Image Processing

Introduction

Fernando Ribeiro

Introduction

Vision consists on the ability to see and recognize objects through the quantity of light those objects reflect to an image and process that image.
The human eye focus the light on the retina where the intensity of light is receive on. This information is then sent to the brain which perceives that image and identifies objects on the scene.

The artificial vision uses computers or others electronic devices to analyze images and recognize relevant objects to the application.

Image Processing consists of working or transforming the image in such way that it could be recognized analyzed and/or perceived, so that decisions can be taken from that image.

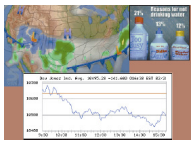
Agenda

- Introduction and History
- Digital Image Representation
- Digital Image Processing System Elements
- Digital image Properties
- Image Fundamentals
- Graphical Algorithms
- Digital Image Properties
  - Metrics
  - Histograms
  - Image Visual Perception
- Colour Images
- Coding and compression
- Graphical Formats

Introduction and History

Digital Image processing has two main applications:

- Human Interpretation - Improving image (data) quality to be analyzed by humans
- Computer Perception - To process the image data (information) to be automatically and autonomously perceived by a machine

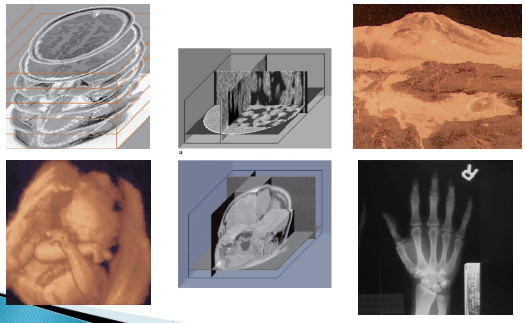


Introduction
Human Interpretation

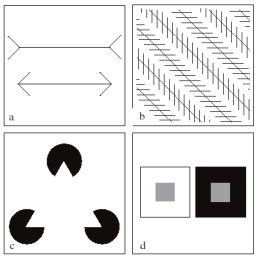
Used in almost all areas:

- Space
- Medicine
- Geology
- Archeology
- Physics
- Astronomy, Biology
- Defense

Introduction
Human Interpretation



Introduction
Human Interpretation – difficulties

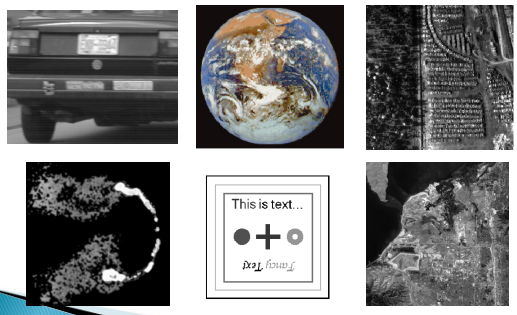


Introduction
Computer perception

Used in most industries:

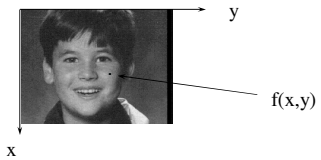
- Optical Characters Recognition
- Robot assembly line
- Industrial Inspection
- Military analysis
- Finger print Processing
- Blood Analysis
- X-Ray
- Weather forecast

Introduction
Computer perception

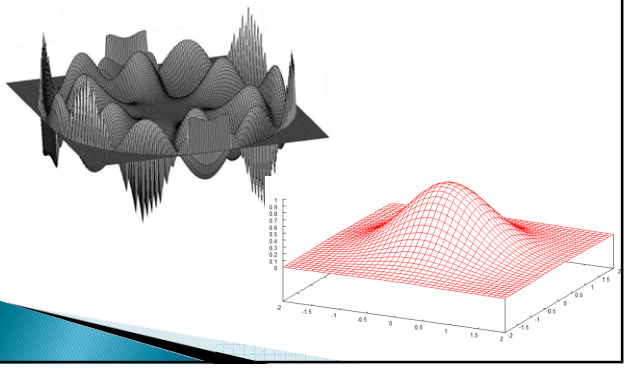


Digital Image Representation

Monochromatic Image or just *Image*, refers to a bi-dimensional mathematical function *f(x,y)* which represents the light intensity, where *x* and *y* are spatial coordinates and the value of *f* in each position *(x,y)* is proportional to the brightness (or gray level) of the image in that position.



Digital Image Representation



Digital Image Representation

Other Representation

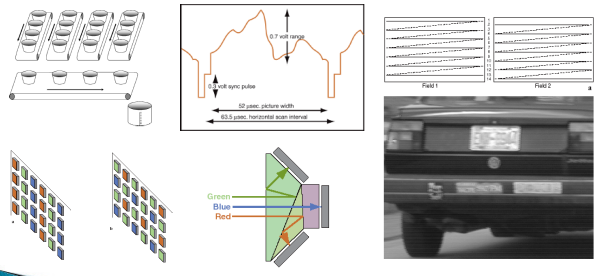
| | | |

D. I. P. System Elements
Digitizers

This device converts an image into a numeric representation able to be used as input in a computer. Examples are:

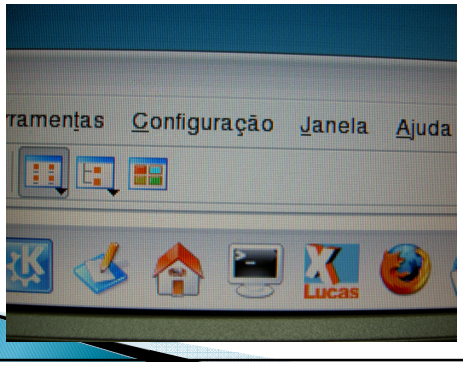
- Video Cameras
- Digital photographic Machine
- *Scanners*
- Photo sensitive Arrays

D. I. P. System Elements
Digitizer

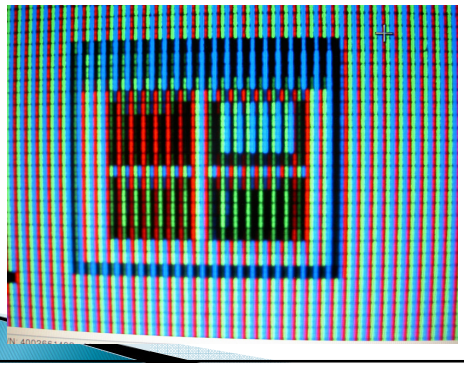


Interlaced / Progressive

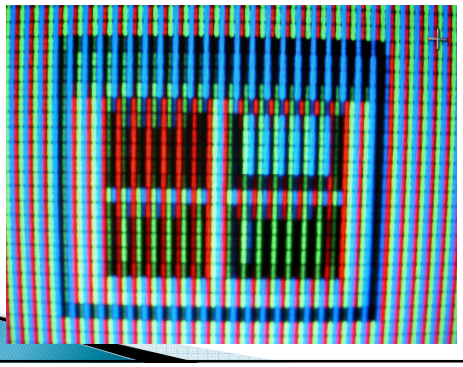
Pixel – LCD



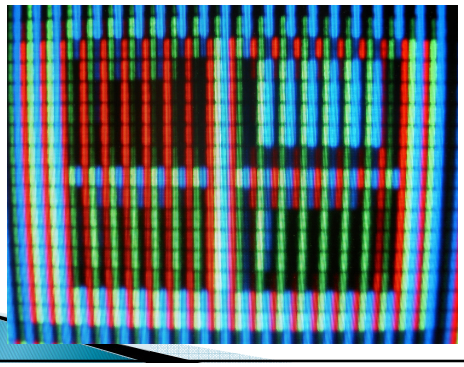
Pixel – LCD



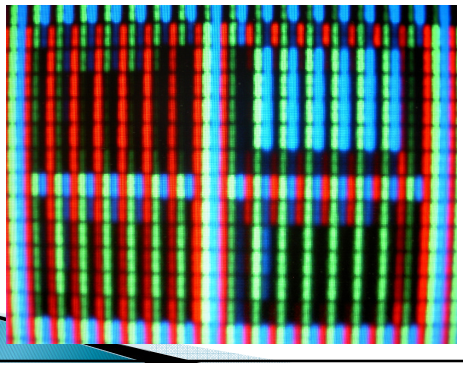
Pixel – LCD



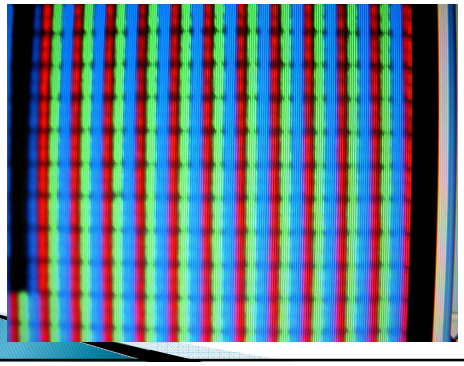
Pixel – LCD



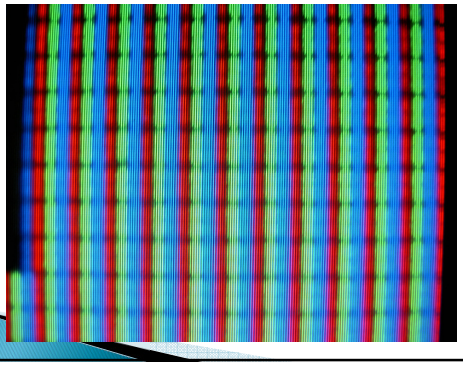
Pixel – LCD



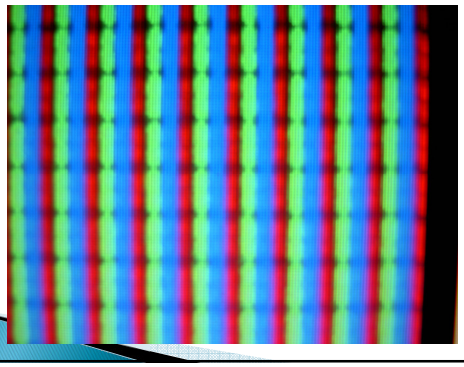
Pixel – LCD



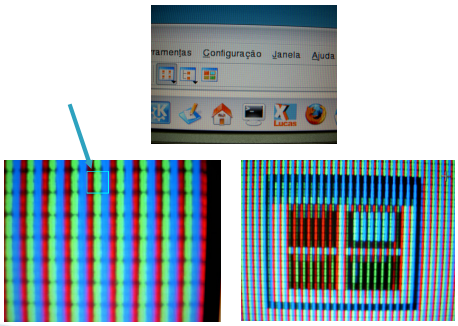
Pixel – LCD



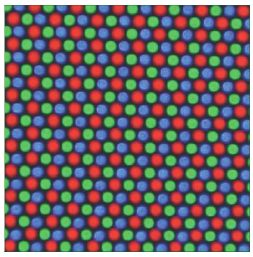
Pixel – LCD



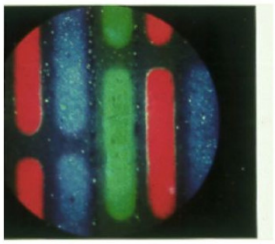
Pixel – LCD



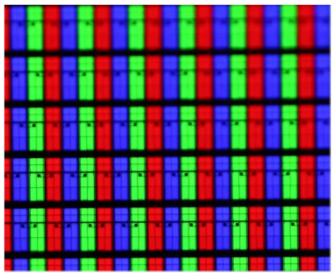
Pixel – CRT



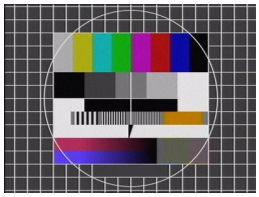
Pixel



Pixel

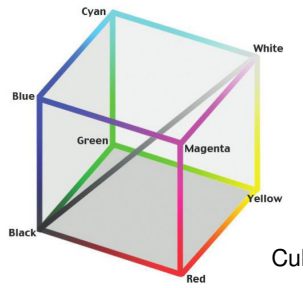


Representação de cor



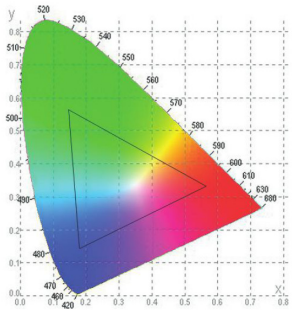
Mira Técnica

Representação de cor



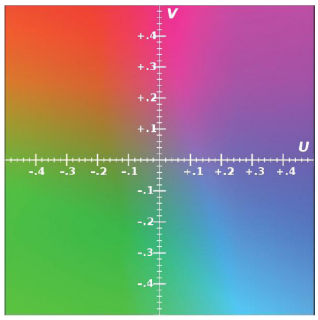
Cubo RGB

Representação de cor

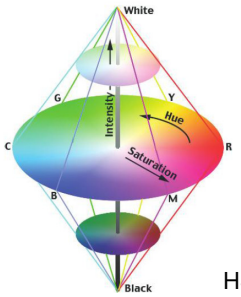


Gamut

Representação de cor

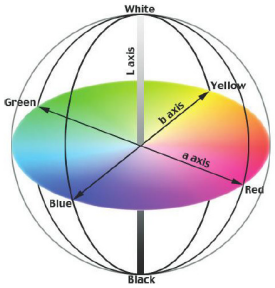


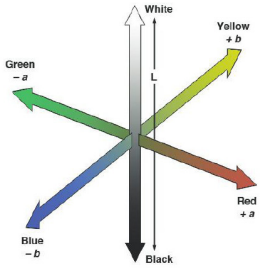
Representação de cor



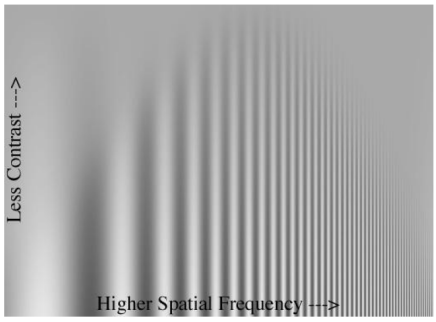
HSL / HLS

Representação de cor

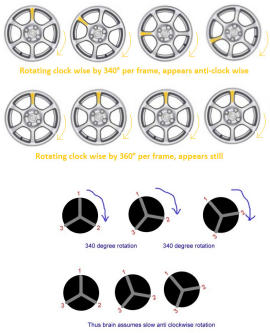




Frequência

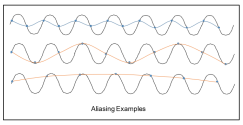

Less Contrast --->
Higher Spatial Frequency --->
Higher Spatial Frequency --->
Higher Spatial Frequency

