



Digital Image Processing Functions - MATLAB

MECH-M-1-SEA-DBV-ILV

(Version September 2023)

© Kurt Niel
University of Applied Sciences Upper Austria
Faculty of Engineering and Environmental Sciences
September 1st, 2023



Literature

- (1) Gonzalez R.C., Woods R., Eddings S.L., Digital Image Processing using MATLAB, Pearson Prentice Hall, 2004, ISBN 978-0-13008-519-7
- (2) Demirkaya O., Asyali M.H., Sahoo P.K., Image Processing with MATLAB, Applications in Medicine and Biology, CRC Press, 2009, ISBN 978-0-8493-9246-7
- (3) MathWorks Documentation, Image Processing Toolbox
<https://de.mathworks.com/help/images>
- (4) Kovesi P., MATLAB and Octave Functions for Computer Vision and Image Processing, The University of Western Australia,
<http://www.peterkovesi.com/matlabfns>



Functions in MATLAB (R2017a / R2021a)

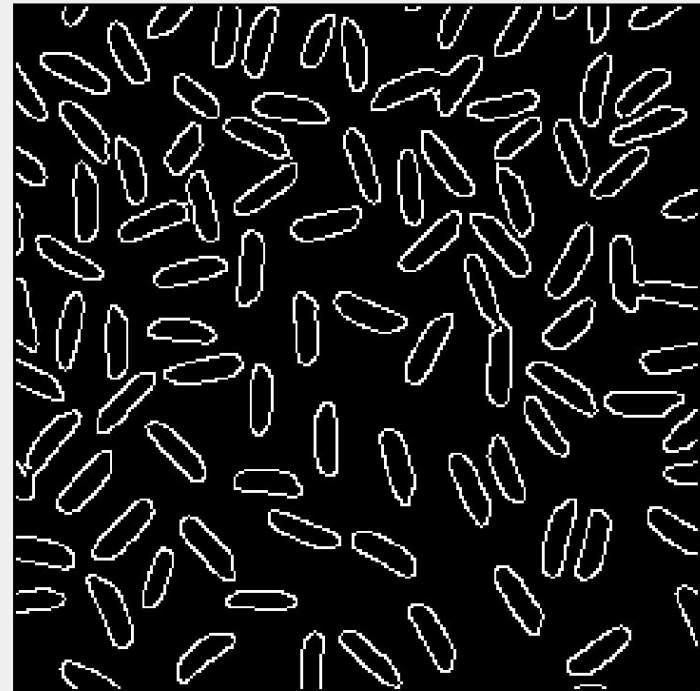
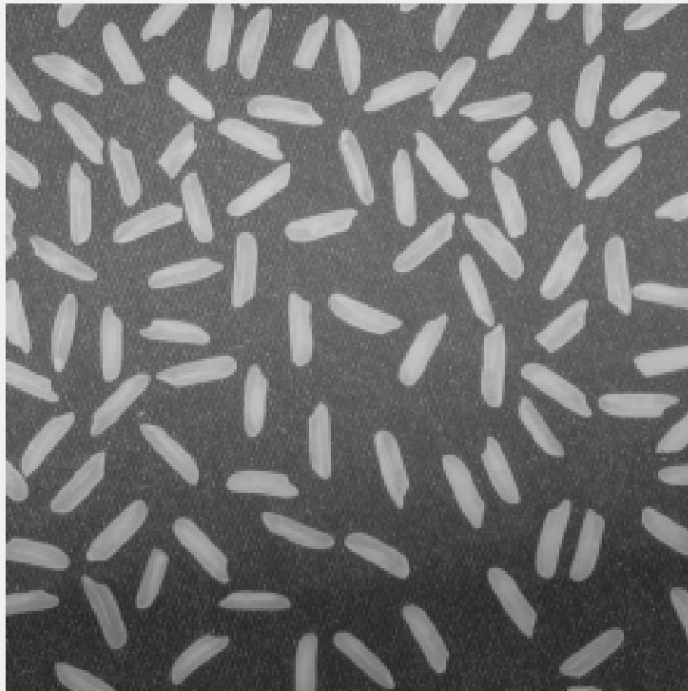
- Preparation
 - load, display, histogram
 - color/gray, resolution
- Metrology
 - edge detection
- Segmentation
 - convolution, denoising, binarization, BLOB analysis, morphology, e.g., watershed
- Identification
 - Hough transformation, projective geometry, image pyramid, Fourier transformation



