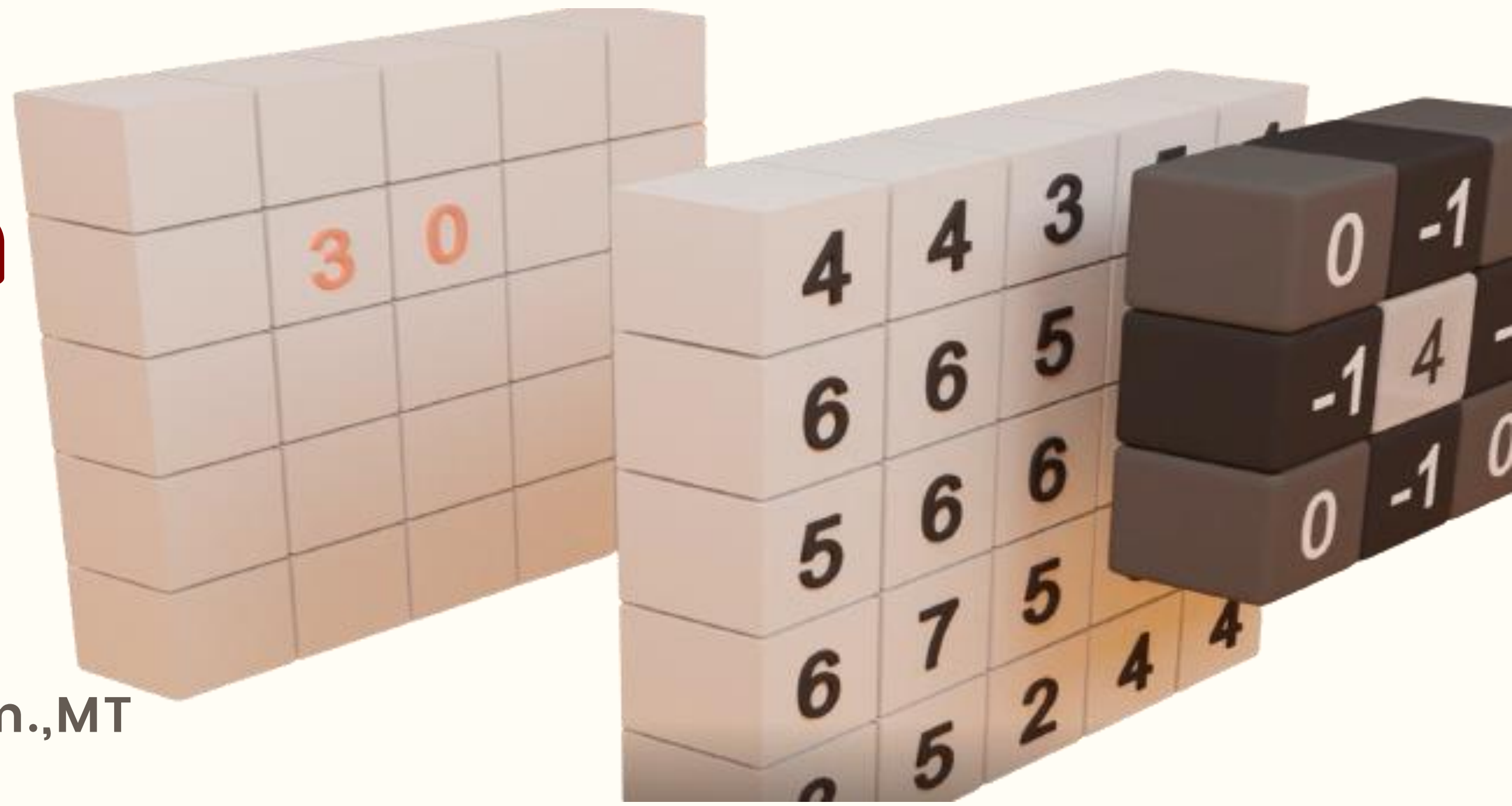


# Cara Kerja Konvolusi

Irma Amelia Dewi.,S.Kom.,MT



# Overview

Filter Kernel Konvolusi

Persamaan Konvolusi

Cara kerja konvolusi

Proses Perhitungan

Contoh Hasil Implementasi Konvolusi



- Dalam proses pengolahan citra digital menggunakan operasi konvolusi 2 dimensi
- Proses operasi konvolusi membutuhkan 2 buah fungsi  $I(u,v)$  adalah citra asli dan  $H(i,j)$  adalah kernel/mask/kernel filter
- Setiap elemen matriks disebut koefisien konvolusi.
- Hasil konvolusi sangat bergantung dari nilai koefisien kernel filter
- Kernel konvolusi dinyatakan dalam bentuk matriks (umumnya  $3 \times 3$ , namun ada juga yang berukuran  $2 \times 2$  atau  $2 \times 1$  atau  $1 \times 2$ ).
- Ukuran matriks ini biasanya lebih kecil dari ukuran citra.

