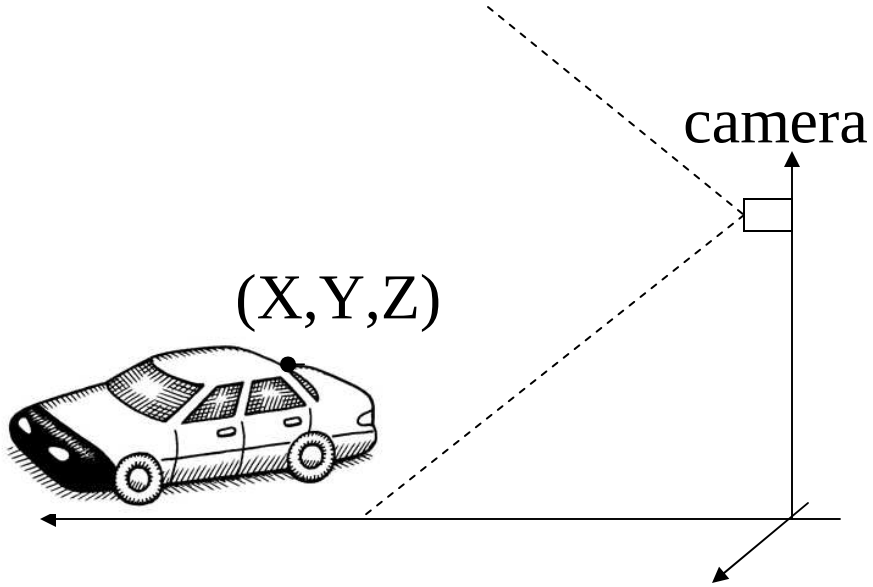
Camera systems on robots

- One fixed camera with horizontal orientation
- One fixed and tilted camera
- Two fixed cameras
- Omni-directional camera with vertical orientation
- Cameras with controlled orientation and focus, …

3D representation

- We need to calculate 3D coordinates of object which we can see on particular pixel on image





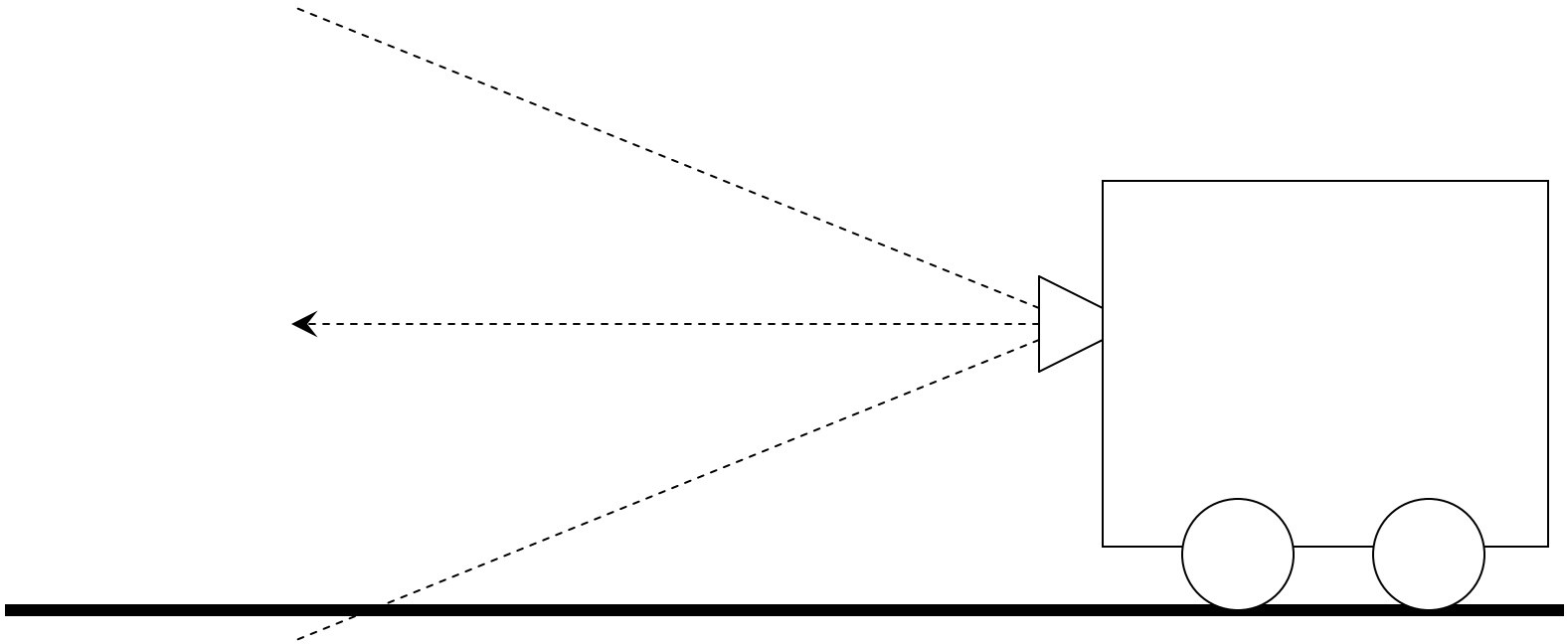
We need to transform (x,y) to (X,Y,Z)

3D reconstruction from one camera

- We need to know something about the object: distance to object or that object is put on surface
- We need to know or calibrate several constants.