Metrology – Edge Detection

```
I = edge( I, method );    % 'canny', 'sobel', etc.
```

- Highlighting edge features

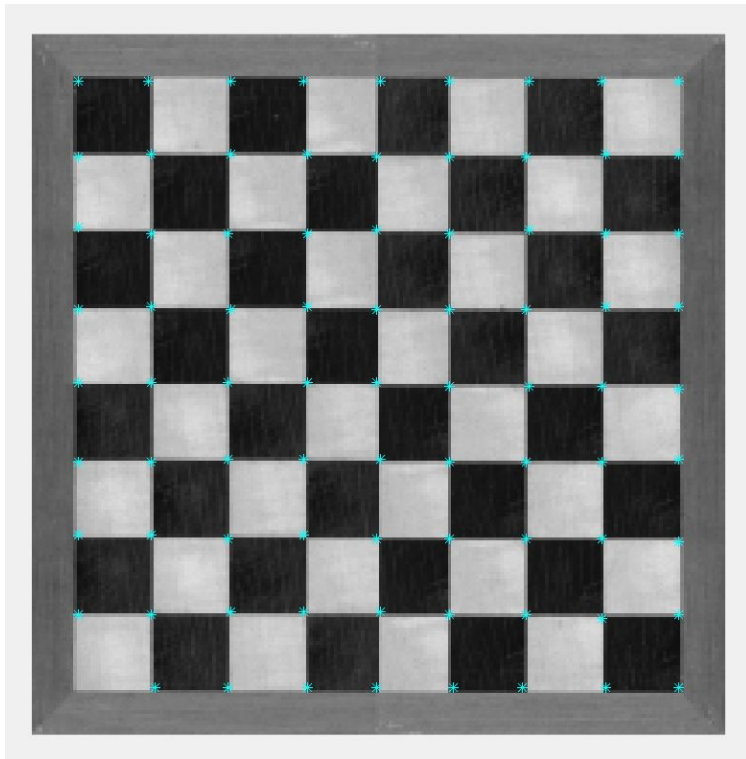




Metrology – Corner

```
C = corner( I );
```

- Finds and locates corners; result is list of corners

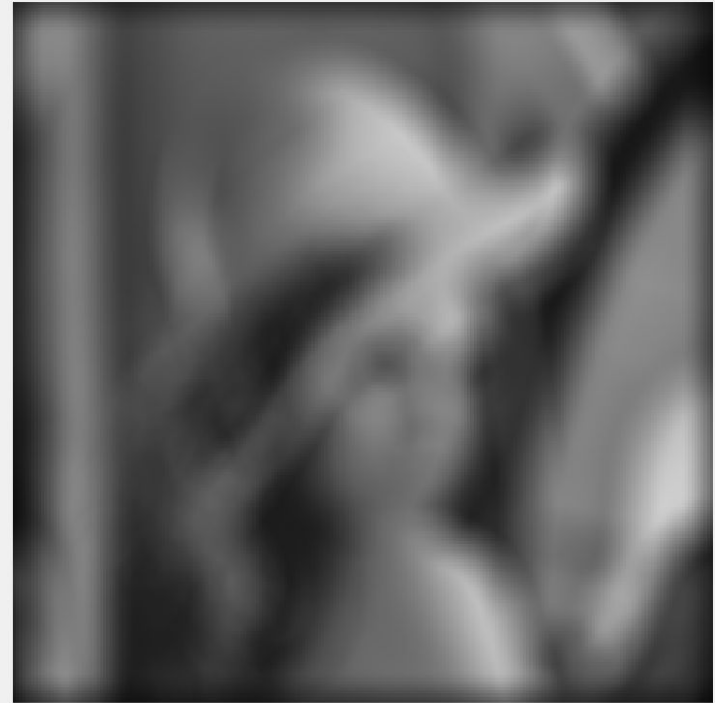




Segmentation – Convolution

```
h = fspecial( type );           % 'average', 'gaussian', 'prewitt', etc  
I = imfilter( I, h );
```

- Modify image properties for highlighting different features: blurring, edges, etc.