$(0,0)$  = Hot Spot/Titik Origin

$H =$

$i-2,j-2$	$i-2,j-1$	$i-2,j$	$i-2,j+1$	$i-2,j+2$
$i-1,j-2$	$i-1,j-1$	$i-1,j$	$i-1,j+1$	$i-1,j+2$
$(i,j-2)$	$(i,j-1)$	$(i,j)$	$(i,j+1)$	$(i,j+2)$
$i+1,j-2$	$i+1,j-1$	$(i+1,j)$	$i+1,j+1$	$i+1,j+2$
$i+2,j-2$	$i+2,j-1$	$(i+2,j)$	$i+2,j+1$	$i+2,j+2$

# Kernel Filter

Koordinat pada kernel filter  
konvolusi



- Untuk fungsi diskrit, konvolusi didefinisikan sebagai berikut:

$$I'(u, v) = \sum_{i=-\infty}^{i=\infty} \sum_{j=-\infty}^{j=\infty} I(u + i, v + j) \cdot H(i, j)$$

## Persamaan Konvolusi 2D

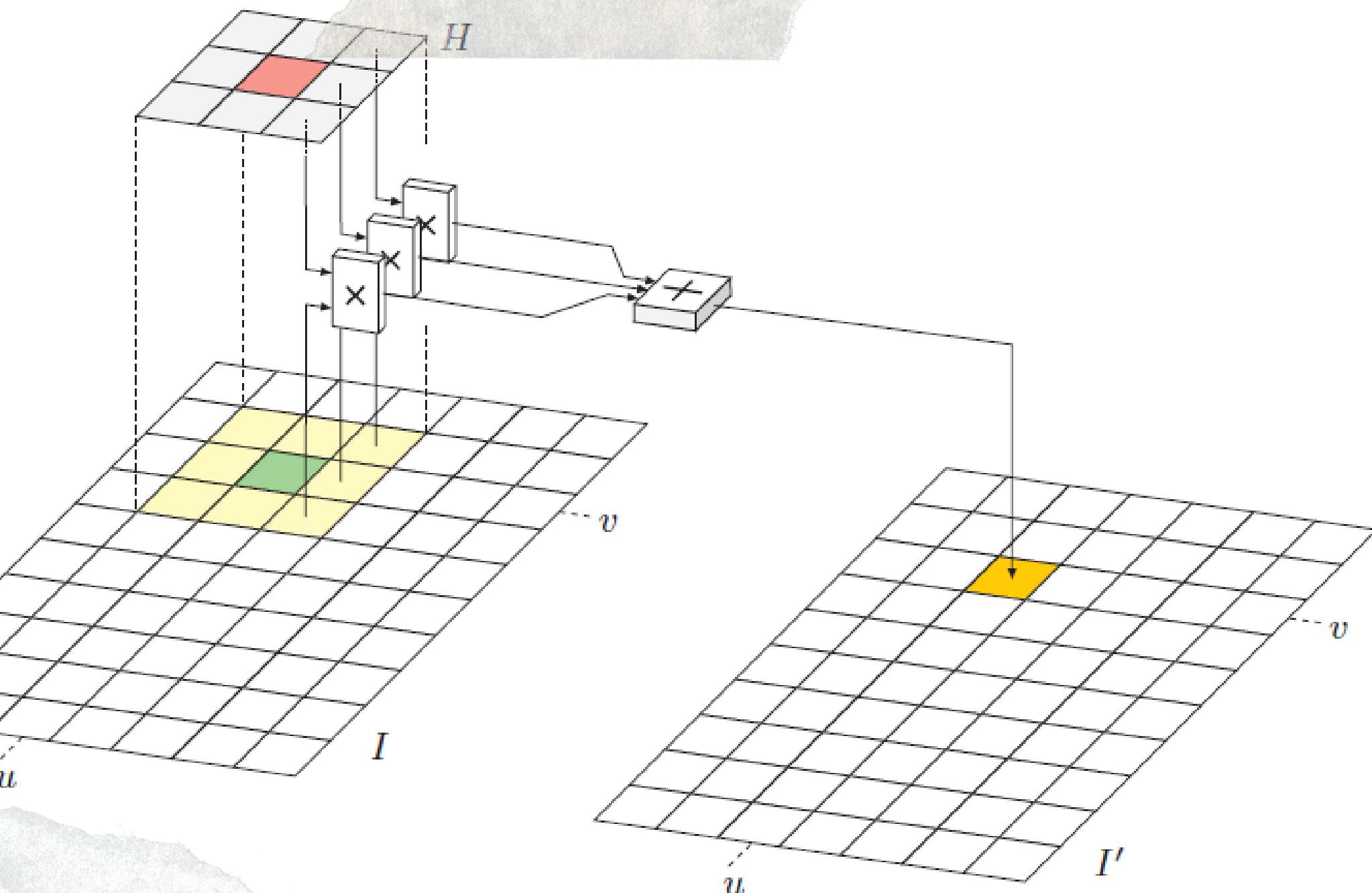
keterangan:

$I'(u, v)$  = Nilai hasil konvolusi di koordinat  $u, v$

$(i, j)$  = koordinat koefisien kernel

$I(u, v)$  = Nilai pixel dari citra sumber

$H(i, j)$  = nilai koefisien kernel di koordinat  $i, j$



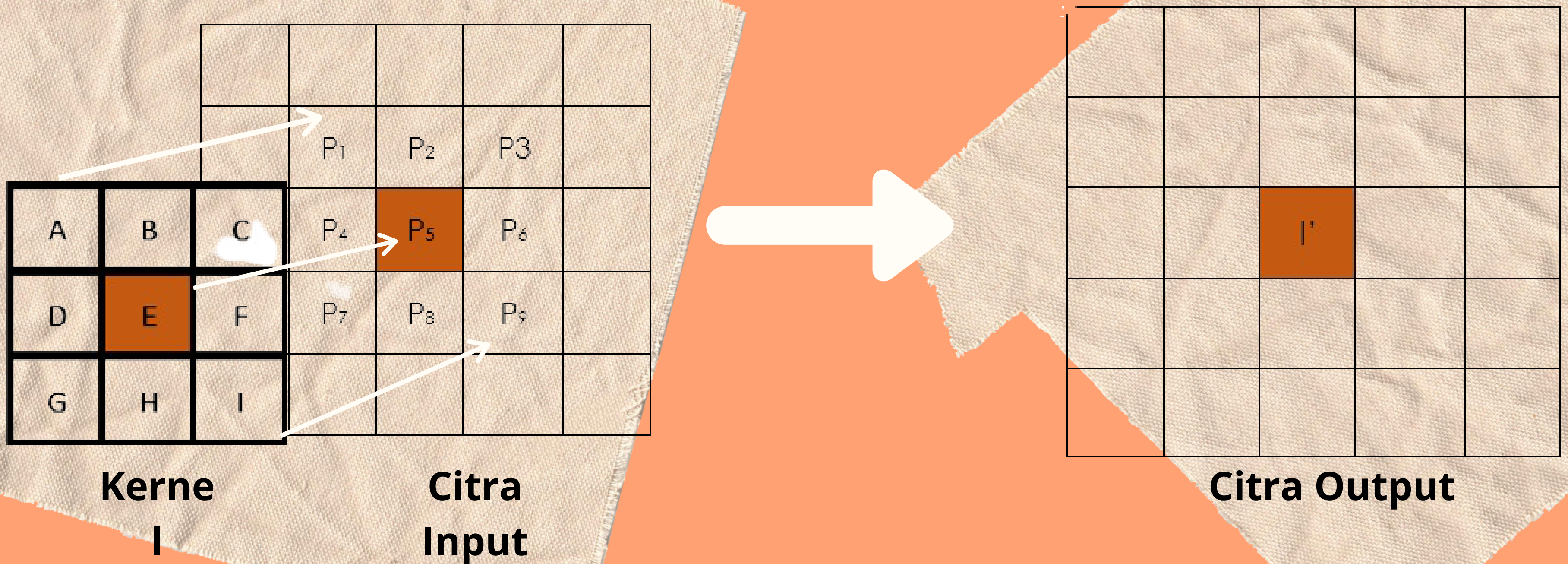
- Operasi konvolusi dilakukan dengan menggeser kernel konvolusi pixel per pixel.
- Hasil konvolusi disimpan di koordinat  $(u, v)$  yang ditandai dengan titik origin dalam matriks yang baru.

## Cara Kerja Konvolusi

Kernel  $H(i, j)$  merupakan suatu jendela yang dioperasikan secara bergeser pada citra masukan  $I(u, v)$ , yang dalam hal ini, jumlah perkalian kedua nilai pada setiap titik merupakan hasil konvolusi yang dinyatakan dengan keluaran  $I'(u, v)$ .



