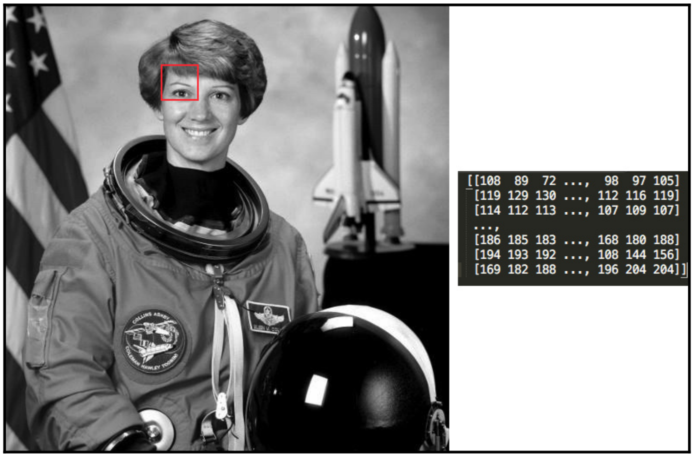
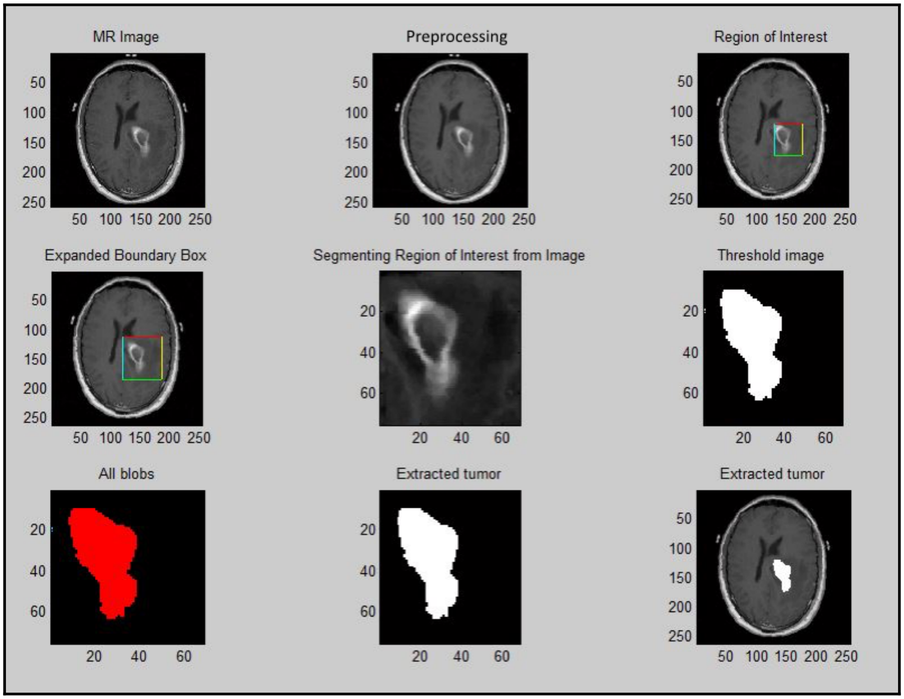
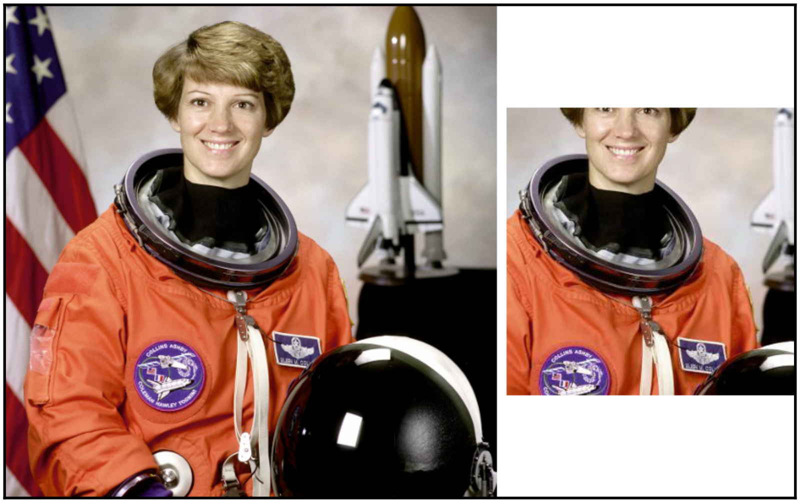
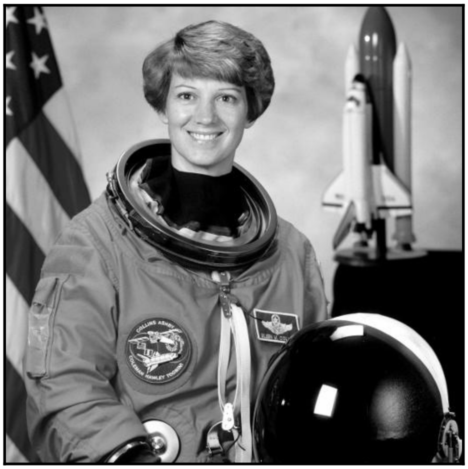
Chapter 01: Introduction to Image Processing

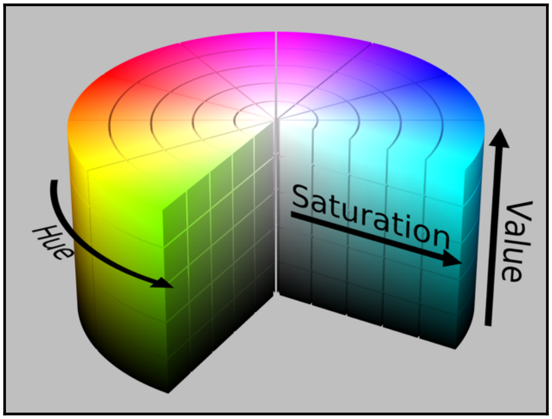


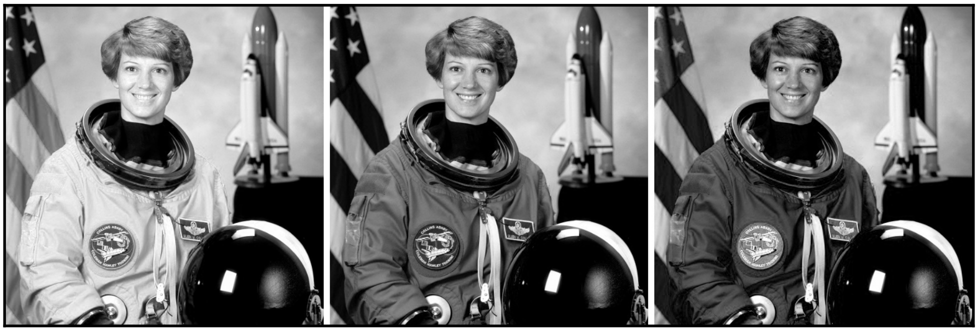
```
[[108  89  72 ...,  98  97 105]
 [119 129 130 ..., 112 116 119]
 [114 112 113 ..., 107 109 107]
 ...,
 [186 185 183 ..., 168 180 188]
 [194 193 192 ..., 108 144 156]
 [169 182 188 ..., 196 204 204]]
```