One camera, horizontal direction

position of camera is
fixed in particular height



One camera, horizontal direction

α - view angle of camera in vertical direction

β - view angle of camera in horizontal direction

Φ - rotation angle of camera in vertical plane

d - focal length of camera

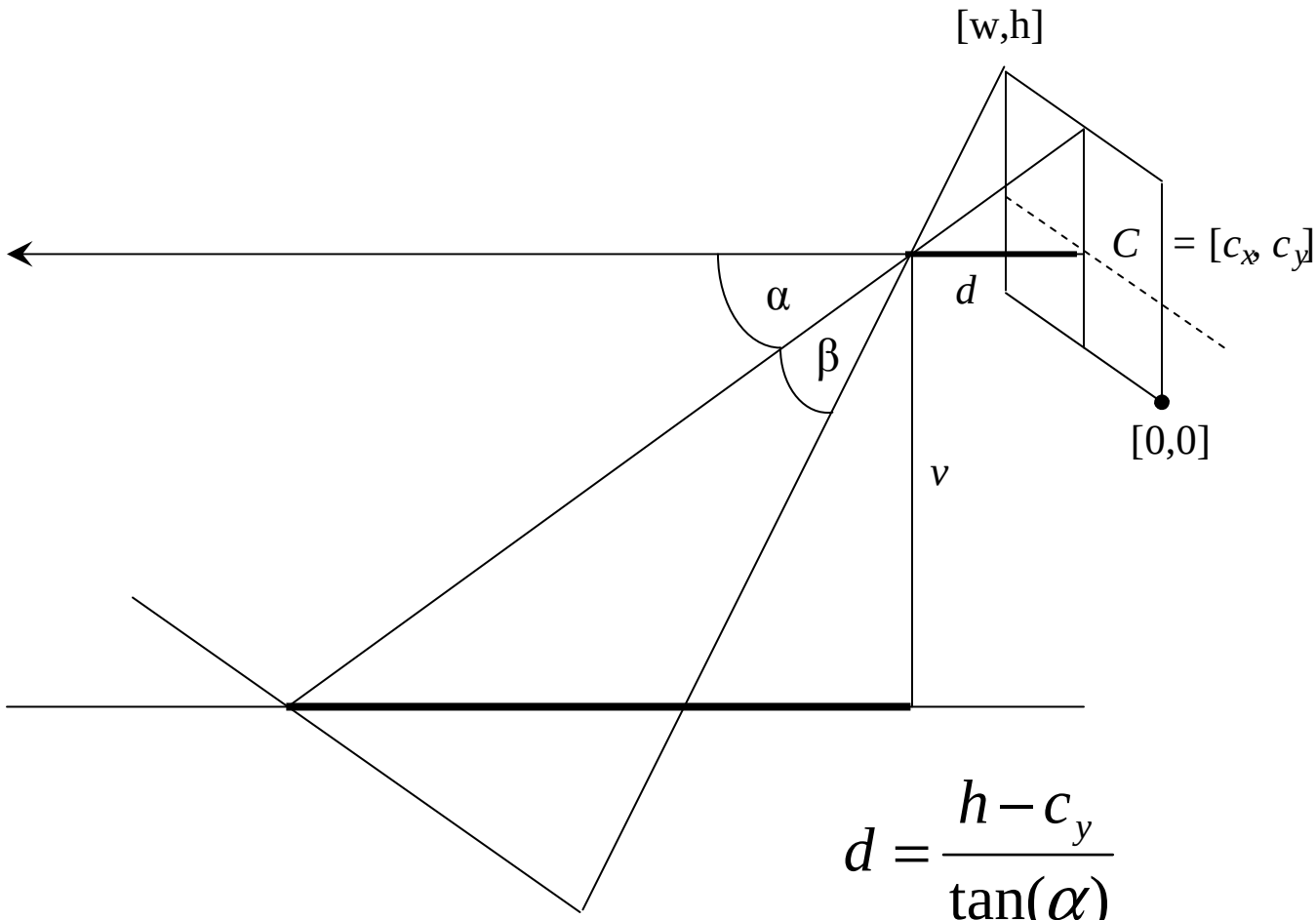
v - height of camera over ground

w - width of the image (in pixels)

h - height of the image (in pixels)

