

# Procesamiento Digital de Imágenes

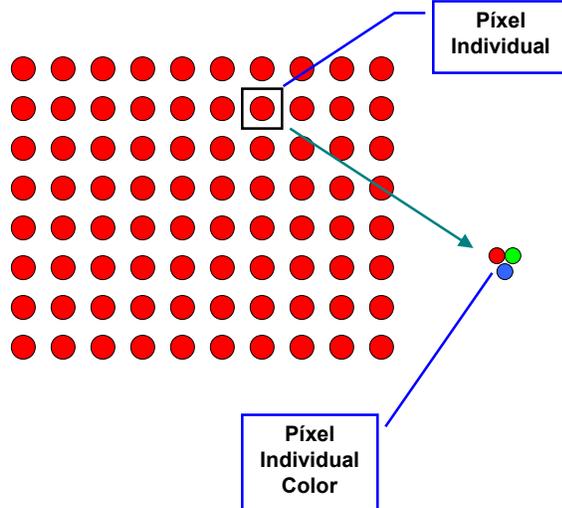
Pablo Roncagliolo B.  
Nº 3



# Procesamiento Básico de Imágenes Digitales



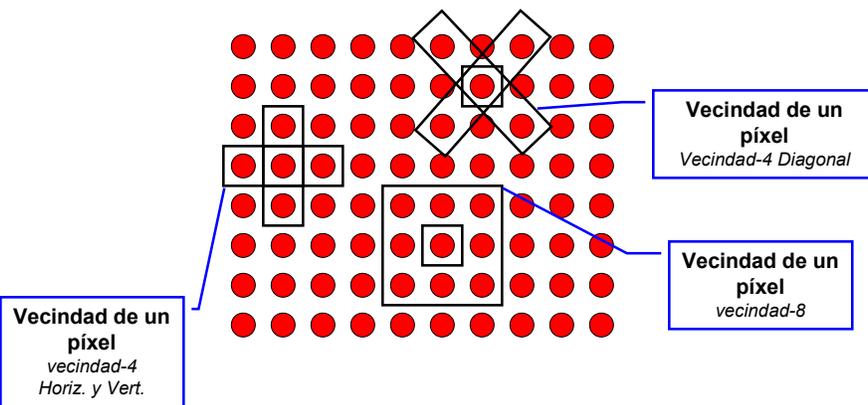
# Representación de imágenes digitales



prb

3

# Vecindad



prb

4

## Vecindad



- Píxel  $p$  con coordenadas  $(x,y)$ 
  - Vecindad-4
    - $V_4(p) = \{ (x+1,y), (x-1,y), (x,y+1), (x,y-1) \}$
    - $V_D(p) = \{ (x+1,y+1), (x-1,y-1), (x-1,y+1), (x+1,y-1) \}$
  - Vecindad-8
    - $V_8(p) = \{ V_4(p) \cup V_D(p) \}$

prb

5

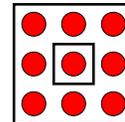
## Ej. Vecindad



- Código para calcular el promedio del punto  $(ff,cc)$  y su Vecindad-8:

```
sum=0;
for f=ff-1:ff+1
    for c=cc-1:cc+1
        sum=sum+IM(f,c);
    end;
end;

prom=sum/9
```



¿Qué ocurre si el punto  $(ff,cc)$  pertenece al borde de la imagen? Proponga un código "robusto" para calcular el promedio de una vecindad 8 para cualquier punto de la imagen.