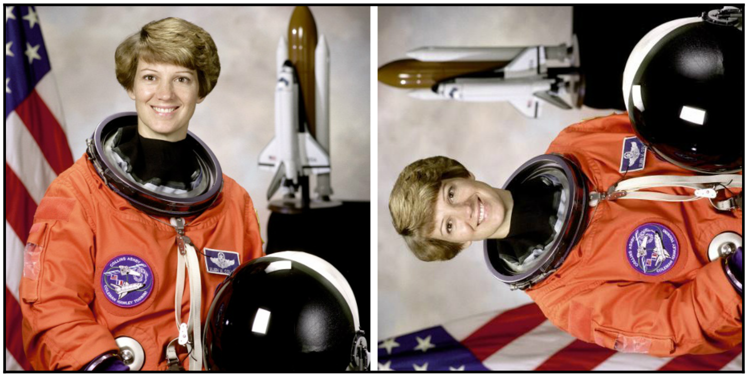


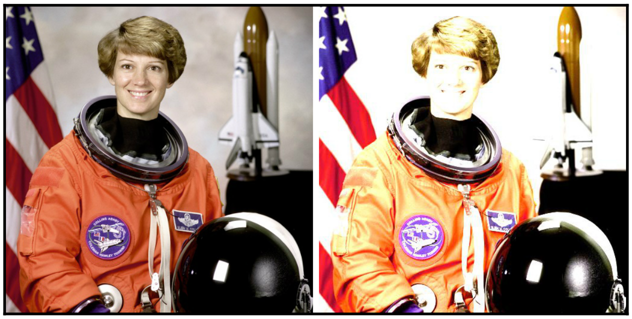


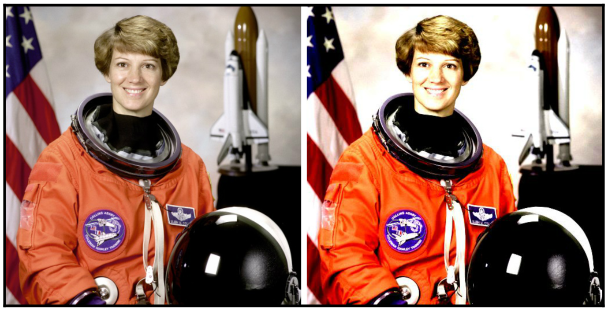


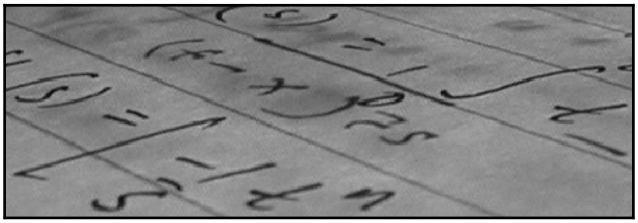

Hue
Saturation
Value

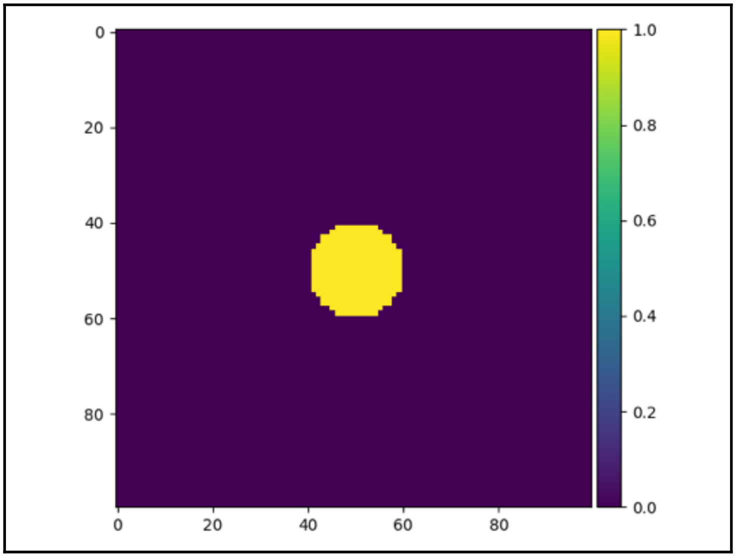


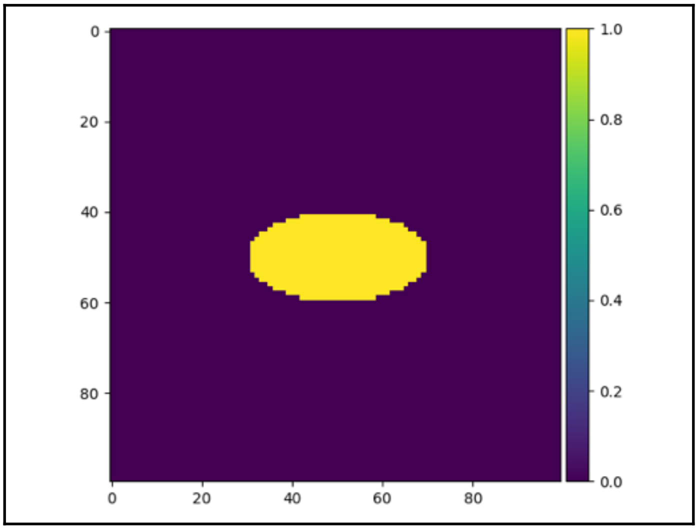


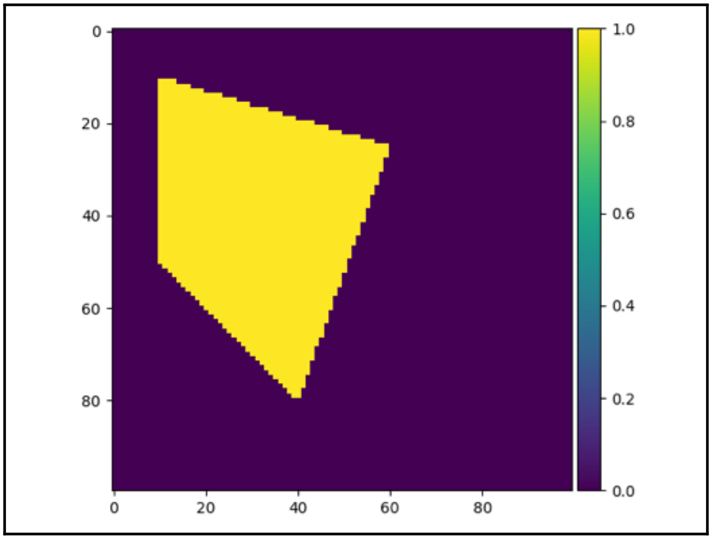







0
20
40
60
80
0
20
40
60
80
1.0
0.8
0.6
0.4
0.2
0.0


0
20
40
60
80
0
20
40
60
80
1.0
0.8
0.6
0.4
0.2
0.0


0
20
40
60
80
0
20
40
60
80
1.0
0.8
0.6
0.4
0.2
0.0

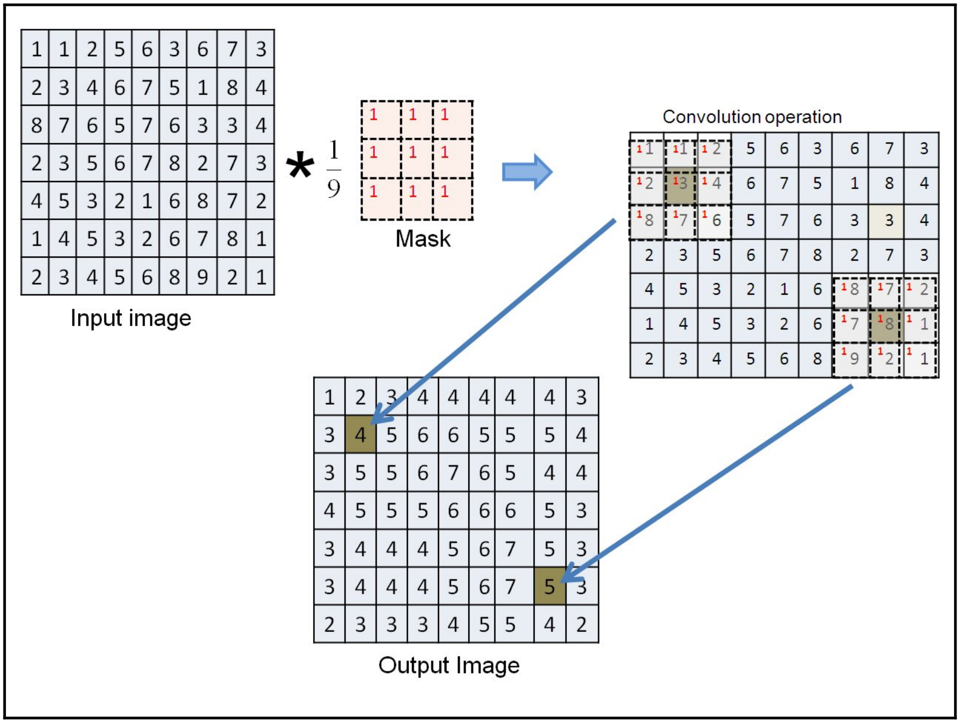
Chapter 02: Filters and Features

$$\lim_{h\to 0}\frac{f(x+h)-f(x)}{h}$$

$$\begin{bmatrix}1 & 0 & -1\\ 1 & 0 & -1\\ 1 & 0 & -1\end{bmatrix}$$

$$\begin{bmatrix}40 & 50 & 60 & 70\\ 40 & 50 & 60 & 70\\ 40 & 50 & 60 & 70\\ 40 & 50 & 60 & 70\end{bmatrix} \qquad \frac{1}{3}\begin{bmatrix}1 & 0 & -1\\ 1 & 0 & -1\\ 1 & 0 & -1\end{bmatrix} \qquad \begin{bmatrix}0 & 0 & 0 & 0 & 0\\ 0 & 20 & 20 & 20 & 0\\ 0 & 20 & 20 & 20 & 0\\ 0 & 20 & 20 & 20 & 0\\ 0 & 0 & 0 & 0 & 0\end{bmatrix}$$

Image Matrix   Derivative Mask   Derivative



1	1	2	5	6	3	6	7	3
2	3	4	6	7	5	1	8	4
8	7	6	5	7	6	3	3	4
2	3	5	6	7	8	2	7	3
4	5	3	2	1	6	8	7	2
1	4	5	3	2	6	7	8	1
2	3	4	5	6	8	9	2	1

Input image

$$* \ \frac{1}{9}$$

1	1	1
1	1	1
1	1	1

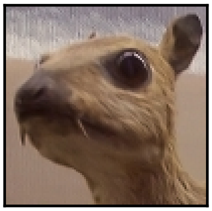
Mask

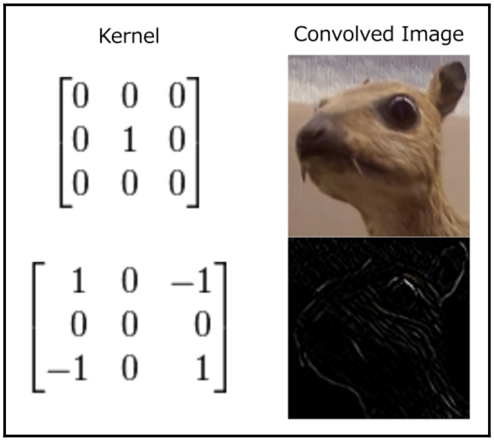
Convolution operation

1 1	1 1	1 2	5	6	3	6	7	3
1 2	1 3	1 4	6	7	5	1	8	4
1 8	1 7	1 6	5	7	6	3	3	4
2	3	5	6	7	8	2	7	3
4	5	3	2	1	6	1 8	1 7	1 2
1	4	5	3	2	6	1 7	1 8	1 1
2	3	4	5	6	8	1 9	1 2	1 1

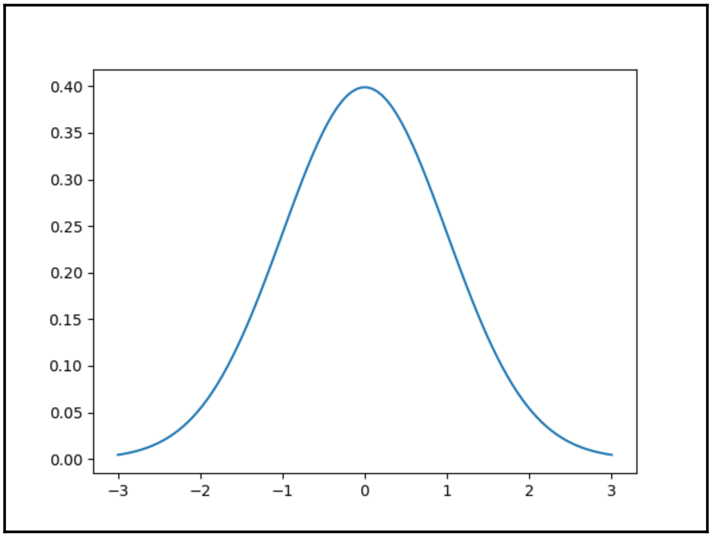
1	2	3	4	4	4	4	4	3
3	4	5	6	6	5	5	5	4
3	5	5	6	7	6	5	4	4
4	5	5	5	6	6	6	5	3
3	4	4	4	5	6	7	5	3
3	4	4	4	5	6	7	5	3
2	3	3	3	4	5	5	4	2

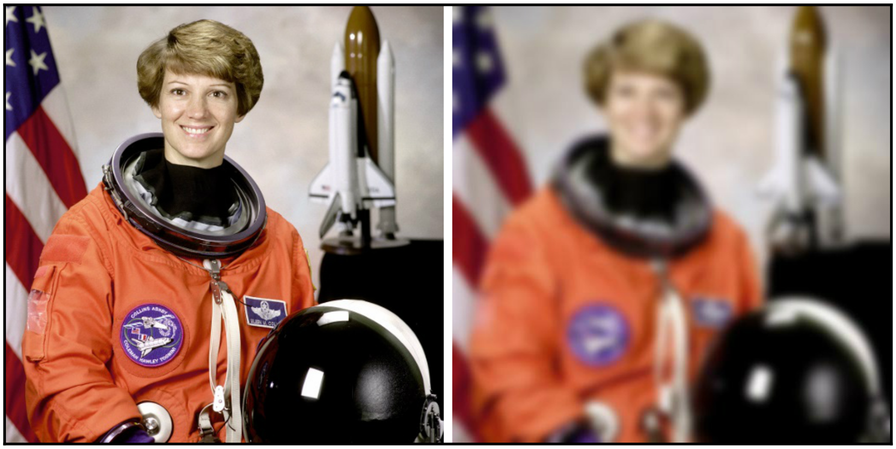
Output Image

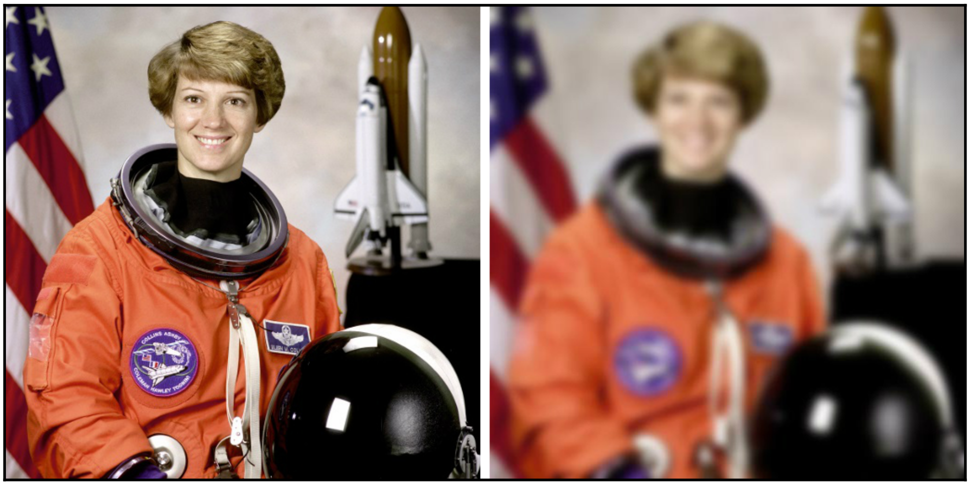


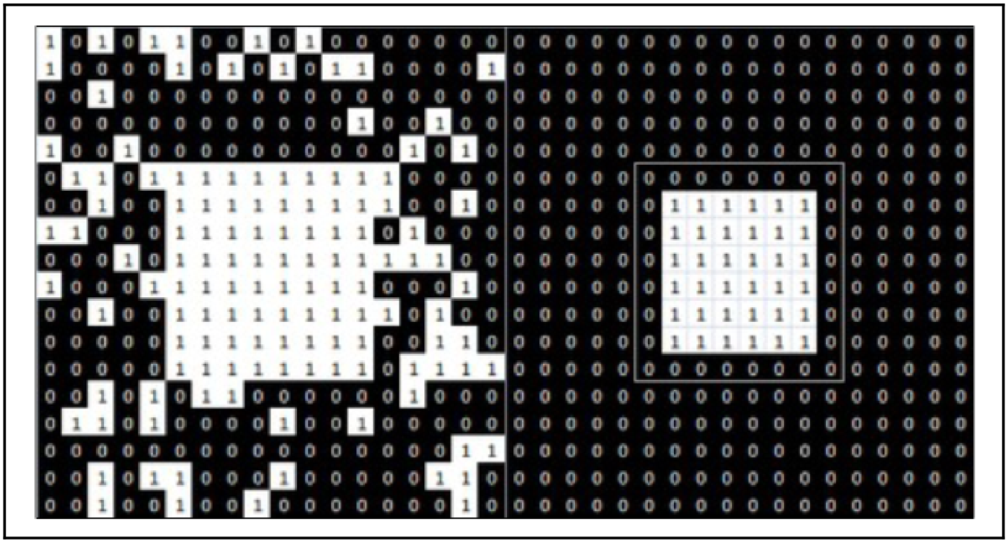

Kernel
Convolved Image
$\begin{bmatrix}0 & 0 & 0\\0 & 1 & 0\\0 & 0 & 0\end{bmatrix}$
$\begin{bmatrix}1 & 0 & -1\\0 & 0 & 0\\-1 & 0 & 1\end{bmatrix}$

$$f\left(x\right)=\frac{1}{\sigma\sqrt{2\pi}}e^{-\left(x-\mu\right)^2/\left(2\sigma^2\right)}$$

