Disclaimer

- All images/contents used in this presentation are copyright of original owners.
- The PPT has prepared for academic use only.

DIGITAL IMAGE PROCESSING

Basic Relationships between Pixels

By:

HARSHAVARDHANREDDY

AIET, GLB

Neighbors of a Pixel

$$
f(x,y) = \begin{bmatrix}
f(0,0) & f(0,1) & f(0,2) & f(0,3) & f(0,4) & ----- \\
f(1,0) & f(1,1) & f(1,2) & f(1,3) & f(1,4) & ----- \\
f(2,0) & f(2,1) & f(2,2) & f(2,3) & f(2,4) & ----- \\
f(3,0) & f(3,1) & f(3,2) & f(3,3) & f(3,4) & ----- \\
| & | & | & | & | & ----- \\
| & | & | & | & | & -----
\end{bmatrix}
$$

Neighbors of a Pixel

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & f(0,2) & f(0,3) & f(0,4) & ----- \\ f(1,0) & f(1,1) & f(1,2) & f(1,3) & f(1,4) & ----- \\ f(2,0) & f(2,1) & f(2,2) & f(2,3) & f(2,4) & ----- \\ f(3,0) & f(3,1) & f(3,2) & f(3,3) & f(3,4) & ----- \\ | & | & | & | & | & ----- \\ | & | & | & | & | & ----- \end{bmatrix}$$

- A Pixel p at coordinates (x, y) has 4 horizontal and vertical neighbors.
- Their coordinates are given by:

(x+1, y)	(x-1, y)	(x, y+1)	&	(x, y-1)
f(2,1)	f(0,1)	f(1,2)		f(1,0)

- This set of pixels is called the ***4-neighbors*** of p denoted by $N_4(p)$.
- Each pixel is unit distance from (x ,y).

Neighbors of a Pixel

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & f(0,2) & f(0,3) & f(0,4) & ----- \\ f(1,0) & f(1,1) & f(1,2) & f(1,3) & f(1,4) & ----- \\ f(2,0) & f(2,1) & f(2,2) & f(2,3) & f(2,4) & ----- \\ f(3,0) & f(3,1) & f(3,2) & f(3,3) & f(3,4) & ----- \\ | & | & | & | & | & ----- \\ | & | & | & | & | & ----- \end{bmatrix}$$

- A Pixel p at coordinates (x, y) has 4 diagonal neighbors.
- Their coordinates are given by:

$(x+1, y+1)$ $f(2,2)$ $\quad (x+1, y-1)$ $f(2,0)$ $\quad (x-1, y+1)$ $f(0,2)$ $\quad$ & $\quad (x-1, y-1)$ $f(0,0)$

- This set of pixels is called the ***diagonal-neighbors*** of p denoted by $N_D(p)$.
- diagonal neighbors + 4-neighbors = 8-neighbors of p.
- They are denoted by $N_8(p)$. So, $N_8(p) = N_4(p) + N_D(p)$

Adjacency, Connectivity

Adjacency: Two pixels are adjacent if they are neighbors and their intensity level 'V' satisfy some specific criteria of similarity.

e.g. V = {1}
V = { 0, 2}
Binary image = { 0, 1}
Gray scale image = { 0, 1, 2, ------, 255}

In binary images, 2 pixels are adjacent if they are neighbors & have some intensity values either 0 or 1.

In gray scale, image contains more gray level values in range 0 to 255.

Adjacency, Connectivity

4-adjacency: Two pixels p and q with the values from set 'V' are 4-adjacent if q is in the set of $N_4(p)$.

e.g. V = { 0, 1}

1	1	2
1	1	0
1	0	1

p in **RED** color
q can be any value in **GREEN** color.

Adjacency, Connectivity

8-adjacency: Two pixels p and q with the values from set 'V' are 8-adjacent if q is in the set of $N_8(p)$.

e.g. $\boxed{V = \{1, 2\}}$

$$\begin{matrix} 0 & 1 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{matrix}$$

p in **RED** color
q can be any value in **GREEN** color

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$ OR

(ii) q is in $N_D(p)$ & the set $\boxed{N_4(p) \cap N_4(q)}$ have no pixels whose values are from 'V'.

e.g. V = { 1 }

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$

e.g. V = { 1 }

(i) b & c

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$

e.g. V = { 1 }

(i) b & c

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Soln: b & c are m-adjacent.

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$

e.g. V = { 1 }

(ii) b & e

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 l

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$

e.g. V = { 1 }

(ii) b & e

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 I

Soln: b & e are m-adjacent.

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$ OR

e.g. V = { 1 }

(iii) e & i

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_D(p)$ & the set $\underline{N_4(p) \cap N_4(q)}$ have no pixels whose values are from 'V'.

e.g. V = { 1 }

(iii) e & i

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_D(p)$ & the set $N_4(p) \cap N_4(q)$ have no pixels whose values are from 'V'.

e.g. V = { 1 }

(iii) e & i

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Soln: e & i are m-adjacent.

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$ OR

(ii) q is in $N_D(p)$ & the set $\underline{N_4(p) \cap N_4(q)}$ have no pixels whose values are from 'V'.

e.g. V = { 1 }

(iv) e & c

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$ OR

(ii) q is in $N_D(p)$ & the set $N_4(p) \cap N_4(q)$ have no pixels whose values are from 'V'.

e.g. V = { 1 }

(iv) e & c

One is common and it is a v{1}. so, it is not m-adjacent.

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Soln: e & c are NOT m-adjacent.

Adjacency, Connectivity

Connectivity: 2 pixels are said to be connected if their exists a path between them.

Let 'S' represent subset of pixels in an image.

Two pixels p & q are said to be connected in 'S' if their exists a path between them consisting entirely of pixels in 'S'.

For any pixel p in S, the set of pixels that are connected to it in S is called a **connected component of S**.

Paths

Paths: A path from pixel p with coordinate (x, y) with pixel q with coordinate (s, t) is a sequence of distinct sequence with coordinates (x_0, y_0), (x_1, y_1), ….., (x_n, y_n) where

$(x, y) = (x_0, y_0)$
& $(s, t) = (x_n, y_n)$

Closed path: $(x_0, y_0) = (x_n, y_n)$

Paths

Example # 1: Consider the image segment shown in figure. Compute length of the ***shortest-4, shortest-8 & shortest-m paths*** between pixels p & q where,
V = {1, 2}.

4	2	3	2 q
3	3	1	3
2	3	2	2
p 2	1	2	3

Paths

Example # 1:

$\boxed{\text{Shortest-4 path:}}$

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2 →	1	2	3	

Paths

Example # 1:

Shortest-4 path:

V = {1, 2}.

	4		2		3		2 q
	3		3		1		3
	2		3		2		2
p	2	→	1	→	2		3

Paths

<u>Example # 1</u>:

<u>Shortest-4 path:</u>

V = {1, 2}.

$$
\begin{array}{ccccccccc}
 & 4 & & 2 & & 3 & & 2 & q \\
 & 3 & & 3 & & 1 & & 3 & \\
 & 2 & & 3 & & 2 & & 2 & \\
 & & & & & \uparrow & & & \\
p & 2 & \rightarrow & 1 & \rightarrow & 2 & & 3 &
\end{array}
$$

Paths

Example # 1:

Shortest-4 path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

Paths

Example # 1:

Shortest-4 path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

Paths

Example # 1:

Shortest-4 path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

So, Path does not exist.

Paths

Example # 1:

Shortest-8 path:

V = {1, 2}.

	4	2	3	2 q
	3	3	1	3
	2	3	2	2
p	2	1	2	3

Paths

Example # 1:

Shortest-8 path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2 →	1	2	3	

Paths

<u>Example # 1</u>:

<u>Shortest-8 path:</u>

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

Paths

Example # 1:

Shortest-8 path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

Paths

<u>Example # 1</u>:

<u>Shortest-8 path:</u>

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

Paths

Example # 1:

Shortest-8 path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

So, shortest-8 path = 4

Paths

Example # 1:

Shortest-m path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

Paths

Example # 1:

Shortest-m path:

V = {1, 2}.

$$\begin{array}{cccccc} & 4 & 2 & 3 & 2 & q \\ & 3 & 3 & 1 & 3 & \\ & 2 & 3 & 2 & 2 & \\ p & 2 \rightarrow & 1 & 2 & 3 & \end{array}$$

Paths

Example # 1:

Shortest-m path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2 →	1 →	2	3	

Paths

Example # 1:

Shortest-m path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2 ↑	2	
p	2 →	1 →	2	3	

Paths

Example # 1:

Shortest-m path:

V = {1, 2}.

$$\begin{array}{cccccccc}
 & 4 & & 2 & & 3 & 2 & q \\
 & 3 & & 3 & & 1 & 3 & \\
 & 2 & & 3 & & 2 & 2 & \\
p & 2 & \rightarrow & 1 & \rightarrow & 2 & 3 &
\end{array}$$

Paths

Example # 1:

Shortest-m path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

Paths

Example # 1:

Shortest-m path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

So, shortest-m path = 5

Regions & Boundaries

Region: Let R be a subset of pixels in an image. Two regions Ri and Rj are said to be adjacent if their union form a connected set.

Regions that are not adjacent are said to be disjoint.

We consider 4- and 8- adjacency when referring to regions.