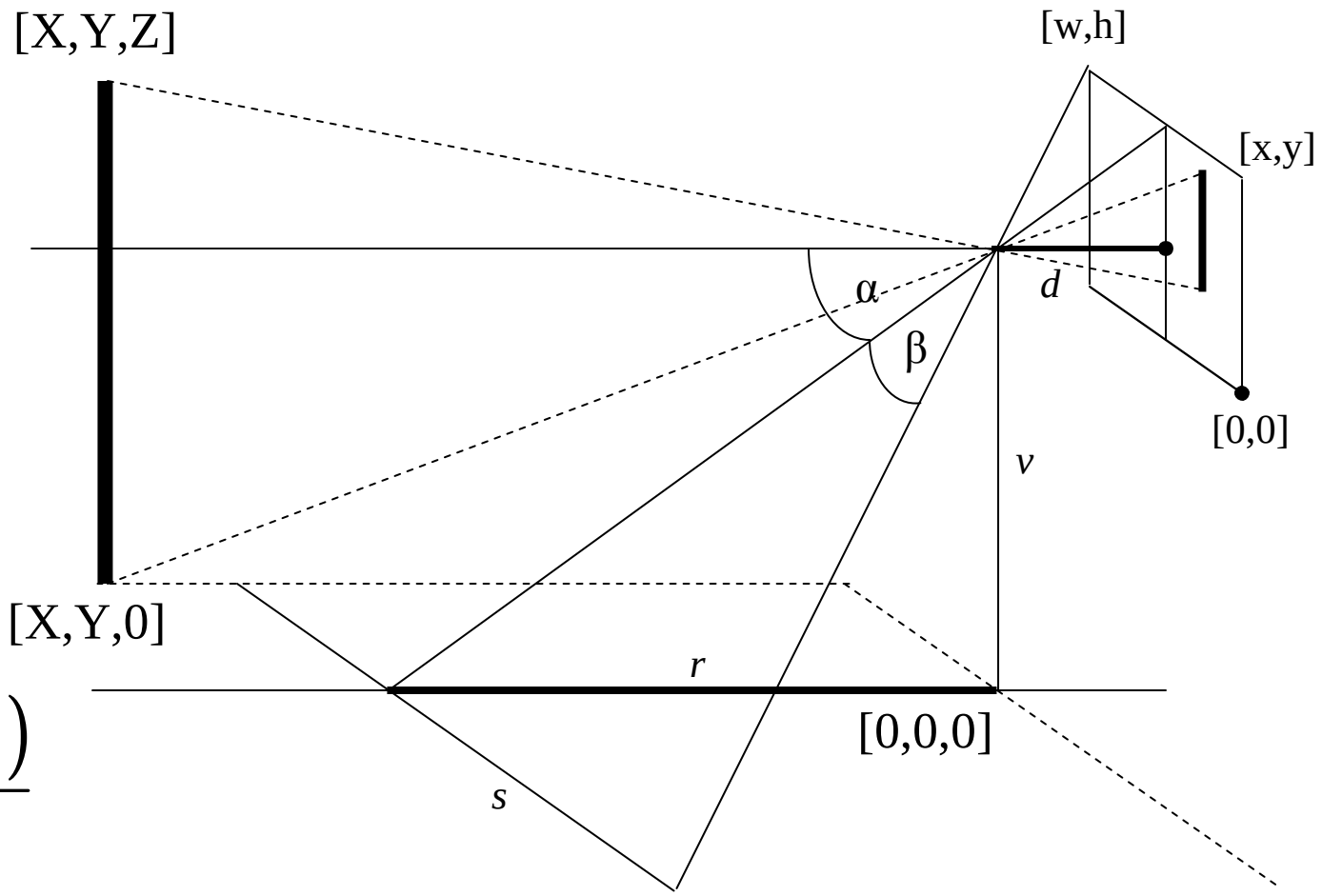
C - vanishing point on image

One camera, horizontal direction



$$d = \frac{h - c_y}{\tan(\alpha)}$$

One camera, horizontal direction



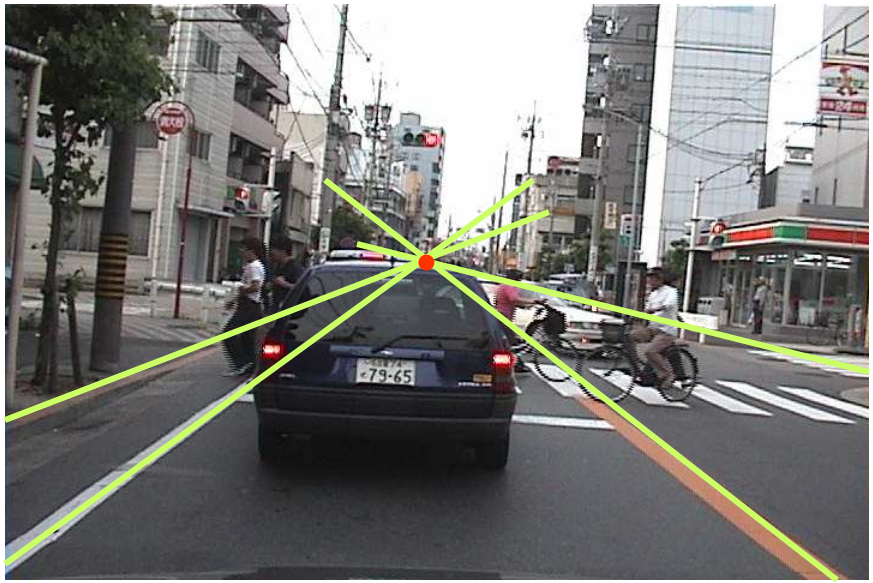
$$Y = \frac{vd}{y - c_y}$$

$$X = \frac{Y\left(x - c_x\right)}{d}$$

$$Z = 0$$

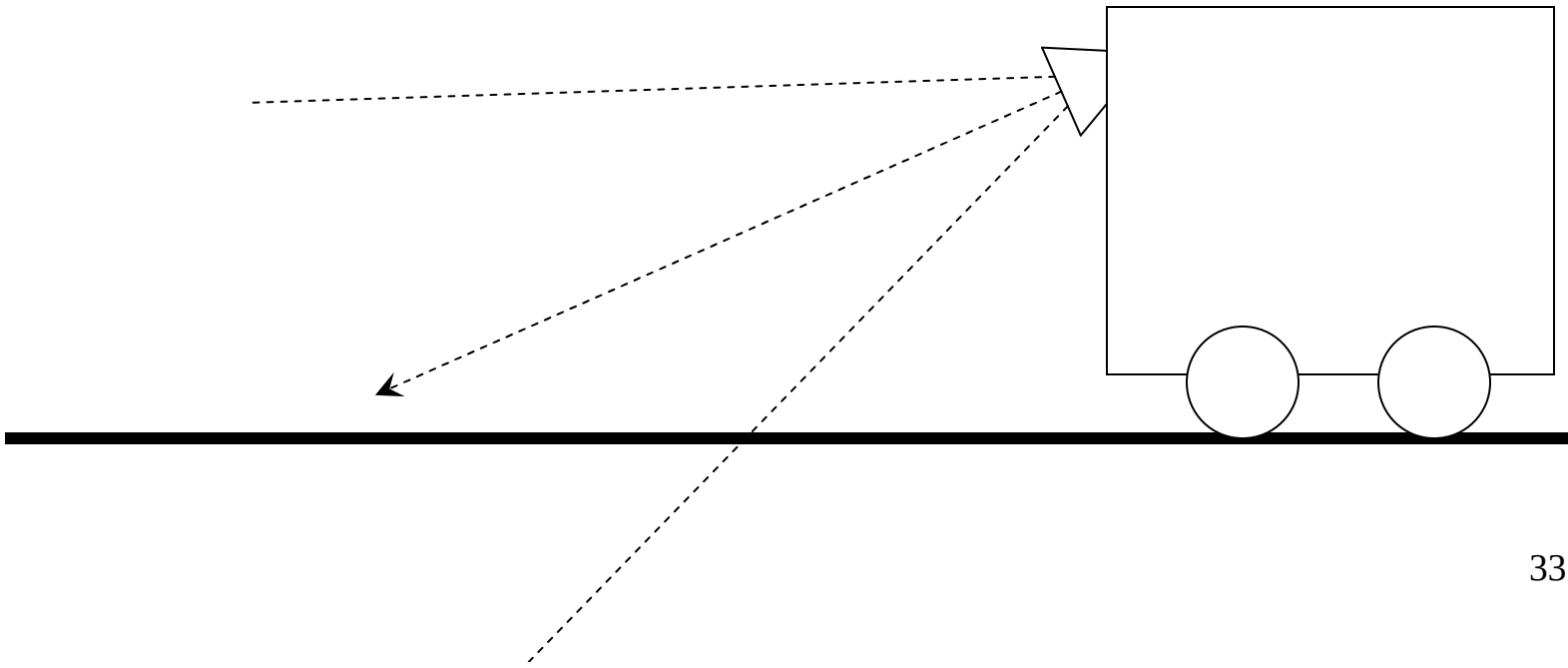
Vanishing point

- is the centre of perspective; a projection of the focal point to image
- for some applications, it can be calculated on-fly from image
- for others it can calculated during calibration