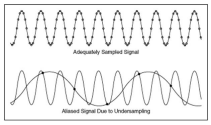
Frequência

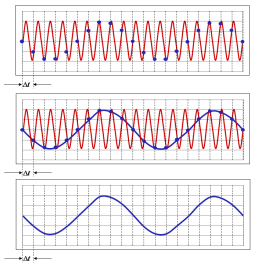


Frequência

Frequência



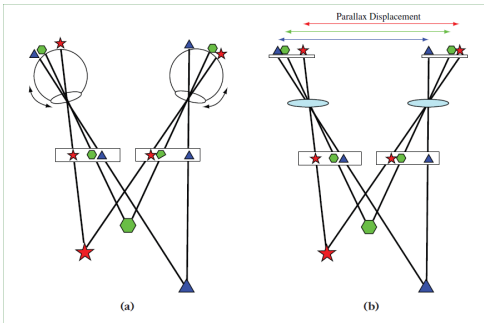




Frequência

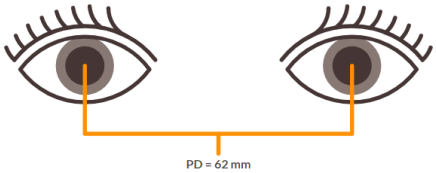
https://www.youtube.com/watch?v=0NPH_udOOek	Ilusões I
http://www.yoism.org/?q=node/71	Ilusões II
https://www.youtube.com/watch?v=VNftf5qLpiA	Explicação (Pneu de lego a rodar)
https://www.youtube.com/watch?v=9MN5MF72PHs	Explicação (gráficos dos pneus a rodar)
https://www.testufo.com/	Testes com UFO (vários parâmetros explicados)
https://www.youtube.com/watch?v=FhSHeYT2U70	Explicação (quantos *frames* os nossos olhos veem)

Visão Estereoscópica



Pupillary Distance

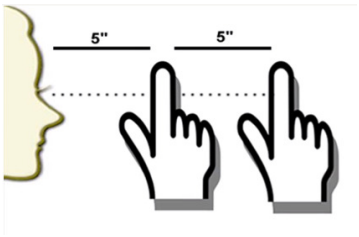
Pupillary Distance (PD) is the measurement from the center of one pupil to the center of the other. If you are getting single vision reading glasses, use your near-vision PD, which is your distance vision PD lowered by 3 mm.



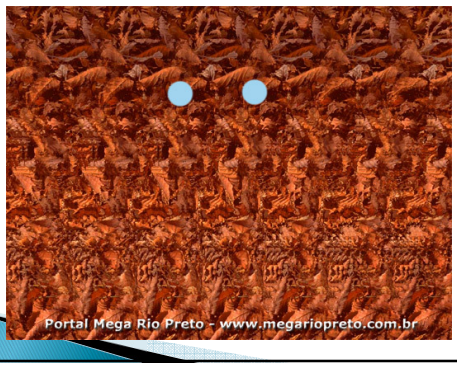
Visão Estereoscópica – câmaras

Estereogramas

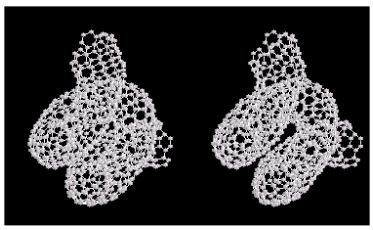
Estereogramas



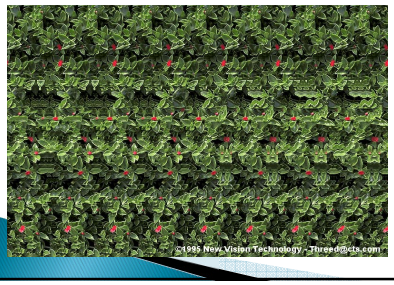
Estereogramas



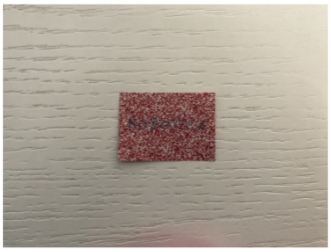
Estereograma



Estereograma com solução

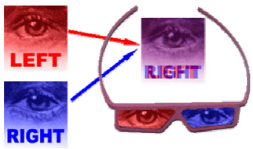


?? Colour ??

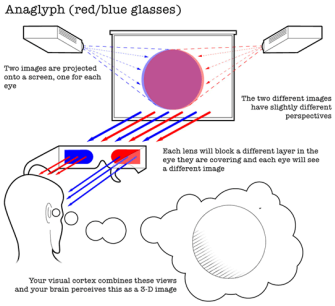


?? Colour ??

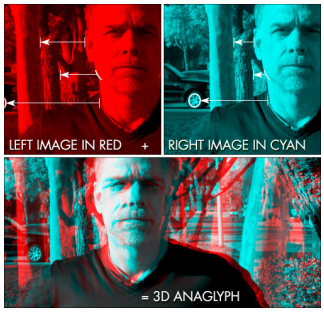
Óculos Passivos



Óculos Passivos



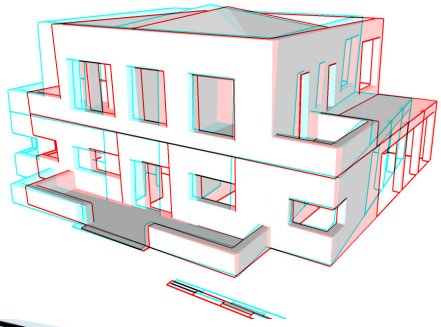
Óculos Passivos



Óculos Passivos

Óculos Passivos

Óculos Passivos

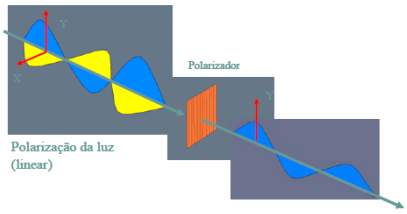


Óculos Activos

Óculos Activos

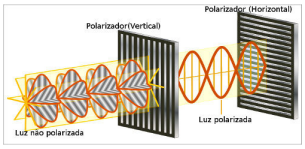
Óculos Activos

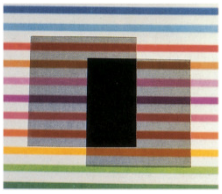
Óculos 3D polarizados



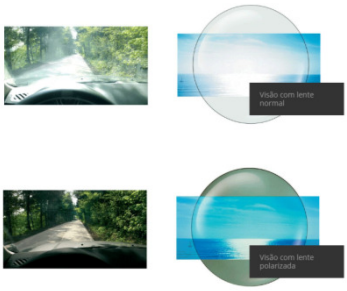
Luz polarizada

Polarização vertical + horizontal



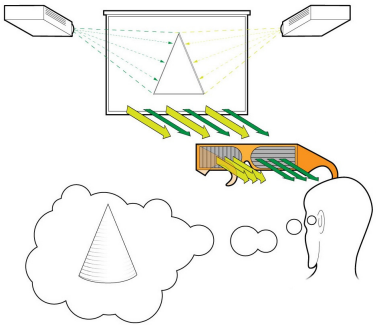


Óculos Polarizados



Óculos 3D polarizados

Óculos 3D polarizados



Para experiências…

http://www.youtube.com/watch?v=qwhGwqOX3E4#t=85

Luz

Importância da Luz:

- As imagens são um registo da quantidade de luz reflectida de uma cena.
- Em Teatro, a luz é usada não só para iluminar uma cena mas também para produzir um efeito dramático.
- Quanto melhor for a iluminação, menos complexos serão os algoritmos de processamento dessa imagem.

Em visão computorizada, o tipo de luz e a forma como esta é usada, depende do tipo de imagem que queremos obter, e da informação que desejamos extrair.

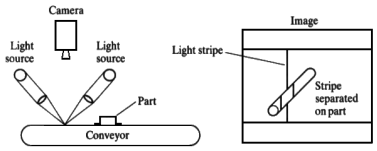
Luz

Imagem Binária:
Todos os *pixels* são ou brancos ou pretos. Alto contraste do objecto com o seu fundo é essencial. O objecto deve estar no centro do campo de visão da câmera. *Backlighting* é usado para obter silhuetas do objecto.

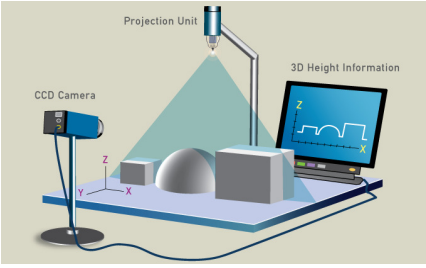
Luz ambiente:
Para objectos com superfícies lisas que não podem usar o *Backlight*, esta técnica reduz bastante as sombras. Numa imagem com graus de cinzento, esta técnica dá-nos cinzentos diferentes para cada superfície proporcionais à intensidade de luz reflectida.

Luz

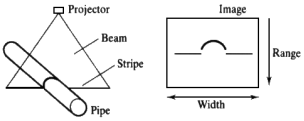
Luz estruturada (*structured light*):
Usado quando visão a 3 Dimensões é necessária. Usado para estimar distâncias entre objectos assim como as suas dimensões, e forma. Pode ser usado para produzir sombras e para realçar arestas de objectos.

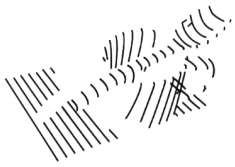


Structured light

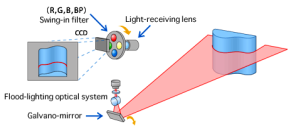


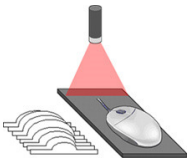
Luz estruturada





Luz estruturada





D. I. P. System Elements

Digital Computers

Although the image processors come with internal processing capacities, this level of processing is low in sophistication. Though, the image processors come connected to a common computer, which offers versatility as well as programming facilities.

D. I. P. System Elements
Storage

Images use lots of space to be stored. A 512 x 512 pixels image with 256 gray levels uses 256 Kbytes. Therefore, it is necessary to take into account the storage type to use. The main storage types are:

- Hard disk
- CD/DVD

D. I. P. System Elements
Output devices

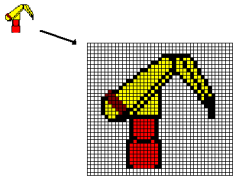
- Black/White TV
- Colour TV
- Slides
- Photographs
- Transparencies

Fundamentals
Gray Levels

Gray level is the value of intensity that one pixel represents. The Grayscale is the list of all possible grays or gray levels, which a monochromatic image can have.

Fundamentals
Pixel

Pixel, or Picture Element, is the minimum unit of an image. The pixel physical size depends on the output where the image is shown.



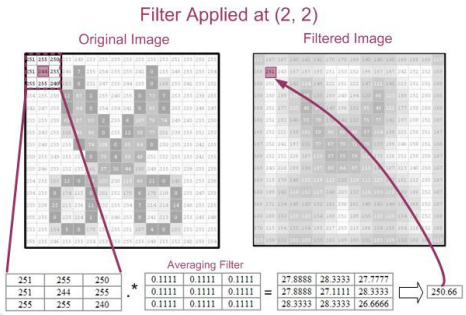
Fundamentals
Mask

Mask defines one portion of the image used to generate parameters and to define the operation area to work on a bigger image.

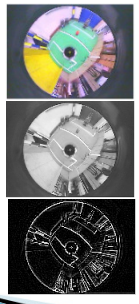
$$\begin{bmatrix} -2 & 0 & +2 \\ -1 & 0 & +1 \\ -2 & 0 & +2 \end{bmatrix} \qquad \begin{bmatrix} -2 & -1 & -2 \\ 0 & 0 & 0 \\ +2 & +1 & +2 \end{bmatrix}$$

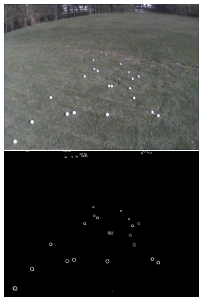
Mask Examples to detect Vertical and Horizontal profiles.

Fundamentals
Mask



Fundamentals
Mask



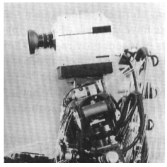


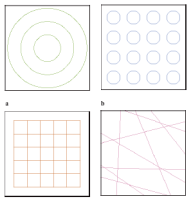
Fundamentals
Quantization

Describes all possible values a pixel can have. Is determined by the digitization of the image and the amount of different bright intensity values that the hardware supports and the software represents.

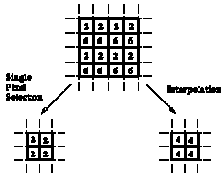
Fundamentals
Sampling

Sampling describes the spatial distribution of the image acquisition. The relationship between the number of pixels in one line or column and the width or height of the image defines the spatial frequency or *sample rate*.





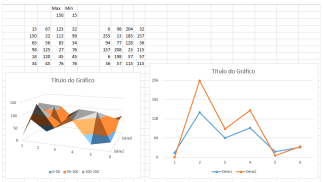
Fundamentals
Sampling



Fundamentals
Scaling (light intensity)

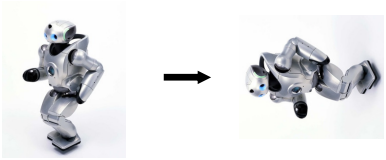
Scaling refers to the attribution of a new value min-max of the image processing result in a quantization scale. For example, an algorithm result could be non integer. To show the pixel in gray level the value needs some adjustment. A formulae to scale a 256 grays image is:

$$OutPixel = \frac{255}{\max - \min}(InPixel - \min)$$



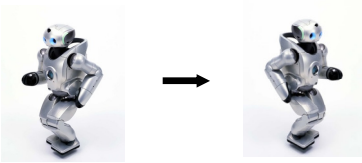
Graphical Algorithms
Rotation

Rotation is used to rotate an image about its centre.
Sometimes used to compare objects or to generate images.



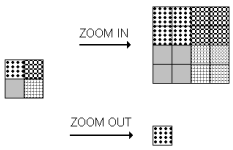
Graphical Algorithms
Flip

Flipping is different from rotation, although a similar operation is carried out. Consists on mirror (vertical or horizontal) the image.