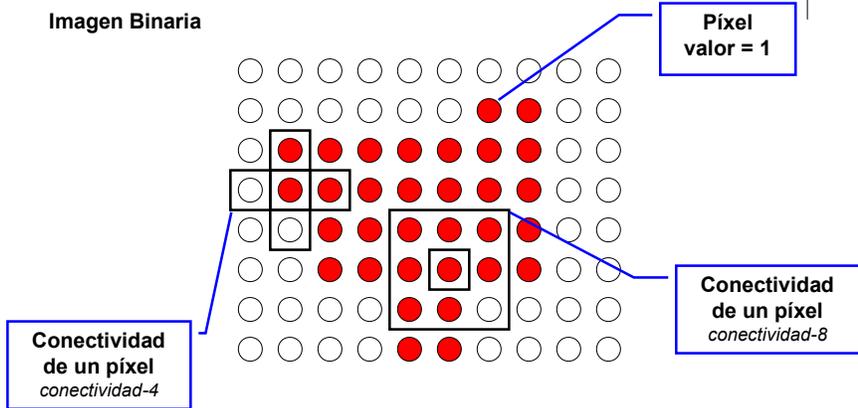
prb

6

# Conectividad



Imagen Binaria



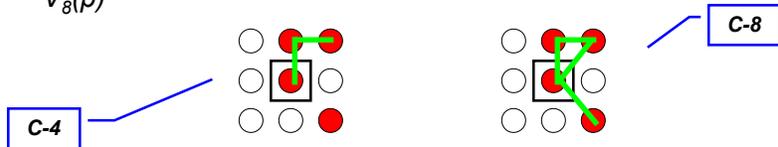
prb

7

# Conectividad



- Sea  $G$  el conjunto de niveles de gris utilizados para determinar la conectividad de dos píxeles.
  - Ej. imagen binaria  $\rightarrow G = \{1\}$
- Conectividad-4
  - $C_4(p, q) = \text{Verdadero}$ , si  $p$  y  $q$  pertenecen a  $G$  y  $q$  pertenece a  $V_4(p)$
- Conectividad-8
  - $C_8(p, q) = \text{Verdadero}$ , si  $p$  y  $q$  pertenecen a  $G$  y  $q$  pertenece a  $V_8(p)$



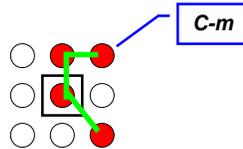
prb

8

# Conectividad



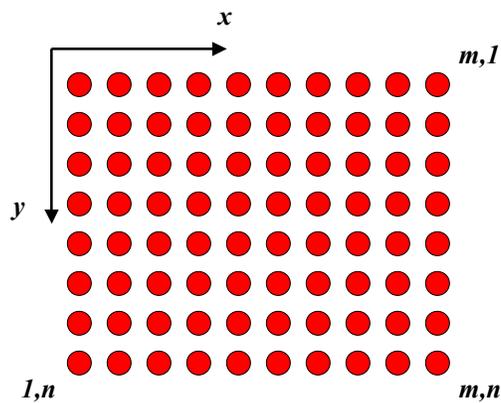
- Conectividad-m (mixta)
  - $C_m(p,q) = \text{Verdadero}$ , si  $p$  y  $q$  pertenecen a  $G$  y:
    - $q \in V_4(p)$  ó
    - $q \in a V_D(p)$  y  $V_4(p) \cap V_4(q) = \emptyset$ , para píxeles  $\in G$



prb

9

# Representación de imágenes digitales



**Nota:**  
\* Array(1:n)  
\*\* Array(0..n-1)  
  
\* Matlab, Pascal  
\*\* C, Delphi

prb

10

# Resolución Espacial



**256x256**  
(65536 byte)  
(64kb)



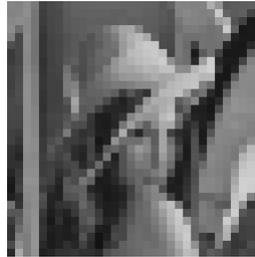
**128x128**  
(16384 byte)  
(16Kb)



**64x64**  
(4096 byte)  
(4Kb)



**32x32**  
(1024 byte)  
(1Kb)



prb

11

# Resolución en Amplitud



**7bpp**



**4bpp**



**2bpp**



**1bpp**



¿Cuántos  
colores en  
cada imagen?

¿Cuánto pesa  
cada imagen  
en memoria y  
en disco?

prb

12

# Resolución en Amplitud



4bpp

Cada Pixel:

$G = \{0..15\}$

(Lenguaje C)

```
for(i=0; i<256;i++)  
  for(j=0; j<256; j=j+2)  
  {  
    P=M(i,j)<<4;  
    P=P & M(i,j+1);  
    write(P)  
  }
```

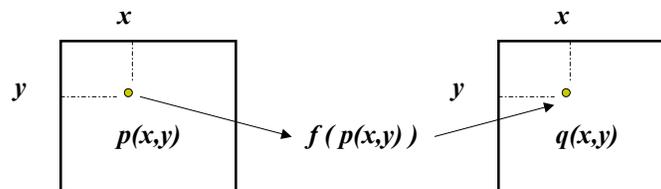
prb

13

# Operadores básicos



## Operaciones Individuales



$$q(x,y) = f(p(x,y))$$

prb

14

# Operadores básicos



## Operaciones Individuales

### a) Operador Identidad

$$q(x,y) = p(x,y)$$

### b) Operador Inverso o negativo

$$q(x,y) = 255 - p(x,y)$$



prb



15

# Operadores básicos



## Operaciones Individuales

### c) Operador Umbral

$$q(x,y) = 0 \text{ para } p(x,y) < u$$
$$q(x,y) = 255 \text{ para } p(x,y) > u$$