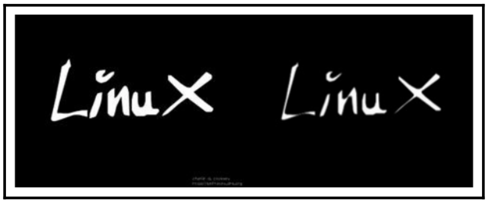


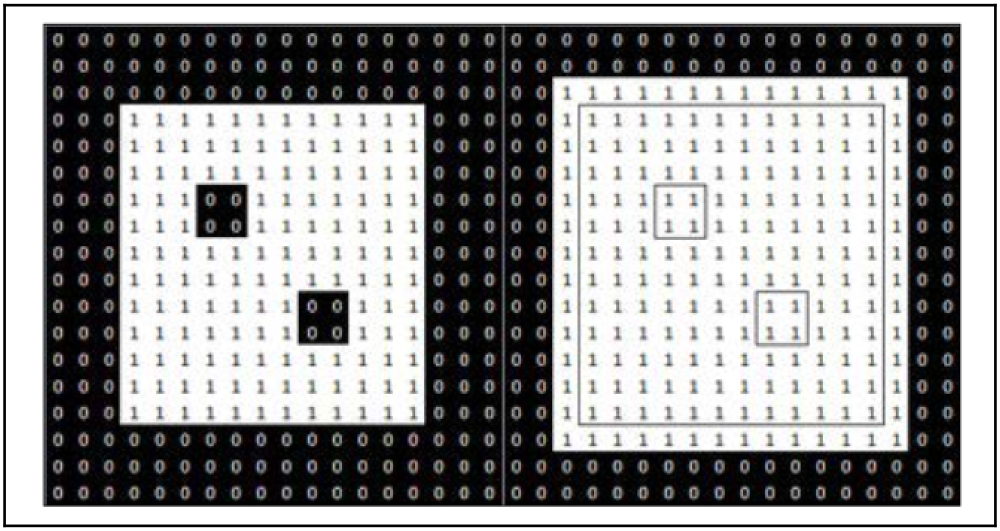




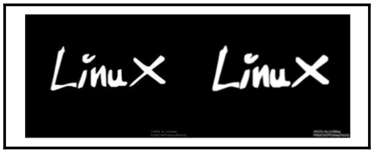


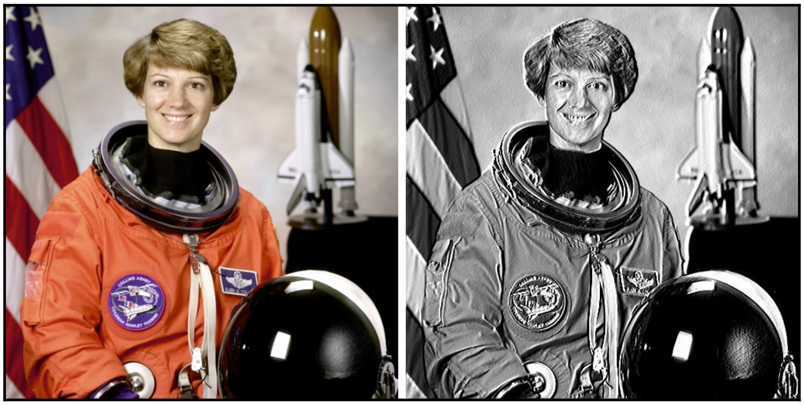
```
1 0 1 0 1 1 0 0 1 0 1 0 0 0 0 0 0 0 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 1 0 1 0 1 0 1 1 0 0 0 0 1 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 1 0 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 0 1 1 1 1 1 1 1 1 1 1 0 0 0 0 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 0 0 1 1 1 1 1 1 1 1 1 0 0 1 0 | 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0
1 1 0 0 0 1 1 1 1 1 1 1 1 0 1 0 0 0 | 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 1 0 1 1 1 1 1 1 1 1 1 1 1 0 0 | 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0
1 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 1 0 | 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0
0 0 1 0 0 1 1 1 1 1 1 1 1 1 0 1 0 0 | 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 1 1 1 0 0 1 1 0 | 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 1 1 1 0 1 1 1 1 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 0 1 0 1 1 0 0 0 0 0 0 1 0 0 0 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 0 1 0 0 0 0 1 0 0 1 0 0 0 0 0 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 0 1 1 0 0 0 1 0 0 0 0 0 1 1 0 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 0 0 1 0 0 1 0 0 0 0 0 0 0 1 0 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```



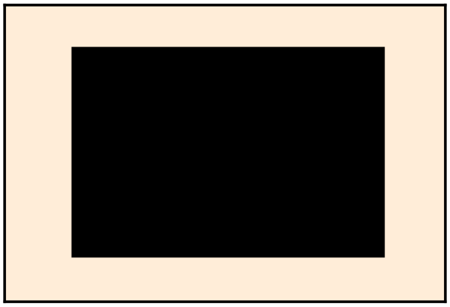


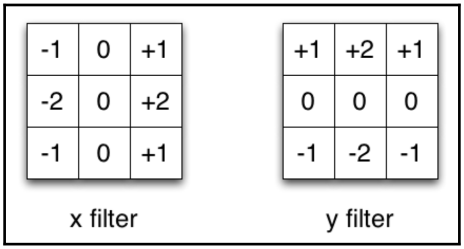
```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 | 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 | 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 | 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 1 1 1 0 0 1 1 1 1 1 1 1 0 0 0 | 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 1 1 1 0 0 1 1 1 1 1 1 1 0 0 0 | 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 | 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 | 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 1 1 1 1 1 1 1 0 0 1 1 1 0 0 0 | 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 1 1 1 1 1 1 1 0 0 1 1 1 0 0 0 | 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 | 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 | 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 | 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```





$$\nabla f = \frac{\partial f}{\partial x} i + \frac{\partial f}{\partial y} j + \frac{\partial f}{\partial z} k$$



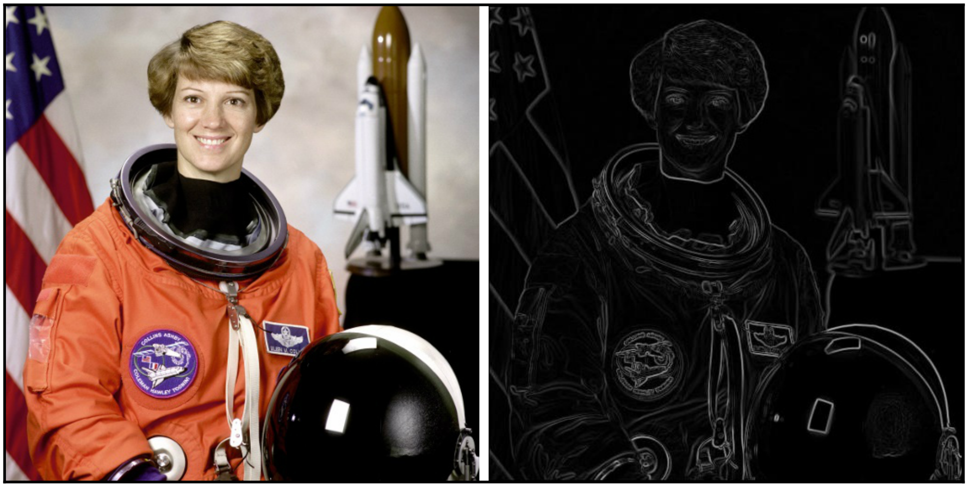


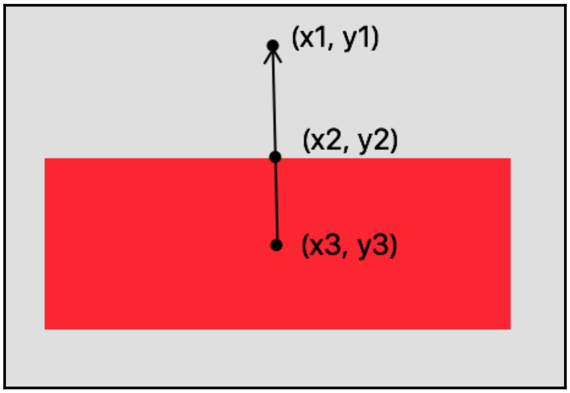
-1	0	+1
-2	0	+2
-1	0	+1

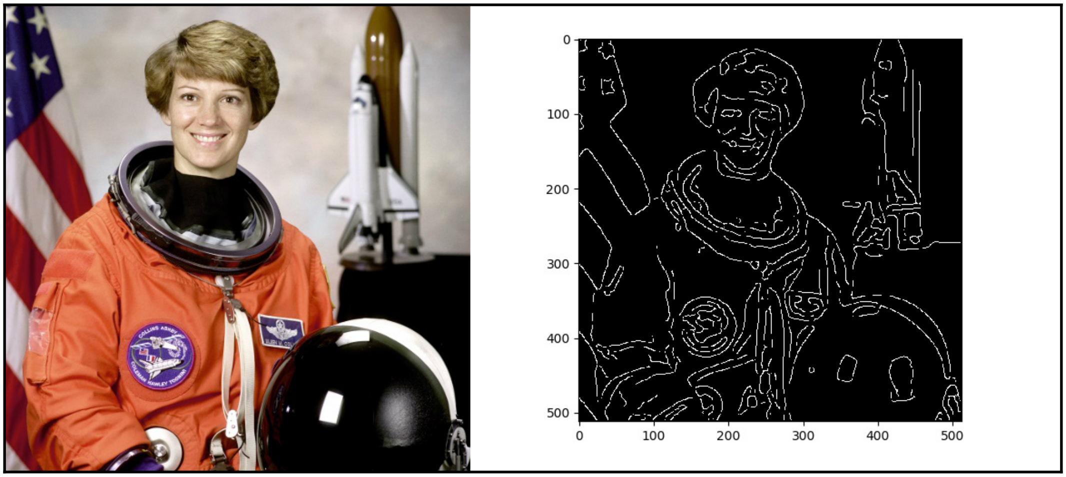
x filter

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

y filter




(x1, y1)
(x2, y2)
(x3, y3)



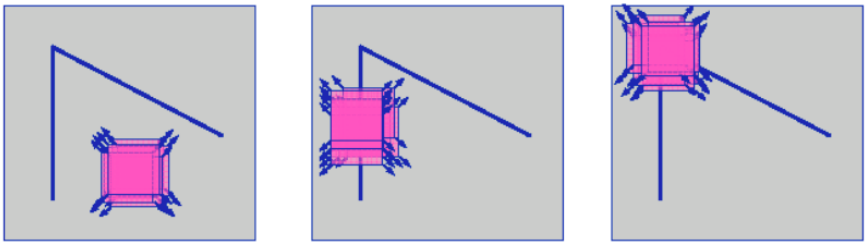
$$\boxed{y = mx + c}$$

$$\boxed{y_1 = mx_1 + c}$$

$$\boxed{y_2 = mx_2 + c}$$

$$\boxed{(x-h)^2 + (y-k)^2 = r^2}$$

Chapter 03: Drilling Deeper into Features - Object Detection

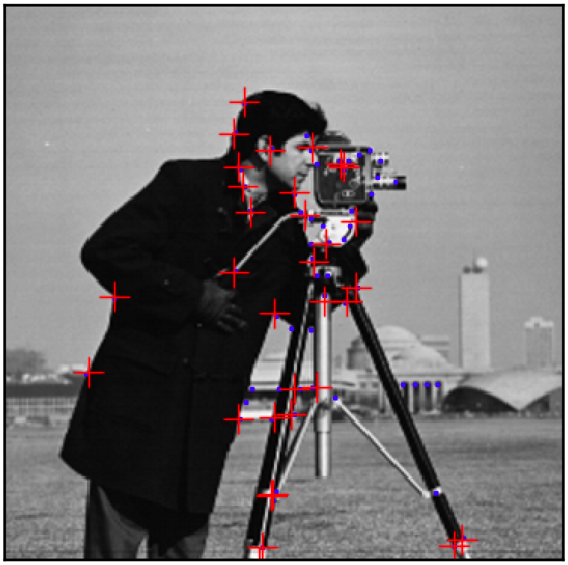


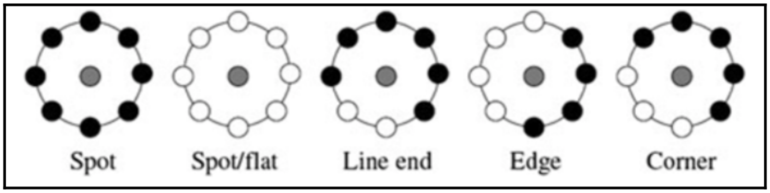
"flat" region:
no change in
all directions

"edge":
no change along
the edge direction

"corner":
significant change
in all directions

$$\boxed{\sum \left[I(x+u, y+v) - I(x,y) \right]^2}$$





$$\boxed{m_{pq} = \sum_{x,y} x^p y^p I(x,y)}$$

$$\boxed{C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}}\right)}$$

$$\boxed{\theta = atan2\left(m_{01}, m_{10}\right)}$$

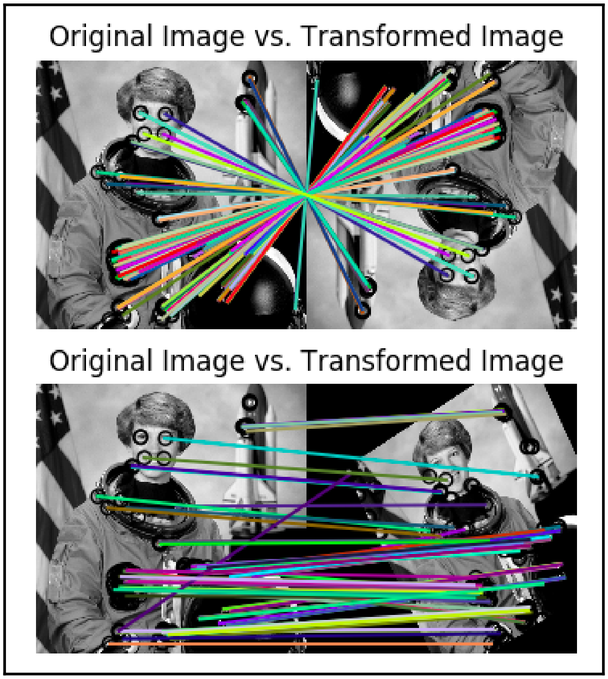
$$\boxed{f(n) = \sum_{1<i<n} 2^{i-1}\tau\left(p; x_i, y_i\right)}$$

$$\boxed{\begin{array}{l} Where\,\tau(p;x,y)\,is\,defined\,as: \\ \qquad\qquad\qquad \tau(p;x,y) = \begin{cases} 1 & : p(x) < p(y) \\ 0 & : p(x) \geq p(y) \end{cases} \\ p(x)\,is\,the\,\mathrm{int}\,ensity\,value\,at\,pixel\,x. \end{array}}$$