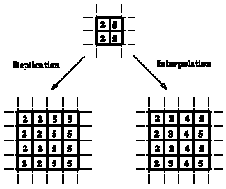
Graphical Algorithms
Zoom

These algorithms amplify or reduce images. Since the image size and pixels number determine the image size, *zoom in* creates pixels and *zoom out* deletes pixels. This technique is simple but requires that the zooming has a factor of (at least) 2.



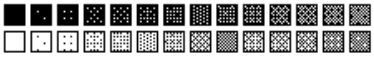
Optical Zoom ≠ Digital Zoom

Graphical Algorithms
Zoom

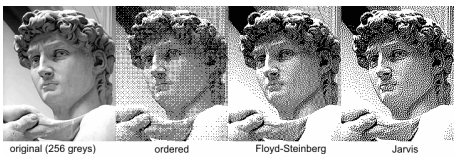


Graphical Algorithms
Dithering

Dithering is the process of producing images in gray levels in printers which are able to print only black dots. It uses patterns of dots to represent a determined gray level. The more concentrated they are in a certain area the darker that area becomes.

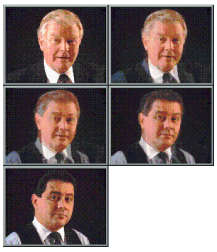


Graphical Algorithms
Dithering

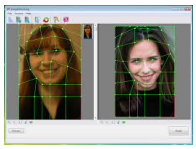


Graphical Algorithms
Morphing

Morphing is used to transform an image into another one. With smooth transitions between images, a visual illusion is created as if a real transformation was occurring. The word was first used by the special effects movie industry and implies the use of an I.P. algorithm to produce a dynamic shape change of one object into another one.



Graphical Algorithms
Morphing

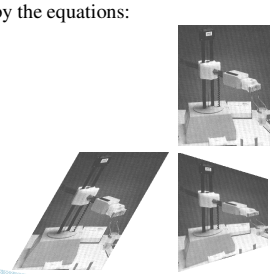


Graphical Algorithms
Warping

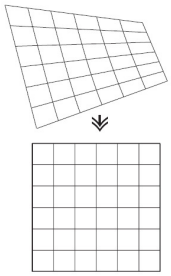
Warping uses image mapping functions to produce geometrical distortions on images. The most common transformations are based on projections, given by the equations:

$$X' = \frac{aX + bY + c}{iX + jY + 1}$$

$$Y' = \frac{dX + eY + f}{iX + jY + 1}$$



Graphical Algorithms
Warping



Geometric distortion occurs when a surface is viewed from a position away from the surface normal. Correcting this distortion to obtain a rectilinear image which can be properly processed and measured, or fitted together with adjoining images, requires knowing the viewing geometry and/or including known fiducial marks in the scene that can be used to determine it.

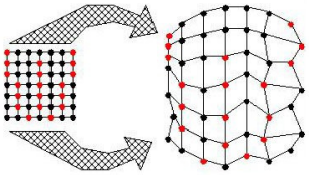
Graphical Algorithms
Warping

Outlining a region parallel to the edges of the tire track defines the image distortion, from which the necessary rectification can be calculated.

(a)

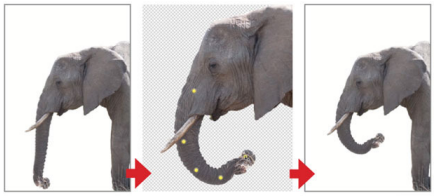
(b)

Graphical Algorithms
Warping



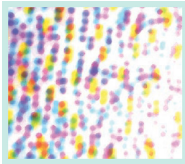
Graphical Algorithms
Warping

Graphical Algorithms
Warping



Resolution

Image Resolution – is the area of the pixel, but normally specified as the number of pixels per line.



DPI (dots per inch) is a measure of how many droplets (dots) of ink an inkjet system can fit into a linear inch. This is not a measure of image detail or resolution as the dots have to be laid down in a matrix pattern to create the appearance of a colour.

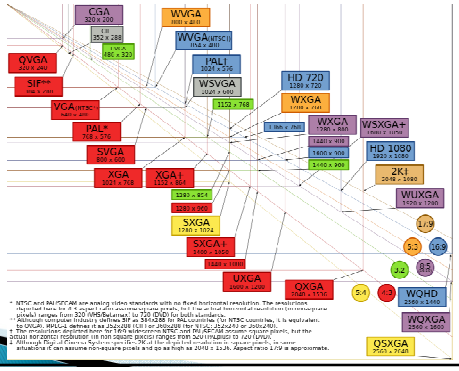
PPI (pixels per inch) is a measure of the resolution of digital images, cameras and scanners and can be thought of as the degree of refinement of the digital sampling process. DPI is commonly and mistakenly used for PPI.

Resolution

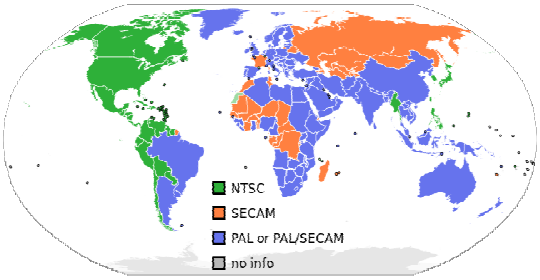
Depends on:
- Camera Scene distance
- Lens size
- Number of *pixels* per line on the image

Resolution:
- Colour TV - 768 x 576 *pixels*
- Film (35mm) - 2000 x 2000 *pixels*
- Human eye - 4000 x 4000 *pixels*

Resolution



Sistemas TV



Normas dos sistemas de TV

- **PAL** (Phase Alternating Line)
  - Developed in 1967 (United Kingdom & Germany)
  - Uses a wider channel bandwidth than NTSC (better picture quality)
  - Lines: 525
  - Frame rate: 25 Hz
  - Picture resolution: 720 x 576
- **NTSC** (National Television Standards Committee)
  - First used in 1954
  - Lines: 625
  - Frame rate: 29.97 Hz
  - Picture resolutions: 720 x 480
- **SECAM** (Squentiel couleurmmoire)
  - Developed in 1967 (in France)
  - Uses the same bandwidth and resolution (720x576) as PAL
  - Transmits colour information sequentially
  - Note: Not many products support SECAM

Digital Image Properties

- Metric
- Topological
- Histograms
- Visual Perception
- Image Quality

Metrics

Pixel Adjacency:

- Any two pixels are called 4-neighbours if they have a distance D_4=1 between them.

- Any two pixels are called 8-neighbours if they have a distance D8=1 between them.

Pixel Distance:

- *Euclidean distance*: $D_E((i,j),(h,k)) = \sqrt{(i-h)^2 + (j-k)^2}$
- Distance in steps (using horizontal/vertical directions)

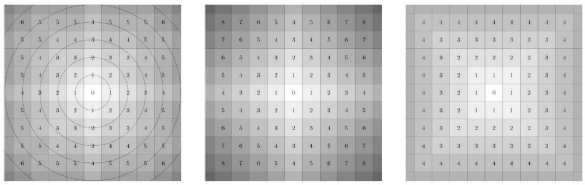
$$D_4((i,j),(h,k)) = |i-h| + |j-k|$$

- Distance in steps (using horizontal/vertical/diagonal directions)

$$D_8((i,j),(h,k)) = \max\{|i-h|, |j-k|\}$$

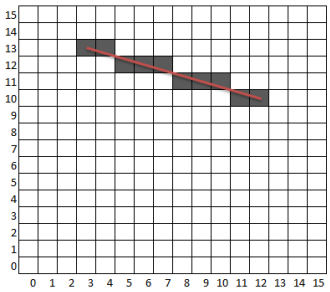
Metrics

Pixel Distance:



D_e D_4 D_8

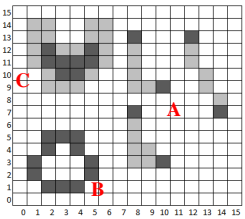
Metrics



Metrics

Path between *pixel* P and *pixel* Q is a sequence of pixels $A_1, A_2, \ldots, A_n$ where A_1=P, A_n=Q e A_{i+1} is neighbour of A_i.

- ***Simple Path*** is a ***Path*** where there are no repeated *pixels* and no *pixel* has more than one neighbour.
- ***Closed Path*** is a ***Path*** where the first *pixel* is neighbour of the last.
- **Region** consists of a group of *pixels* where there is always one ***path*** between any pair of *pixels*. Sometimes it is also called **object**.

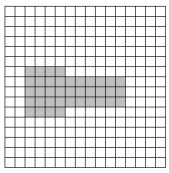


If there is a ***path*** between two *pixels*, then those *pixels* are called contiguous.

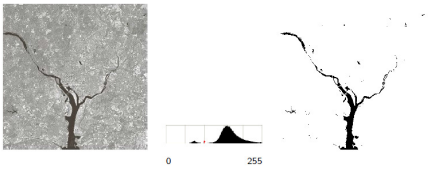
Metrics
Area

Algorithms are used to measure area. These count the number of pixels of the desired object on the image. The object border must be well defined. It is then necessary to know the pixel size, usually through a calibration.

$$Area = Number_pixels * Area\ per\ pixel$$



Metrics
Example of Area Measurement

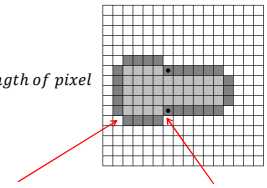


- Identify pixels with the river colour
- Select the connected pixels
- Consider regions aligned

Metrics
Perimeter

The perimeter is given by the number of pixels of the object border. The border can be calculated by subtracting the **erosion** of the object by its original, or by the use of a **border detector**. The function to calculate the area can be the count of pixels of the border (perimeter).

$$Perimeter = Number_pixels_border * Length\ of\ pixel$$

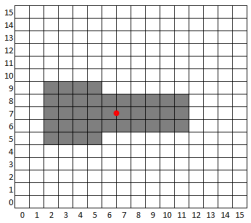


Metrics
Centroid

The centroid refers to the object centre and consists on the geographical centre. It is given in coordinates and it is calculated by using the formulas:

$$X_c = \left(\frac{X_{\max} + X_{\min}}{2}\right)$$

$$Y_c = \left(\frac{Y_{\max} + Y_{\min}}{2}\right)$$



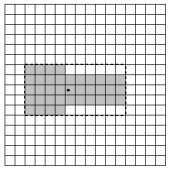
Xmax	11
Xmin	2
Ymax	9
Ymin	5
Xc	6.5
Yc	7

Metrics
Centre of mass

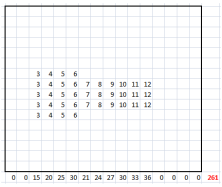
The centre of mass (or barycenter) is the average location of all of its mass. It is given in coordinates and it is calculated by using the following formulas:

$$X_c = \frac{1}{A}\sum_{i=1}^{N} X$$

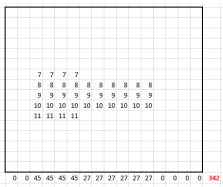
$$Y_c = \frac{1}{A}\sum_{i=1}^{N} Y$$



Metrics
Centre of mass



X



Y

$$X_c = \frac{261}{38} = 6.87$$

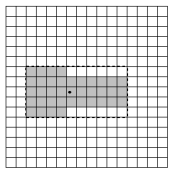
$$Y_c = \frac{342}{38} = 9$$

Metrics
Moments

The moments are the values distributed by the axes. In 2D images, these are used to describe the distribution of the gray values. Moments give measurements of an object for analysis and pattern recognition. The two moments are given by the formulas:

$$m_{pq} = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} x^p y^q f(x,y)$$

$$\bar{x} = \frac{m_{10}}{m_{00}}, \bar{y} = \frac{m_{01}}{m_{00}}$$

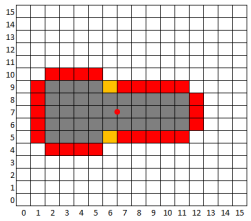


Metrics
Circularity

This measurement gives the proximity of the object to a circular shape. Can be given as the variance of the border pixels distance to the centroid. Should the object be perfectly circular, this variance is zero.

- Border (Object - Erosion)
- Centroid
- Calculate the distances from centroid to each border pixel
- Accumulated values
- Variance calculation

Metrics
Circularity



6.5 7

2	10	5.41
3	10	4.61
4	10	3.91
5	10	3.35
6	9	2.06
6	9	2.06
7	9	2.06
8	9	2.50
9	9	3.20
10	9	4.03
11	9	4.92
1	9	5.85
1	8	5.59
1	7	5.50
1	6	5.59
1	5	5.85

2	4	5.41
3	4	4.61
4	4	3.91
5	4	3.35
6	5	2.06
6	5	2.06
7	5	2.06
8	5	2.50
9	5	3.20
10	5	4.03
11	5	4.92
12	8	5.59
12	7	5.50
12	6	5.59

Soma	121,30
Média	4,04
Variância	1,966

Metrics
Circularity

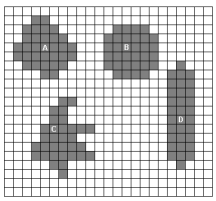
This measurement gives the proximity of the object to a circular shape. Can be given as the variance of the border pixels distance to the centroid. Should the object be perfectly circular, this variance is zero.

$$Circularity = 4 \times \pi \times \left(\frac{Area}{Perimeter^2} \right)$$

Metrics
Circularity

Compactness is a measurement of distribution of an object identical to *circularity*, but defined as the square of the perimeter divided by the area. This value will be minimum for a circular object but is sensitive to noise on the perimeter.