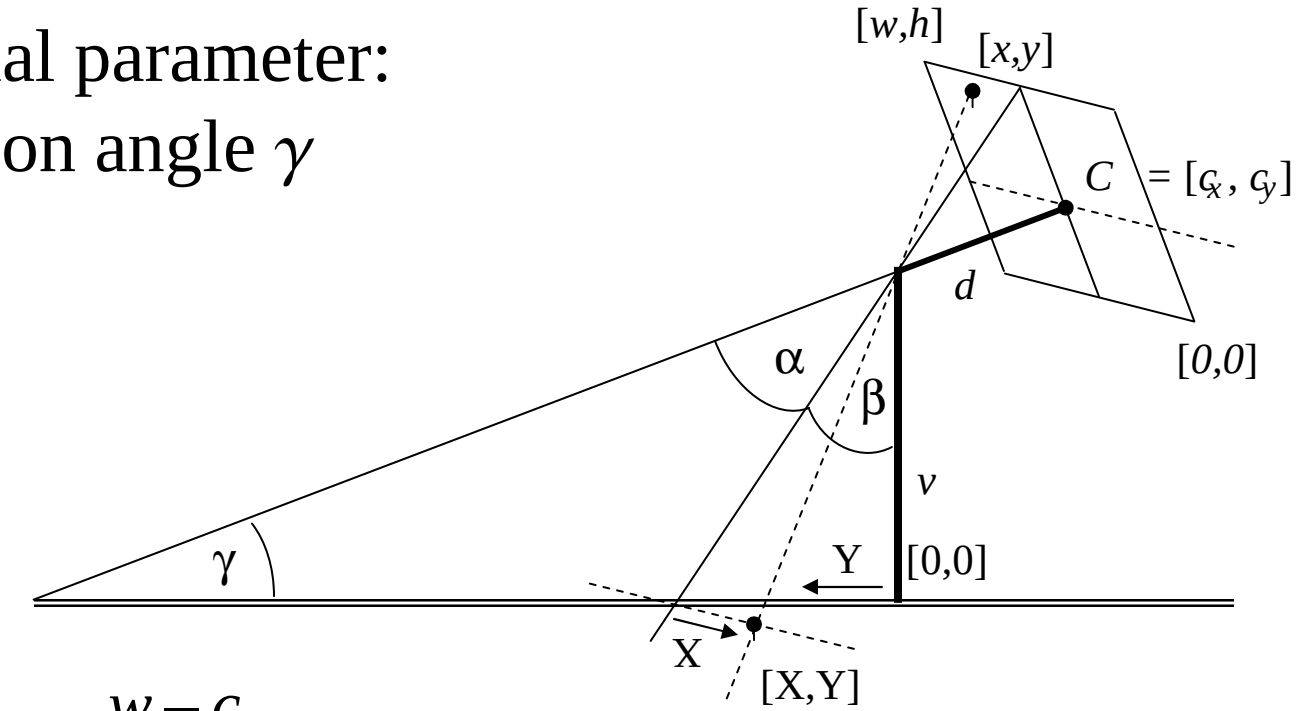
One camera, sloped direction

camera is fixed at
particular height and
has fixed inclination



One camera, sloped direction

- additional parameter: inclination angle γ



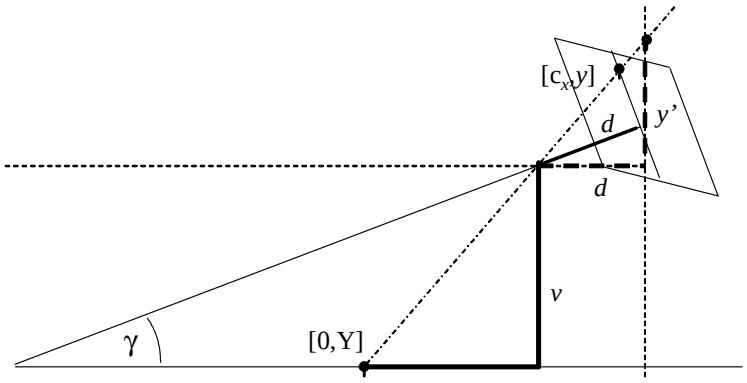
$$d = \frac{h - c_y}{tg\alpha} = \frac{w - c_x}{tg\beta}$$

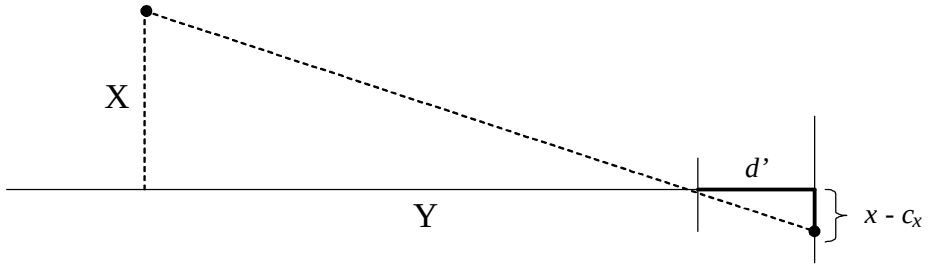
One camera, sloped direction

$$X = Y\frac{x - c_x}{d'} \qquad d' = d\cos\gamma$$

$$Y = \frac{vd}{y'} \qquad y' = d\frac{d\sin\gamma + (y - c_y)\cos\gamma}{d\cos\gamma - (y - c_y)\sin\gamma}$$

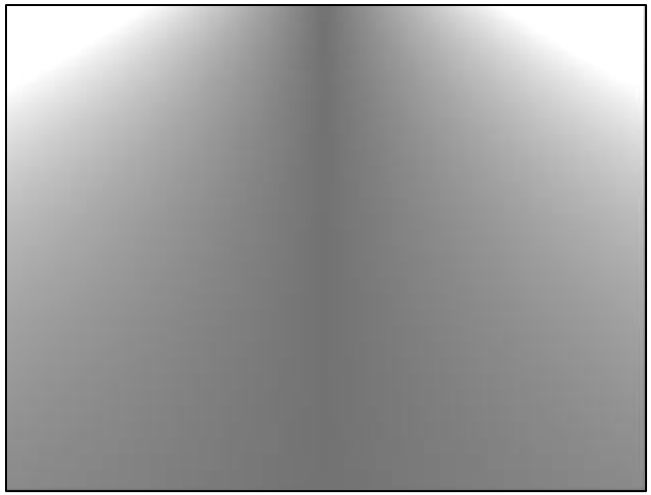
$$Z = 0$$





One fixed camera (straight or tilted)