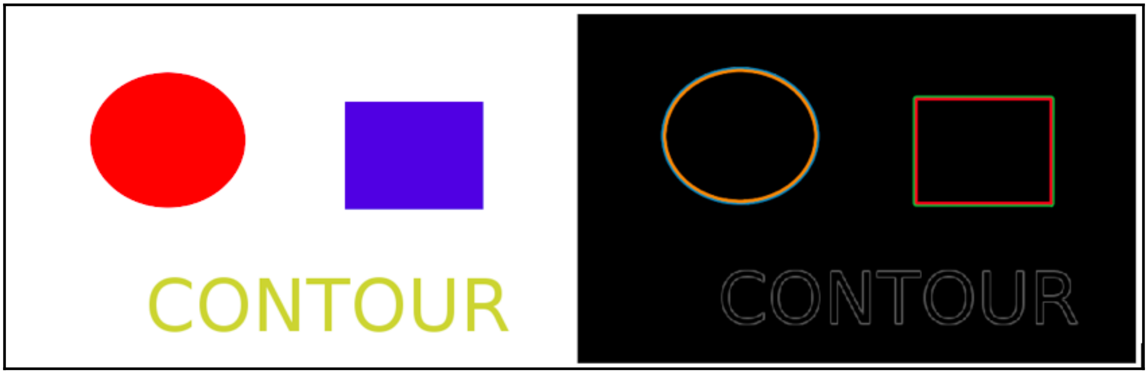
$$\boxed{S = \begin{pmatrix} x1,\dots x_n \\ y1,\dots y_n \end{pmatrix}}$$

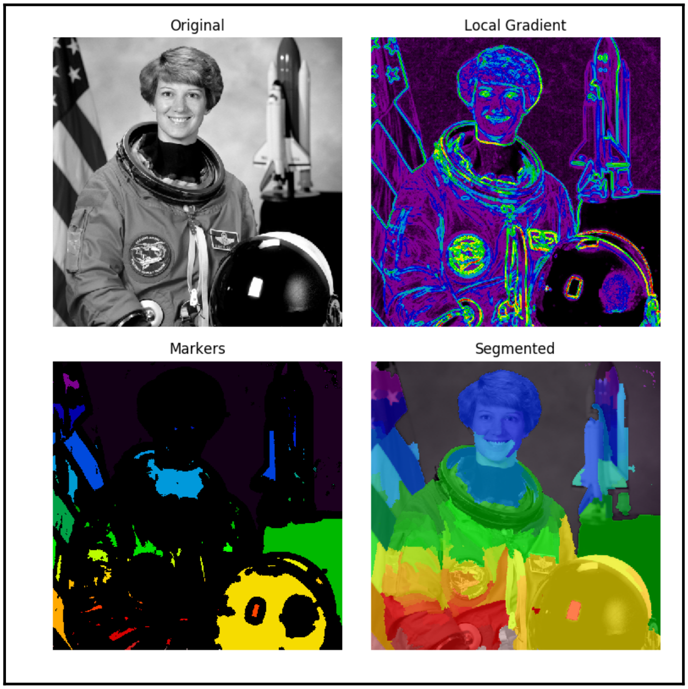
$$\boxed{S_\theta = R_\theta S}$$

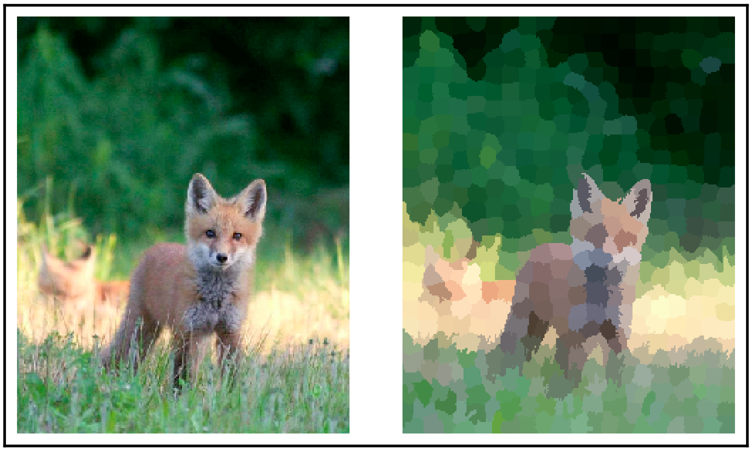
$$\boxed{g_n(p,\theta) = f_n(p) \mid \left(\mathrm{x}_i, \mathrm{y}_i\right) \in S_\theta}$$

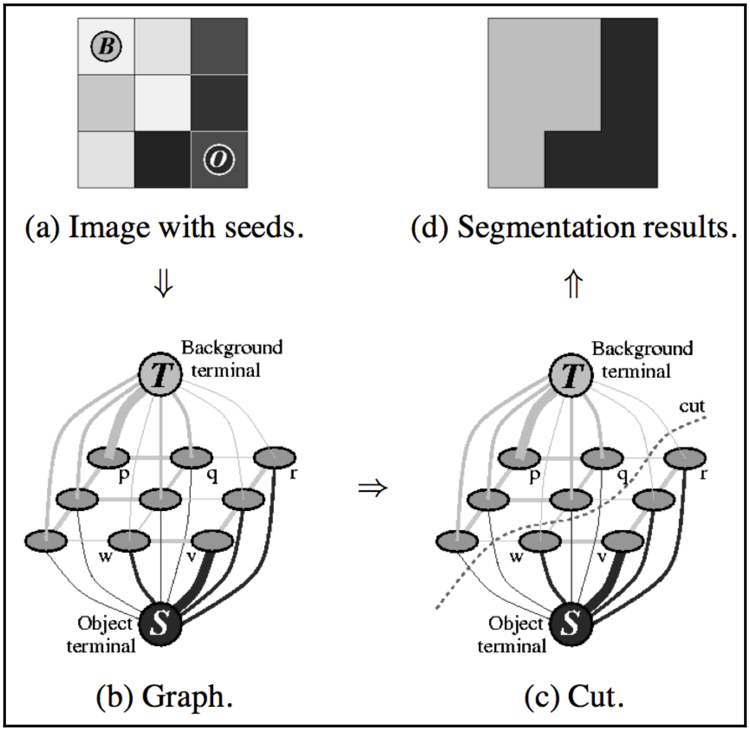

Original Image vs. Transformed Image
Original Image vs. Transformed Image

Chapter 04: Segmentation - Understanding Images Better



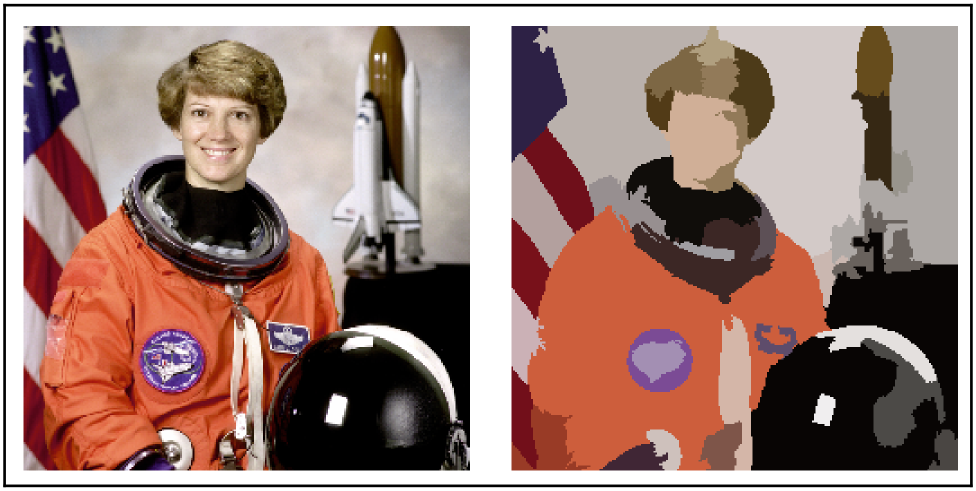

Original
Local Gradient
Markers
Segmented



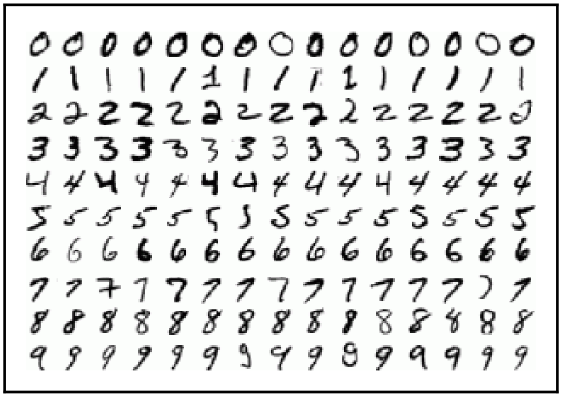


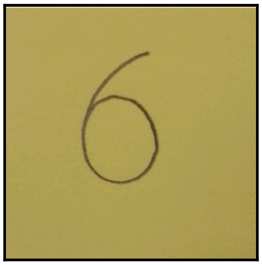
(a) Image with seeds. ⇓ (b) Graph. ⇒ (c) Cut. ⇑ (d) Segmentation results.

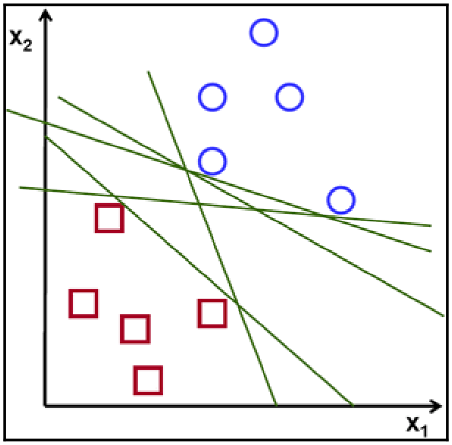
$$\boxed{\text{edge weight} = e^{-d^2/sigma}}$$

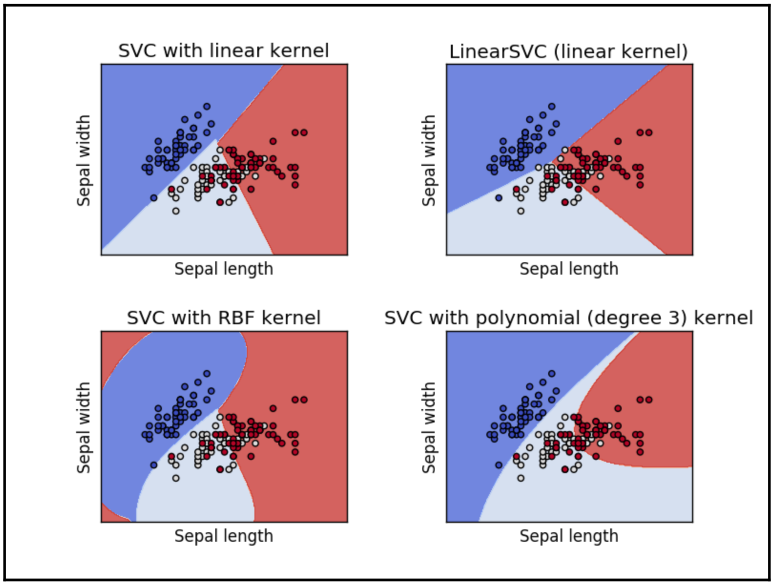


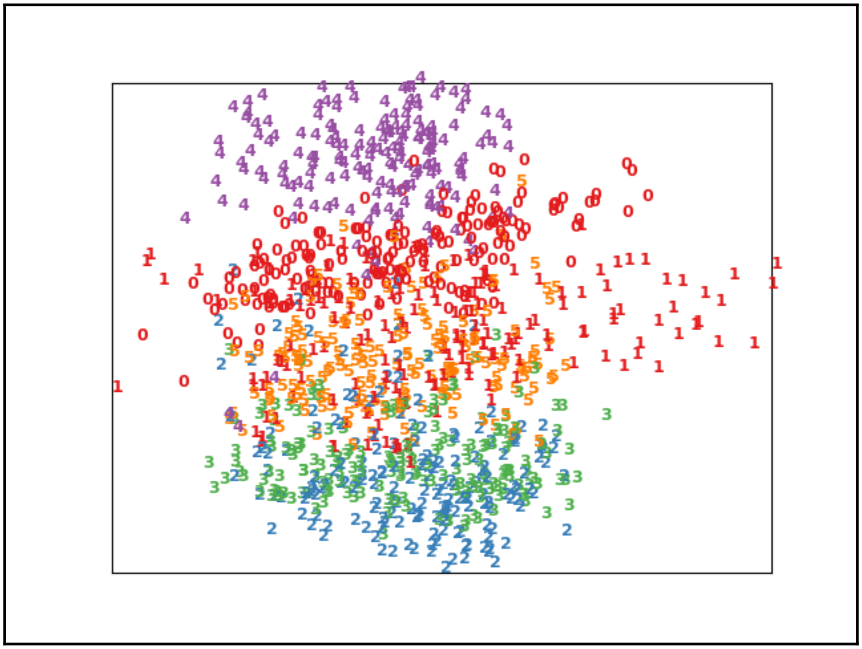
Chapter 05: Integrating Machine Learning with Computer Vision