Segmentation – Denoising

```
D = medfilt2( I );
```

- Removes local extremes, e.g. 'salt & pepper noise'

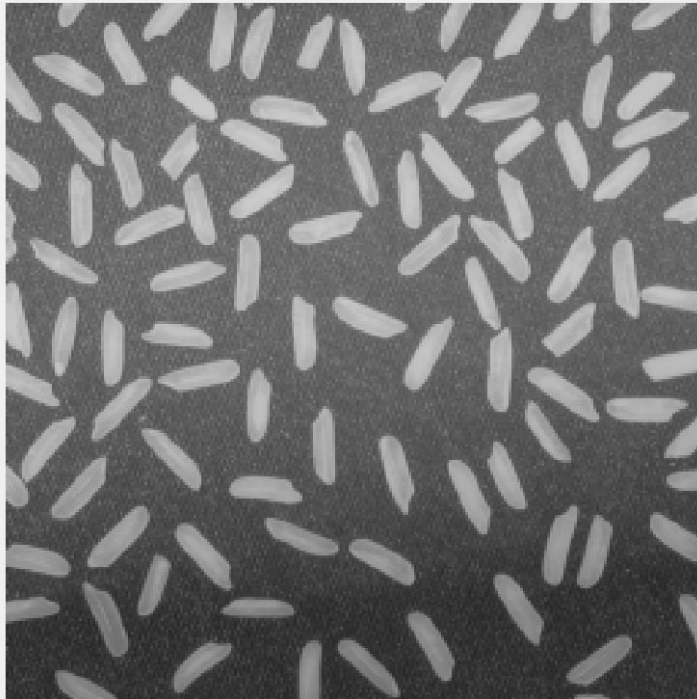




Segmentation – Binarization

```
BW = imbinarize( I, method );           % 'global', 'adaptive'
```

- Simple segmentation method

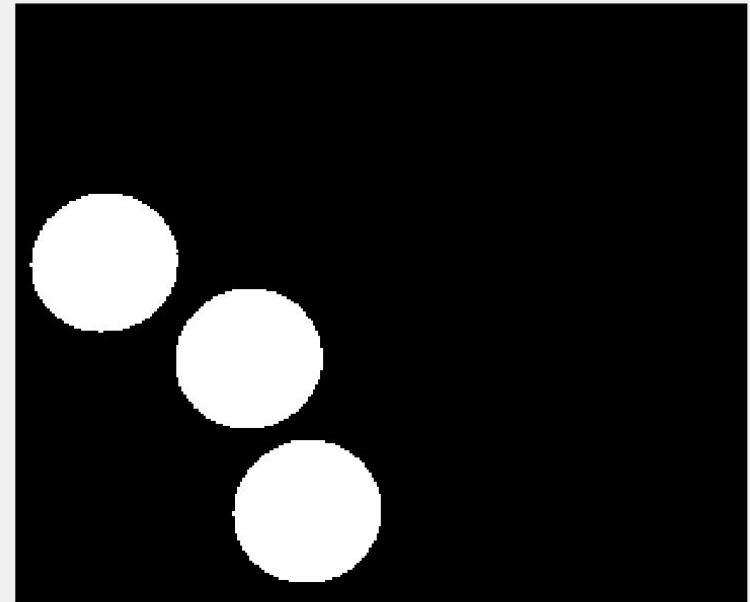
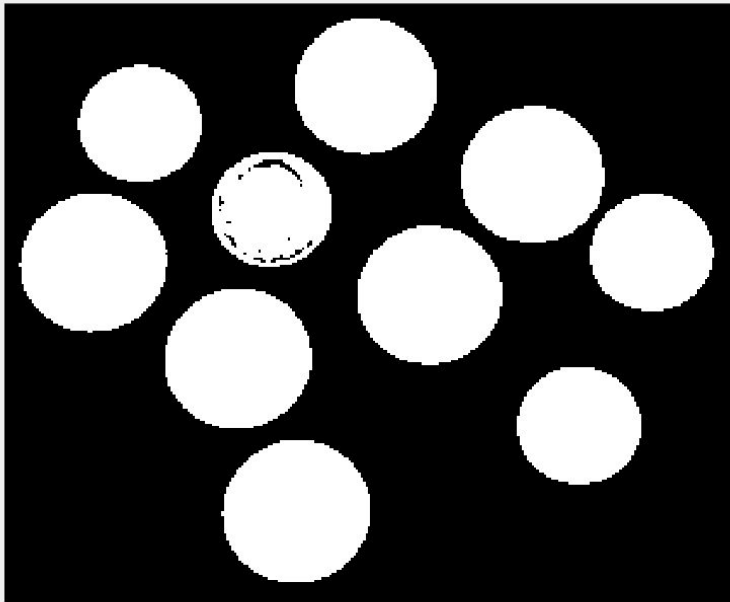