Circularity	0,513
Circularity	0,278

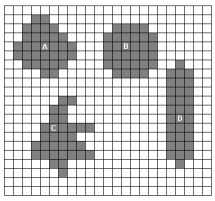


Circularity	0,698
Circularity	0,447

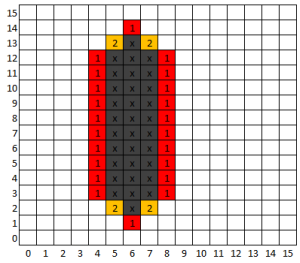
Metrics
Compactness

Compactness is a measurement of distribution of an object identical to *circularity*, but defined as the square of the perimeter divided by the area. This value will be minimum for a circular object but is sensitive to noise on the perimeter.

$$Compactness = \frac{perimeter^2}{area}$$

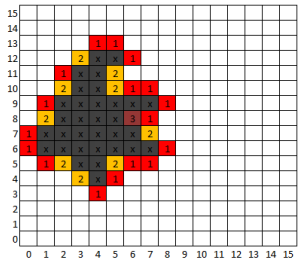


Metrics
Compactness



Área	32
Perimetro	30
Compactness	28.125

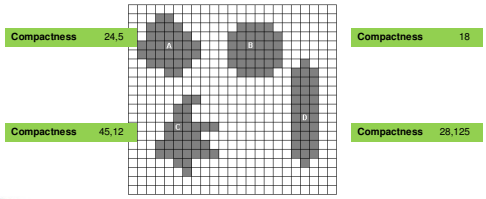
Metrics
Compactness



Área	32
Perimetro	38
Compactness	45.125

Metrics
Compactness

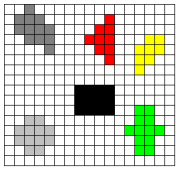
Compactness is a measurement of distribution of an object identical to *circularity*, but defined as the square of the perimeter divided by the area. This value will be minimum for a circular object but is sensitive to noise on the perimeter.



Metrics
Clustering

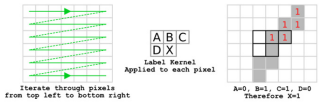
Clustering is the process of counting and separating individual objects in an image. One *cluster* consists of a group of pixels related (or linked) and non distant more than a certain pre-defined value. This measurement can be defined as the distance between *clusters*. The objects can then be shown on a binary image as independent objects.

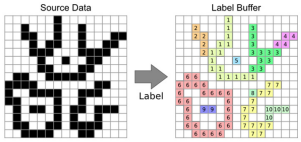
- Define distance between *clusters*
- For all non-zero pixels:
  - For all existing clusters:
    - calculate distance from pixel to cluster
    - If distance < distance_between_clusters
      - then add pixel to current cluster
      - else create new cluster
- Change colour of each *cluster*



Blob Detection

```
l = 1                    // Initial Label number
for each pixel
 if pixel X is foreground
  if neighbours A,B,C & D are unlabelled (equal to zero)
   label pixel X with l
   increment l
  else
   num = neighbour label A,B,C & D with least value (not 0)
   label pixel X and pixels A, B, C & D if foreground with num
  end if
 end if
done
```





http://www.labbookpages.co.uk/software/imgProc/blobDetection.html

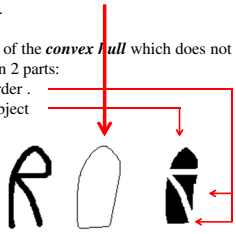
Topological

Euler-Poincaré Characteristic - defined as the number of regions less the number of ***holes***.

Convex Hull – describes topological properties of an object. It consists on the minimum region which contains an object.

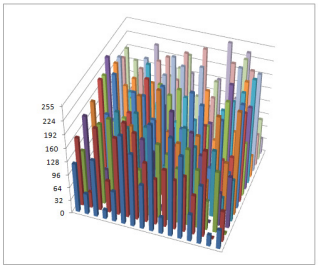
Deficit of convexity - consists of the inner of the ***convex hull*** which does not belong to the object. This can be divided in 2 parts:

- ***Bays*** - contiguous to the ***convex hull*** border .
- ***Lakes*** – completely surrounded by the object



Histogramas
Imagem como matriz

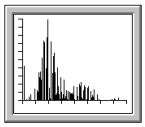
120	51	140	22	75	214	177	109	254	41	236	200	67	143	28	47
167	39	200	79	188	228	210	146	113	141	122	157	97	208	2	76
121	101	195	212	71	107	108	67	180	162	134	185	142	73	149	4
189	147	70	99	11	107	166	141	10	245	47	208	161	2	82	127
50	95	56	200	168	207	180	84	98	7	199	160	110	162	53	103
195	1	64	59	158	233	83	255	165	143	5	171	68	226	123	247
6	156	253	26	222	219	169	250	158	163	250	227	84	203	83	249
219	65	10	156	137	244	149	30	101	136	18	166	3	94	106	216
216	110	124	177	135	128	223	205	221	129	108	151	5	218	32	131
245	22	43	75	84	24	160	72	88	206	154	143	151	243	68	201
80	30	160	38	234	148	87	111	176	126	37	21	165	191	216	251
206	157	179	125	178	143	45	62	11	47	188	72	75	104	228	54
208	130	36	218	100	188	235	218	113	225	76	84	154	242	0	237
194	125	177	140	204	218	168	236	5	254	144	19	235	144	227	78
204	180	118	78	39	160	184	219	125	71	170	98	74	236	101	102
4	129	82	213	26	163	175	163	164	90	162	253	125	239	144	39



Histogram

Consists of a graphic which represents the amount of each gray level. The algorithm to calculate this histogram consists on the following steps:

1. Set zero to all values of h_f array
   (this has as many elements as gray levels).
2. For all *pixels* (x,y) of the image *f*, increment h_f (*f*(x,y)) of one unit.

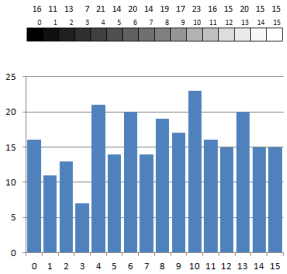


Histogram

7	3	8	1	4	13	11	6	15	2	14	12	4	8	1	2
10	2	12	4	11	14	13	9	7	8	7	8	6	13	0	4
7	6	12	13	4	6	6	4	11	10	8	11	8	4	9	0
11	9	4	6	0	6	10	8	0	15	2	13	10	0	5	7
3	5	3	12	10	12	11	5	6	0	12	10	6	10	3	6
12	0	4	3	9	14	5	15	10	8	0	10	4	14	7	15
0	9	15	1	13	13	10	15	9	10	15	14	5	12	5	15
13	4	0	9	8	15	9	1	6	8	1	10	0	5	6	13
13	6	7	11	8	8	13	12	13	8	6	9	0	13	2	8
15	1	2	4	5	1	10	4	5	12	9	8	9	15	4	12
5	1	10	2	14	9	5	6	11	7	2	1	10	11	13	15
12	9	11	7	11	8	2	3	0	2	11	4	4	6	14	3
13	8	2	13	6	11	14	13	7	14	4	5	9	15	0	14
12	7	11	8	12	13	10	14	0	15	9	1	14	9	14	4
12	11	7	4	2	10	11	13	7	4	10	6	4	14	6	6
0	8	5	13	1	10	10	10	10	5	10	15	7	14	9	2

Image (16x16 pixels)
16 colours

Values sum

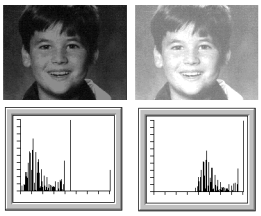


Histogram

Histograms
Brightness

Brightness correction is used to highlight the visual aspect of an image. A change in bright is defined by:

$$s_i = g_i + \text{bright}$$



Histograms
Brightness

Brightness correction is used to highlight the visual aspect of an image. A change in bright is defined by:

$$s_i = g_i + \text{bright}$$

Original

	0	1	2	3	4	5	6	7
0	181	69	225	139	161	60	110	54
1	121	174	73	121	77	21	197	113
2	250	40	251	146	247	230	57	4
3	32	110	141	215	20	159	69	26
4	247	38	65	173	115	12	205	173
5	94	52	92	80	26	36	182	214
6	190	199	80	177	23	153	216	198
7	167	87	118	60	56	58	208	19

40

After Bright increase

	0	1	2	3	4	5	6	7
0	221	109	255	179	201	100	150	94
1	161	214	113	161	117	61	237	153
2	255	80	255	186	255	255	97	44
3	72	150	181	255	60	199	109	66
4	255	78	105	213	155	52	245	213
5	134	92	132	120	66	76	222	254
6	230	239	120	217	63	193	255	238
7	207	127	158	100	96	98	248	59

Histograms
Brightness

Brightness correction is used to highlight the visual aspect of an image. A change in bright is defined by:

$$s_i = g_i + \text{bright}$$

Original

	0	1	2	3	4	5	6	7
0	181	69	225	139	161	60	110	54
1	121	174	73	121	77	21	197	113
2	250	40	251	146	247	230	57	4
3	32	110	141	215	20	159	69	26
4	247	38	65	173	115	12	205	173
5	94	52	92	80	26	36	182	214
6	190	199	80	177	23	153	216	198
7	167	87	118	60	56	58	208	19

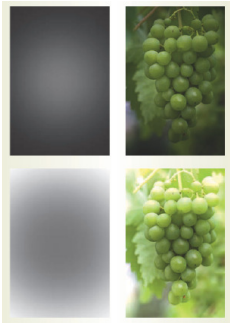
40

After Bright increase

	0	1	2	3	4	5	6	7
0	221	109	255	179	201	100	150	94
1	161	214	113	161	117	61	237	153
2	255	80	255	186	255	255	97	44
3	72	150	181	255	60	199	109	66
4	255	78	105	213	155	52	245	213
5	134	92	132	120	66	76	222	254
6	230	239	120	217	63	193	255	238
7	207	127	158	100	96	98	248	59

Values saturated

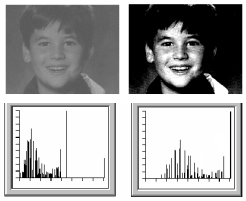
Histograms
Brightness



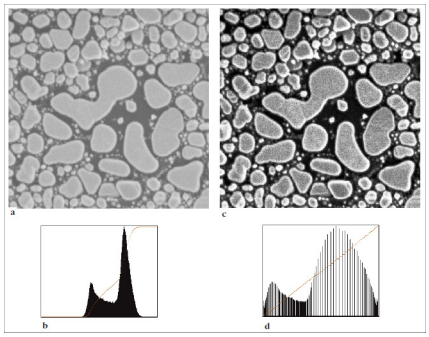
Histogram
Contrast

A contrast correction is used to highlight the visual aspect of an image. A contrast change of an image is defined by:

s_i = contrast * (g_i - average) + average $\quad average = \frac{1}{N_x N_y}\sum_{X=0}^{N_X}\sum_{Y=0}^{N_Y} P(X,Y)$



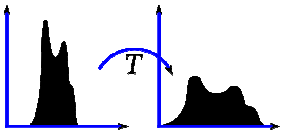
Histogram
Contrast



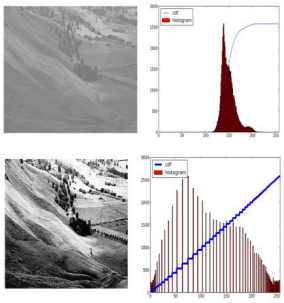
Histograms
Equalization

A histogram equalization distributes evenly the pixel gray values in such way that the number of pixels on each gray level is approximately the same.

$$g_i = \frac{M-1}{n_t}\sum_{j=0}^{i} n_i$$



Histograms
Equalization



Histograms
Histogram Specification

The histogram specification can be used to darken, lighten or improve contrast of an image. If we consider h_{di} the desired histogram, then the first step is equalize it:

$$s_i = (m-1)\sum_{j=0}^{i} h_{di}$$

Then, the original image histogram is also equalized:

$$g_i = P(d_i) = (m-1)\sum_{j=0}^{i} h_{fi}$$

The last step is to find the inverse transform of the desired histogram,

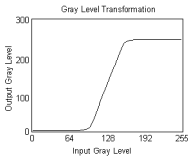
$$d_i = P^{-1}(s_i)$$

And apply this inverse transformation to the pixel gray values on the equalized image.

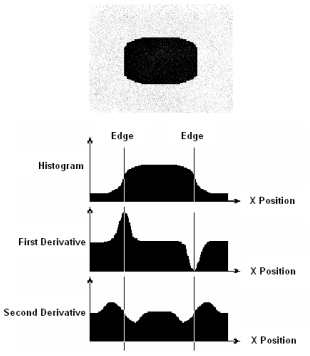
Histograms
Non linear transformations

Non linear transformation of an image in gray level is used to increase the contrast for certain values of gray, while it decreases other values of gray. The transformation is defined by:

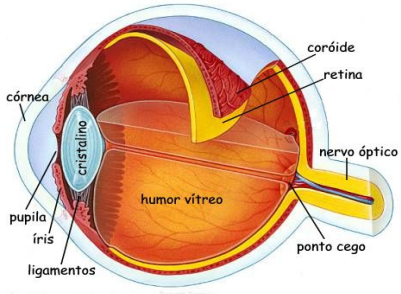
$$s_i = T(g_i)$$



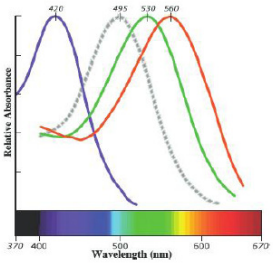
Edge searching



Características da visão



Características da visão



Sensitivity of the rods (shown in gray) and three kinds of cones (shown in red, green, and blue) as a function of wavelength (Dartnall et al., 1983). Human vision detects roughly the range from about 400 nm (blue) to 700 nm (red).

Image Visual Perception

No matter who creates algorithms or devices for digital image processing, basic principles of human image perception should be taken into account. If an image is to be analyzed by a human, it should be expressed using easy to understand variables. Examples are:

- Contrast
- Border
- Shape
- Texture
- Colour

Image Visual Perception

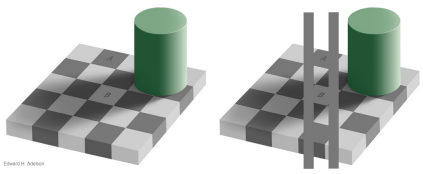
Contrast is a local change in brightness and is defined as the ratio between the average brightness of one object and the background brightness. The apparent brightness depends very much on the background brightness.

The **Border** of an object carries much information and consists on the profile shape of the object.

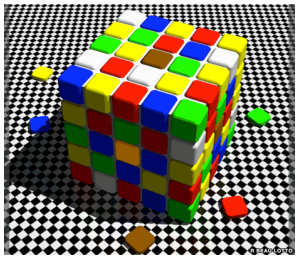
Cuity is the ability to detect detail on an image. This decreases with distance.

Colour is other important property for human perception since the human eye is more sensitive to colour than to contrast.