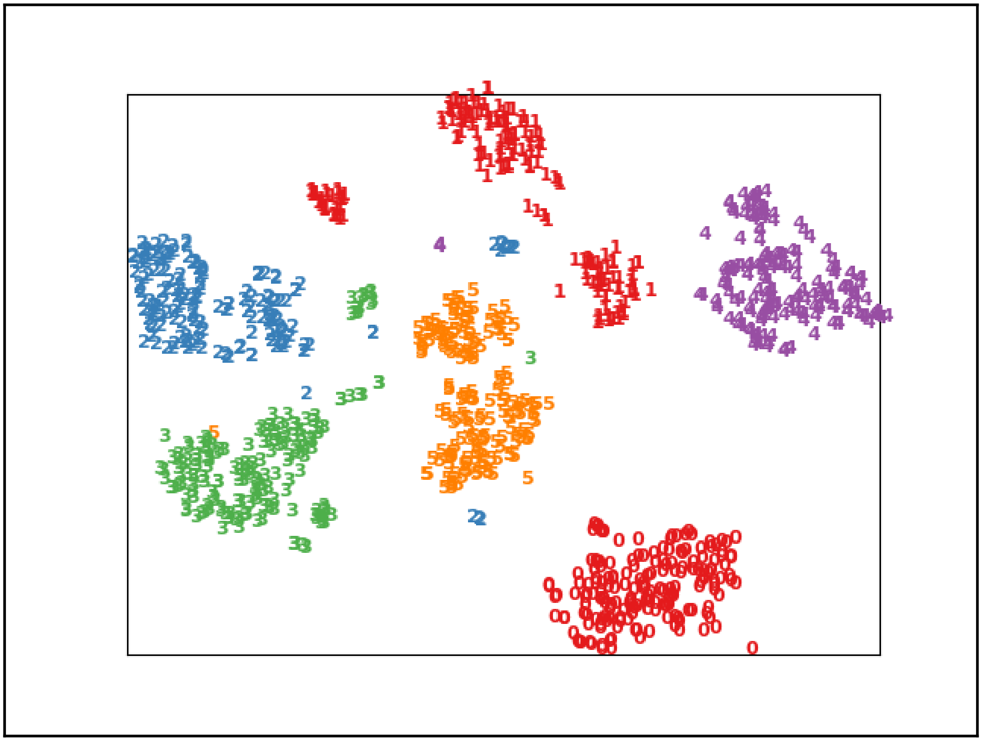




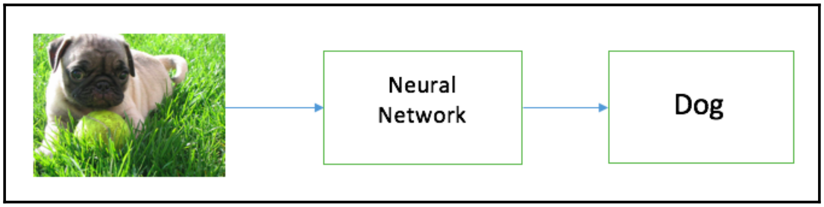

X2
X1

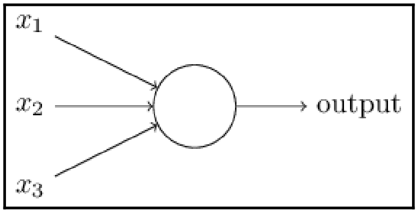

SVC with linear kernel
Sepal width
Sepal length
LinearSVC (linear kernel)
Sepal width
Sepal length
SVC with RBF kernel
Sepal width
Sepal length
SVC with polynomial (degree 3) kernel
Sepal width
Sepal length

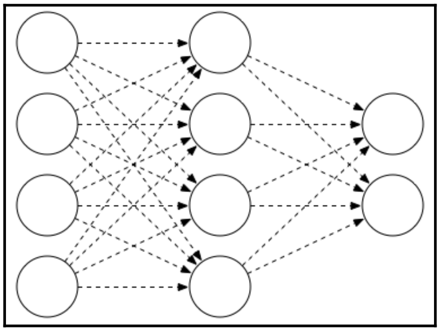


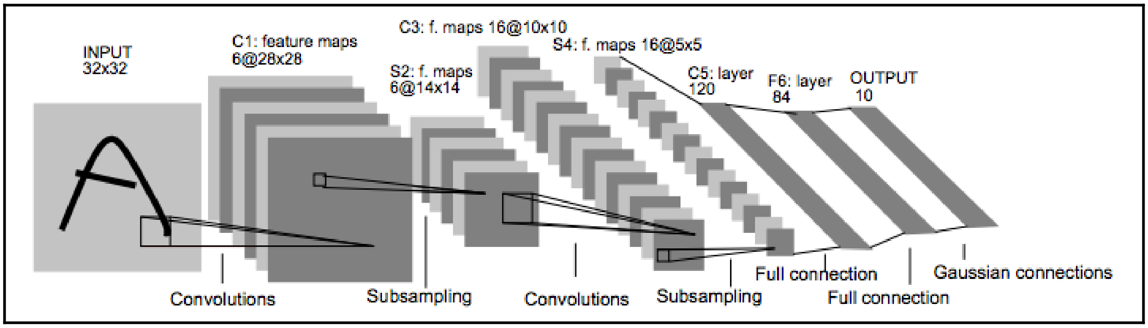


Chapter 06: Image Classification Using Neural Networks

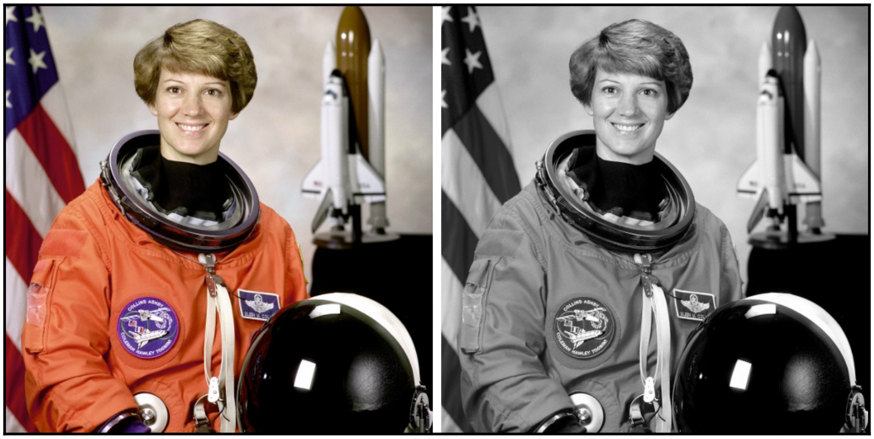


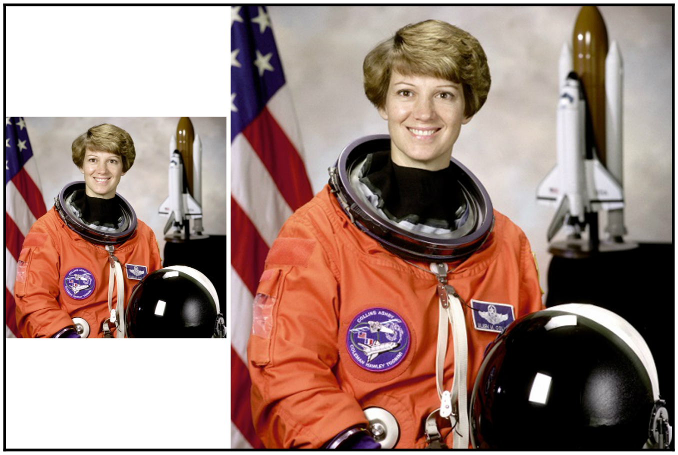


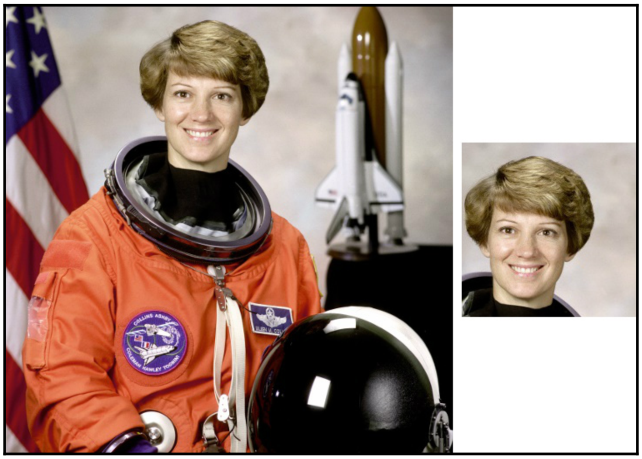




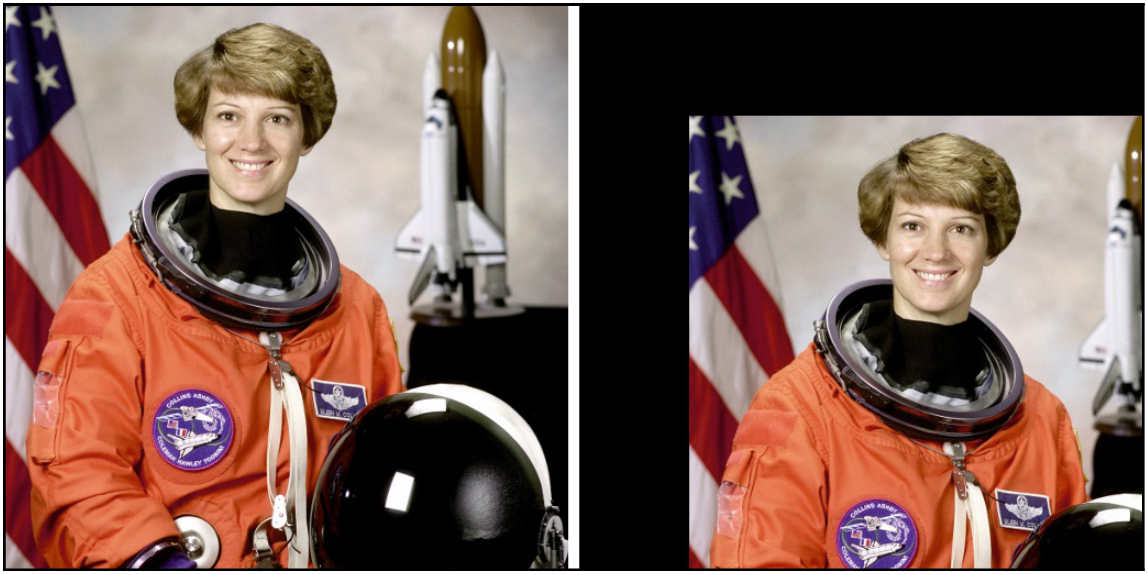
Chapter 07: Introduction to Computer Vision using OpenCV