Segmentation – BLOB Analysis

```
Y = bwprofilt( I, type );           % 'Area', 'Perimeter', etc.
```

- Analyze and filter BLOBs (binary large objects) by features



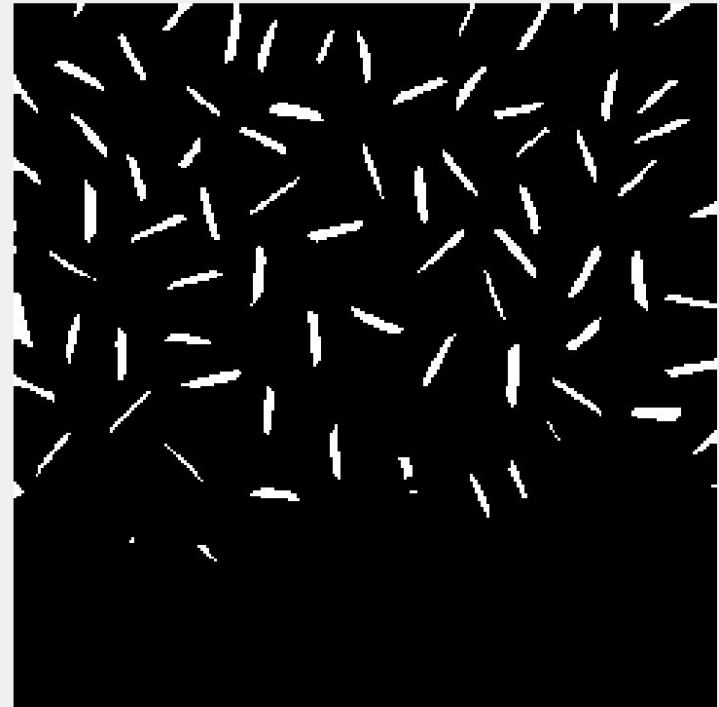
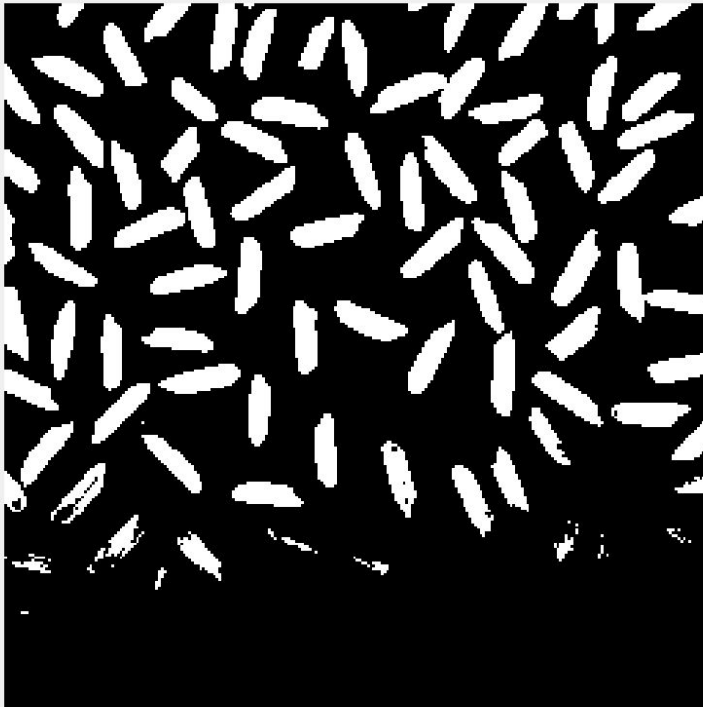


Segmentation – Morphology

```
Y = imerode( I, se );
```

% disk, rectangle, etc.