# Proses Konvolusi



$$I'(u,v) = (P1 * A) + (P2 * B) + (P3 * C) + (P4 * D) + (P5 * E) + (P6 * F) + (P7 * G) + (P8 * H) + (P9 * I)$$

Konvolusi diawali dari koordinat pojok kiri atas, hasil konvolusi disimpan pada koordinat yang bersesuaian dengan titik origin kernel



# Proses Perhitungan Konvolusi

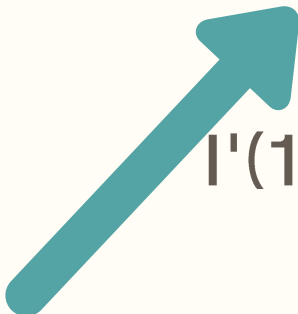
Misalkan diketahui citra input dengan ukuran 5 x 5, dikonvolusikan dengan kernel berukuran 3 x 3

$I(u,v) =$

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

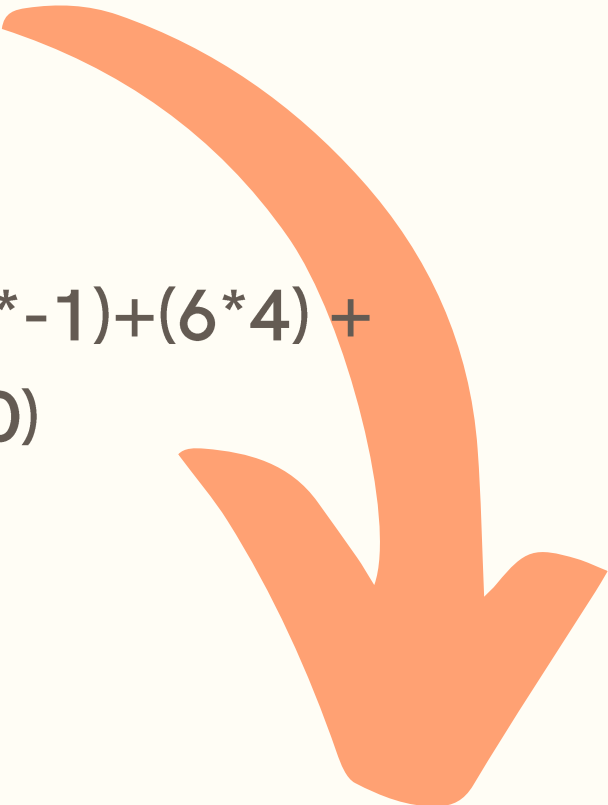
$H(i,j) =$

0	-1	0