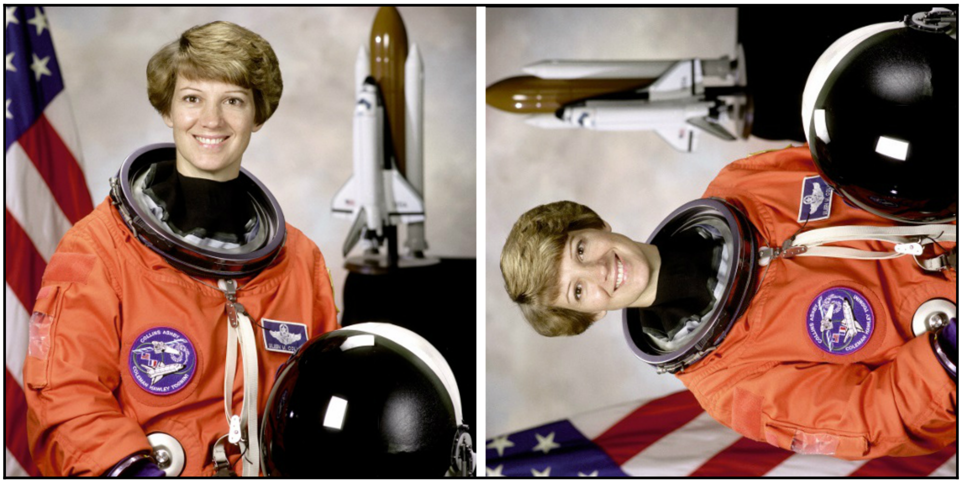


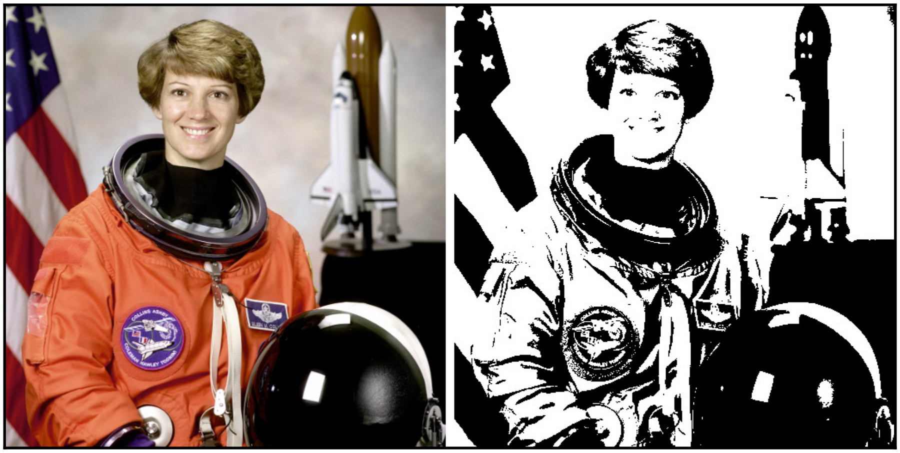


$$\boxed{M = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \end{bmatrix}}$$

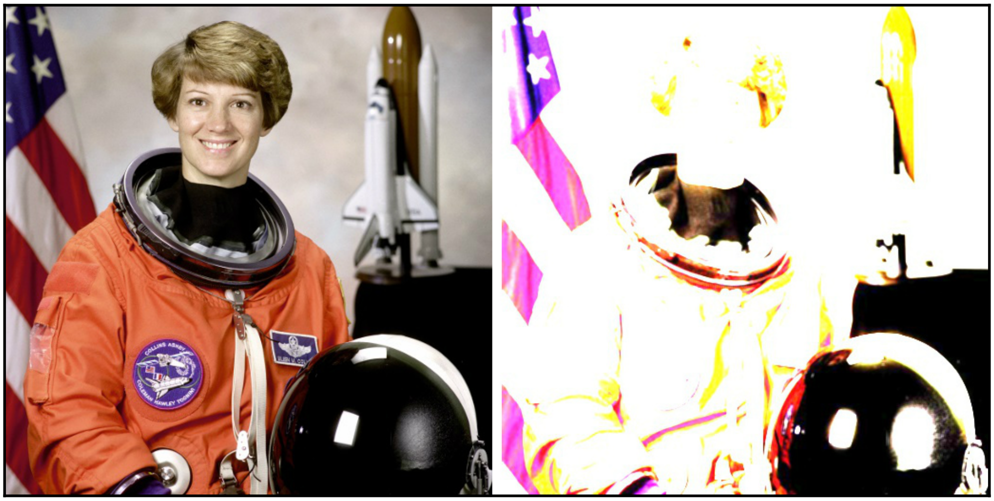


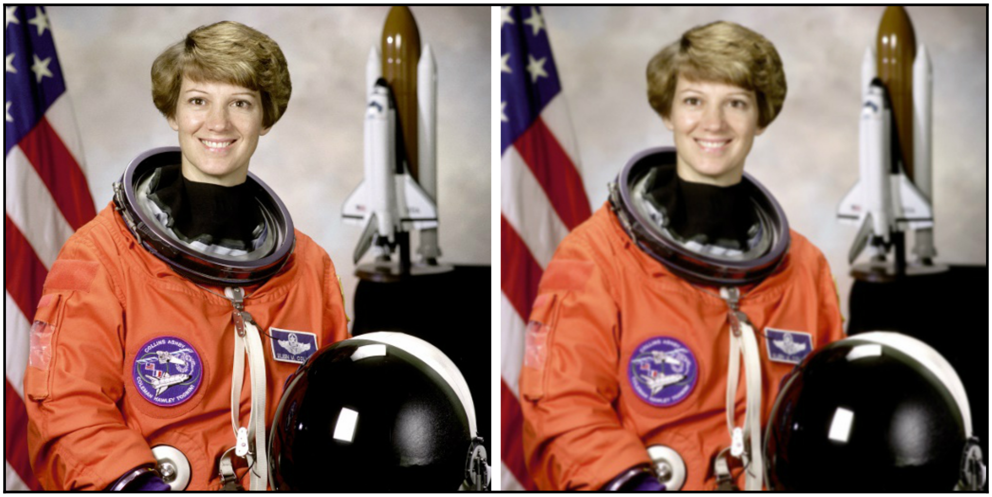
$$\boxed{M = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}}$$

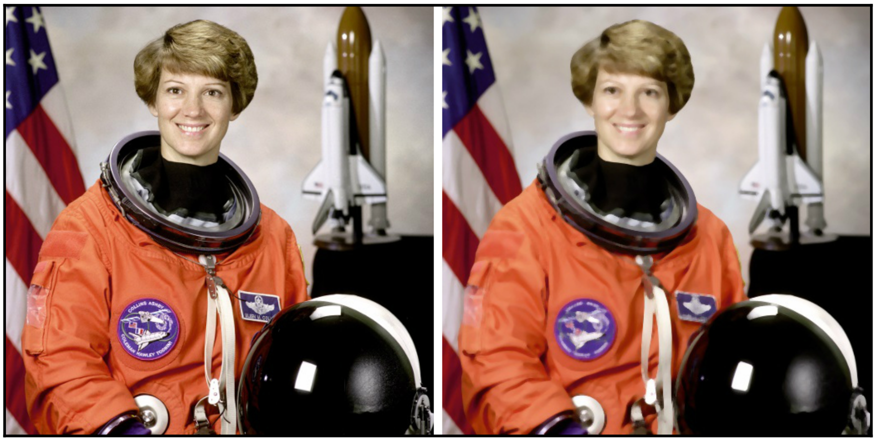


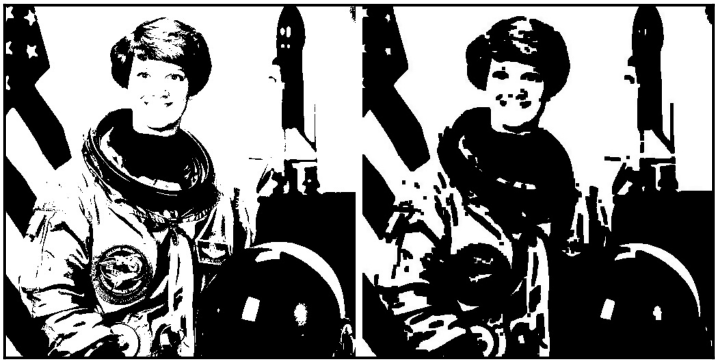


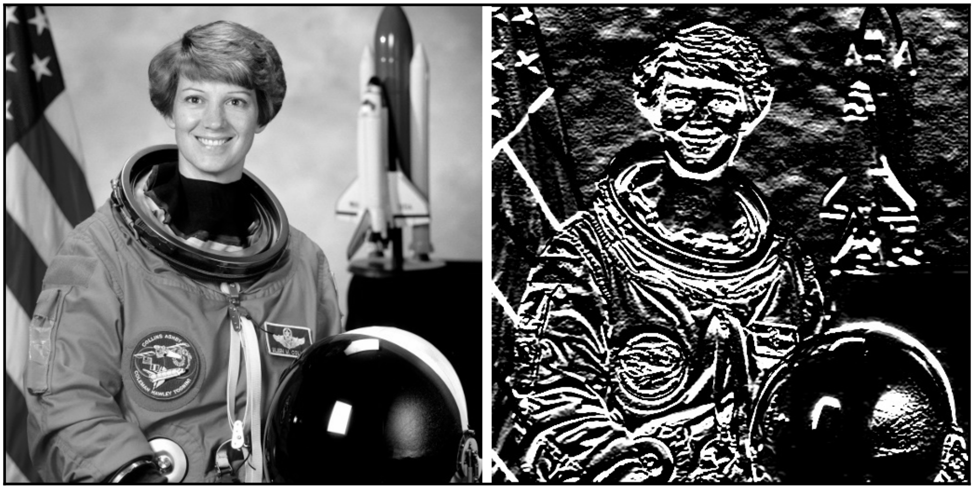
$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