Below regions are adjacent only if 8-adjacency is used.

$$\begin{array}{cccl} 1 & 1 & 1 & \\ 1 & 0 & 1 & R_i \\ 0 & 1 & 0 & \\ 0 & 0 & 1 & \\ 1 & 1 & 1 & R_j \\ 1 & 1 & 1 & \end{array}$$

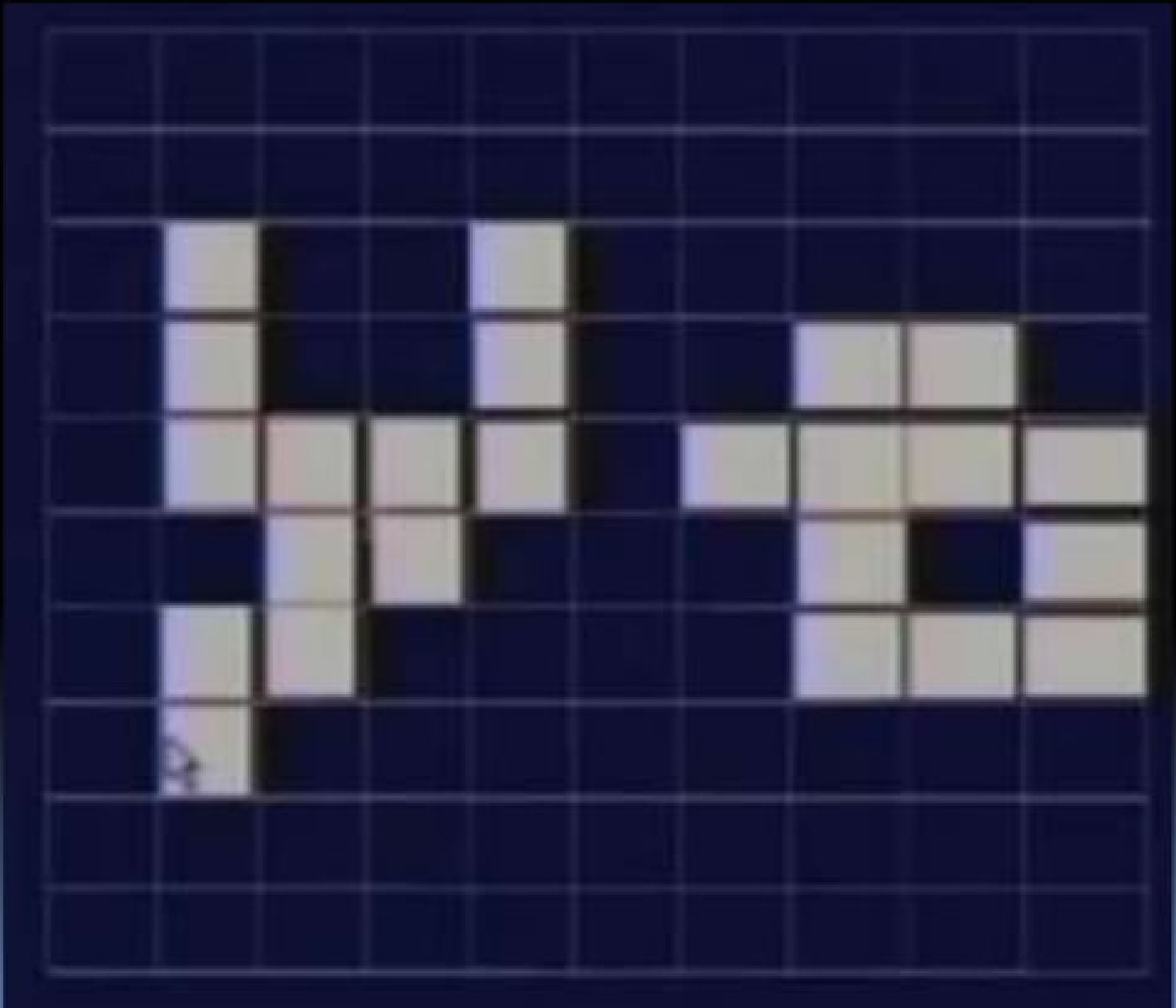
Regions & Boundaries

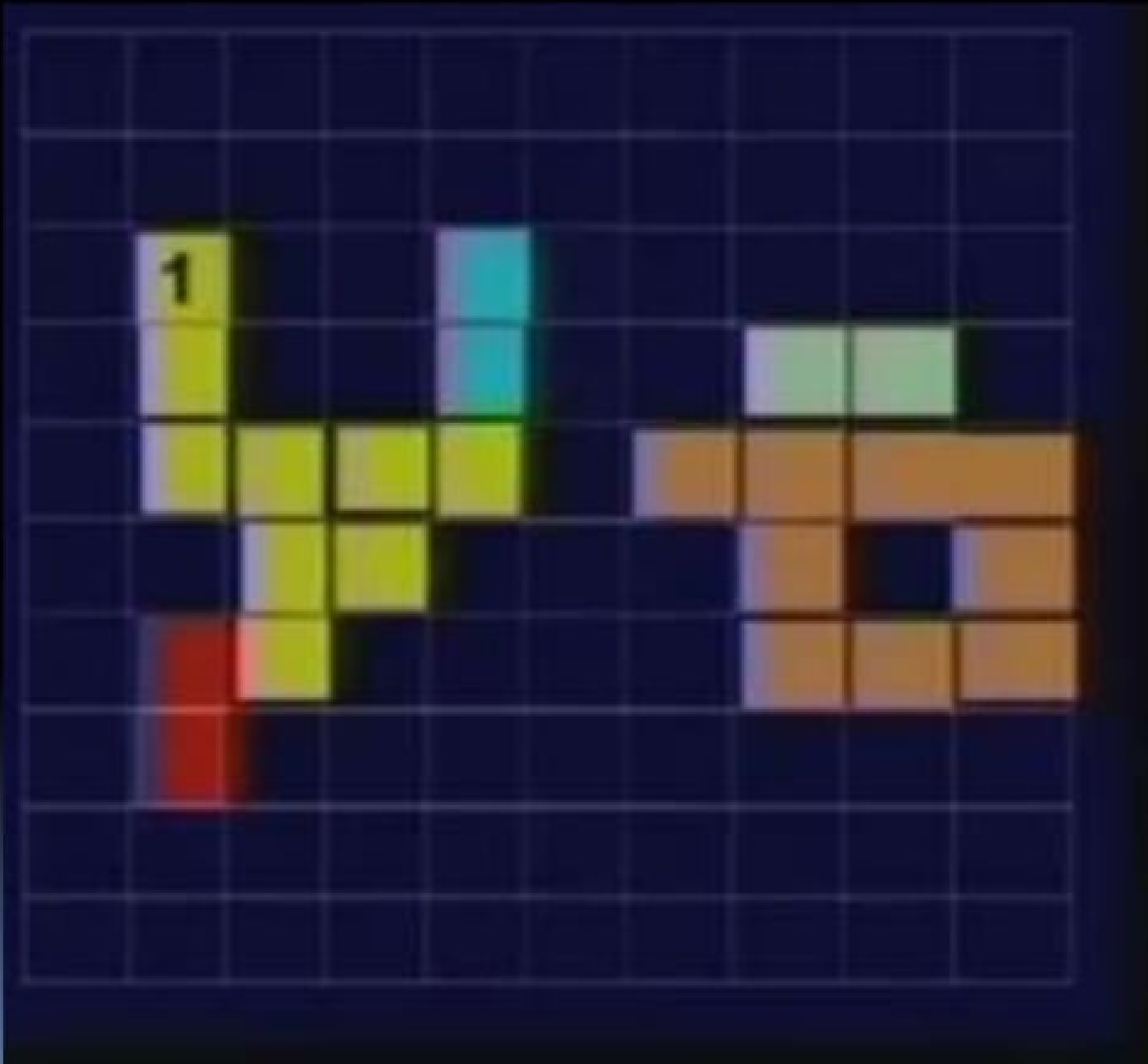
Boundaries (border or contour): The boundary of a region R is the set of points that are adjacent to points in the compliment of R.

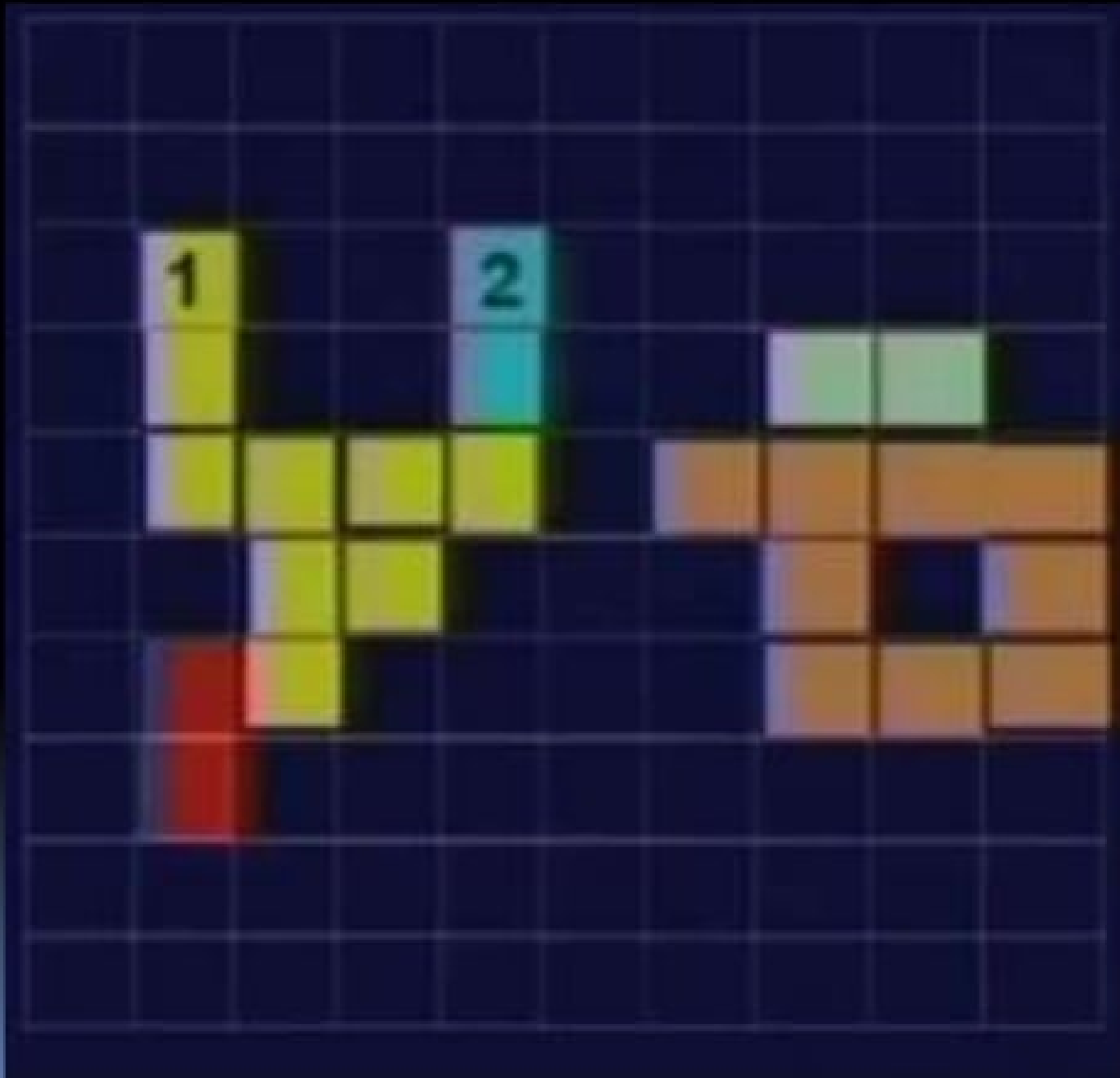
0	0	0	0	0
0	1	1	0	0
0	1	1	0	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