-1	● 4	-1
0	-1	0



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

$I'(1,1) = (4 * 0) + (4 * -1) + (3 * 0) + (6 * -1) + (6 * 4) +$   
 $(5 * -1) + (5 * 0) + (6 * -1) + (6 * 0)$   
 $= 3$



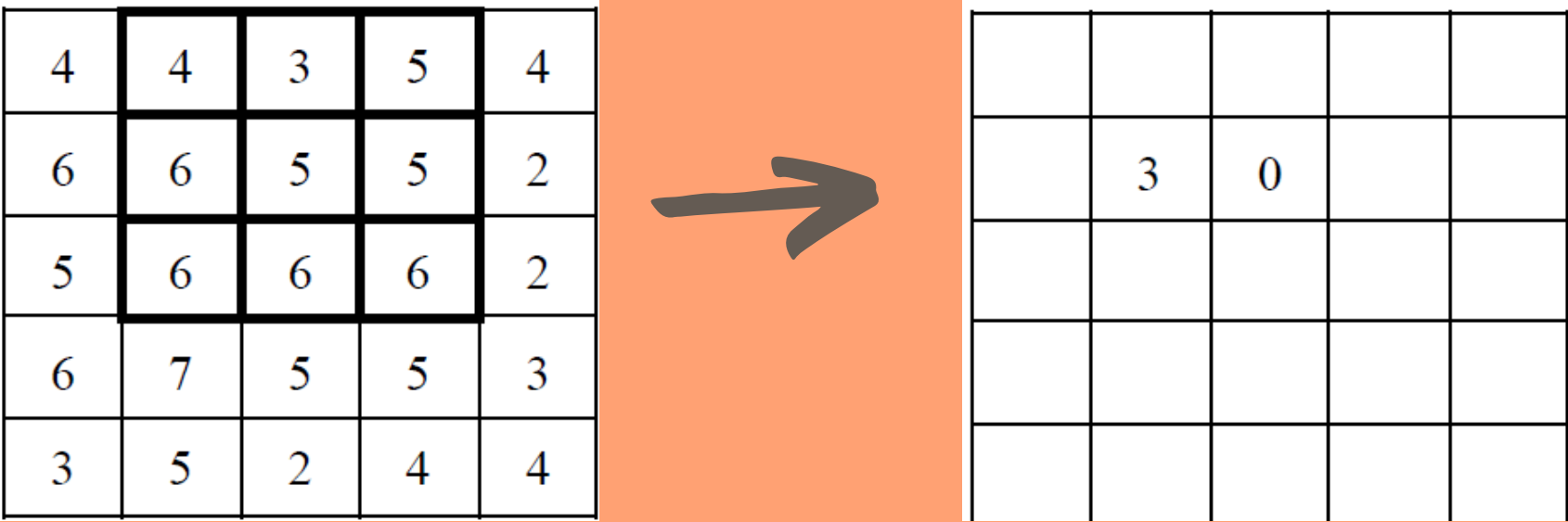
	3			

Citra hasil konvolusi

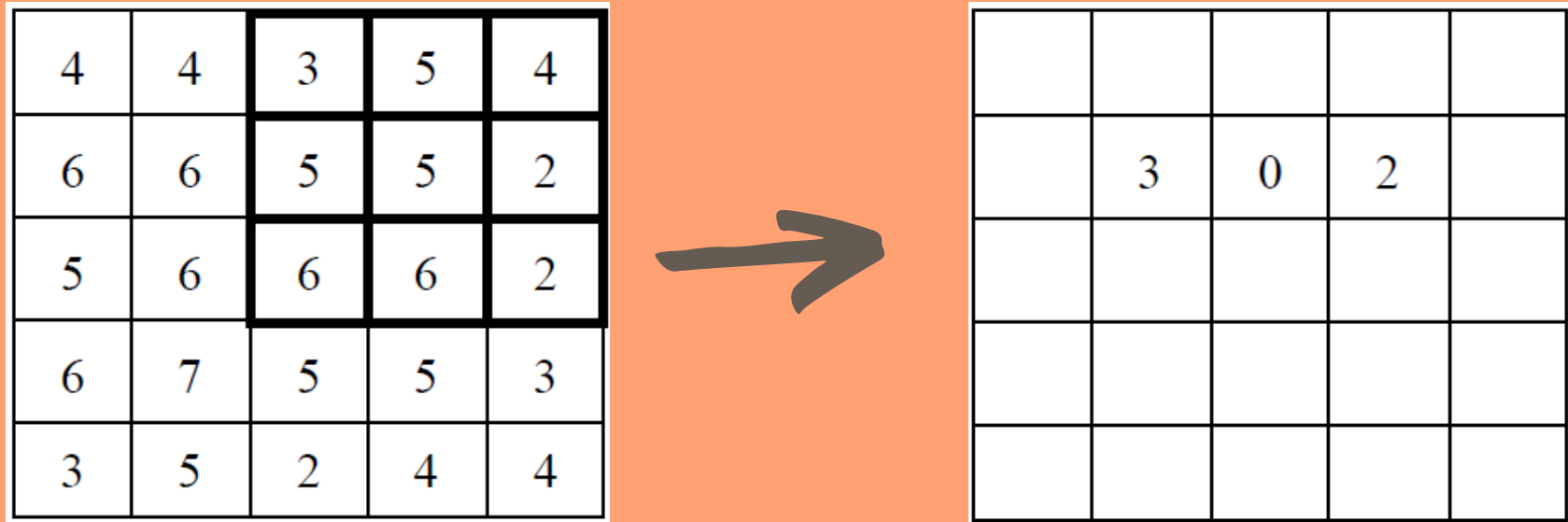
Titik hitam pada kernel menandakan titik  
origin.  
Tempatkan kernel pada sudut kiri atas, kemudian  
hitung nilai pixel pada posisi (0, 0) dari citra input



Geser kernel satu pixel ke kanan, kemudian hitung nilai pixel pada posisi (0,1) dari citra input:



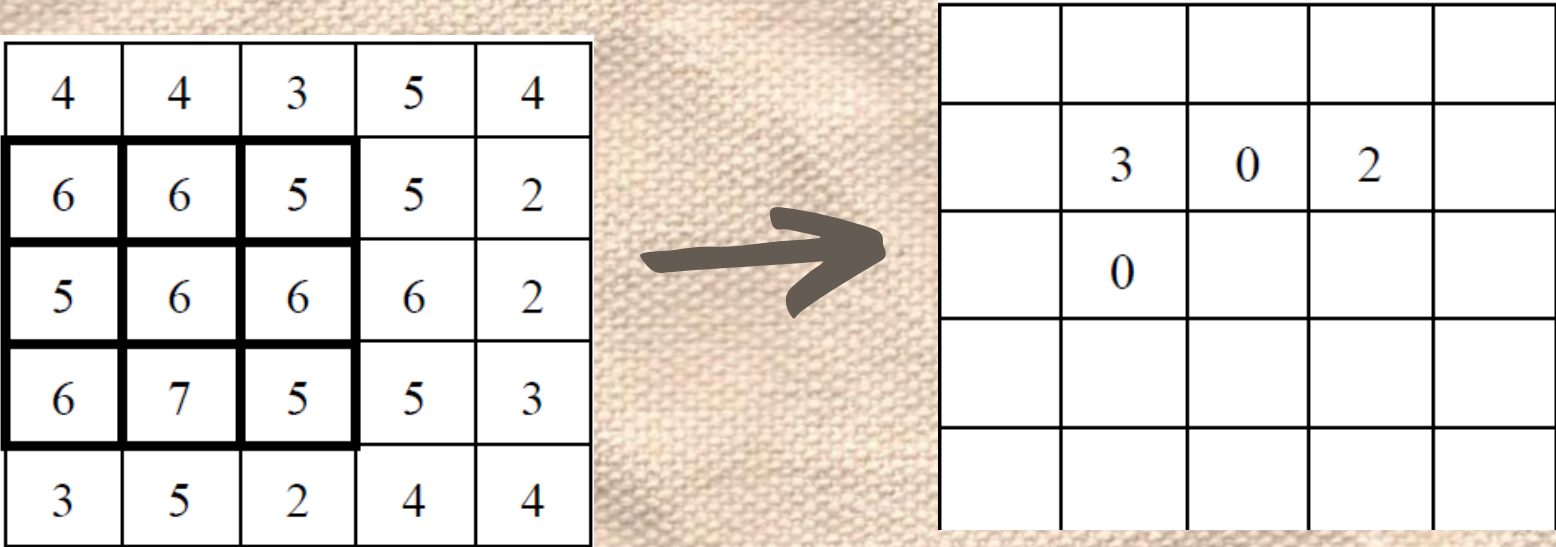
$$I'(1,2) = (4 * 0) + (3 * -1) + (5 * 0) + (6 * -1) + (5 * 4) + (5 * -1) + (6 * 0) + (6 * -1) + (6 * 0) = 0$$



$$I'(1,3) = (3 * 0) + (5 * -1) + (4 * 0) + (5 * -1) + (5 * 4) + (2 * -1) + (6 * 0) + (6 * -1) + (2 * 0) = 2$$

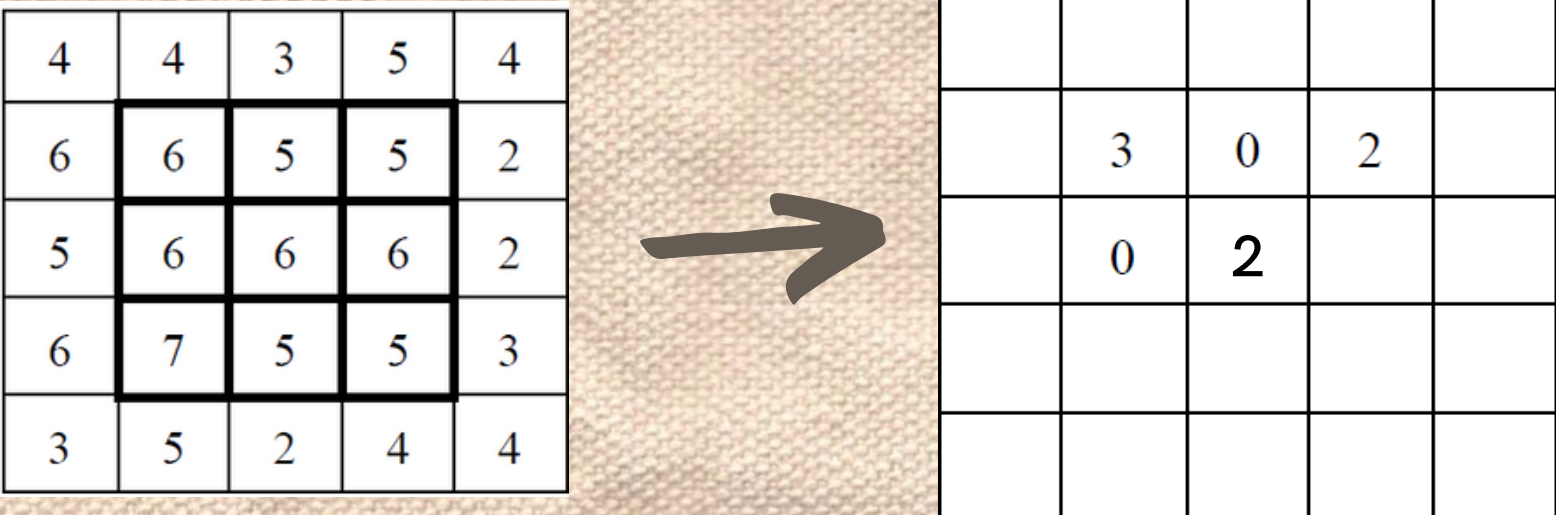
Geser kernel satu pixel ke kanan, kemudian hitung nilai pixel pada posisi (0,2) dari citra input:

Selanjutnya, geser kernel satu pixel ke bawah, lalu mulai lagi melakukan konvolusi dari sisi kiri citra. Setiap kali konvolusi, geser kernel satu pixel ke kanan:



$$I'(2,1) = (6 * 0) + (6 * -1) + (5 * 0) + (5 * -1) + (6 * 4) + (6 * -1) + (6 * 0) + (7 * -1) + (5 * 0) = 0$$

proses selanjutnya dilakukan seperti tahap sebelumnya



$$I'(2,2) = (6 * 0) + (5 * -1) + (5 * 0) + (6 * -1) + (6 * 4) + (6 * -1) + (7 * 0) + (5 * -1) + (5 * 0) = 2$$



Geser kernel satu pixel ke kanan, kemudian hitung nilai pixel pada posisi (1,2) dari citra input, dan disimpan di koordinat (2,3) di matriks citra baru:

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



	3	0	2	
	0	2	6	

$$I'(2,3) = (5 * 0) + (5 * -1) + (2 * 0) + (6 * -1) + (6 * 4) + (2 * -1) + (5 * 0) + (5 * -1) + (3 * 0) = 6$$

Dengan cara yang sama seperti di atas, maka pixel-pixel pada baris ketiga dikonvolusi sehingga menghasilkan

$$I'(3,1) = (5 * 0) + (6 * -1) + (6 * 0) + (6 * -1) + (7 * 4) + (5 * -1) + (3 * 0) + (5 * -1) + (2 * 0) = 3$$

$$I'(3,2) = (6 * 0) + (6 * -1) + (6 * 0) + (7 * -1) + (5 * 4) + (5 * -1) + (5 * 0) + (2 * -1) + (4 * 0) = 0$$

$$I'(3,3) = (6 * 0) + (6 * -1) + (2 * 0) + (5 * -1) + (5 * 4) + (3 * -1) + (2 * 0) + (4 * -1) + (4 * 0) = 2$$

Berikut ini matriks citra hasil proses konvolusi

	3	0	2	
	0	2	6	
	3	0	2	

- jika hasil konvolusi menghasilkan nilai pixel negatif, maka nilai tersebut dijadikan 0,
- jika hasil konvolusi menghasilkan nilai pixel lebih besar dari nilai keabuan maksimum, maka nilai tersebut dijadikan ke nilai keabuan maksimum (operasi clipping).



Jika diperhatikan hasil konvolusi, bagian pinggir (border) matriks pixel citra baru tidak memiliki nilai dikarenakan beberapa koefisien konvolusi tidak dapat diposisikan pada pixel-pixel citra (efek “menggantung”), seperti contoh di bawah ini:

4	4	3	5	4	?
6	6	5	5	2	?
5	6	6	6	2	?
6	7	5	5	3	
3	5	2	4	4	

Terdapat 3 alternatif cara yang dapat dilakukan:

- Pixel-pixel pinggir diabaikan, tidak di-konvolusi. Solusi ini banyak dipakai, maka pixel-pixel pinggir nilainya tetap sama seperti citra asal.
- Duplikasi elemen citra, misalnya elemen kolom pertama disalin ke kolom M+1, begitu juga sebaliknya, lalu konvolusi dapat dilakukan terhadap pixel-pixel pinggir tersebut.
- Elemen yang ditandai dengan “?” diasumsikan bernilai 0 atau konstanta yang lain, sehingga konvolusi pixel-pixel pinggir dapat dilakukan.

Berikut ini matriks citra hasil proses konvolusi menambahkan pixel tepian menggunakan cara 1.

