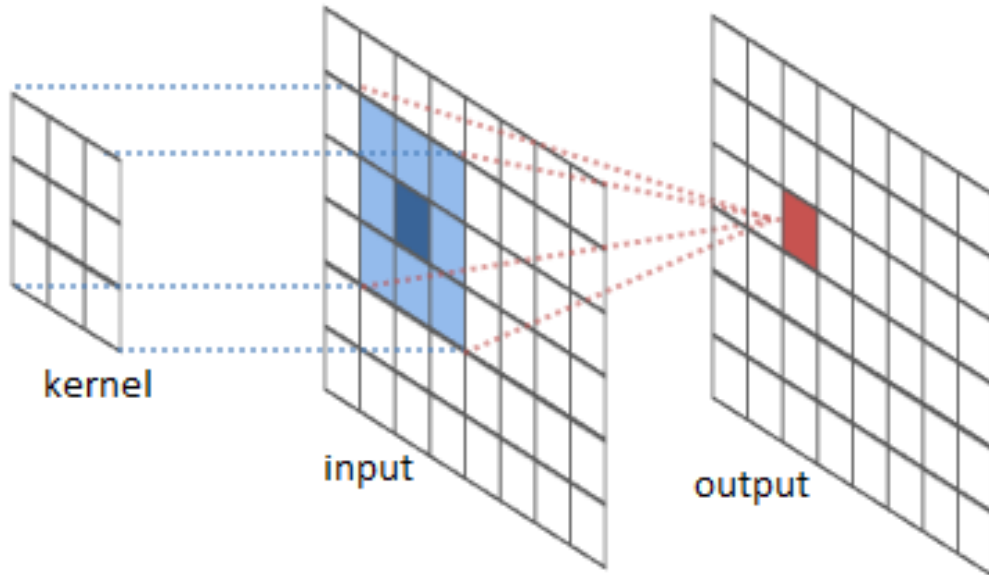


# Practica 3. Filtro de color en paralelo

---

# Convolución



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

**I**

\*

1	0	1
0	1	0
1	0	1

**K**

=

1	4	3	4	1
1	2	4	3	3
1	2	3	4	1
1	3	3	1	1
3	3	1	1	0

**I \* K**

# Convolución



$1/9$	$1/9$	$1/9$
$1/9$	$1/9$	$1/9$
$1/9$	$1/9$	$1/9$

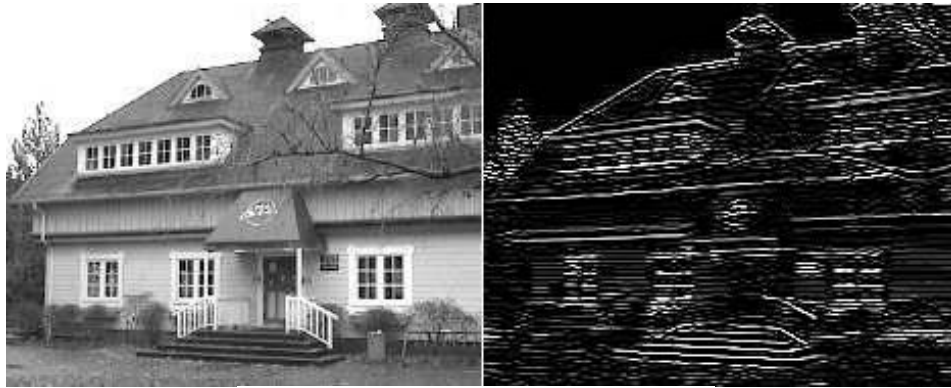
Desenfoque simple



0	0	0	5	0	0	0
0	5	18	32	18	5	0
0	18	64	100	64	18	0
5	32	100	100	100	32	5
0	18	64	100	64	18	0
0	5	18	32	18	5	0
0	0	0	5	0	0	0

Desenfoque Gaussiano

# Convolución



-1	-1	-1
2	2	2
-1	-1	-1

Horizontal lines

-1	2	-1
-1	2	-1
-1	2	-1

Vertical lines

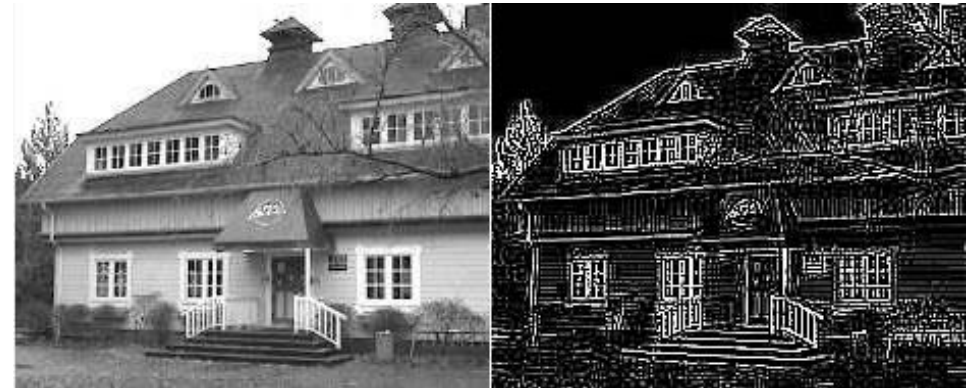
-1	-1	2
-1	2	-1
2	-1	-1

45 degree lines

2	-1	-1
-1	2	-1
-1	-1	2

135 degree lines

Detección de líneas



-1	-1	-1
-1	8	-1
-1	-1	-1

Detección de aristas

# Convolución



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

Horizontal

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

Vertical

Detección de aristas Sobel

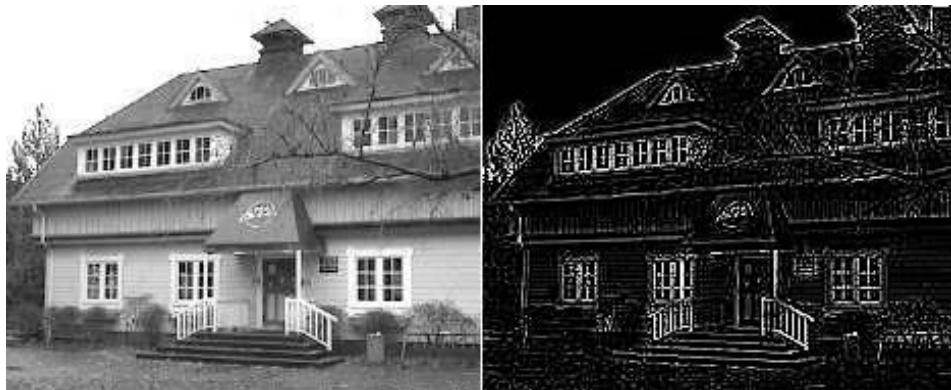


-1	-1	-1
-1	8	-1
-1	-1	-1

Operador laplaciano

# Convolución

---



0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

Laplaciano-Gaussiano

# Práctica

---

- Crear filtro en CPU
- Reservar y copiar memoria en GPU
- Convolución (GPU)
  - Separar imagen en RGB
  - Aplicar filtro
  - Volver a juntar la imagen
- Parte 1: ejecutar en una sola GPU
- Parte 2: distribuir el sistema utilizando MPI aplicando el esquema maestro-esclavo