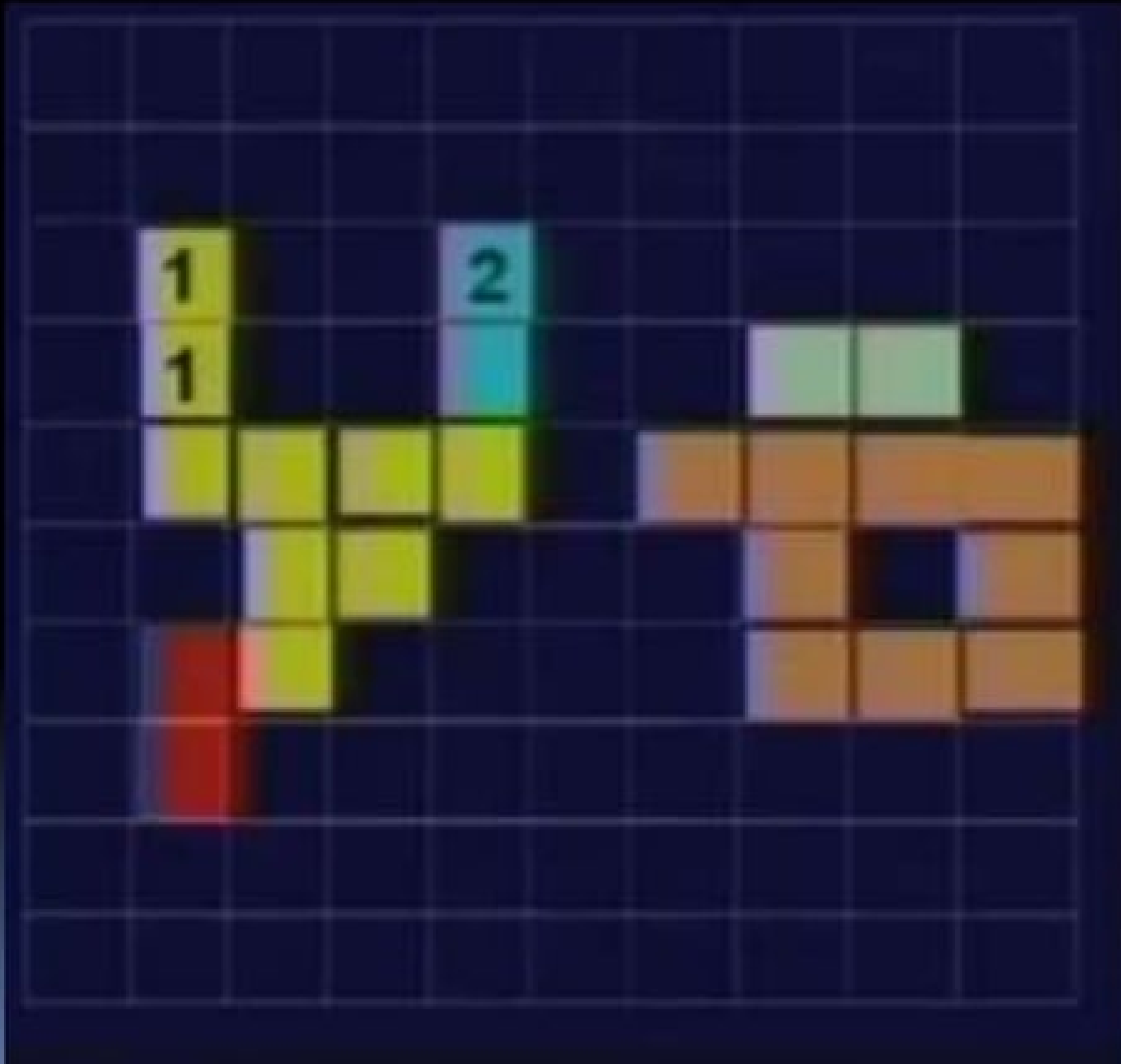
RED colored 1 is NOT a member of border if 4-connectivity is used between region and background. It is if 8-connectivity is used.

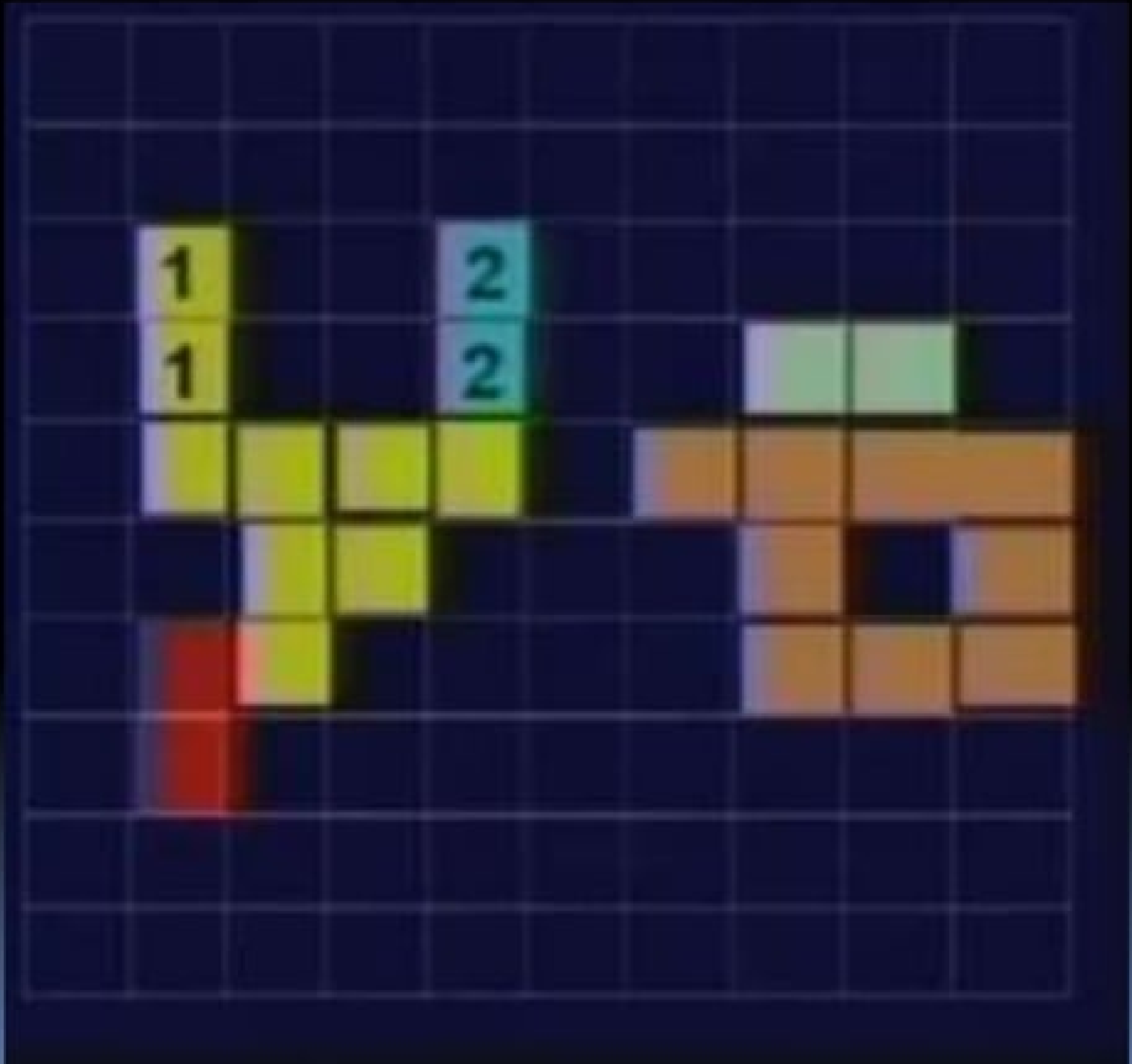
Example:

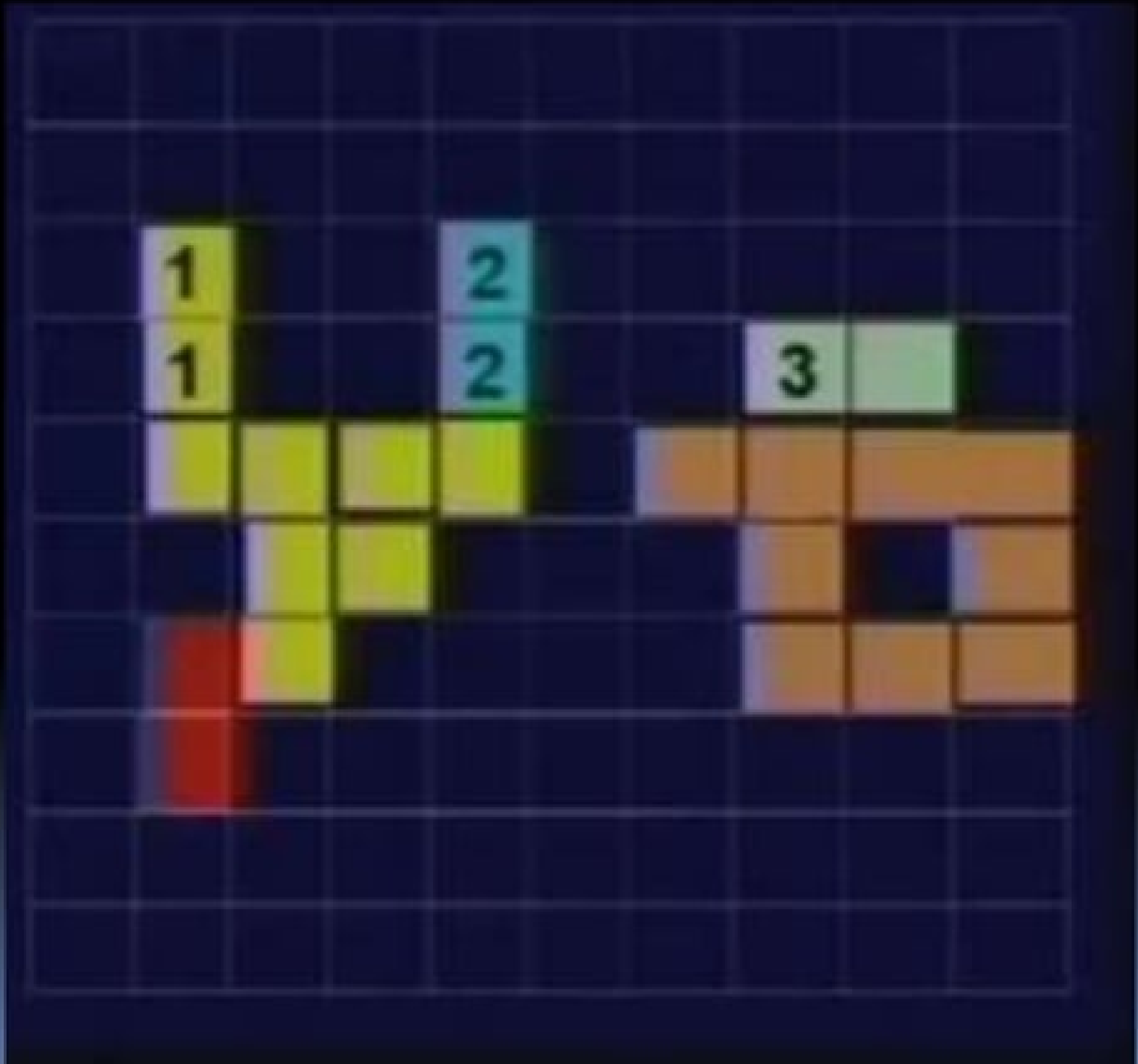


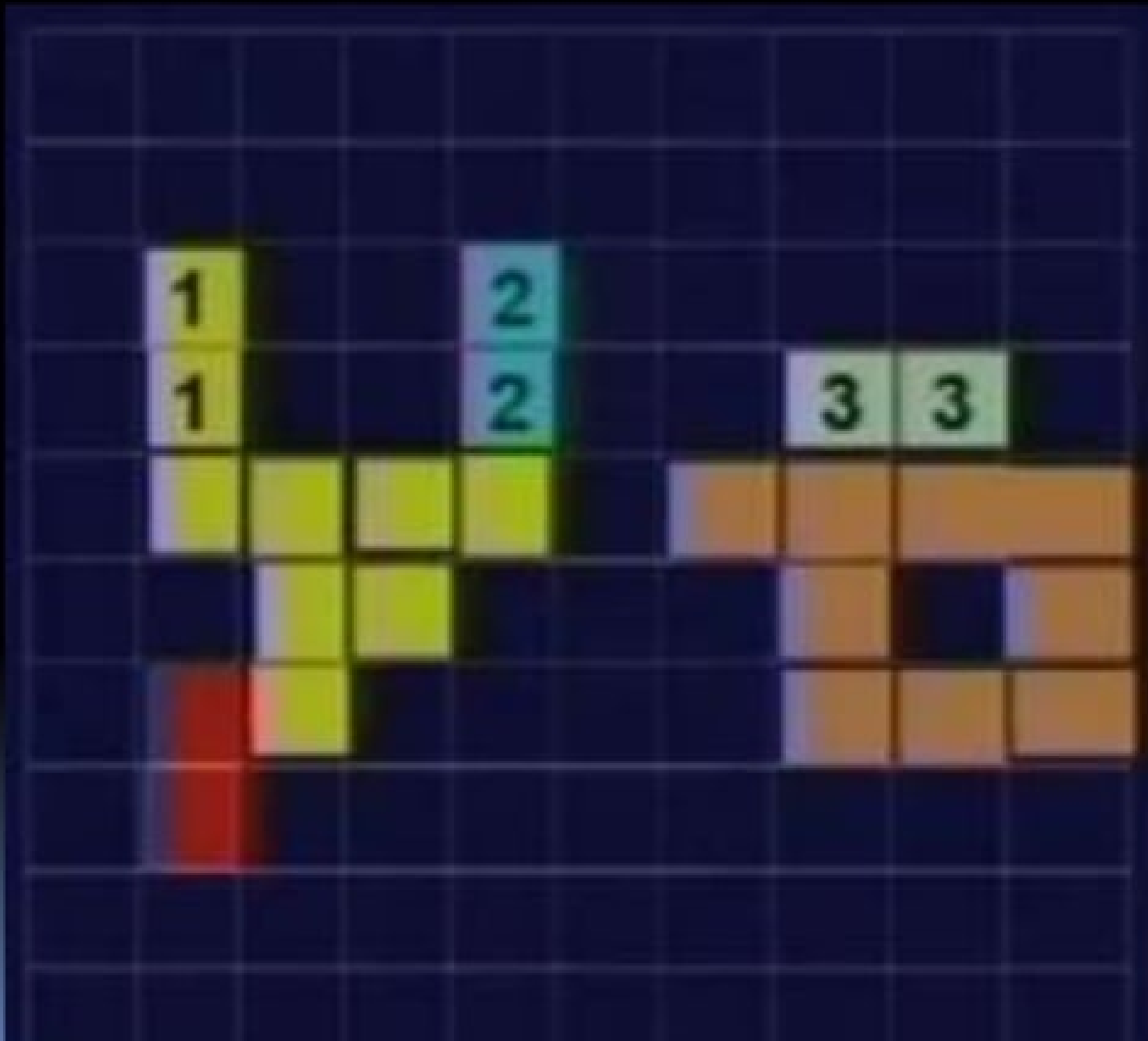

1


1
2

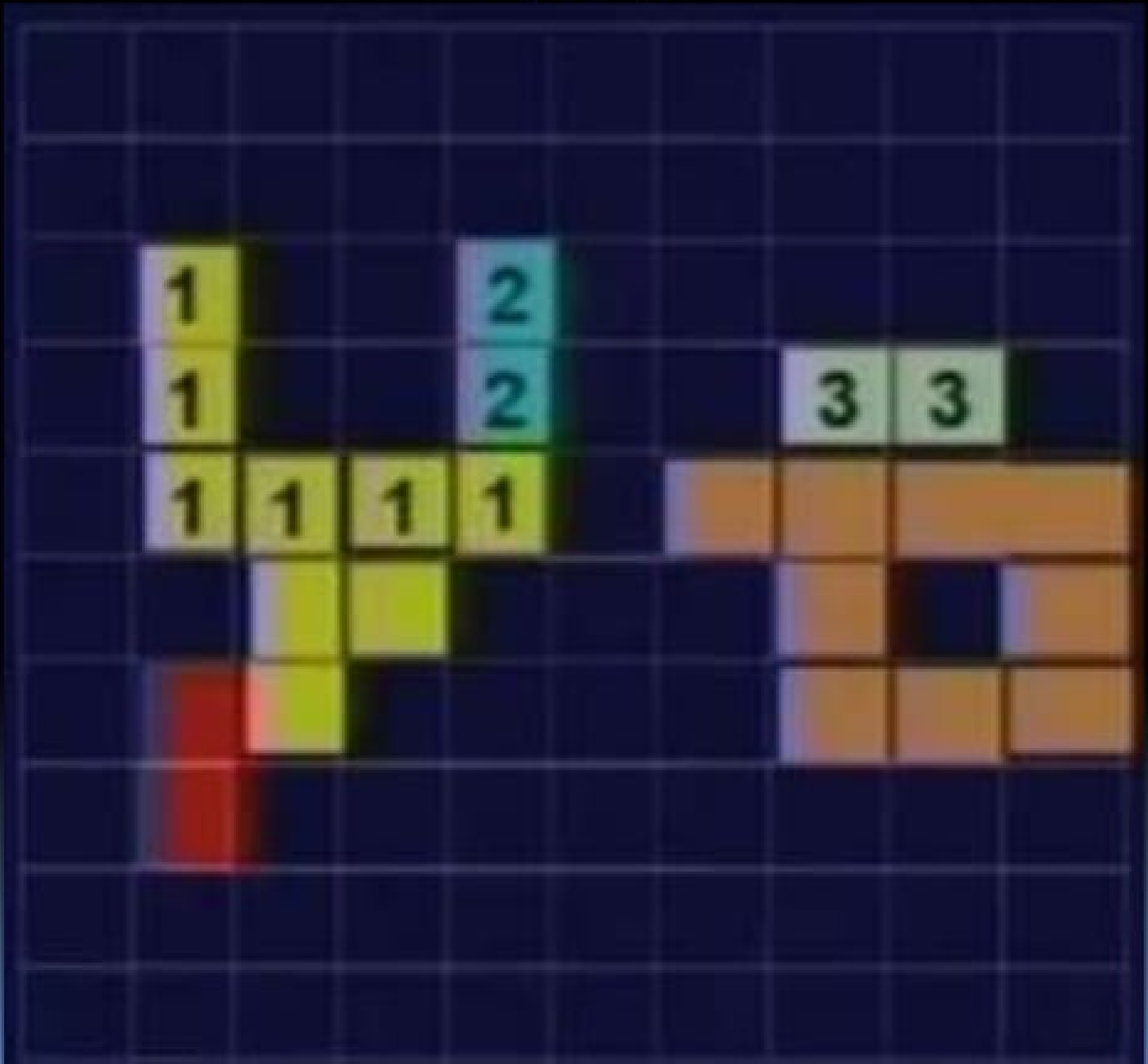

1
1
2