### d) Operador Umbral Inverso



prb



16

# Operadores básicos



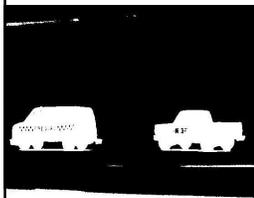
## Operaciones Individuales



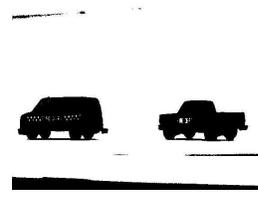
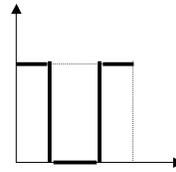
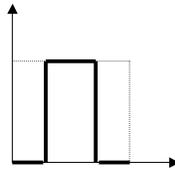
### e) Operador Intervalo de Umbral binario

$$q(x,y) = 0 \text{ para } p(x,y) < u1 \text{ ó } p(x,y) > u2$$

$$q(x,y) = 255 \text{ para } u1 > p(x,y) < u2$$



prb



17

# Operadores básicos



## Operaciones Individuales

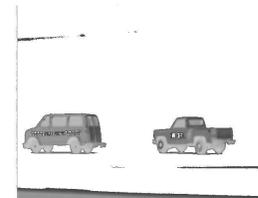
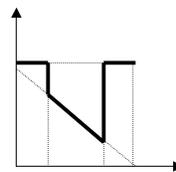
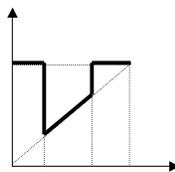
### h) Operador Intervalo de Umbral en Gris

$$q(x,y) = 255 \text{ para } p(x,y) < u1 \text{ ó } p(x,y) > u2$$

$$q(x,y) = p(x,y) \text{ para } u1 > p(x,y) < u2$$



prb



18

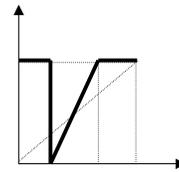
# Operadores básicos



## Operaciones Individuales

### j) Operador de extensión

$$q(x,y) = 255 \text{ para } p(x,y) < u1 \text{ ó } p(x,y) > u2$$
$$q(x,y) = 255 * (p(x,y) - u1) / (u2 - u1) \text{ para } u1 > p(x,y) < u2$$



prb

19

# Operadores básicos



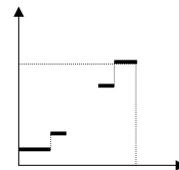
## Operaciones Individuales

### k) Operador reducción niveles de gris

$$q(x,y) = 0 \text{ para } p(x,y) < u1$$
$$q(x,y) = q1 \text{ para } u1 > p(x,y) < u2$$

....

$$q(x,y) = qn \text{ para } u_{n-1} > p(x,y) < 255$$



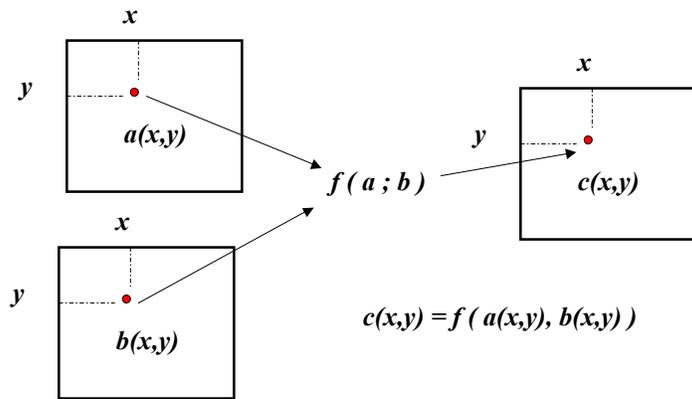
prb

20

# Operadores básicos



Operaciones de 2 imágenes punto a punto



prb

21

# Operadores básicos



Operaciones de  $K$  imágenes punto a punto

A) SUMA

$$c(x,y) = (a(x,y) + b(x,y)) / k, \text{ en este caso } k=2$$



prb

22

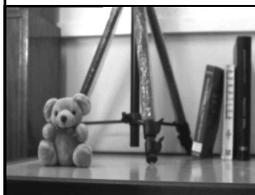
# Operadores básicos



Operaciones de  $K$  imágenes punto a punto

## B) RESTA

$$c(x,y) = \text{abs}( a(x,y) - b(x,y) )$$



prb



23

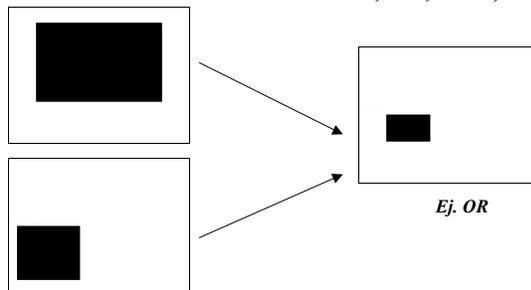
# Procesamiento básico de imágenes digitales



Operaciones de  $K$  imágenes punto a punto

## C) Transformaciones Lógicas

*AND, OR, NOT, XOR*



prb

Considere "blanco"=verdadero y "negro"=falso

24

## Práctica:



**Sistema básico de seguridad, que permite determinar los objetos ausentes o desplazados en un recinto.**

**Descargar Demo del sitio web.**