4	4	3	5	4
6	3	0	2	2
5	0	2	6	2
6	3	0	2	3
3	5	2	4	4



# HASIL IMPLEMENTASI KONVOLUSI



Citra input

$$\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Kernel

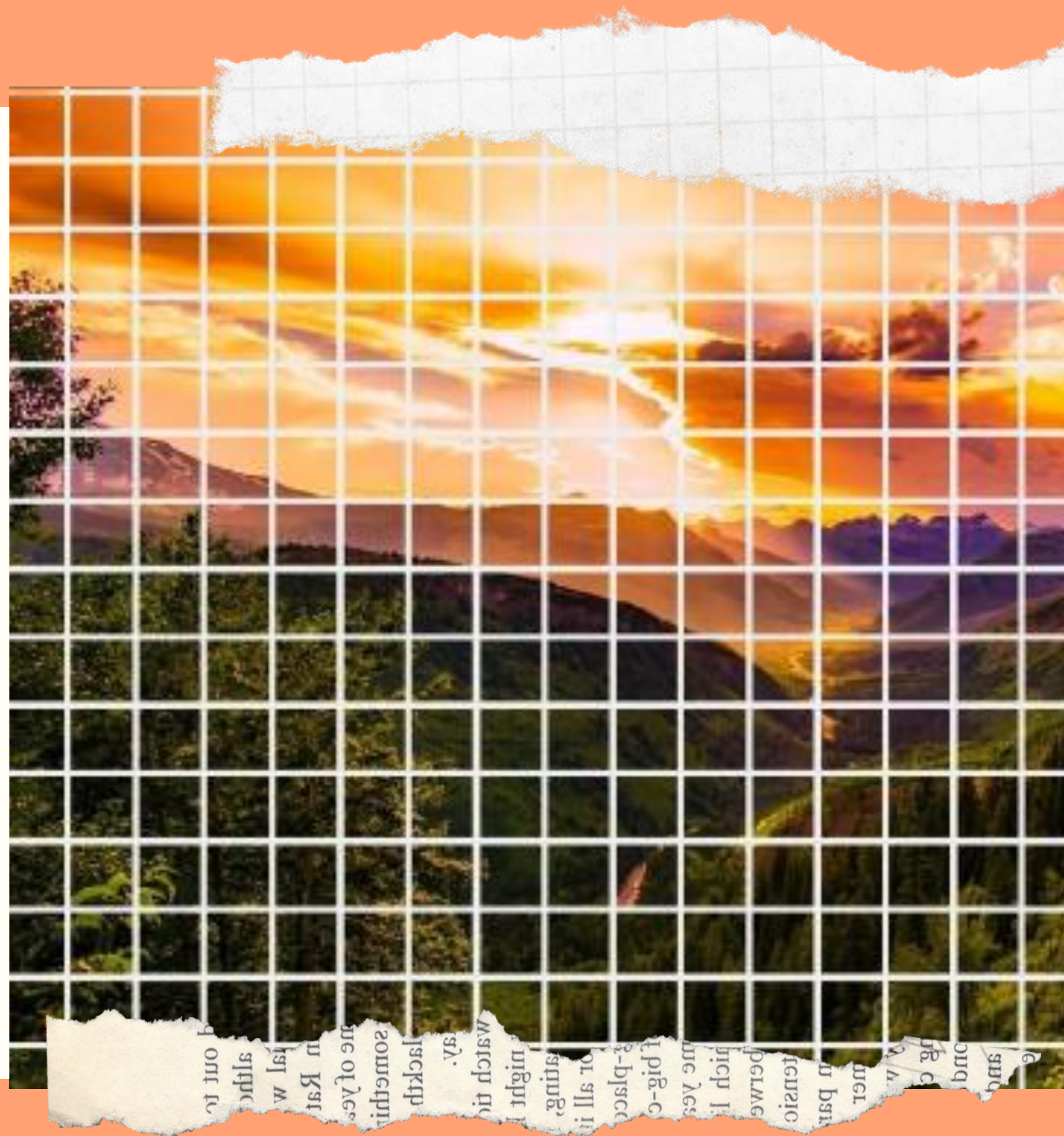


Citra hasil konvolusi





# Quiz





# Referensi

- Gonzales, Rafael C, Digital Image Processing, Addison-Wisley

Ballard,Dana.H, Computer Vision,Prentice Hall,1982

Munir, Rinaldi, Pengolahan Citra Digital,Informatika,2004

- Burger, Wilhelm.Burge,J,Mark. Principle of Digital Image Processing (Fundamental Techniques). Springer, 2009



A top-down view of a workspace. In the center is a silver laptop with a black keyboard. A person's hands are visible; the left hand is on the trackpad and the right hand is on the keyboard. To the left of the laptop is a black smartphone with a brown case. Below the smartphone is a red notebook with a black pen resting on it. To the right of the laptop are two glasses of water. The entire scene is set on a light-colored surface. The image is framed by torn paper edges at the top and bottom, and a textured beige background on the right side.

Thank You