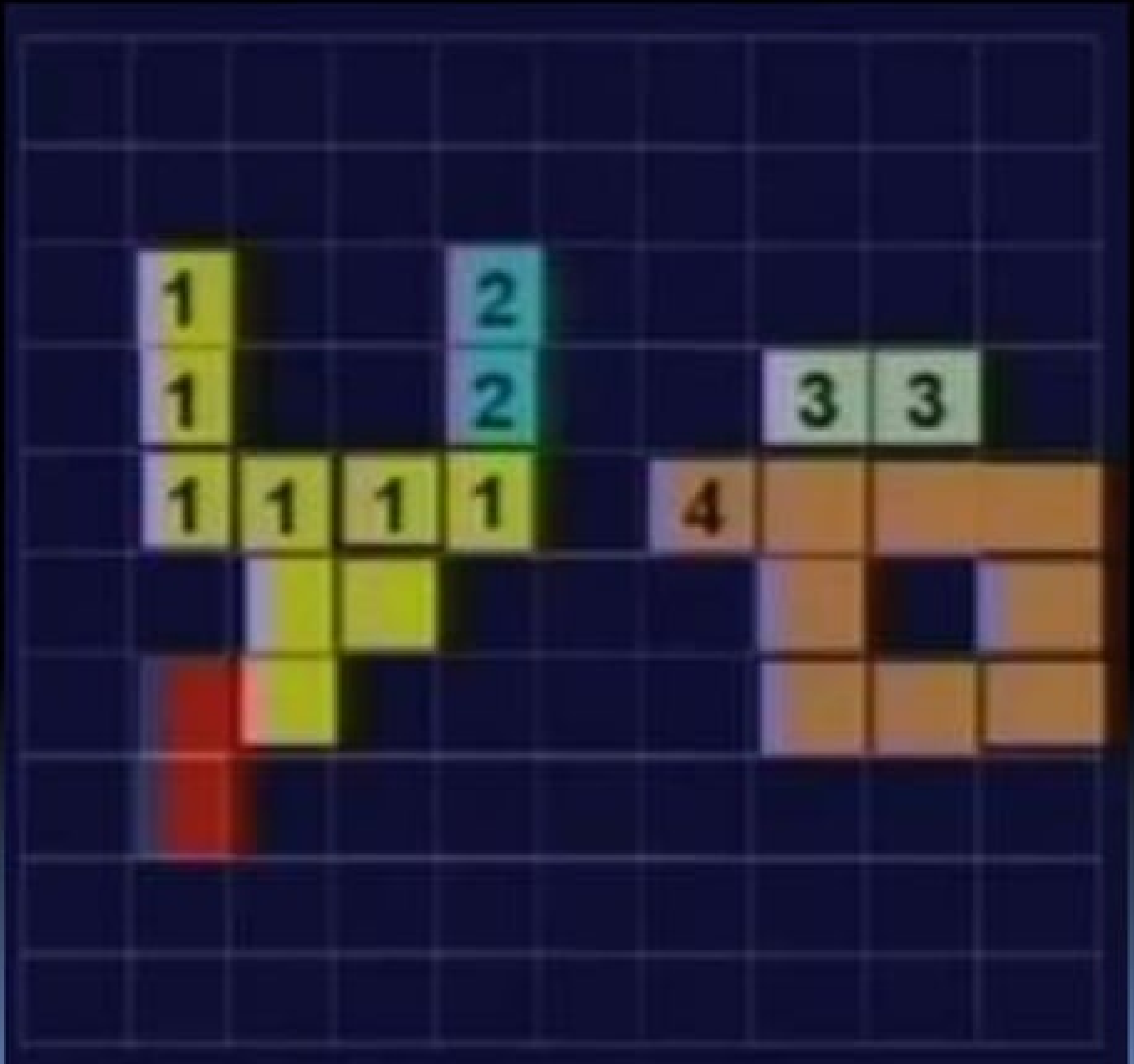

1
1
2
2


1
1
2
2
3

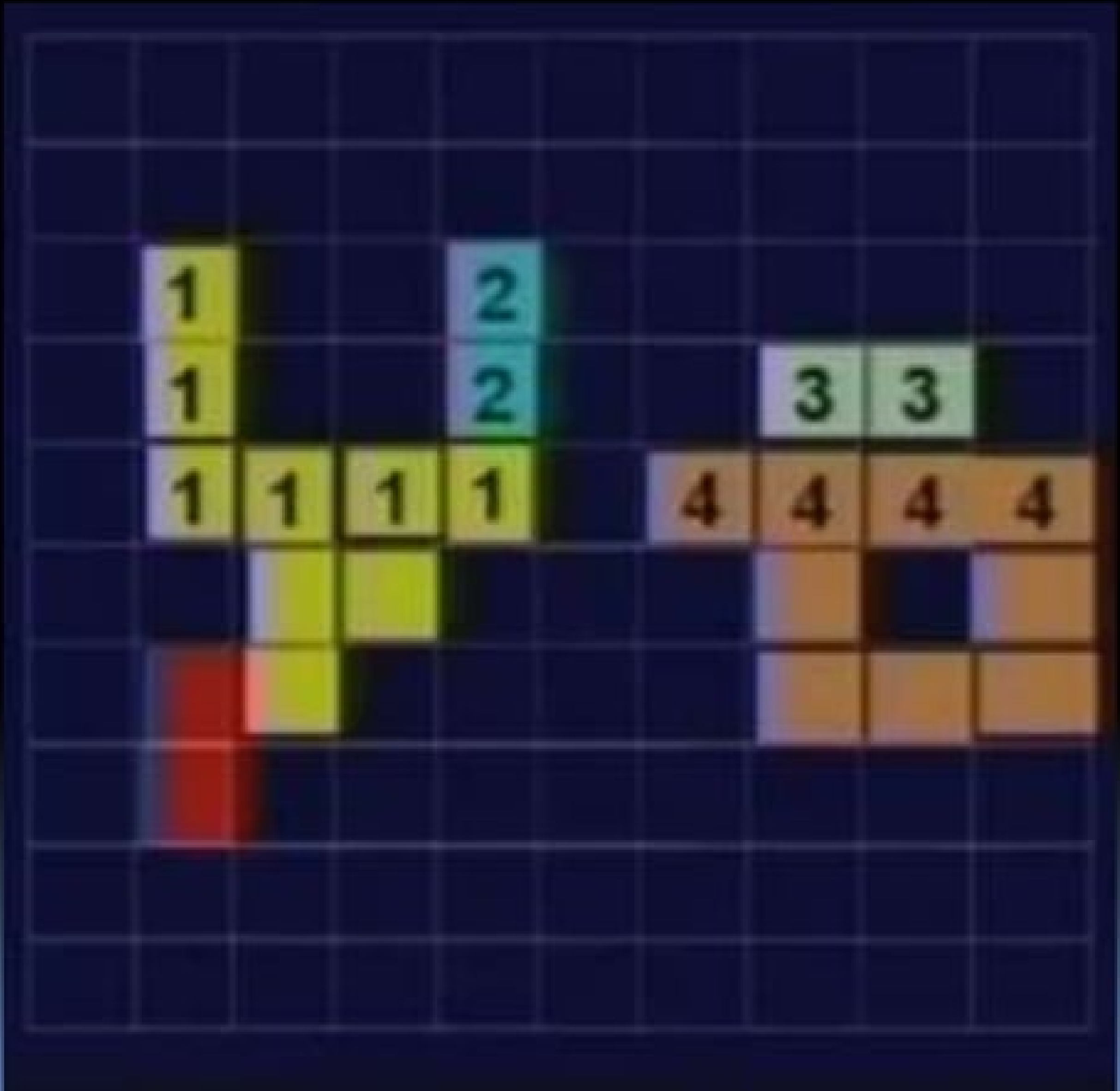

1
1
2
2
3 3

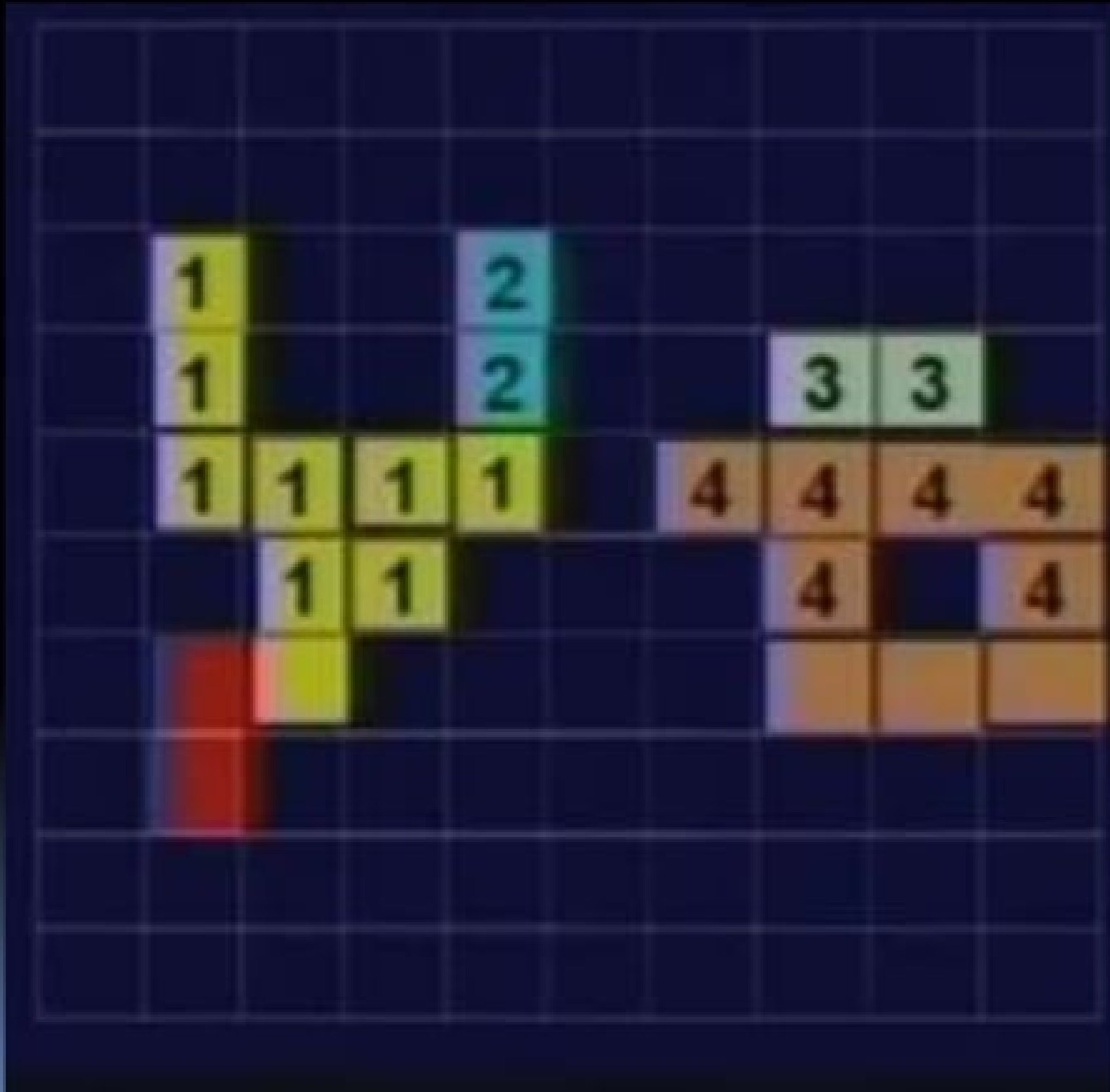
(1=2)