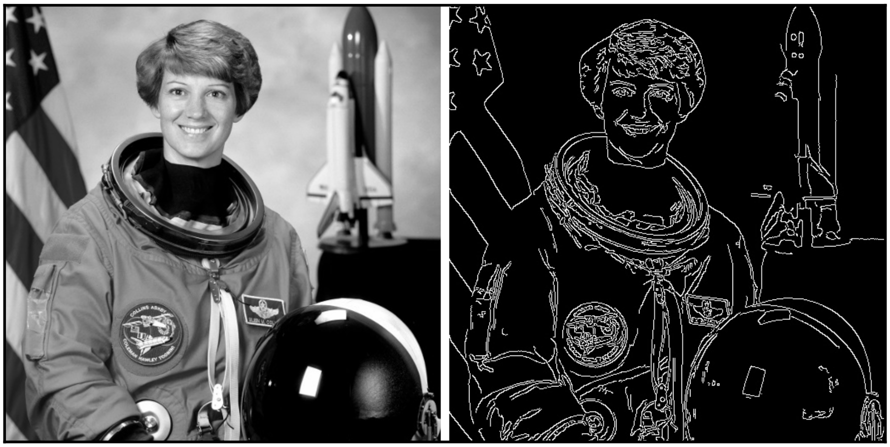


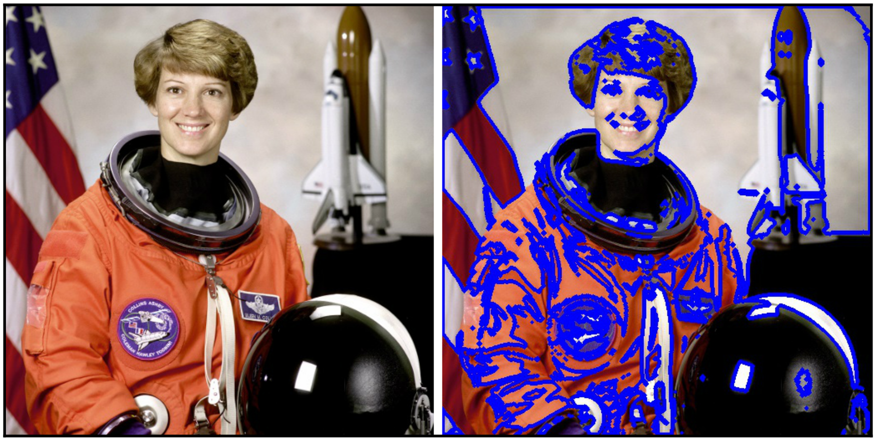


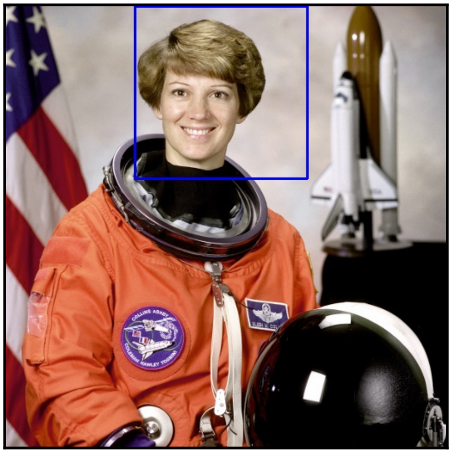




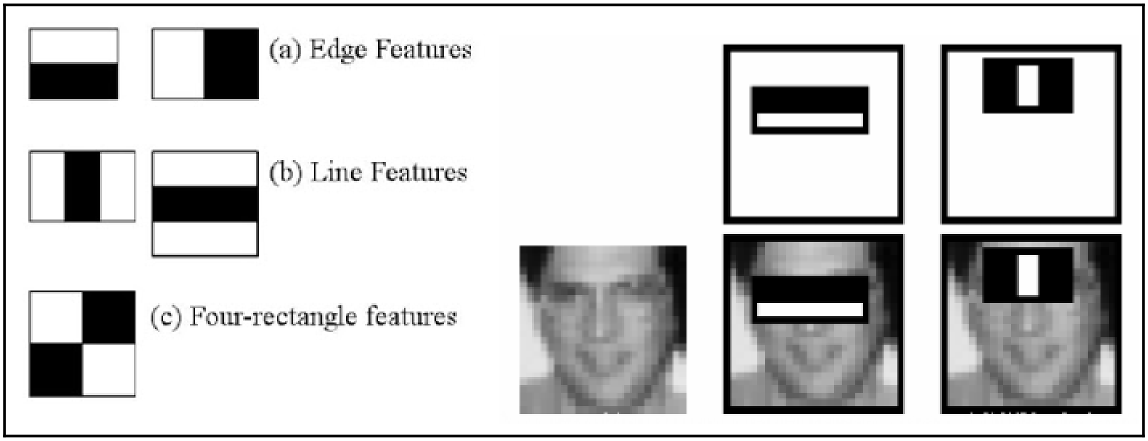






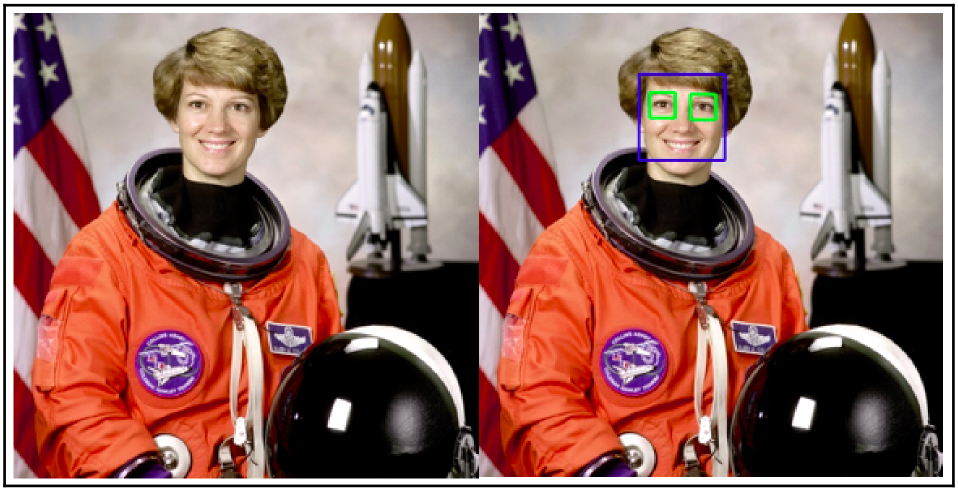


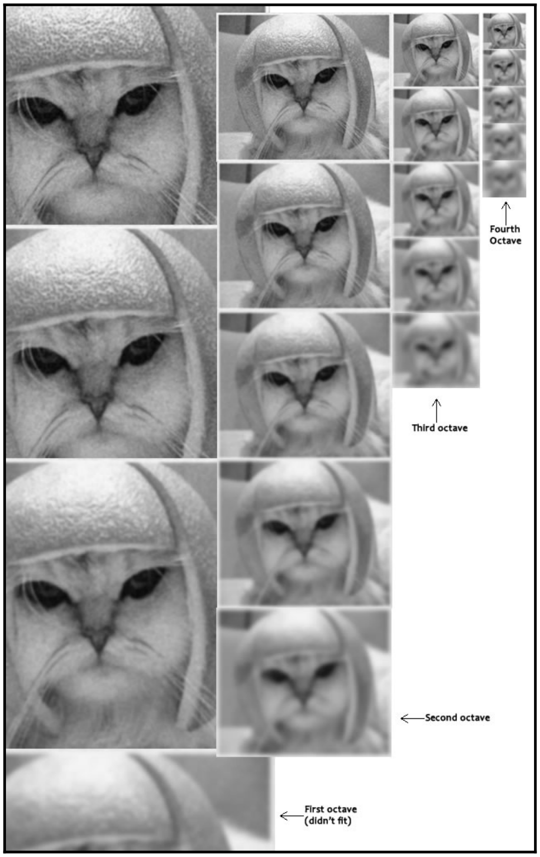
Chapter 08: Object Detection Using OpenCV



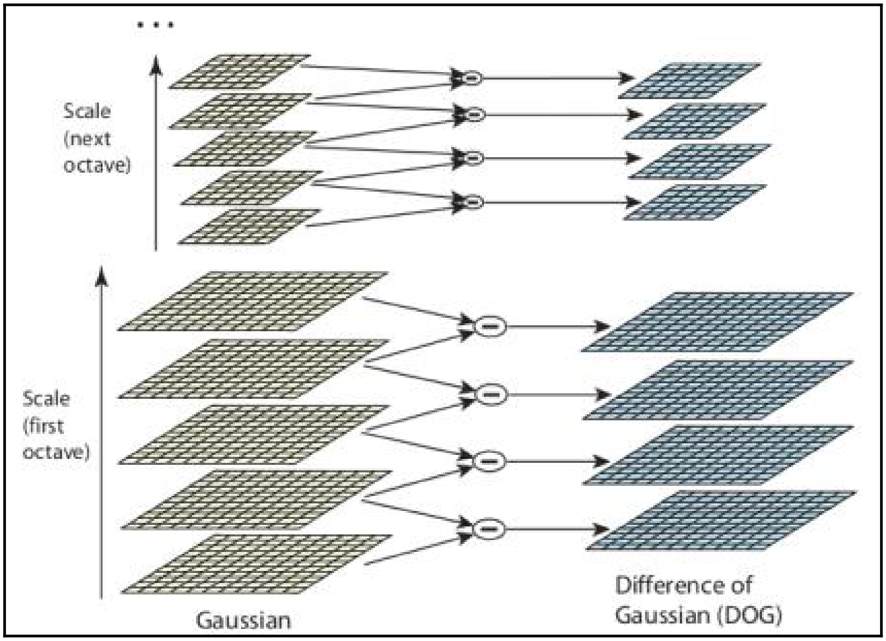
$$I_{\Sigma}(x,y) = \sum_{\substack{x' \le x \\ y' \le y}} i(x',y')$$

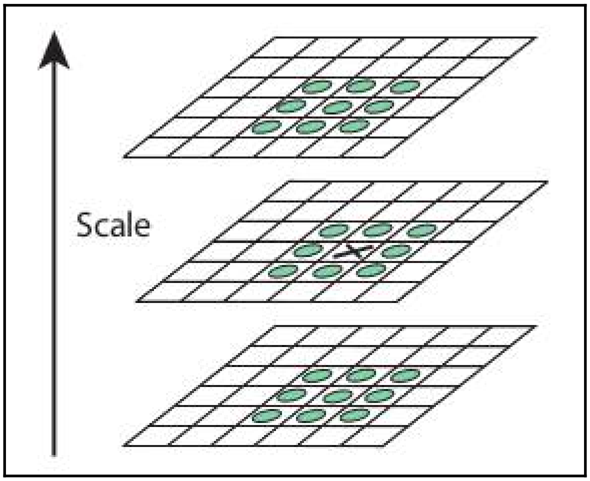
$$I(x,y) = i(x,y) - I(x-1,y-1) + I(x,y-1) + I(x-1,y)$$





$$\boxed{\begin{aligned} D(x,y,\sigma) &= \left(G(x,y,k\sigma) - G(x,y,\sigma)\right) * I(x,y) \\ &= L(x,y,k\sigma) = L(\mathrm{x},\mathrm{y},\sigma) \end{aligned}}$$

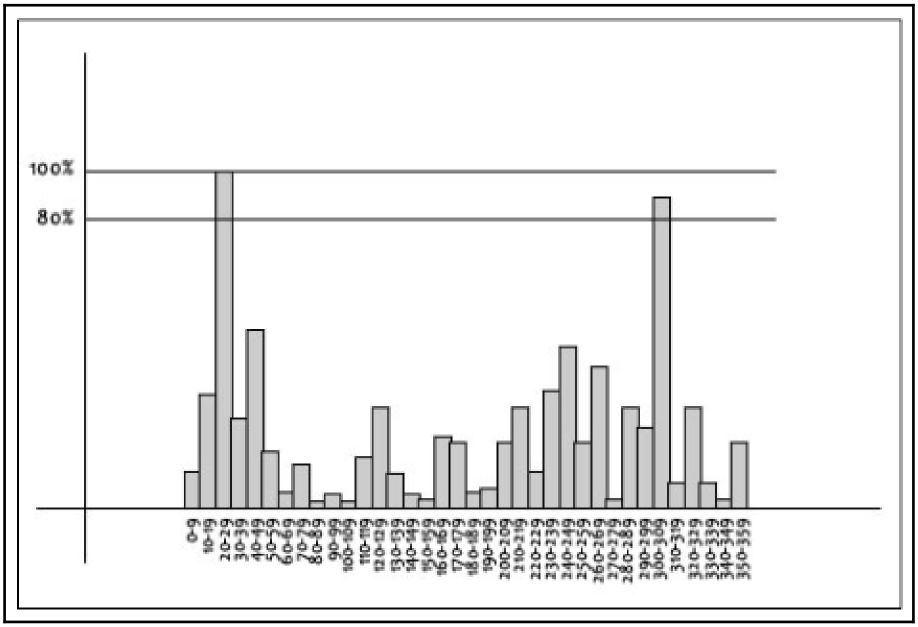


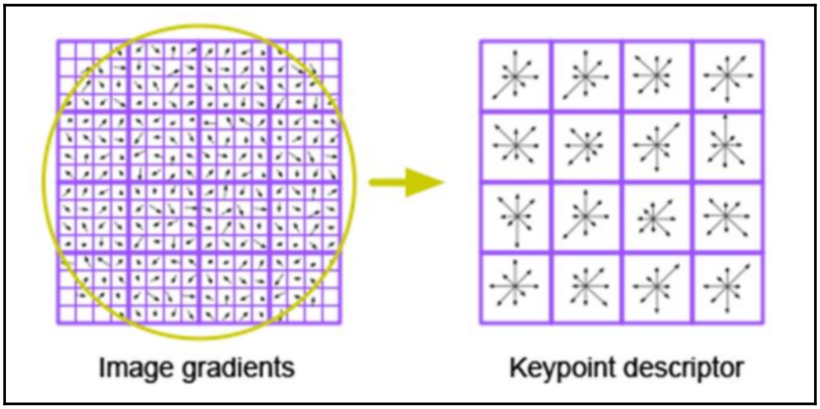


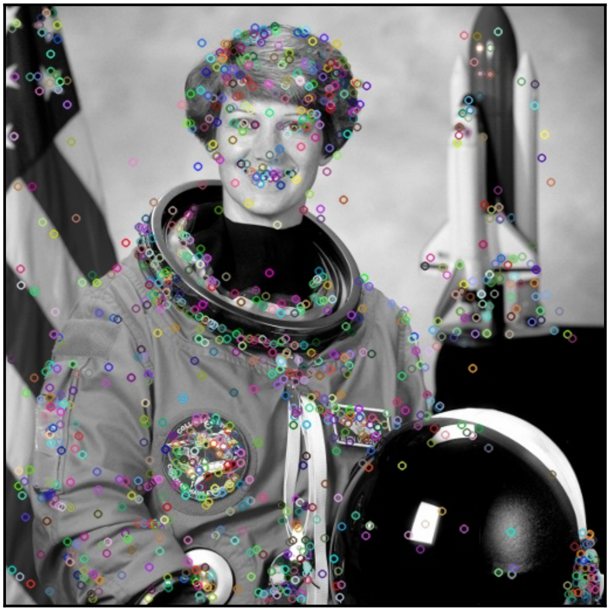
$$\boxed{D\left(x\right) = D + \frac{\partial D^{T}}{\partial x} x + \frac{1}{2} x^{T} \frac{\partial^{2} D}{\partial x^{2}} x}$$

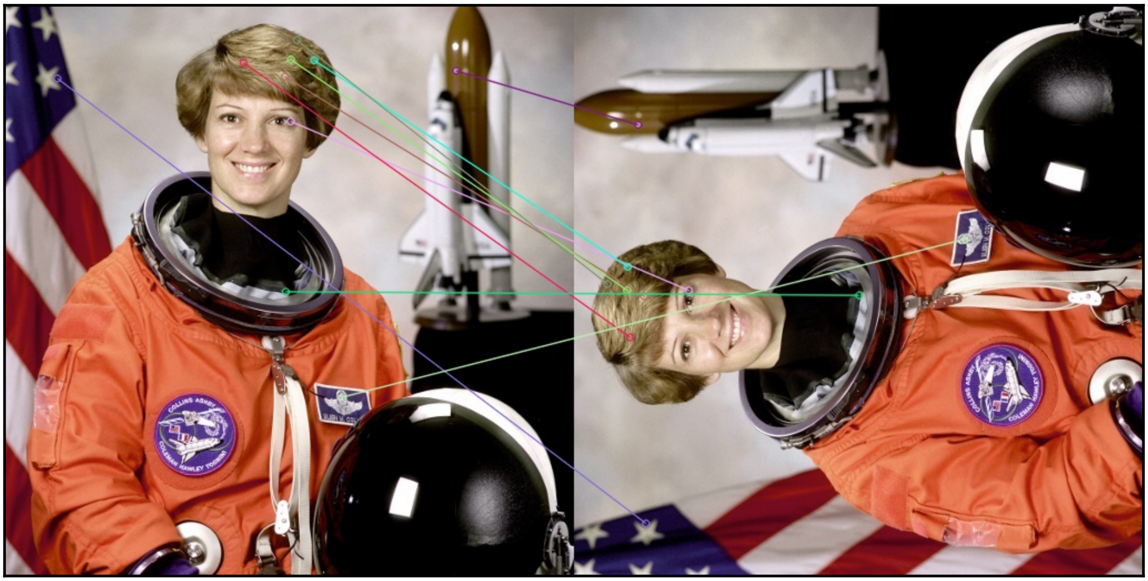
$$m(x,y)=\sqrt{\left(L(x+1,y)-L(x-1,y)\right)^2+\left(L(x,y+1)-L(x,y-1)\right)^2}$$