- Removes small objects, separates objects

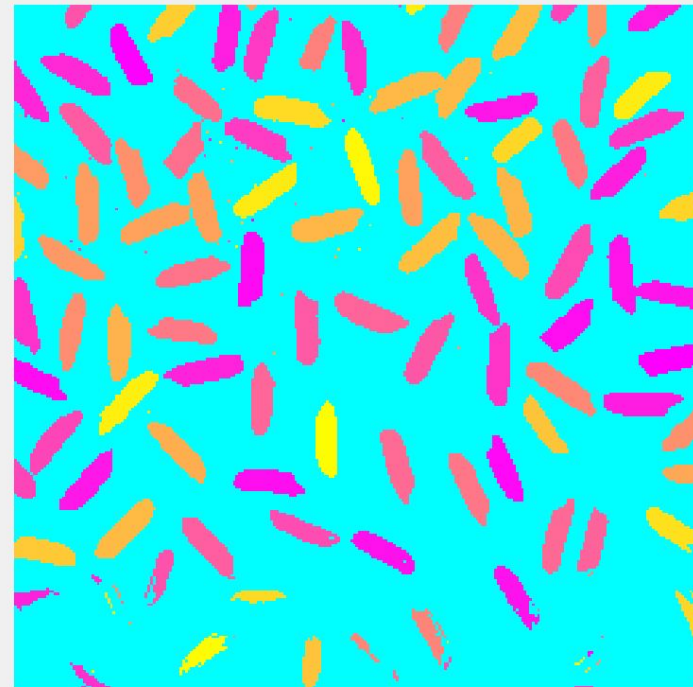
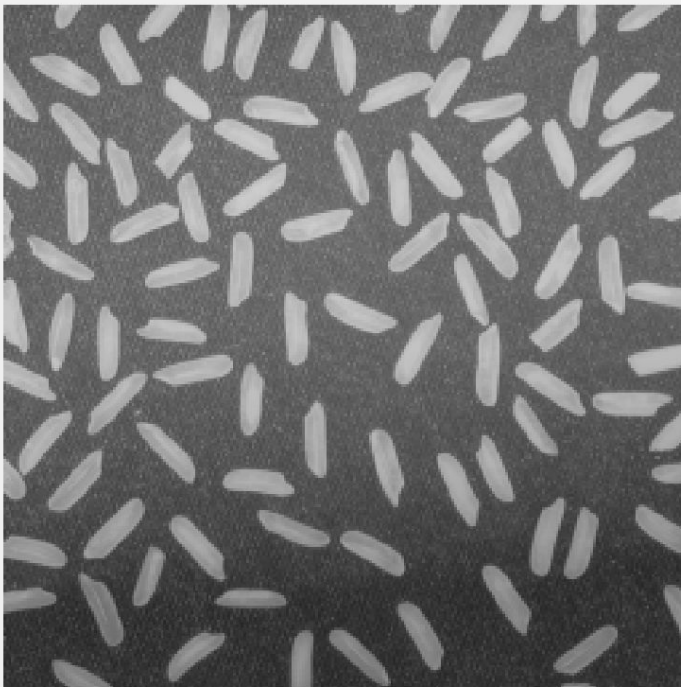




Segmentation – e.g., Watershed

```
CC = bwconncomp( I );  
L = labelmatrix( CC );  
RGB = label2rgb( L, map, zerocolor, order );
```