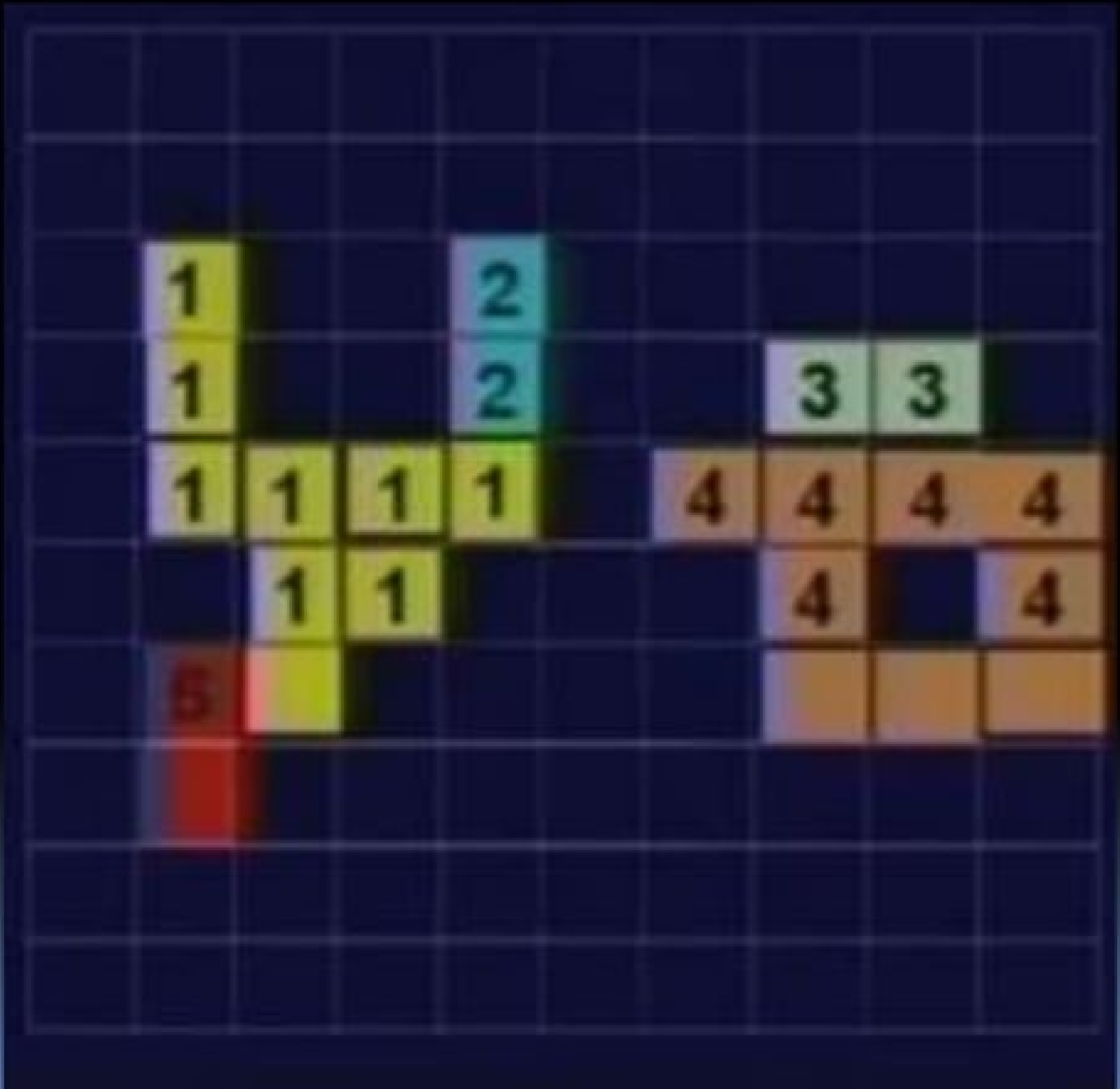

1
2
1
2
3 3
1 1 1 1
4

(3=4)

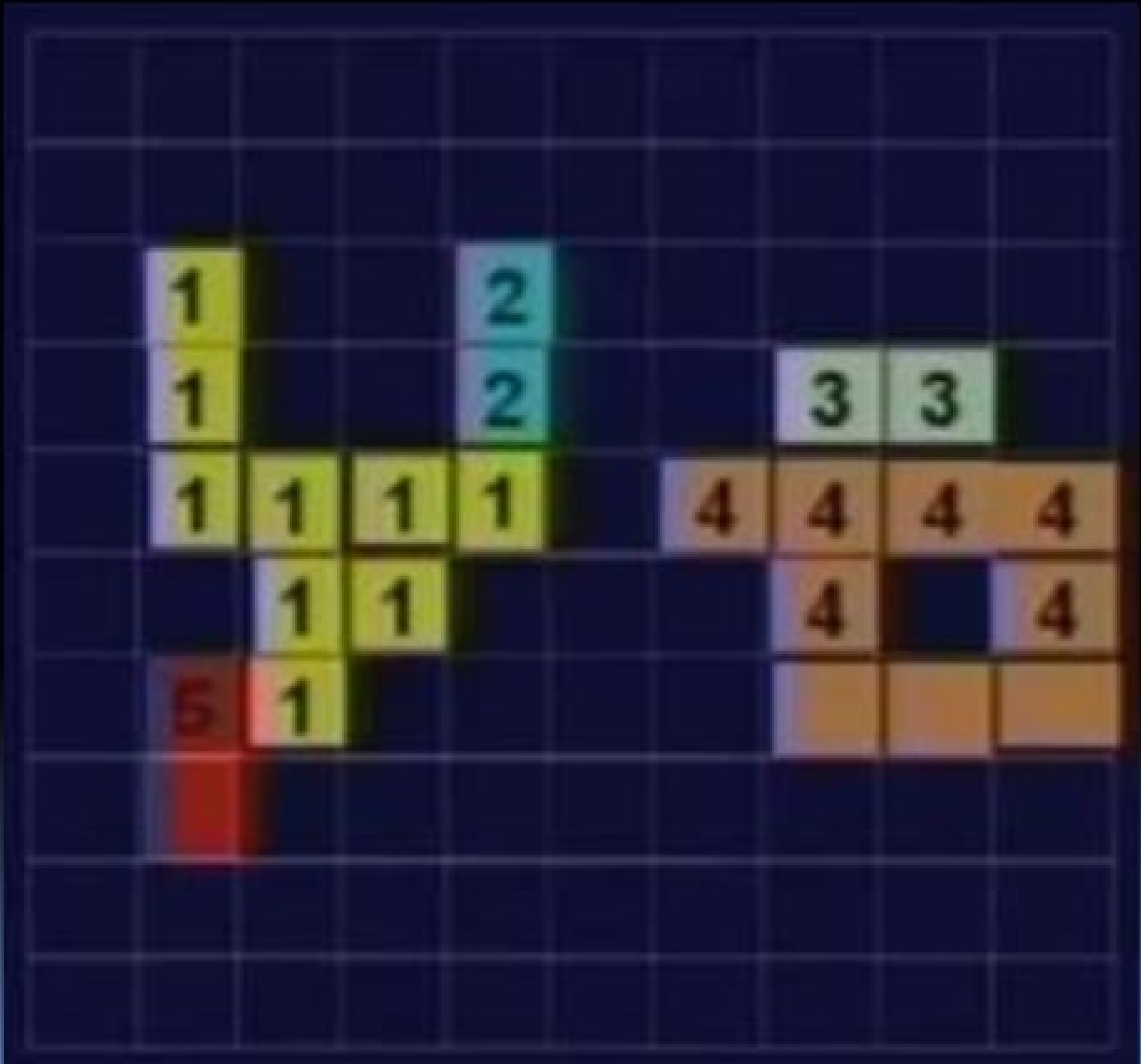




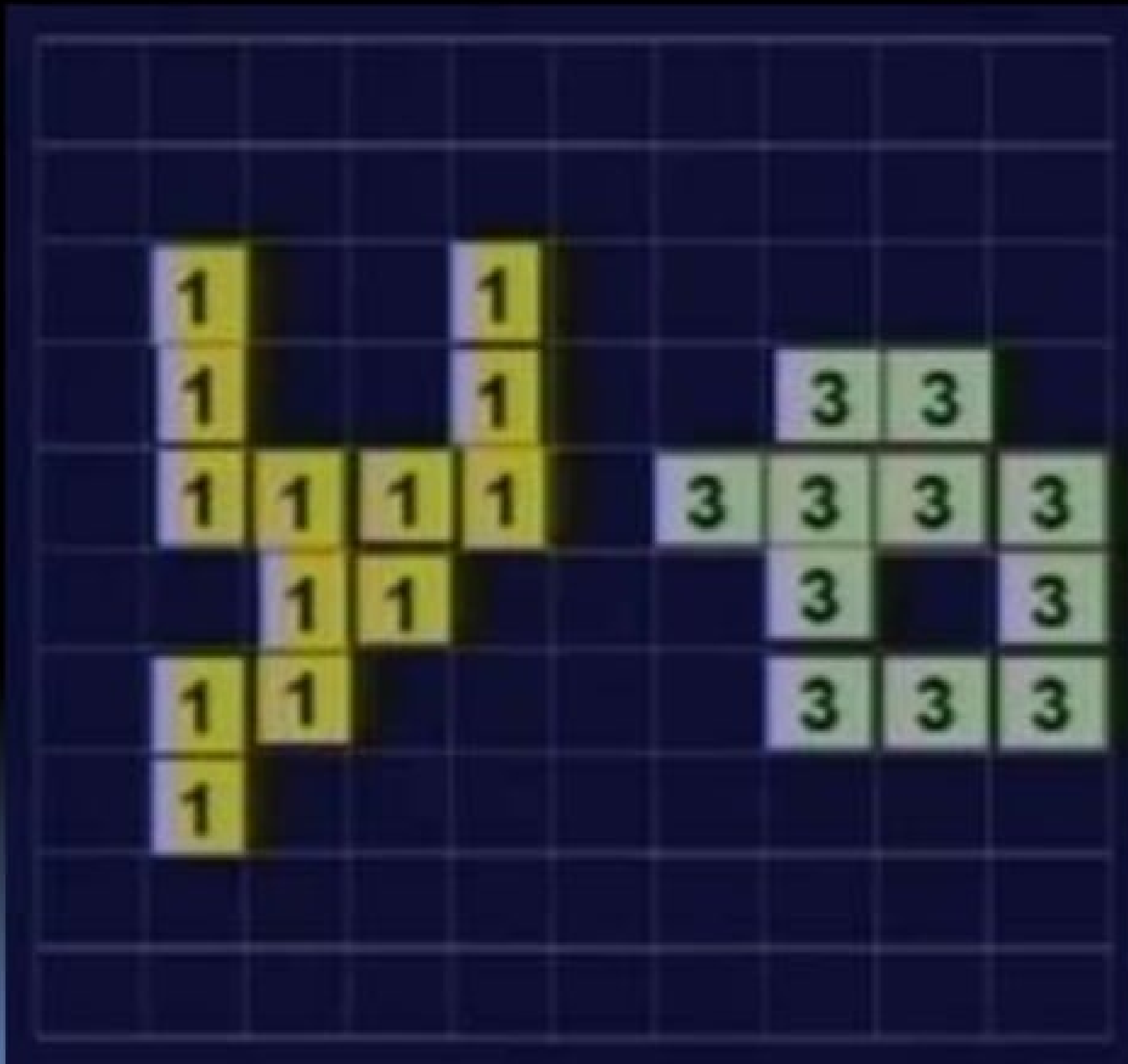
1 2
1 2 3 3
1 1 1 1 4 4 4 4
1 1 4 4


1 2
1 2 3 3
1 1 1 1 4 4 4 4
1 1 4 4

(1=5)