- Regardless geometry is complicated, finally calibration provides X and Y coordinate for each pixel (x,y)



X

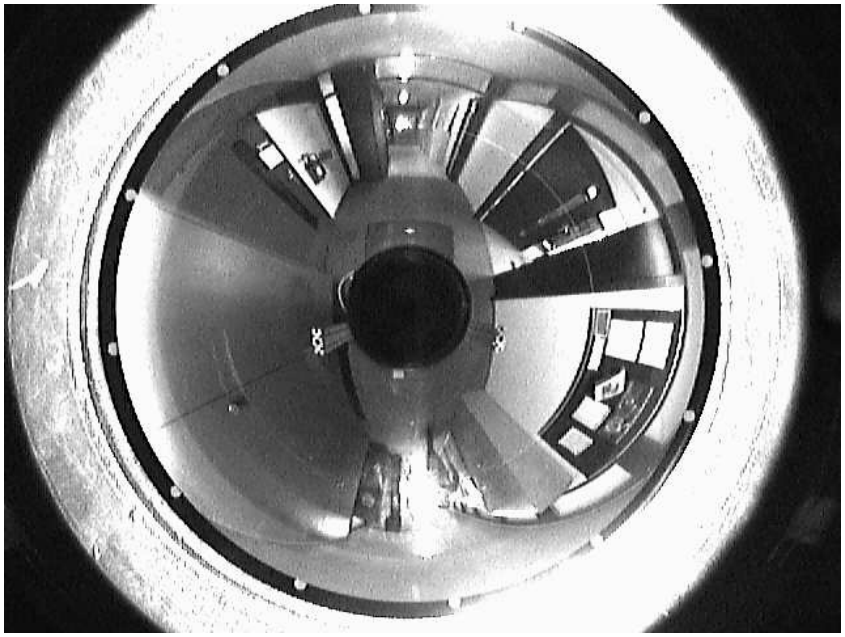
Y

Two fixed cameras

- We have two images taken from other point
- We find corresponding points (for each point we look for point of the same color on epipolar line in the other image)
- We can use corresponding points for triangulation
- Thus we are able to calculate all 3D coordinates for any corresponding point

Other camera systems

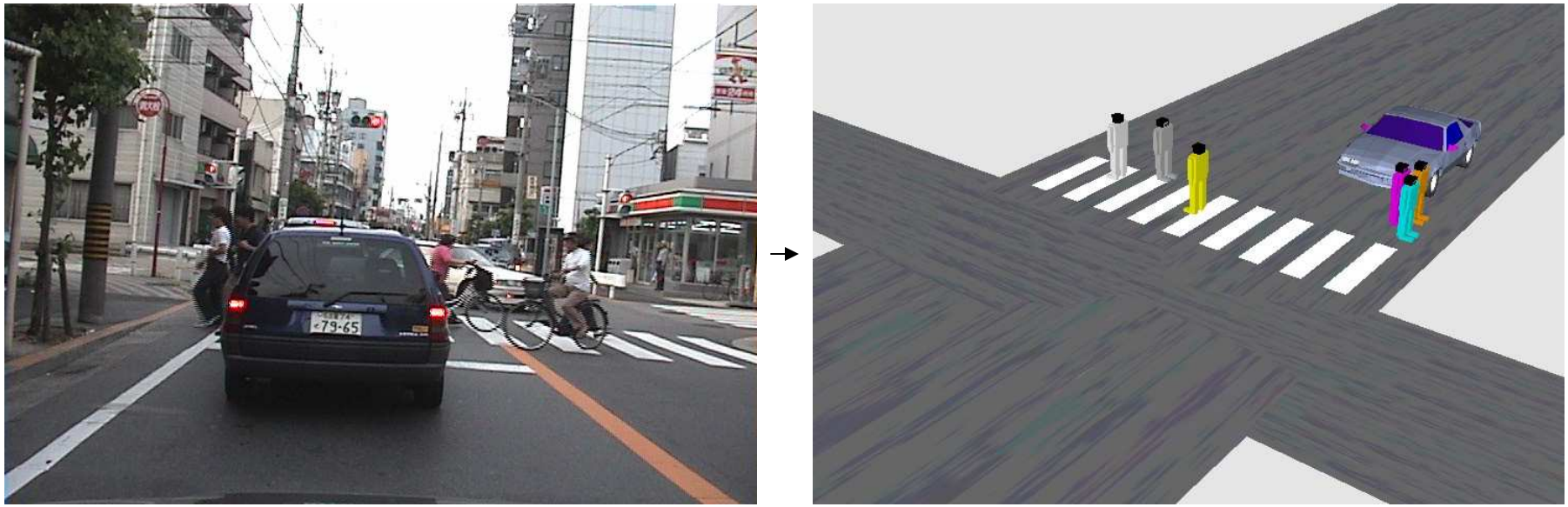
- More complicated geometry
- Provides better service
- But, more performance or time is needed for processing