- Label connected regions



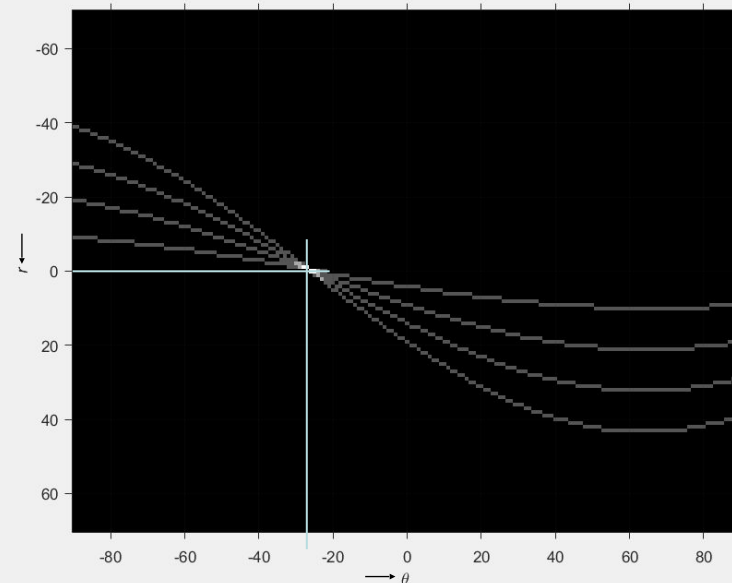
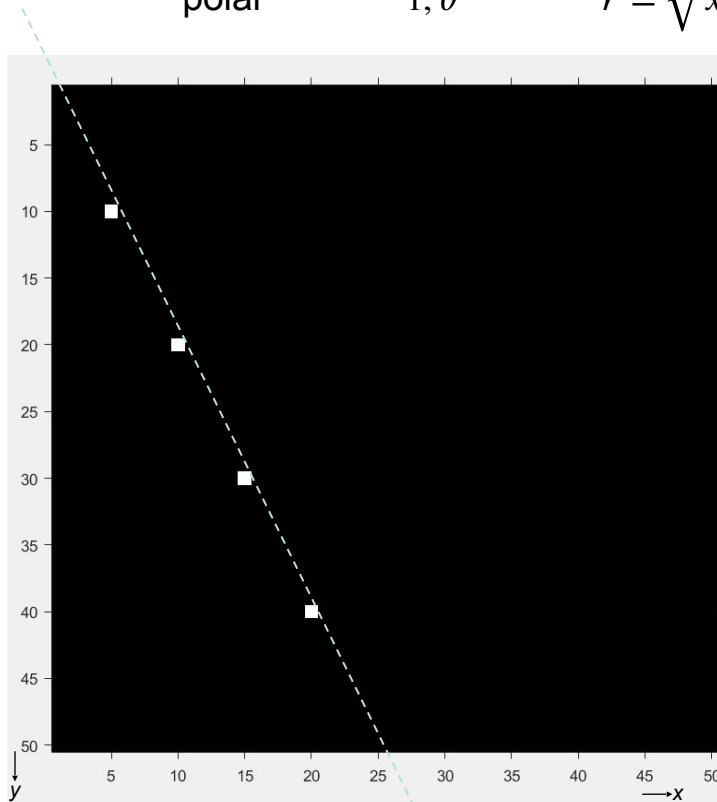


Identification – Hough Transformation

```
[H, theta, radius] = hough( I );
```

- lines in polar coordinates → intersection at similar lines within the image

cartesian	x, y	$x = r \cos \theta \quad / \quad y = r \sin \theta$	$y = kx + d$
polar	r, θ	$r = \sqrt{x^2 + y^2} \quad / \quad \theta = \operatorname{atan} \frac{y}{x}$	$r = -\frac{p}{\cos(\theta - \theta_0)}$

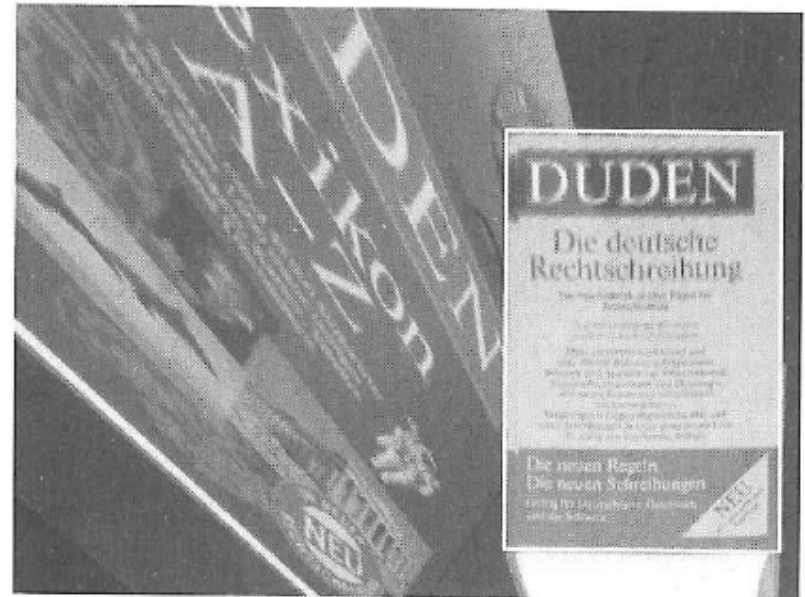
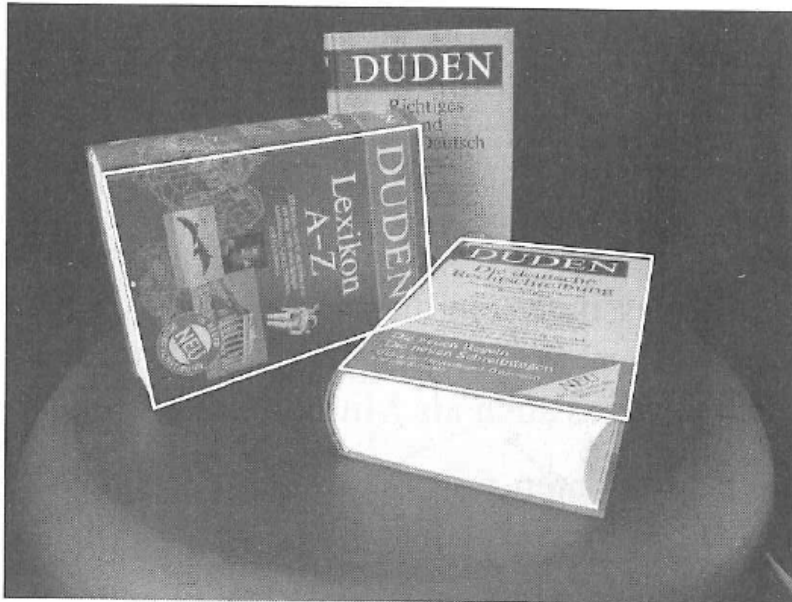