1 2
1 2 3 3
1 1 1 1 4 4 4 4
1 1 4 4
5 1 4 4 4
5


1 1
1 1 3 3
1 1 1 1 3 3 3 3
1 1 3 3
1 1 3 3 3
1

Distance Measures

<u>*Distance Measures:*</u> Distance between pixels p, q & z with co-ordinates (x, y), (s, t) & (v, w) resp. is given by:

a) $D(p, q) \geq 0$ [$D(p, q) = 0$ if $p = q$] ....called reflexivity
b) $D(p, q) = D(q, p)$ ....called symmetry
c) $D(p, z) \leq D(p, q) + D(q, z)$ ....called transmitivity

Euclidean distance between p & q is defined as-

$$D_e(p, q) = [(x - s)^2 + (y - t)^2]^{1/2}$$

Distance Measures

City Block Distance: The D_4 distance between p & q is defined as

$$D_4(p, q) = |x - s| + |y - t|$$

In this case, pixels having D_4 distance from (x, y) less than or equal to some value r form a diamond centered at (x, y).

$$\begin{array}{ccccc} & & 2 & & \\ & 2 & 1 & 2 & \\ 2 & 1 & 0 & 1 & 2 \\ & 2 & 1 & 2 & \\ & & 2 & & \end{array}$$

Pixels with D_4 distance $\leq$ 2 forms the following contour of constant distance.

Distance Measures

<u>Chess-Board Distance</u>: The D_8 distance between p & q is defined as

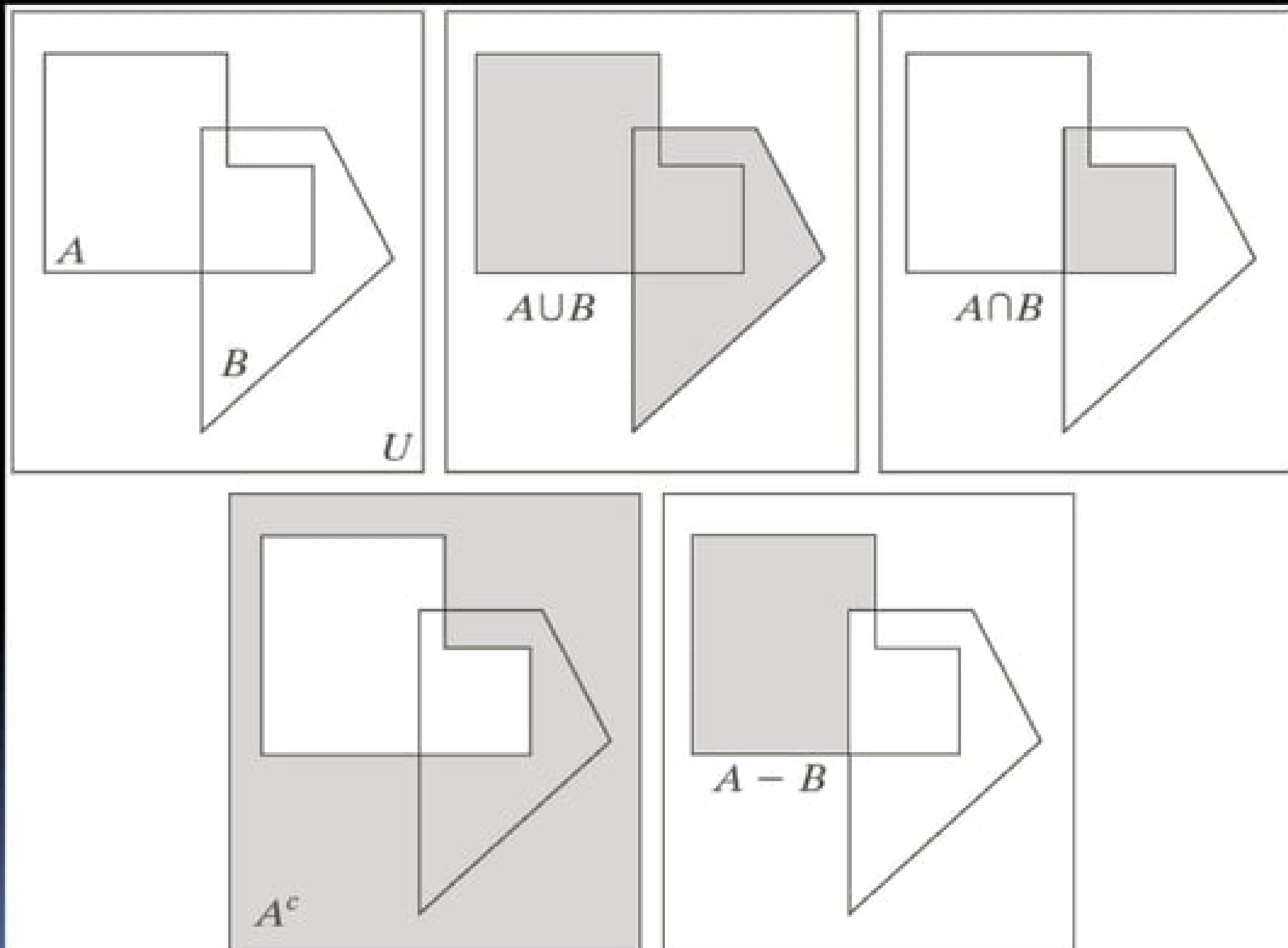
$$D_8(p, q) = \max(|x - s|, |y - t|)$$

In this case, pixels having D_8 distance from (x, y) less than or equal to some value r form a square centered at (x, y).

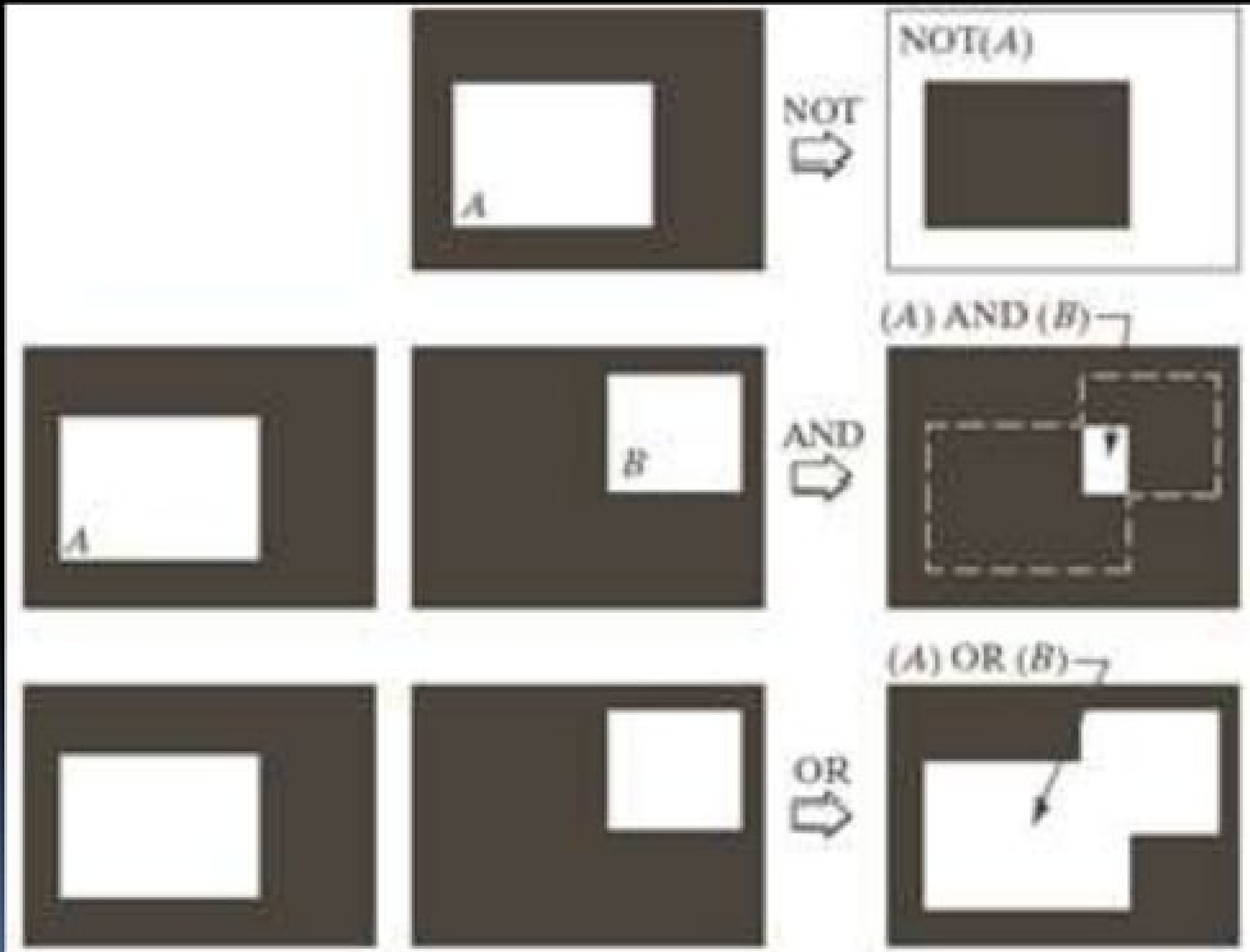
2	2	2	2	2
2	1	1	1	2
2	1	0	1	2
2	1	1	1	2
2	2	2	2	2

Pixels with D_8 distance $\leq$ 2 forms the following contour of constant distance.