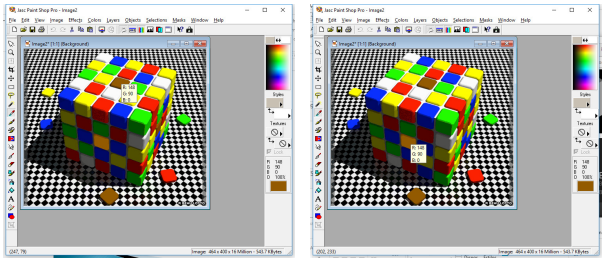
?? Contrast ??



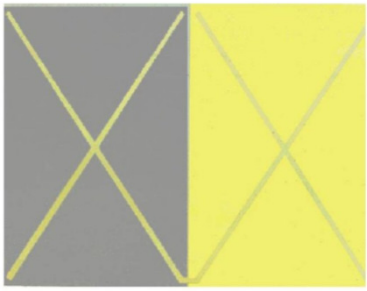
?? Contrast ??



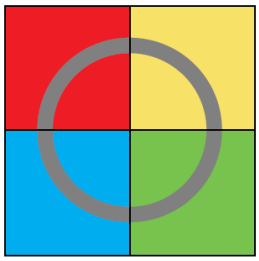
?? Contrast ??



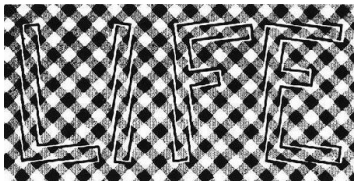
?? Contrast ??



?? Contrast ??

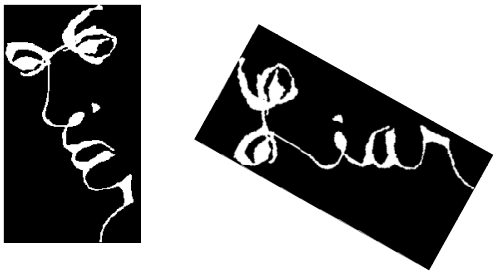


?? Border ??

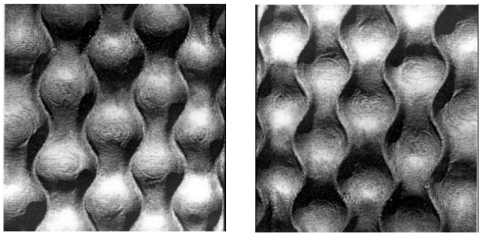


?? Shape ??

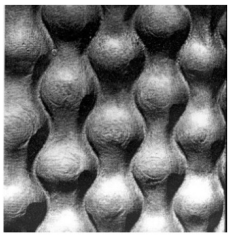
?? Shape ??



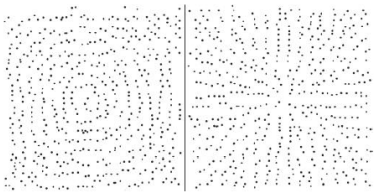
?? Shape ??



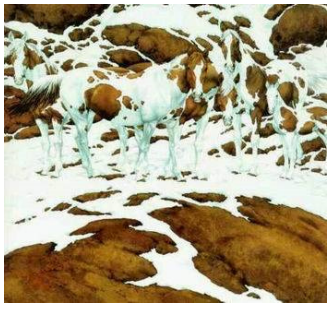
?? Shape ??



?? Shape ??

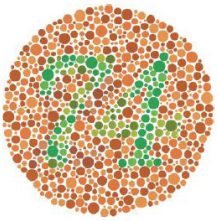


?? Acuity ??



?? Acuity ??

?? Acuity ??

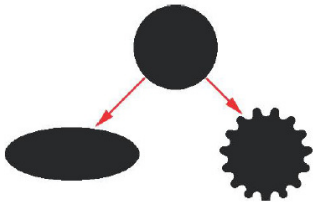


?? Acuity ??

?? time ??

?? space ??

?? space ??



?? space ??

?? meaning ??

5 FIVE 5

five 5 0101

V cinco

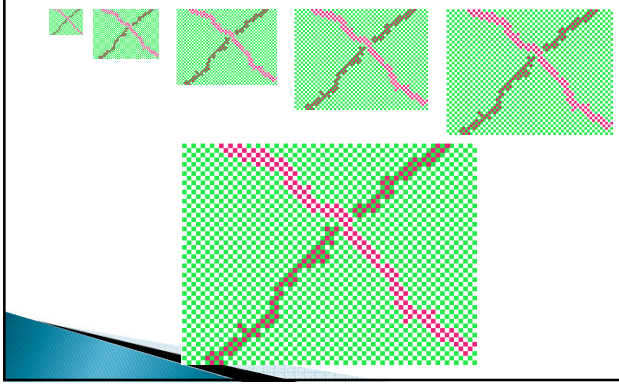
Fünf Πεντε

5 QUINTUS 5

?? Colour ??

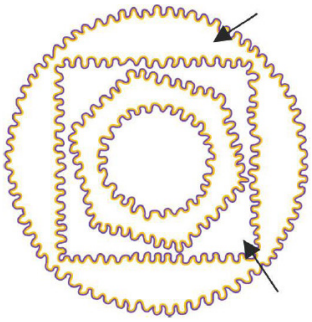
AMARELO AZUL LARANJA
PRETO VERMELHO VERDE
ROXO AMARELO VERMELHO
LARANJA VERDE PRETO
AZUL VERMELHO ROXO
VERDE AZUL LARANJA

?? Colour ??



?? Colour ??

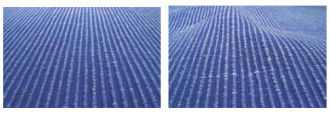
?? Colour ??



?? Colour ??

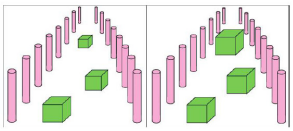
?? Colour ??

?? Perspective ??



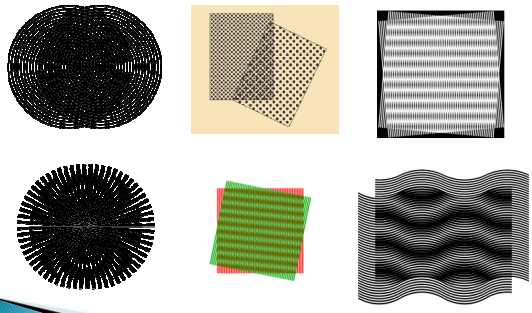
(a) (b)

*Converging lines are interpreted as parallel lines that converge according to the rules of perspective and so represent the surface perceived as receding from the viewer. Straight lines imply a flat surface **(a)**, while irregularities are interpreted as bumps or dips in the perceived surface **(b)**.*



"Ver" infra-vermelhos

Moiré Pattern



Moiré Pattern

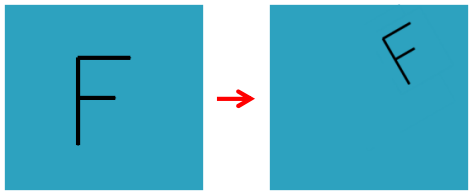
Image Quality

One image can become downgraded during its capture, transmission or processing. Image quality measurements can be used to assess the degradation degree. The necessary quality depends on the practical application. There are two types of image assessment:

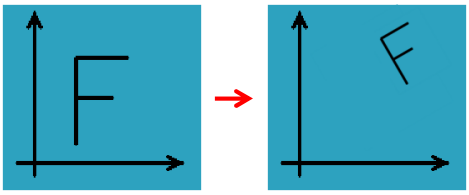
- Objective
- Subjective

Subjective Assessment

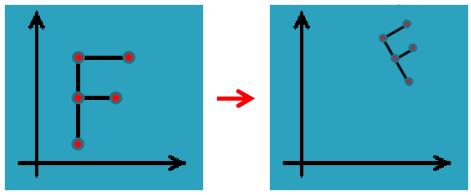
Transformações Geométricas



Transformações Geométricas



Transformações Geométricas

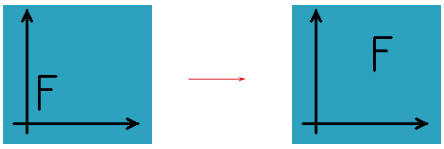


Tipos de Transformações

- Translação
- Escalamento
- Rotação
- Reflexão
- Distorção

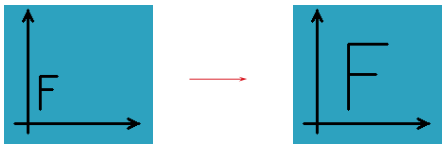
Transformações Geométricas

Transformação 2D
Translação



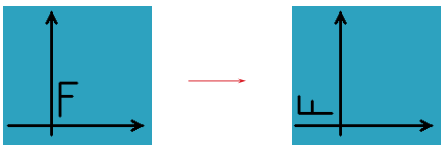
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} dx \\ dy \end{bmatrix} + \begin{bmatrix} x \\ y \end{bmatrix}$$

Transformação 2D
Escalamento



$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} ax + 0y \\ 0x + by \end{bmatrix}$$

Transformação 2D
Rotação

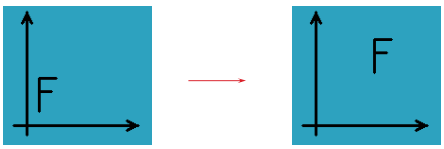


$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos(\emptyset) & -\sin(\emptyset) \\ \sin(\emptyset) & \cos(\emptyset) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Transformação 2D

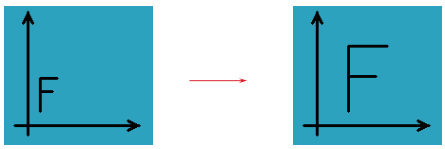
- Translation cannot be expressed using matrix multiplication
- move from Cartesian to homogeneous coordinate space
- Homogeneous Matrix

Transformação 2D
Translação



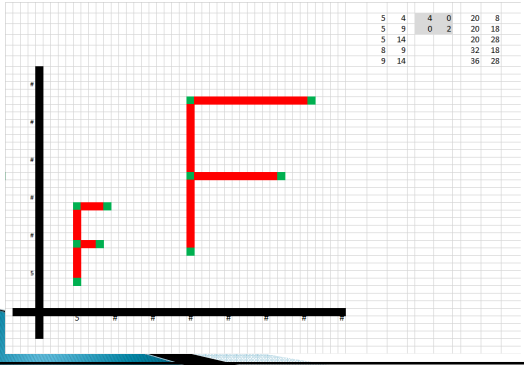
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & d_x \\ 0 & 1 & d_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Transformação 2D
Escalamento



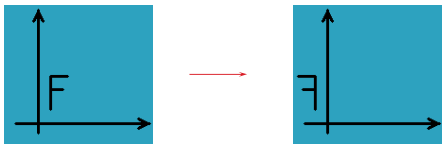
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Exemplo de Escalamento



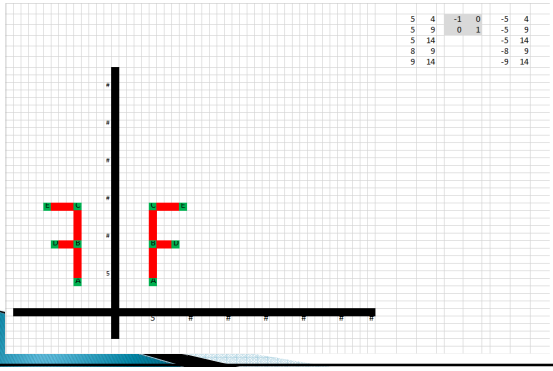
5	4	4	0	20	8
5	9	0	2	20	18
5	14			20	28
8	9			32	18
9	14			36	28

Transformação 2D
Reflexão



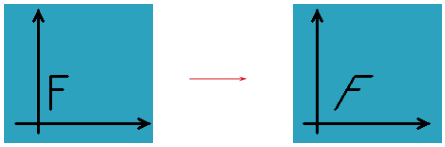
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Exemplo de Reflexão



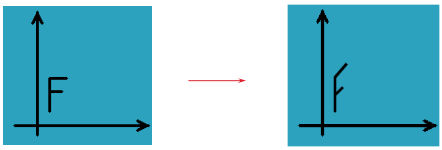
5	4	-1	0	-5	4
5	9	0	1	-5	9
5	14			-5	14
8	9			-8	9
9	14			-9	14

Transformação 2D
Distorção em X



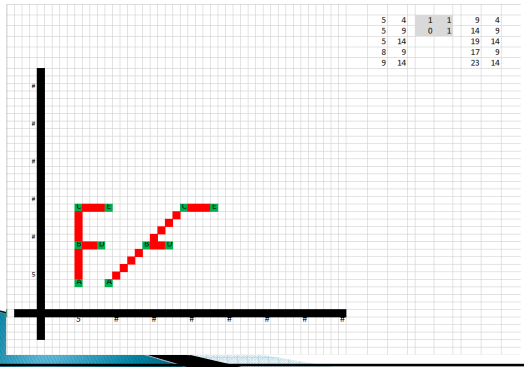
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & a & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Transformação 2D
Distorção em Y



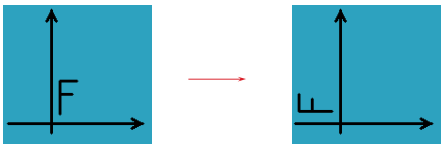
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ b & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Exemplo de Esticar



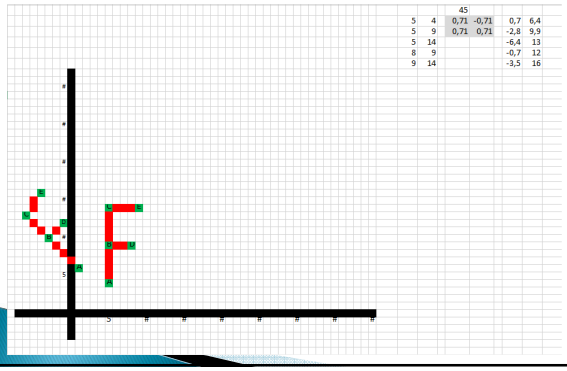
5	4	1	1	9	4
5	9	0	1	14	9
5	14			19	14
8	9			17	9
9	14			23	14

Transformação 2D
Rotação



$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\emptyset) & -\sin(\emptyset) & 0 \\ \sin(\emptyset) & \cos(\emptyset) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Exemplo de Rotação



		45			
5	4	0,71	-0,71	0,7	6,4
5	9	0,71	0,71	-2,8	9,9
5	14			-6,4	13
8	9			-0,7	12
9	14			-3,5	16

Transformações Matriciais a 3D

$$T = \left[\begin{array}{ccc|c} 3 & & & 3 \\ & \times & & \times \\ & & 3 & 3 \\ \hline 1 & \times & 3 & 1\times 1 \end{array}\right] = \left[\begin{array}{c|c} rotação & trans \\ \hline 0 & 1 \end{array}\right]$$

Translação a 3D

$$Trans(d_x, d_y, d_z) = \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotações a 3D

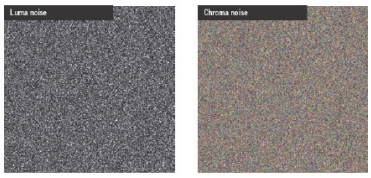
$$Rot(x,\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) & 0 \\ 0 & \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Rot(y,\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Rot(z,\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 & 0 \\ \sin(\theta) & \cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Noise

- *Noise* is to digital photography as grain to film
- Pixels out of place, in regular pattern of pixel mosaic



Colour Image
RGB model

Gray levels:

uses 1 byte <=> 8 bits

$2^8 = 256$ combinations

Colours:

uses 1 byte for each primary colour

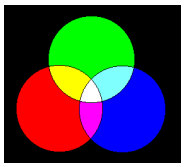
3 bytes <=> 24 bits

$2^{24} = 16{,}777{,}216$ combinations

$$r = \frac{R}{R+G+B}$$

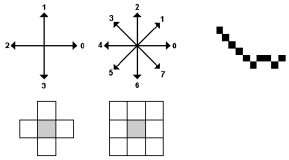
$$g = \frac{G}{R+G+B}$$

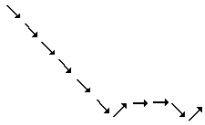
$$b = \frac{B}{R+G+B}$$



Coding and compression
Chain Codes

Chain Coding is used to define the border of an object in order to measure and identify that object. The *chain code* follows the object border and indicates changes in direction according to the pattern type desired of 4 or 8 directions.