Identification – Projective Geometry

$$I = \text{ImTrans}(I, H_{3 \times 3});$$

- Euclidean, Affine, Projective transformation of images - rectification

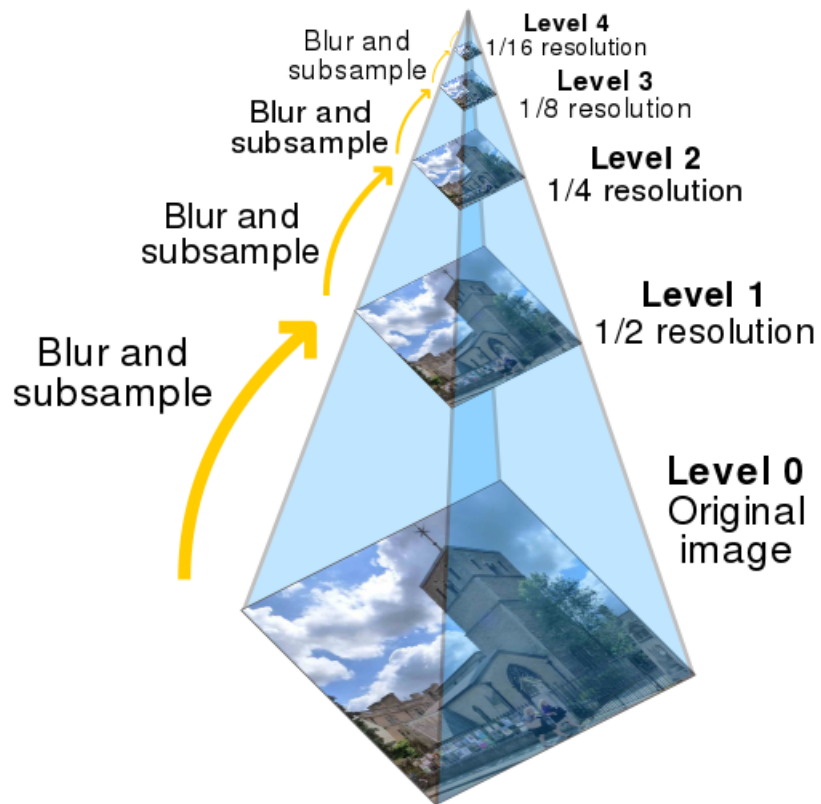




Identification – Image Pyramid

```
[ pyr ] = genPyr( I, type, level );
```

- Generating multiscale images for speeding up segmentation/identification



Von Cmglee - Eigenes Werk, CC BY-SA 3.0,
<https://commons.wikimedia.org/w/index.php?curid=42549151>



Identification – Fourier Transformation

```
F = fft2( I );
```

- Analyze frequency domain