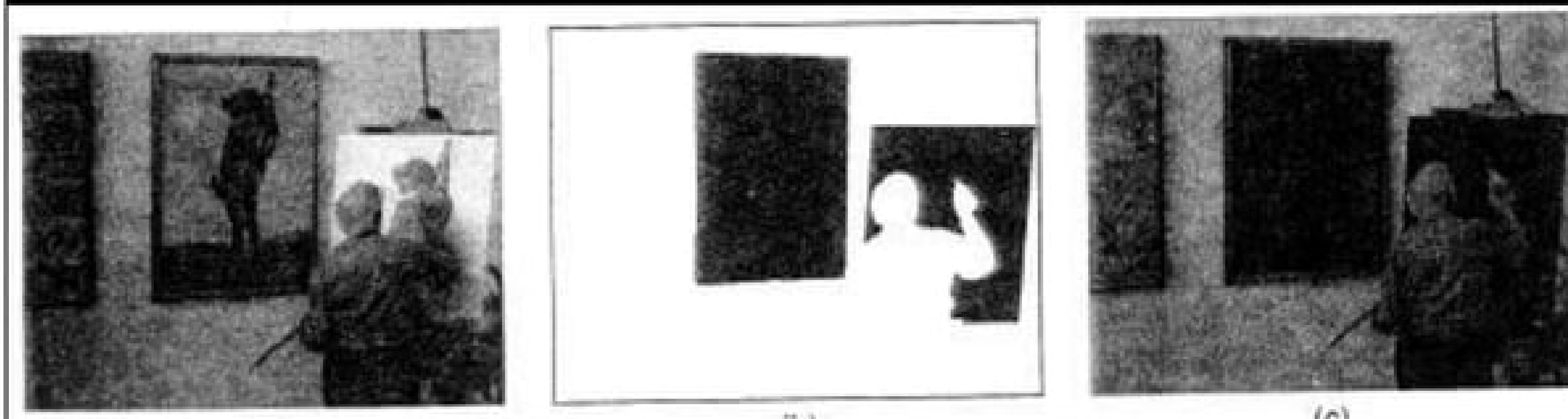
Set operations



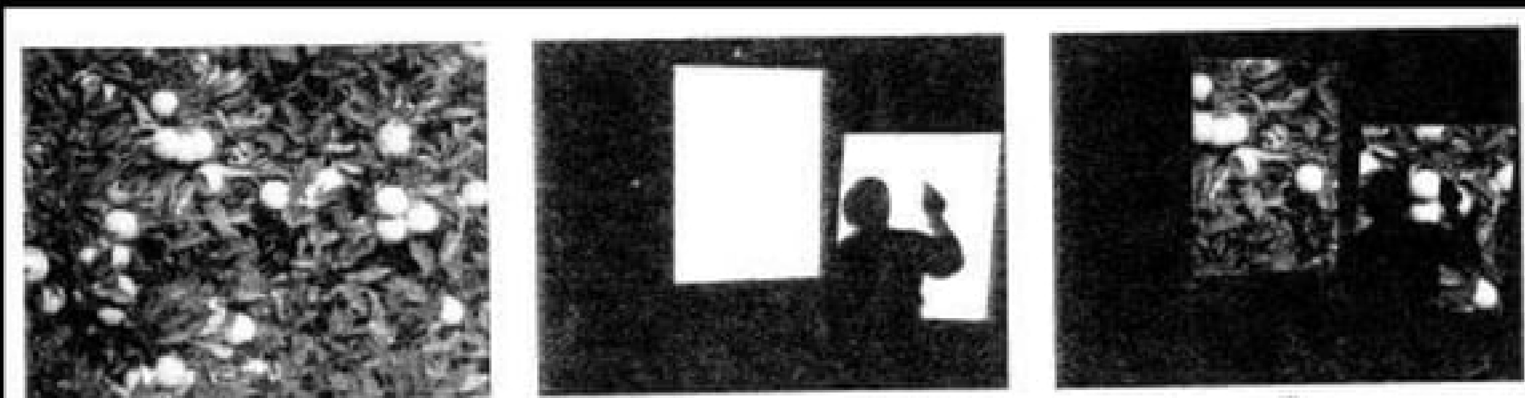
Logical operations



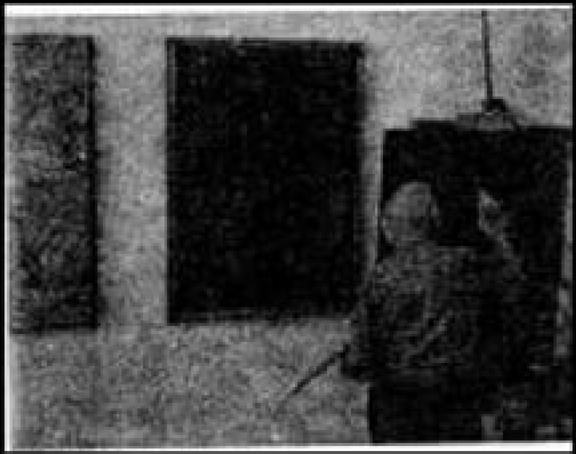
- The AND operator is usually used to mask out part of an image.



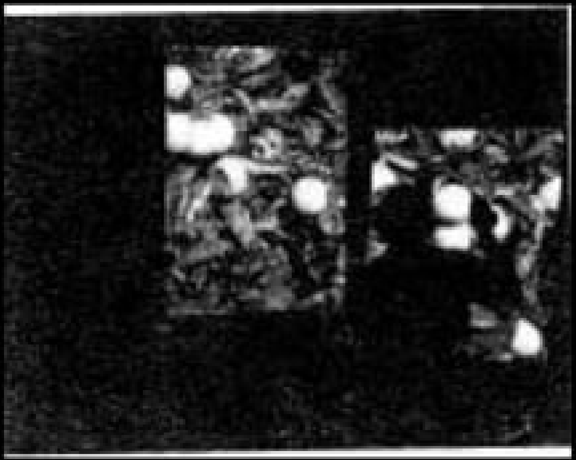
- Parts of another image can be added with a logical OR operator.



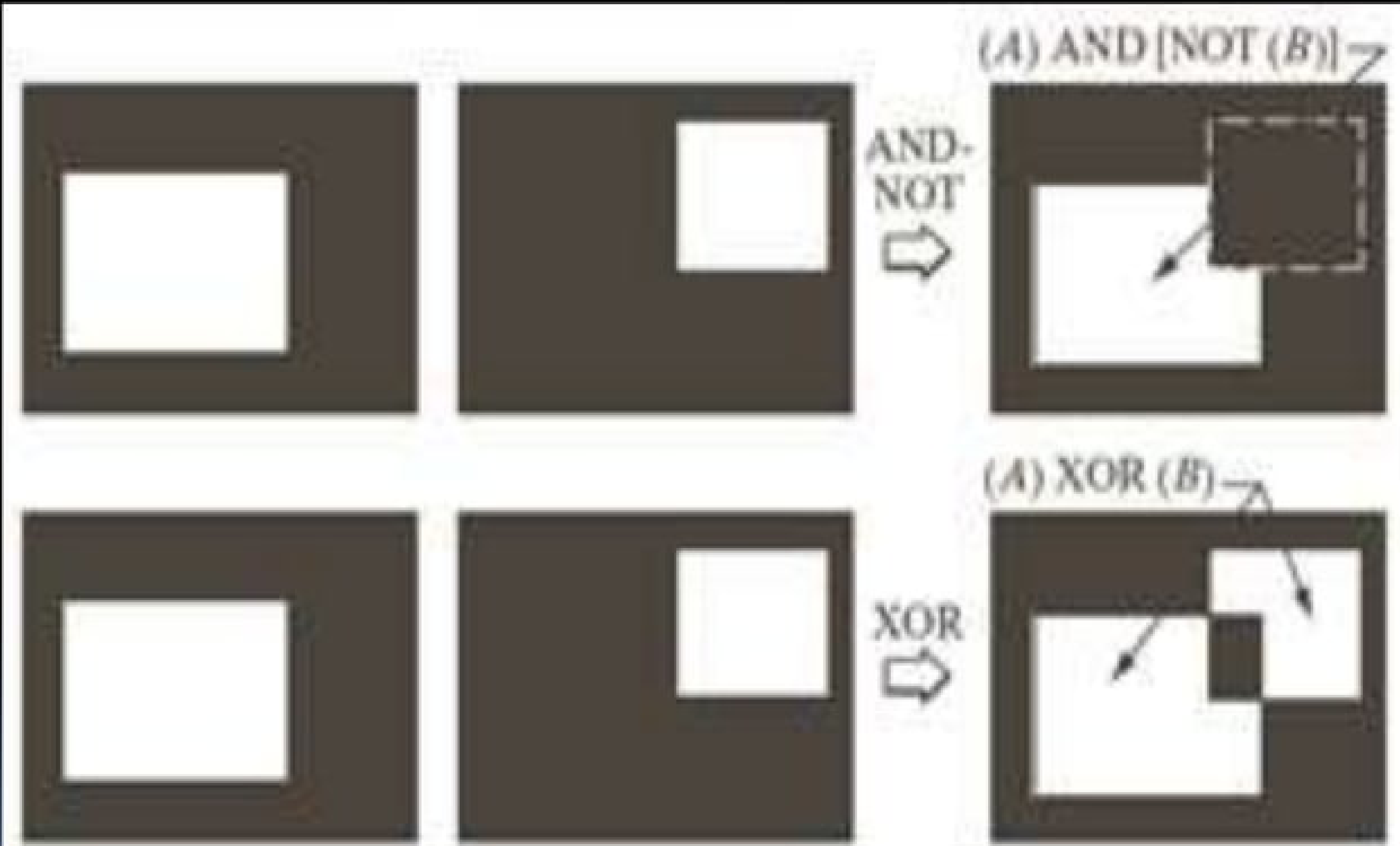
Result of AND



Result of OR



OR


(A) AND [NOT (B)]
AND-
NOT
(A) XOR (B)
XOR