7 7 7 7 7 7 1 0 0 7 1

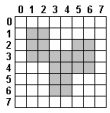
Coding and compression
Run Length Encoding

Run Length Encoding (RLE) is a very simple and powerful scheme used to compress images. In its simplest form, it consists on the repeated characters counting (called index) followed by the number of occurrences. RLE takes advantages on homogeneous areas of the image with long chains of identical pixels .

0	0	0	0	0	0
0	0	0	0	0	0
0	0	255	255	0	0
0	0	255	255	0	0
0	0	0	0	0	0
0	0	0	0	0	0

(14,0)
(2,255)
(4,0)
(2,255)
(14,0)

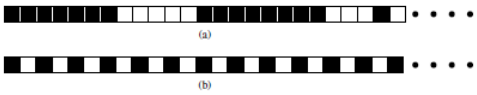
Coding and compression
Run Length Encoding



	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	1	0	0	0	0	0
2	0	1	1	0	0	1	1	0
3	0	1	1	1	1	1	1	0
4	0	0	0	1	1	1	1	0
5	0	0	0	1	1	0	0	0
6	0	0	0	1	1	0	0	0
7	0	0	0	0	0	0	0	0

0	**0**	-> [0,8,0]
1	**1**	-> [0,1,0] [1,2,1] [3,5,0]
2	**4**	-> [0,1,0] [1,2,1] [3,2,0] [5,2,1] [7,1,0]
3	**9**	-> [0,1,0] [1,6,1] [7,1,0]
4	**12**	-> [0,3,0] [3,4,1] [7,1,0]
5	**15**	-> [0,3,0] [3,2,1] [5,3,0]
6	**18**	-> [0,3,0] [3,2,1] [5,3,0]
7	**21**	-> [0,8,0]

Coding and compression
Run Length Encoding



Coding and compression
Huffman Coding

Huffman Coding is an image compression technique that uses statistical data from the image to determine a variable length code. The occurrences probabilities are calculated in all possible forms and are then sorted. The shortest word is given to the most used value and so on.

The probabilities are accumulated according to the tree side. The **bold** numbers are the *Huffman codes*. The tree is generated from left to right but the *Huffman* codes are generated from right to left. the codes are unique.

Coding and compression
Huffman Coding

Example:
Assuming that one image has 6 gray levels and the probabilities are distributed as:
0.4, 0.3, 0.1, 0.1, 0.06, 0.04

Input Level	Input Prob.	Step 1	Step 2	Step 3	Step 4
W1	0.4 **1**	0.4 **1**	0.4 **1**	0.4 **1**	0.6 **0**
W2	0.3 **00**	0.3 **00**	0.3 **00**	0.3 **00**	0.4 **1**
W3	0.1 **011**	0.1 **011**	0.2 **010**	0.3 **01**	
W4	0.1 **0100**	0.1 **0100**	0.1 **011**		
W5	0.06 **01010**	0.1 **0101**			
W6	0.04 **01011**				

The average word length for this coding is:
R=1(0.4)+2(0.3)+3(0.1)+4(0.1)+5(0.06)+5(0.04)=2.2 bits

Coding and compression
Huffman Coding

Example:
Assuming that one image has 4 gray levels and the probabilities are distributed as:
0.4, 0.3, 0.2, 0.1

Probability p_k	Assigned codeword
0.1	111
0.3	10
0.4	0
0.2	110

The average word length for this coding is:
R=1(0.4)+2(0.3)+3(0.2)+3(0.1) = 1.9 bits (por pixel)

Graphical Formats

Images contain a great amount of information which derive from huge sources in several different formats. Naturally and necessarily, specific image file formats came up to interpretate, work, store and transmit that information. The amount of file formats is almost as much as the number of acquisition and output devices. The top five most used file formats (and more efficient) are:

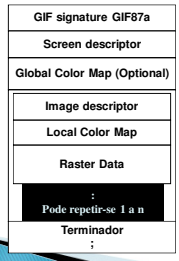
GIF, JPEG, MAC, PCX, TIFF

But there are others:

BMP	TGA	DIB	DCX	WMF	PICT
WPG	EPS	KFX	RLE	LV	CALS
G3	G4	ATT	CLP	XWD	IMG
IFF	SUN	XBM	ICO	ICA	GX2
XPM	ASCII	CUT	BRK	PSD	MSP
PCD	PCC	HGL	CDR	AI	CGM
DXF	DWG	DRW	GEM	PPM	PGM
LBM	PBM	RAS	RAW	FLC	FLI

Graphical Formats
GIF

The *Graphics Interchange File* (GIF) file format is a registered mark from Compuserve who developed it for an efficient storage and file transfer. Its vast use is due to a well defined format and due to being compressed. O seu formato é como se segue:




```
GIF_evaluation(FILE *fp)
{
unsigned char buf[6], flag, bg;
int wd, ht;
        fread(buf, 6, 1, fp);
        if (buf[0]!='G' && buf[1]!='I' && buf[2]!='F' )
                printf("Not a GIF File\n"),
                return (-1);
        printf("Evaluating a GIF File\n");

        fread(&wd, 2, 1, fp);   wd>>=8;  printf("Screen Width:  %d\n", wd);
        fread(&ht, 2, 1, fp);   ht>>=8;  printf("Screen Height: %d\n", ht);
        fread(&flag, 1, 1, fp);  printf("Global Flag:   %x\n", flag);
        fread(&bg, 1, 1, fp);  printf("Background:    %d\n", bg);

        fread(&buf, 1, 1, fp);
        if (buf[0])
                printf("Problem in GIF Header\n"),
                return (-1);
}
```

Graphical Formats
JPEG

JPEG images (*Joint Photographic Experts Group*) are compressed image files which use a loss algorithm and are identified as JPEG *File Interchange Format* (JFIF). Loss Algorithm means that there is a compromise between the image quality and the compression ratio. This type of images are for human perception, and therefore the humans weak spots were taken into consideration so that the information lost is not critical to the human perception. The compression amount is variable and depends on a user defined value. Since a compression loss is used, many compressions result in many degradations and this technique is not reversible.

Graphical Formats
MAC

This file format was first used in beginning 80's with the introduction of Macintosh and it is still very used. They are images with 576 x 720 binary pixels (8 x 10 inches at 72 dpi) and most of them are '*clip art*'. In other computers, the extension used is '.MAC'.

Block of 128 bytes with the name (optional).
Block of 512 bytes with information about filling in patterns.
720 lines in RLE with the information about the image itself

Graphical Formats
PCX

PC Paintbrush, or PCX file format, are images created to or read by the Paintbrush software from Zsoft. This program is classified as a drawing program and allows the user to draw and fill in objects, add texture to figures and manipulate colours or patterns in grays levels.

```
char  PCX_id;          Identifies a PCX (always 0x0A)
char  version;         Version of PB where PCX was created (def. 5)
char  encoding;        Type of compression. 0 none, 1 RLE
char  Bits_per_pixel;  Number of bits per pixel in each plane
short Left;            Left column   where the image is shown
short Top;             Top most line where the image is shown
short Right;           right column  where the image is shown
short Bottom;          Bottom line   where the image is shown
short Hor_Resol;       Horizontal screen resolution
short Ver_Resol;       Vertical screen resolution
char  colour_map[48];  Colour Palette used for 16 colours
char  Reserved;        Reserved
char  num_of_planes;   No. of colour planes used by the screen
short Bytes_per_line;  No. of bytes (PAR)/'scan line'/colour planes
short palette_type;    0 no palette, 1 palette included, 2 grey-levels
char  filler[58];      Left empty to make up the 128 bytes
```

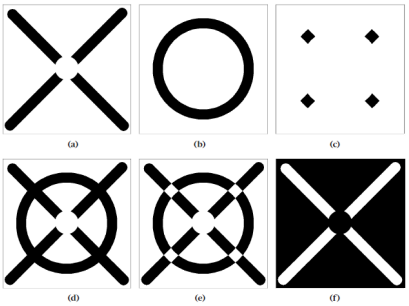
The rest of the information is in RLE.

Graphical Formats
TIFF

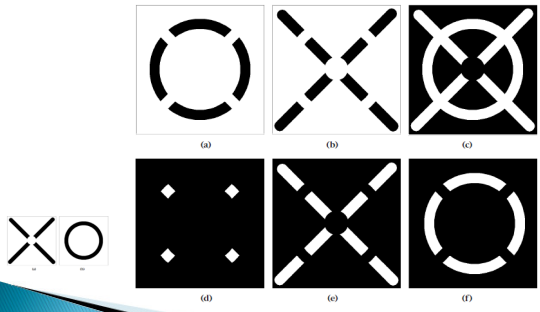
TIFF (*Tagged Interchange File Format*) file format is a file format used to store an image based on a sequence of individual fields defined by unique '*tags*'. These pointers are descriptions to define fields that contain image data. This structure of pointers allows a description of virtually any type of image and is extensible to future new formats which become available. This file consists of a sequence of up to 2^{32} bytes which a reading program interpretates.

- 8 bytes header
  - 2 bytes define Intel/Motorola bytes order
  - 2 bytes define the version number
  - 4 bytes define the offset to the beginning of the image
- Body (1 or more IFD's)
  - 2 bytes counter
  - Counter x 12 bytes of IFD fields
    - 2 bytes specify a '*Tag*'
    - 2 bytes specify the field type (1=byte,2=ascii,3=short,4=long,5=rational)
    - 2 bytes count the IFD field types
    - 4 bytes contain the IFD data offset
  - 4 bytes point to the next IFD (or 0).

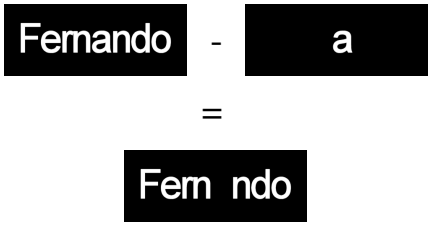
Operações Aritméticas/Lógicas



Operações Aritméticas/Lógicas



Exemplo de Subtracção



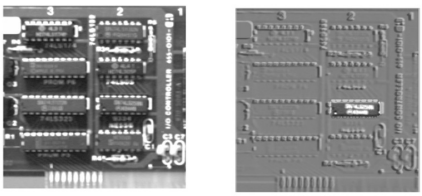
Exemplo de Subtracção

Fernando - a

=

Fernando

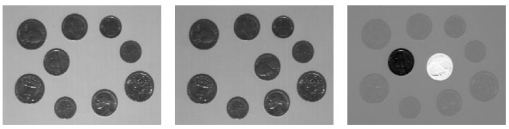
Exemplo de Subtracção



Difference images for quality control. A master image is subtracted from images of each subsequent part. In this example, the missing chip in a printed circuit board is evident in the difference image.

Exemplo de Subtracção

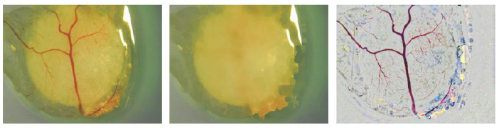
Procura de movimento



(a) (b) (c)

*Showing image differences by subtraction: **(a)** original image; **(b)** image after moving one coin; **(c)** difference image after pixel by pixel subtraction.*

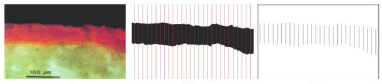
Subtração de cores



(a) (b) (c)

*Example of rank leveling in a color image as described in the text: **(a)** original image (courtesy Kathy Spencer, The Scripps Research Institute, La Jolla, CA); **(b)** background produced by color opening; **(c)** image **(a)** minus image **(b)**.*

Medida de espessura



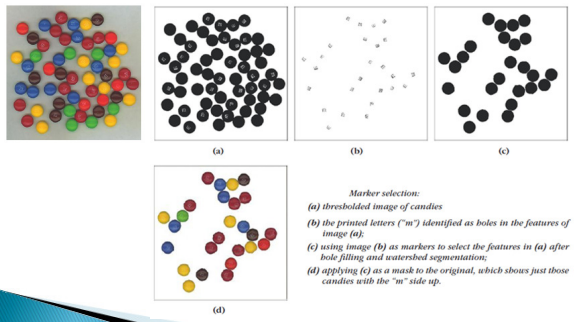
(a) (b) (c)

*Measurement of layer thickness: **(a)** paint layer viewed in cross-section; **(b)** thresholded layer, with superimposed grid of vertical lines; **(c)** AND of lines with layer producing line segments for measurement.*

(a) (b) (c)

*Measurement of layer thickness: **(a)** cross-section of vein in tissue; **(b)** thresholded wall with superimposed grid of radial lines; **(c)** AND of lines with layer producing line segments for measurement (note the errors introduced by non-perpendicular alignment of grid lines to wall, which is not exactly circular).*

Selecção de elementos



Marker selection:
(a) thresholded image of candies
(b) the printed letters ("m") identified as holes in the features of image (a);
(c) using image (b) as markers to select the features in (a) after hole filling and watershed segmentation;
(d) applying (c) as a mask to the original, which shows just those candies with the "m" side up.