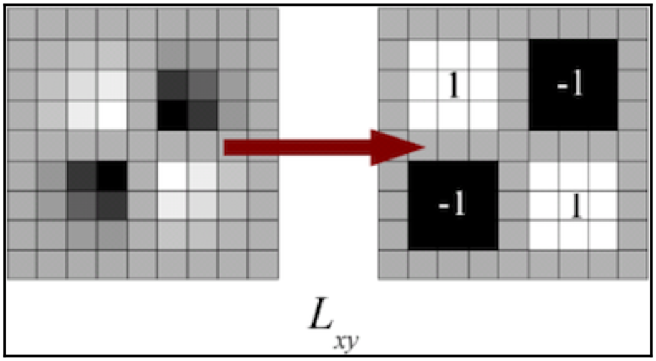
$$\theta(x,y)=\tan^{-1}\left(\left(L(x,y+1)-L(x,y-1)\right)/\left(L(x+1,y)-L(x-1,y)\right)\right)$$

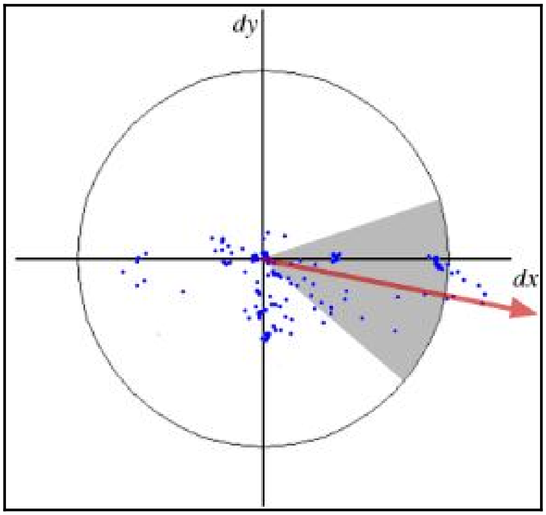


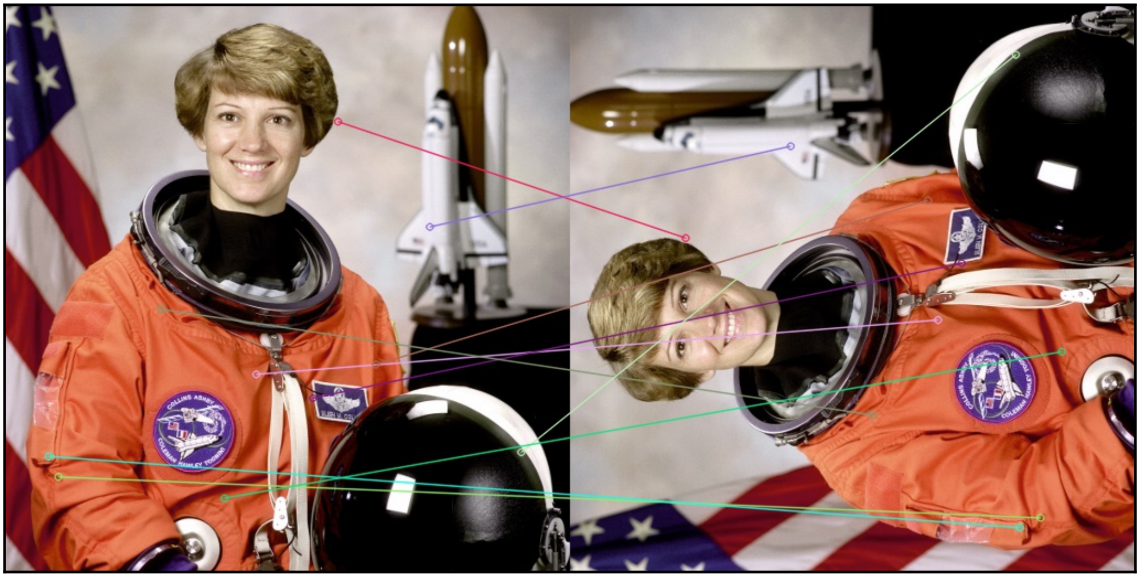




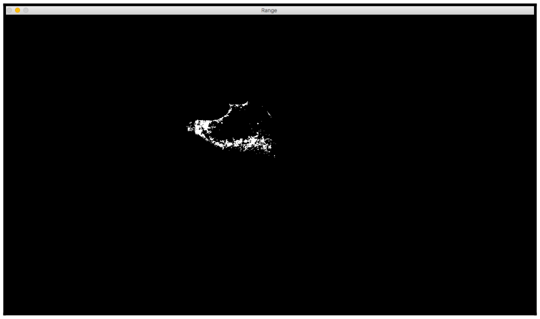



1
-1
-1
1
L_{xy}


dy
dx

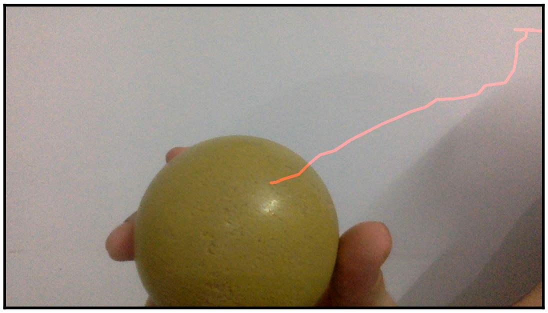


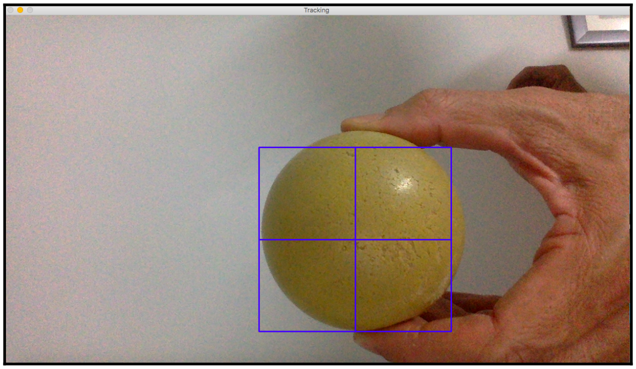
Chapter 09: Video Processing Using OpenCV

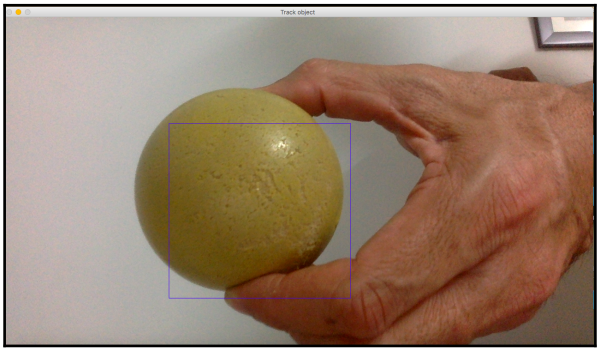


$$\boxed{m_{ji} = \sum_{x,y} \left(array(x,y) \cdot x^{j} \cdot y^{i} \right)}$$

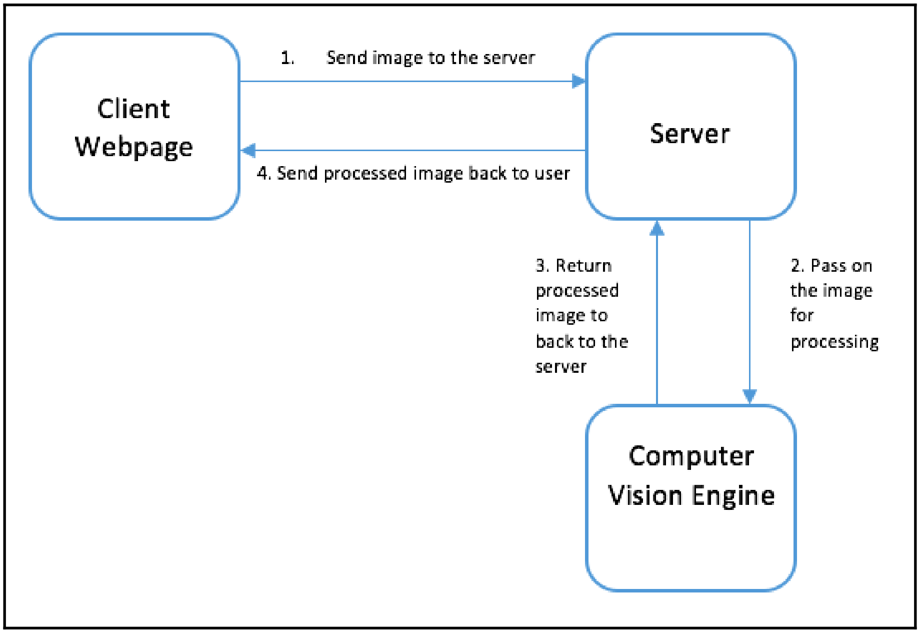
$$\boxed{x = \frac{m10}{m00}} \boxed{y = \frac{m01}{m00}}$$

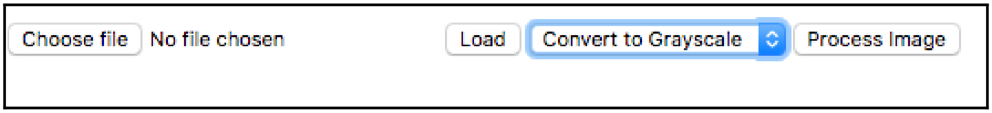


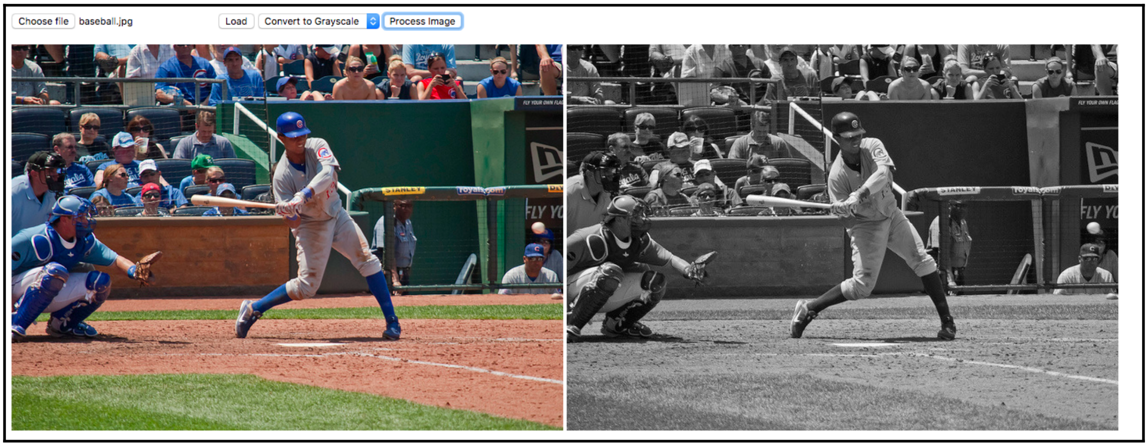

Tracking

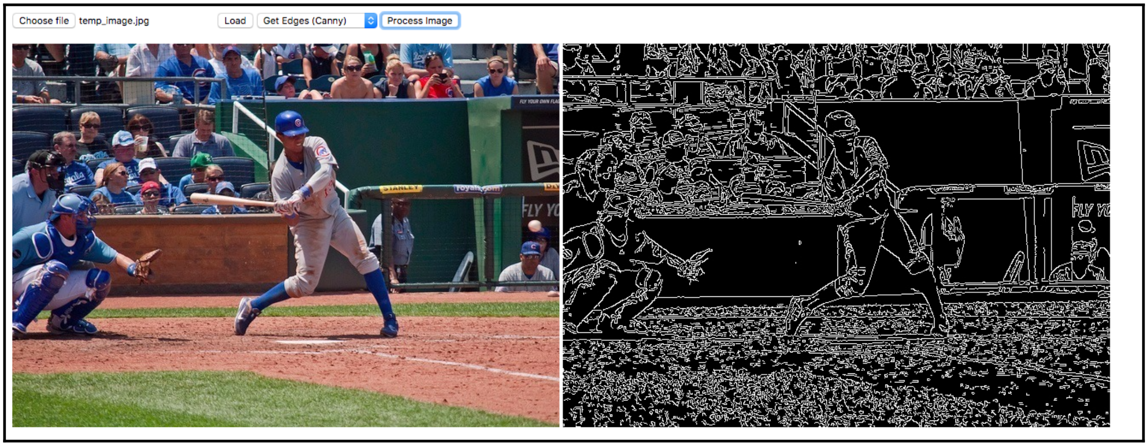

Track object

Chapter 10: Computer Vision as a Service


Client Webpage
1. Send image to the server
4. Send processed image back to user
Server
3. Return processed image to back to the server
2. Pass on the image for processing
Computer Vision Engine






Choose file
temp_image.jpg
Load
Get Edges (Canny)
Process Image