Object recognition

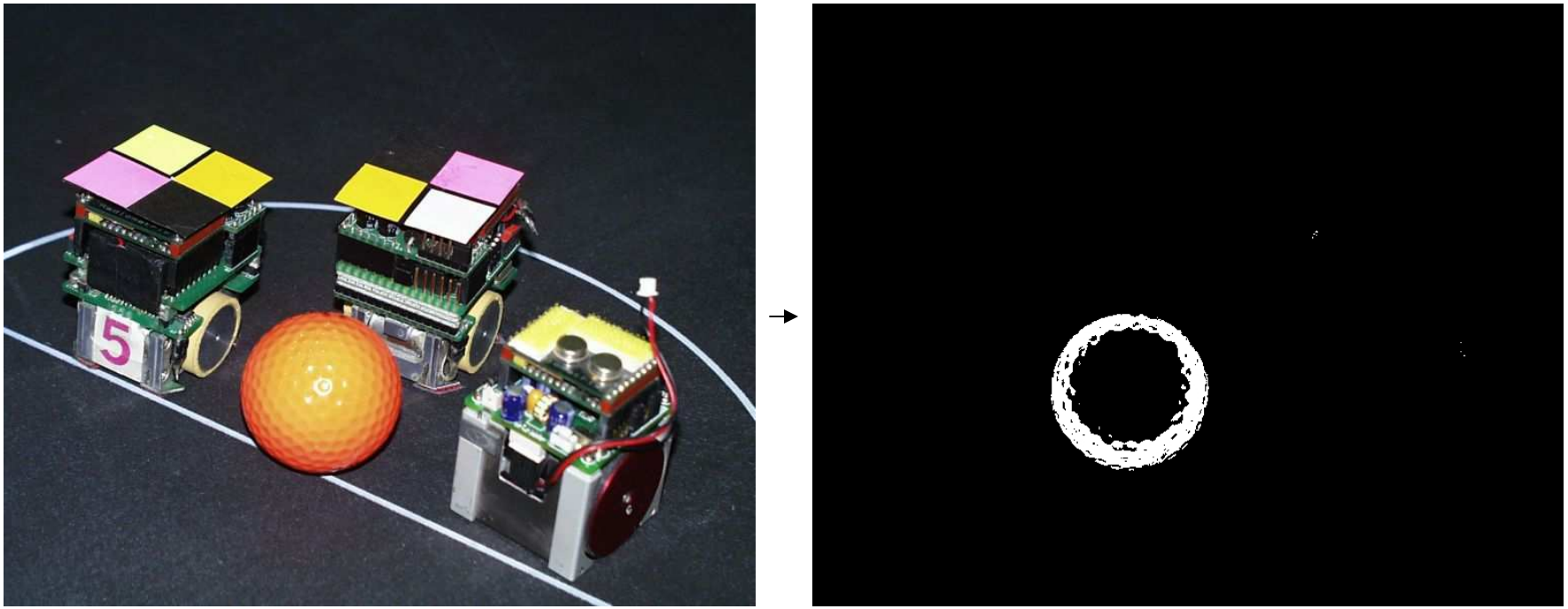
Task: scene reconstruction

- Building model (representation) of what is seen
- Both 2D and 3D processing is involved



Color recognition

Sometimes it is enough to construct condition on red, green and blue ingredients: c = r/(g+1);
if (r > 200 && c > 1.1 && c < 3.3 && b < 75) ...

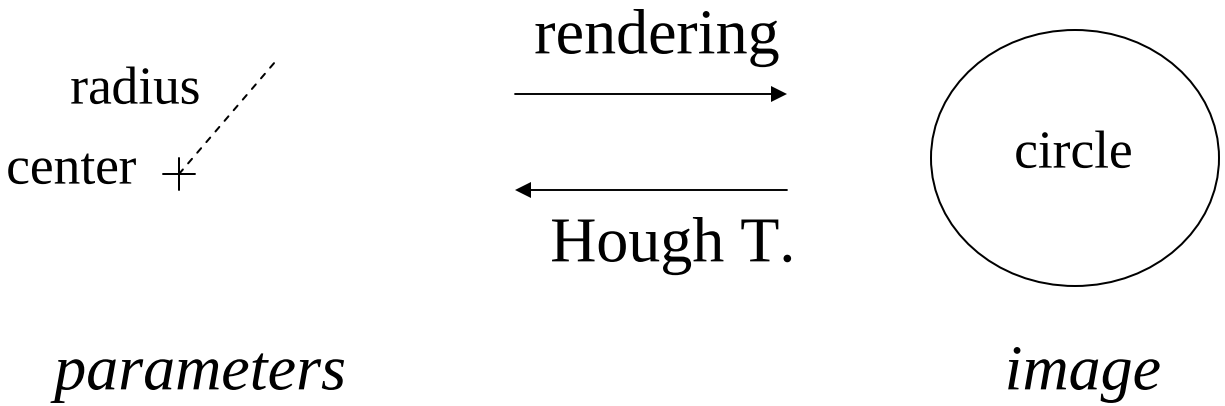


Shape recognition

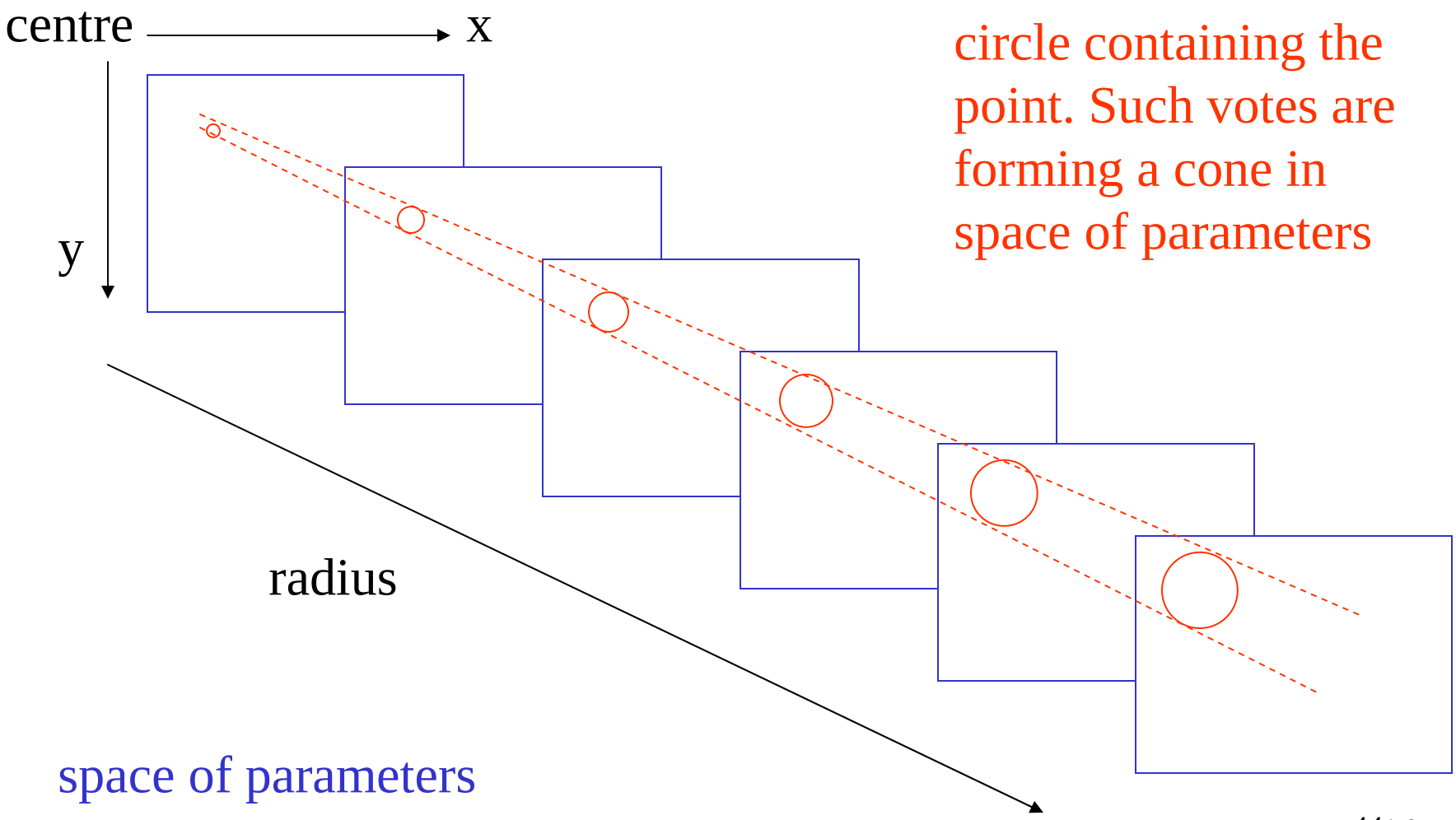
- Geometric shapes which can be generated from few parameters can be recognized by Hough transformation: lines, circles, ellipses … or by ad-hoc methods.
- Compounded objects can be recognized as a join of simpler objects: angles, triangles, …
- Other objects can be recognized by skeletonization if the skeleton can be recognized: person