Segmentação

Uma área importante do processamento de imagem é a segmentação de uma imagem em vários componentes para reconhecimento de objectos. O objectivo da segmentação é separar numa imagem os pixels dos objectos dos pixels do fundo. Técnicas de *Thresholding* separam um objecto do seu fundo baseadas no histograma dos graus de cinzento da imagem. Se os valores de cinzento de um objecto numa imagem são muito diferentes dos cinzentos do fundo da imagem, então torna-se bastante fácil encontrar o valor óptimo para fazer o *threshold*.

Outras técnicas de segmentação usam a descontinuidade de cinzentos dentro de uma imagem para detectar linhas ou pontos nessa imagem. Estas descontinuidades são depois usadas para separar objectos do seu fundo.

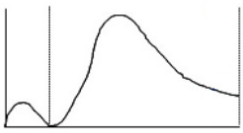
Segmentação
Thresholding

Thresholding é usado no processamento de imagem para separar os pixels de um objecto dos pixels do seu fundo. *Thresholding* converte uma imagem em graus de cinzento numa imagem binária (ou dois graus de cinzento). Esta operação é dedinida por:

$$g(x,y) = \begin{cases} G_{obj} & if \;\; f(x,y) > T \\ G_{back} & if \;\; f(x,y) \le T \end{cases}$$

Segmentação
Optimum Thresholding

O *Thresholding* óptimo de uma imagem assume que o histograma contem dois picos predominantes, um devido ao objecto e o outro devido à cor do fundo. O objectivo do *thresholding* óptimo é encontrar o 'vale' entre estes dois picos e separar a imagem a partir desse valor de cinzento. O histograma que se segue mostra a localização do valor óptimo de *thresholding*.



Segmentação
Multi-Graylevel *Thresholding*

Multi-Graylevel Thresholding é usado no processamento de imagem para separar diferentes regiões de cinzentos numa imagem. *Thresholding* converte uma imagem em graus de cinzento numa outra imagem com menos graus de cinzento do que a primeira. Para um exemplo de 4 graus de cinzento, esta operação é definida por:

$$g(x,y) = \begin{cases} G_3 & if \;\; f(x,y) > T_3 \\ G_2 & if \;\; T_2 < f(x,y) \le T_3 \\ G_1 & if \;\; T_1 < f(x,y) \le T_2 \\ G_0 & if \;\; f(x,y) < T_1 \end{cases}$$

Segmentação
Detector de Linhas

A detecção de linhas horizontais, verticais, diagonal esquerda e direita podem ser usadas para encontrar descontinuidades numa imagem. A detecção de linhas numa imagem é feita executando filtros espaciais na imagem usando as seguintes máscaras:

Horizontal
$$\begin{bmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{bmatrix}$$

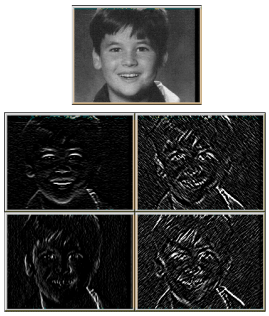
Vertical
$$\begin{bmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{bmatrix}$$

Diagonal Esquerda
$$\begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}$$

Diagonal Direita
$$\begin{bmatrix} -1 & -1 & 2 \\ -1 & 2 & -1 \\ 2 & -1 & -1 \end{bmatrix}$$

$$g(x, y) = \sum_{i=-1}^{1} \sum_{j=-1}^{1} f(x+i, y+j) * M(i, j)$$

Segmentação
Detector de Linhas



Segmentação
Detector de Pontos

A detecção de pontos pode ser usada para encontrar descontinuidades de pontos numa imagem. A detecção de pontos numa imagem é feita executando filtros espaciais na imagem usando a seguinte máscara:

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

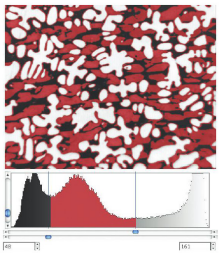
$$g(x, y) = \sum_{i=-1}^{1} \sum_{j=-1}^{1} f(x+i, y+j) * M(i, j)$$

Segmentação
blur

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \div 9$$

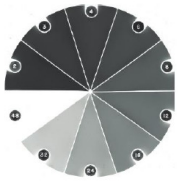
$$g(x, y) = \sum_{i=-1}^{1} \sum_{j=-1}^{1} f(x+i, y+j) * M(i, j)$$

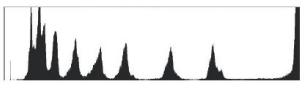
Thresholding

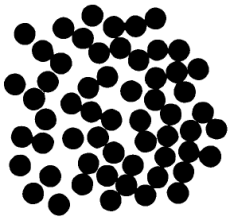


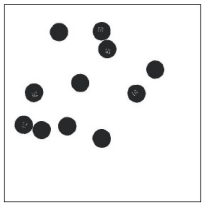
Example of thresholding an image by setting brightness limits on the histogram. A mouse is used to drag the sliders, the selected pixels are shaded in color, and the numeric values of the limits are shown.

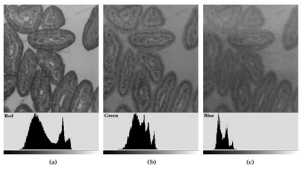
Thresholding



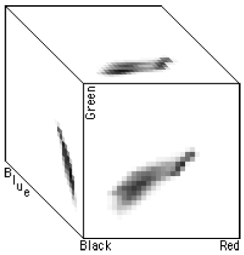


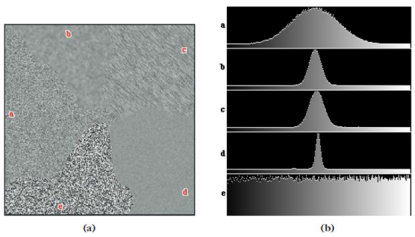




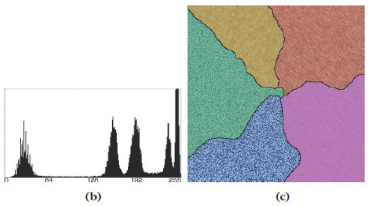


*Red, green and blue color channels from the image in **Figure** 7.7, with their brightness histograms.*

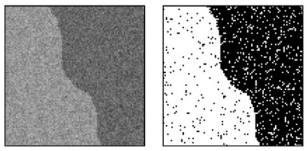


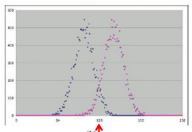


*Test image containing five different regions to be distinguished by differences in the textures. The brightness histograms are shown **(b)**; the average brightness of each region is the same.*



*the textures in **Figure 7.19a** to brightness differences: **(a)** result of **b)** histogram showing that the five regions are distinct in brightness; y thresholding the histogram in **(b)**, shown using colored overlays.*



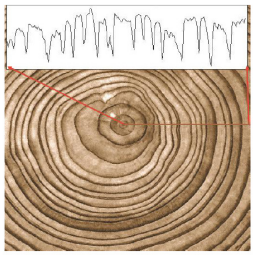


A test image containing two regions whose mean brightness levels are different, but which have variations in individual pixels that overlap.

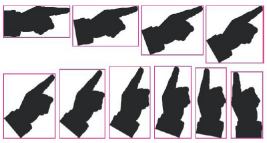
(a) original image (enlarged to show pixels);

(b) histogram showing overlap between regions.

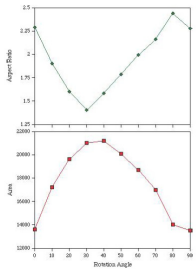
(c) result of setting a simple threshold at the minimum point.



Intensity scan across growth rings in a tree. The density and spacing of the rings are used for age-dating the wood.



(a)



(b)

Rotation of a feature and the bounding box ***(a)*** *and the changes in the size and shape of the bounding box plotted against rotation* ***(b)****.*

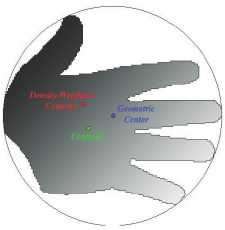


Figure 10.10 *Centroid (green), density weighted centroid (red) using the pixel density values, and geometric center (blue) determined as the center of the minimum bounding circle, for an irregular feature.*

Rácios sem Unidades

$$\text{Form factor} = \frac{4\pi \cdot \text{Area}}{\text{Perimeter}^2}$$

$$\text{Roundness} = \frac{4 \cdot \text{Area}}{\pi \cdot \text{Max Diameter}^2}$$

$$\text{Aspect Ratio} = \frac{\text{Max Caliper Dimension}}{\text{Min Caliper Dimension}}$$

$$\text{Elongation} = \frac{\text{Fiber Length}}{\text{Fiber Width}}$$

$$\text{Curl} = \frac{\text{Length}}{\text{Fiber Length}}$$

$$\text{Convexity} = \frac{\text{Convex Perimeter}}{\text{Perimeter}}$$

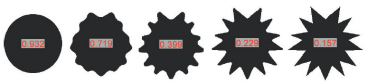
$$\text{Solidity} = \frac{\text{Area}}{\text{Convex Area}}$$

$$\text{Compactness} = \frac{\sqrt{\left(\frac{4}{\pi}\right)\text{Area}}}{\text{Max Diameter}}$$

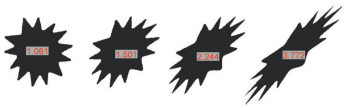
$$\text{Modification Ratio} = \frac{\text{Inscribed Diameter}}{\text{Circumscribed Diameter}}$$

$$\text{Extent} = \frac{\text{Net Area}}{\text{Bounding Rectangle}}$$

Rácios sem Unidades

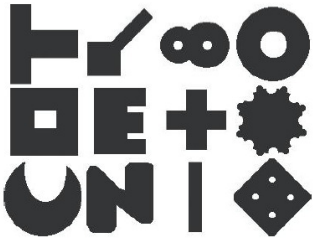


Formfactor: the shapes have the same area, but the increasing perimeter changes the measured formfactor as shown.



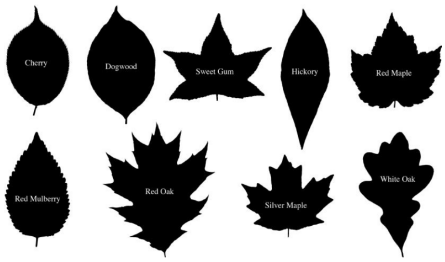
Aspect Ratio: the shapes have the same area and vary only slightly in formfactor, but greatly in aspect ratio, as shown.

Rácios sem Unidades

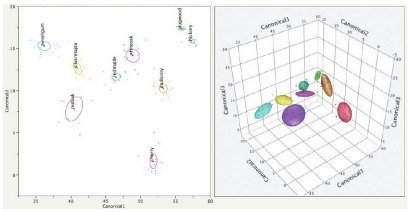


A set of features that are visually distinct, but have identical values for the formfactor (= 0.44).

Rácios sem unidades – exemplo



Examples of leaf shapes (not at the same scale).



Discriminant analysis results for the leaf samples using three classic size and shape measurements.

Roundness, solidity and convexity

Filtros Morfológicos

O processamento de uma imagem antes do seu reconhecimento e identificação normalmente envolve filtros morfológicos para mudar formas geométricas de objectos dessa imagem. O objectivo dos filtros morfológicos é amaciar os contornos de objectos e decompor uma imagem nas suas formas geométricas fundamentais para reconhecer a imagem.

Operações de filtragem morfológica podem ser separadas em duas categorias:
- Imagens binárias
- Imagens em graus de cinzento

Todas as operações de filtragem para ambos os tipos são derivadas de duas operações morfológicas fundamentais: Dilatação e Erosão.

Morfologia

Morphology - Conjunto de funções que filtram ou alteram imagens num variado número de maneiras. O utilizador especifica a operação morfológica assim como o elemento estruturante a ser aplicado. Uma imagem final é criada.

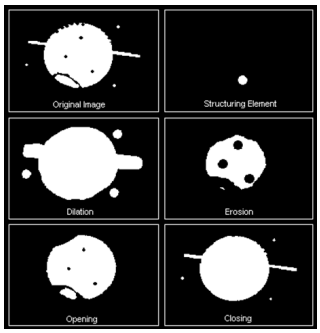
Dilation - Este processo expande regiões brancas da imagem encolhendo portanto as pretas. O efeito deste processo é preencher pequenos vazios dentro e à volta de um objecto.
Erosion - Ao contrário da *Dilation*, apaga áreas na imagem alargando assim as zonas escuras. Isto realça os vazios e fracturas dentro de um objecto enquanto que elimina pequenas manchas negras, tais como ruído e fundo.

Morfologia

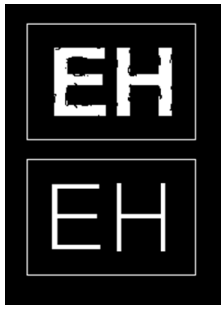
Opening - Consiste numa *Erosion* seguida de uma *Dilation*. Neste processo a maior parte dos objectos mantêm a sua forma original, ainda que pequenas áreas claras sejam apagadas. Este processo é usado para eliminar ruído no fundo enquanto que preserva o tamanho e forma.

Closing - Consiste numa *Dilation* seguida de uma *Erosion*. Este processo também mantém a sua forma e tamanhos originais, para além de juntar objectos adjacentes e preencher pequenos vazios ou falhas. Este processo é usado para alisar extremos ou para ligar objectos que estão ligeiramente desligados.

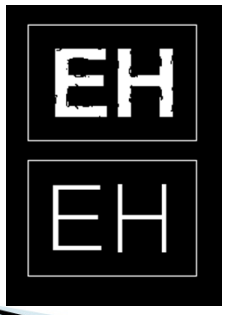
Morfologia

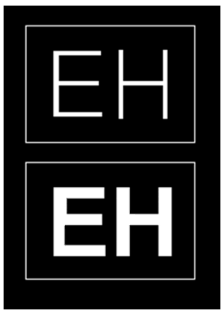


Processamento de Imagem Avançado

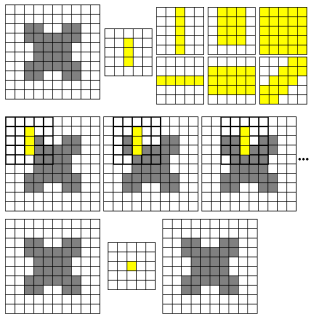


Processamento de Imagem Avançado





Filtros Morfológicos
Princípio de funcionamento

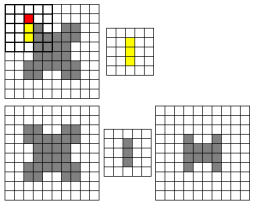


Filtros Morfológicos
Erosion (Binary) Filter

A Erosão binária de um objecto, retirada da subtracção de Minkowski, reduz a sua área geométrica através da transformação dos seus pixéis da borda para o valor do fundo da imagem. Erosão é definida como o complementar da dilatação do complementar do objecto.

Erosão

$$A \Theta B = (A^c \oplus B)^c$$



Filtros Morfológicos
Erosion (Graylevel) Filter

A Erosão de um objecto com níveis de cinzento, é usada para amaciar pequenas regiões de cinzentos positivos. Erosão de níveis de cinzento é definida como o mínimo da diferença entre uma região da imagem e a sua máscara de cinzentos.

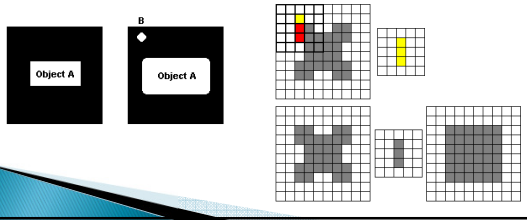
$$A \Theta B = \min[(A(x+i, y+j) - B(i, j)]$$

Filtros Morfológicos
Dilation (Binary) Filter

A Dilatação binária de um objecto, retirada da adição de Minkowski, aumenta a sua área geométrica através da transformação dos pixéis adjacentes à borda para o valor da cor do objecto na imagem. Dilatação é definida como a união de todos os vectores adição entre os pixéis do objecto A e da função estruturante B.

Dilatação

$$A \oplus B = \{t \in Z^2 : t = a + b, a \in A, b \in B\}$$

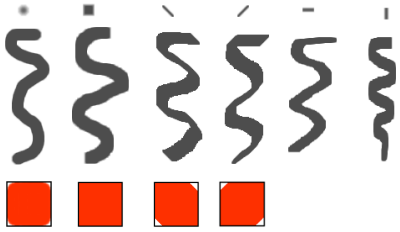


Filtros Morfológicos
Dilation (Graylevel) Filter

A Dilatação de um objecto com níveis de cinzento, é usada para amaciar pequenas regiões de cinzentos negativos. Dilatação de níveis de cinzento é definida como o máximo da soma entre uma região da imagem e a sua máscara de cinzentos.

$$A \oplus B = \max[(A(x+i, y+j) - B(i, j)]$$

Filtros Morfológicos
Elemento estruturante



Filtros Morfológicos
Closing (Binary) Filter

Fechar Morfológico de um objecto binário, é definido como a dilatação (binária) desse objecto seguido da erosão (binária) desse mesmo objecto. Esta operação de filtragem reduz pequenas falhas internas (ou ilhas) ou nas bordas (baias) do objecto.

$$Close(A, B) = (A \oplus B) \Theta B$$

Filtros Morfológicos
Closing (Graylevel) Filter

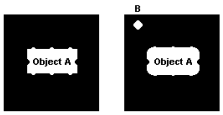
Fechar Morfológico de um objecto em níveis de cinzento, é definido como a dilatação (em níveis de cinzento) desse objecto seguido da erosão (em níveis de cinzento) desse mesmo objecto. Esta operação de filtragem reduz pequenas regiões de cinzentos orientadas negativamente e ainda regiões de ruido de cinzentos negativos.

$$Close(A, B) = (A \oplus B) \Theta B$$

Filtros Morfológicos
Opening (Binary) Filter

Abrir morfológico de um objecto binário, é definido como a erosão (binária) desse objecto seguido da dilatação (binária) desse mesmo objecto. Esta operação de filtragem reduz pequenos excessos (marcas) externas nas bordas objecto.

$$Open(A, B) = (A \Theta B) \oplus B$$

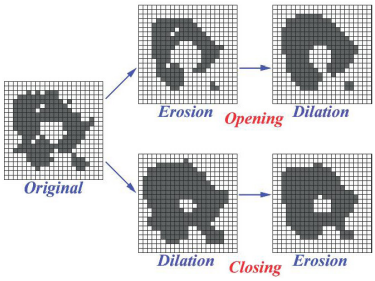


Filtros Morfológicos
Opening (Graylevel) Filter

Abrir Morfológico de um objecto em níveis de cinzento, é definido como a erosão (em níveis de cinzento) desse objecto seguido da dilatação (em níveis de cinzento) desse mesmo objecto. Esta operação de filtragem reduz pequenas regiões de cinzentos orientadas positivamente e ainda regiões de ruido de cinzentos positivos.

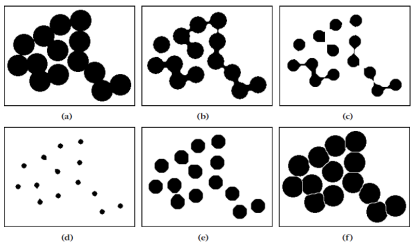
$$Open(A,B) = (A \Theta B) \oplus B$$

Exemplos



Combining erosion and dilation to produce an opening or a closing. The result is different depending on the order of application of the two operations. Since the original image is ambiguous, it is necessary to use a priori knowledge to select the proper combination.

Exemplos



*Separation of touching features by erosion/dilation: **(a)** original test image; **(b)** after two cycles of erosion; **(c)** after four cycles; **(d)** after seven cycles (features are now all fully separated); **(e)** four cycles of dilation applied to image **(d)**; **(f)** additional cycles of non-merging dilation restricted to the original pixel locations, which restores the feature boundaries.*

Exemplos

*Removing power lines: **(a)** original image; **(b)** result of applying an opening and closing using a vertical neighborhood. Only the horizontal lines are removed; a different neighbor pattern is needed to remove the other wires.*

Exemplos

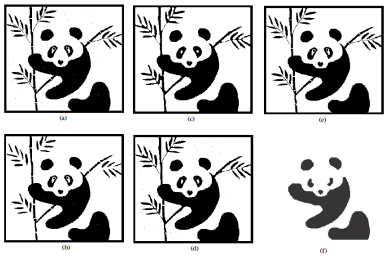


*Illustration of the effect of different neighbor coefficients and number of iterations. **(a)** original test image; **(b)** Erosion, neighbor coefficient = 3, 1 iteration, removes isolated lines and points; **(c)** Closing, neighbor coefficient = 2, 2 iterations, fills in gaps to connect features while removing isolated points; **(d)** Closing using classical operations (neighbor coefficient = 0, 1 iteration) connects most features but leaves isolated points; **(e)** opening, neighbor coefficient = 7, 1 iteration, removes point noise without affecting anything else; **(f)** opening, neighbor coefficient = 1, 4 iterations, removes all small features including the frame of the picture.*

Exemplos

Filtros Morfológicos
Skeleton (Binary) Filter

Skeletonization de um objecto, proposto inicialmente por H. Blum, define uma representação geométrica comprimida única de um objecto. É por vezes referido como *Medial Axis Transform*. A 'esqueletização' morfológica é definida como a união de um grupo de pixels obtidos pela diferença entre a erosão da N-ésima imagem e a sua 'abertura'.

$$K_n(A \quad = Erosion_n(\quad - Open(Eroc \quad _n(A), B)$$

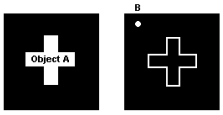
Filtros Morfológicos
Outline (Binary) Filter

O contorno binário de um objecto é o processo de mudar todos os pixels do objecto para a côr do fundo excepto os pixels que estão na borda do objecto. É definido como a erosão da imagem subtraida da imagem original ou a imagem original subtraida da sua dilatação. A grossura do contorno é dada pelo tamanho da função estruturante B.

$$Outline(A,B) = A - (A \Theta B)$$

ou

$$Outline(A,B) = (A \oplus B) - A$$

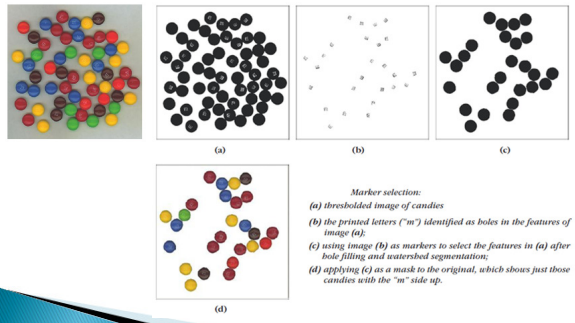


Filtros Morfológicos
Top-Hat (Graylevel) Filter

O *Top-Hat* Morfológico de uma imagem em níveis de cinzento, é definido como a diferença entre a imagem original e a sua versão 'aberta'. Este filtro é usado para realçar características de baixo contraste e alta frequência espacial numa imagem.

$$TopHat(A,B) = A - Open(A,B)$$

Selecção de elementos



Marker selection:
(a) thresholded image of candies
(b) the printed letters ("m") identified as holes in the features of image (a).
(c) using image (b) as markers to select the features in (a) after hole filling and watershed segmentation;
(d) applying (c) as a mask to the original, which shows just those candies with the "m" side up.

Filtros Morfológicos
Hit-Miss (Binary) Filter

A operação *Hit-Miss* binária, é usada para extrair características geométricas de um objecto binário. A máscara usada nesta operação determina o tipo de características que são extraídas. Esta operação é definida por:

$$HitMiss\,(A,B,C) = (A \Theta B) \cap (A^c \Theta C)$$

Filtros Morfológicos
Thickening (Binary) Filter

A operação de 'engrossamento' binário é usada para aumentar o tamanho geométrico de um objecto. Esta operação é normalmente usada interactivamente até que a imagem pretendida seja obtida. Esta operação é definida por:

for I=1 to 8

$$A^{i+1} = A^i \cup [HitMiss(A^i, B^i, C^i)]$$

Thickening(A)=A^9

$$B^1\begin{bmatrix}1&1&1\\0&0&0\\0&0&0\end{bmatrix}\quad B^5\begin{bmatrix}0&1&1\\0&0&1\\0&0&0\end{bmatrix}\qquad C^1\begin{bmatrix}0&0&0\\1&1&1\\1&1&1\end{bmatrix}\quad C^5\begin{bmatrix}1&0&0\\1&1&0\\1&1&1\end{bmatrix}$$

$$B^2\begin{bmatrix}0&0&0\\0&0&0\\1&1&1\end{bmatrix}\quad B^6\begin{bmatrix}0&0&0\\1&0&0\\1&1&0\end{bmatrix}\qquad C^2\begin{bmatrix}1&1&1\\1&1&1\\0&0&0\end{bmatrix}\quad C^6\begin{bmatrix}1&1&1\\0&1&1\\0&0&1\end{bmatrix}$$

$$B^3\begin{bmatrix}1&0&0\\1&0&0\\1&0&0\end{bmatrix}\quad B^7\begin{bmatrix}0&0&0\\0&0&1\\0&1&1\end{bmatrix}\qquad C^3\begin{bmatrix}0&1&1\\0&1&1\\0&1&1\end{bmatrix}\quad C^7\begin{bmatrix}1&1&1\\1&1&0\\1&0&0\end{bmatrix}$$

$$B^4\begin{bmatrix}0&0&1\\0&0&1\\0&0&1\end{bmatrix}\quad B^8\begin{bmatrix}1&1&0\\1&0&0\\0&0&0\end{bmatrix}\qquad C^4\begin{bmatrix}1&1&0\\1&1&0\\1&1&0\end{bmatrix}\quad C^8\begin{bmatrix}0&0&1\\0&1&1\\1&1&1\end{bmatrix}$$

Filtros Morfológicos
Thinning (Binary) Filter

A operação de 'emagrecimento' binário é usada para reduzir o tamanho geométrico de um objecto. Esta operação é normalmente usada interactivamente até que a imagem pretendida seja obtida ou até que não ocorram mais alterações na imagem. Esta operação é definida por:

for I=1 to 8

$$A^{i+1} = A^i \cap [HitMiss(A^i, B^i, C^i)]^c$$

Thickening(A)=A^9

$$B^1\begin{bmatrix}0&0&0\\1&1&1\\1&1&1\end{bmatrix}\quad B^5\begin{bmatrix}1&0&0\\1&1&0\\1&1&1\end{bmatrix}\qquad C^1\begin{bmatrix}1&1&1\\0&0&0\\0&0&0\end{bmatrix}\quad C^5\begin{bmatrix}0&1&1\\0&0&1\\0&0&0\end{bmatrix}$$

$$B^2\begin{bmatrix}1&1&1\\1&1&1\\0&0&0\end{bmatrix}\quad B^6\begin{bmatrix}1&1&1\\0&1&1\\0&0&1\end{bmatrix}\qquad C^2\begin{bmatrix}0&0&0\\0&0&0\\1&1&1\end{bmatrix}\quad C^6\begin{bmatrix}0&0&0\\1&0&0\\1&1&0\end{bmatrix}$$

$$B^3\begin{bmatrix}0&1&1\\0&1&1\\0&1&1\end{bmatrix}\quad B^7\begin{bmatrix}1&1&1\\1&1&0\\1&0&0\end{bmatrix}\qquad C^3\begin{bmatrix}1&0&0\\1&0&0\\1&0&0\end{bmatrix}\quad C^7\begin{bmatrix}0&0&0\\0&0&1\\0&1&1\end{bmatrix}$$

$$B^4\begin{bmatrix}1&1&0\\1&1&0\\1&1&0\end{bmatrix}\quad B^8\begin{bmatrix}0&0&1\\0&1&1\\1&1&1\end{bmatrix}\qquad C^4\begin{bmatrix}0&0&1\\0&0&1\\0&0&1\end{bmatrix}\quad C^8\begin{bmatrix}1&1&0\\1&0&0\\0&0&0\end{bmatrix}$$

(a)

(c)

(b)

Surveillance video image with horizontal scan line noise (a), a custom neighborhood to perform median filtering primarily in a vertical direction (b), and the result (c).