Idea of Hough Transformation

- transformation of image to space of parameters which describe the object we are looking for.

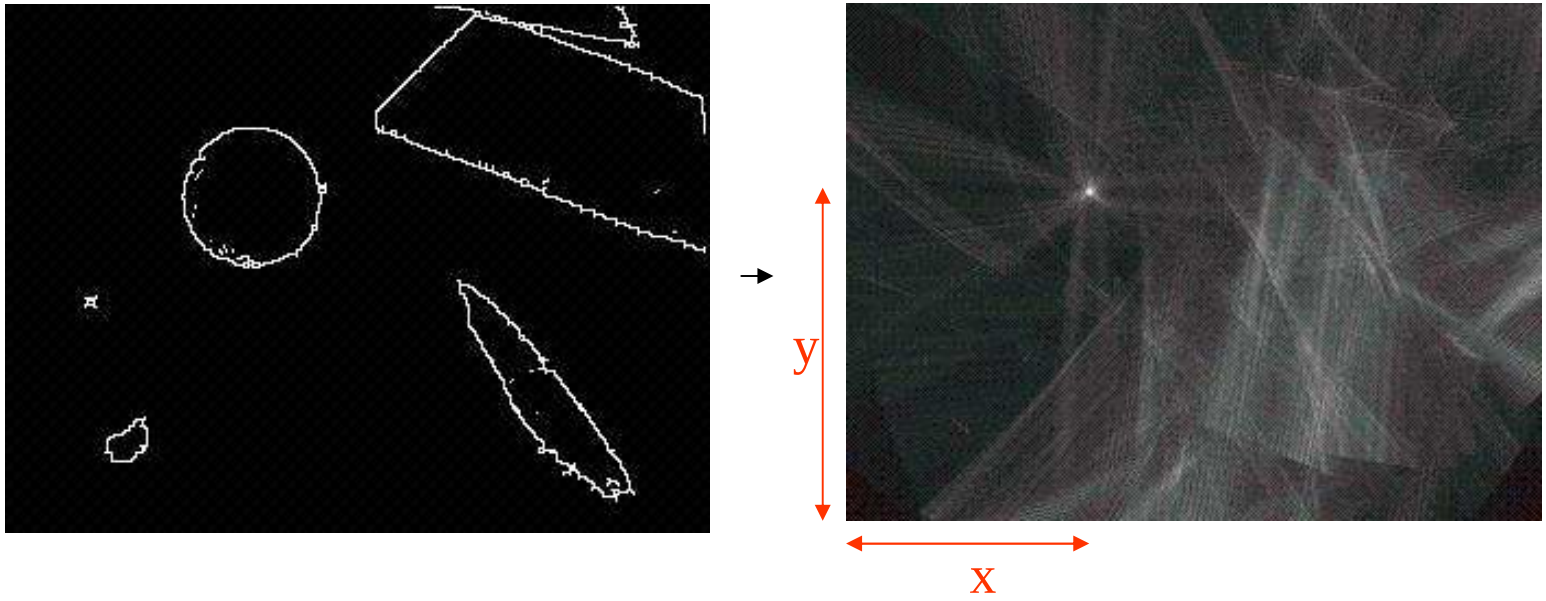


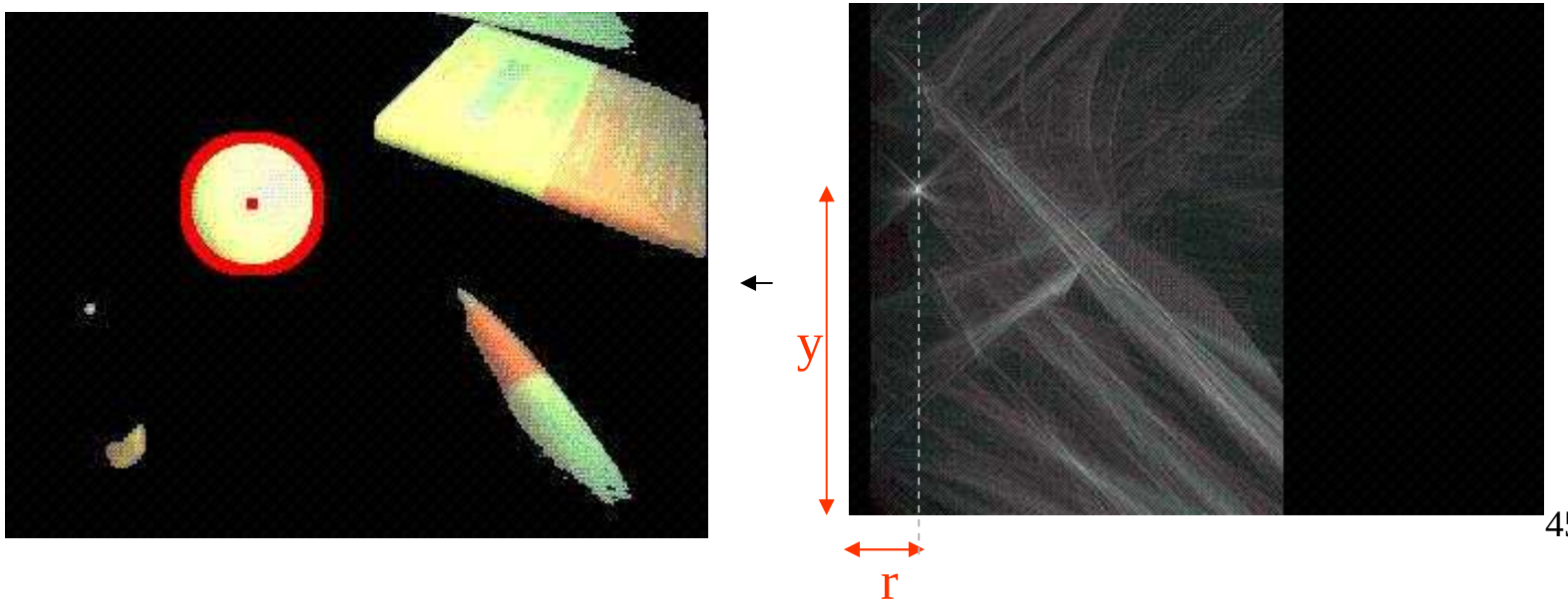
Hough T. (for circle)



Each white point votes for any parameters which could render circle containing the point. Such votes are forming a cone in space of parameters

Hough transformation (for circle)

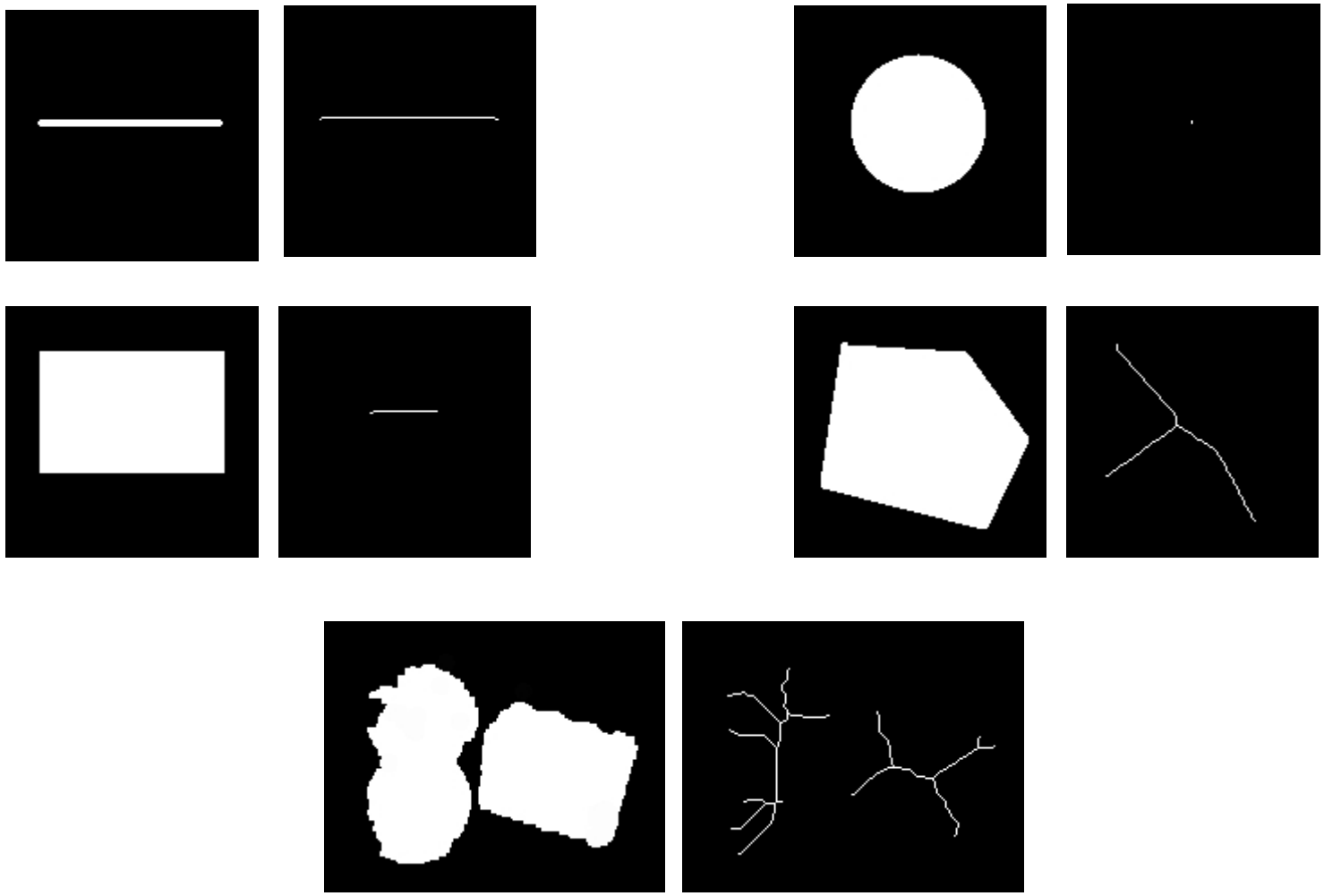




Skeletonization

- Each object has skeleton, which represents its topology
- Skeleton is set of centers of disks which touch object boundary in two points at least
- Skeleton is usually simpler than the represented object and it is easier to detect it

Skeletons



Other methods of recognition

- Methods based on statistical processing of significant points at object boundary
- Methods based on motion detection and object tracking
- Methods based on mapping video sequences into less dimensional space representing scene processes into trajectory in the space
- Many others

Thank you for your attention !

Andrej Lúčny

Department of Applied Informatics

FMFI UK Bratislava

andy@microstep-mis.com

www.microstep-mis.com/~andy