

# Processamento de imagem

---

Por Iago Nicolas

[iago.nicolas@ufpe.br](mailto:iago.nicolas@ufpe.br)

# O que é o processamento de imagens?

Podemos definir como um método de realizar operações em uma imagem para obter uma versão melhorada de forma a extrair informações úteis a partir dela.

# Por quê processar imagens?

Fazemos para traduzir o que a visão humana captura de forma a permitir que máquinas consigam extrair informações para entender imagens compreensíveis pela mente humana.

# A imagem problema



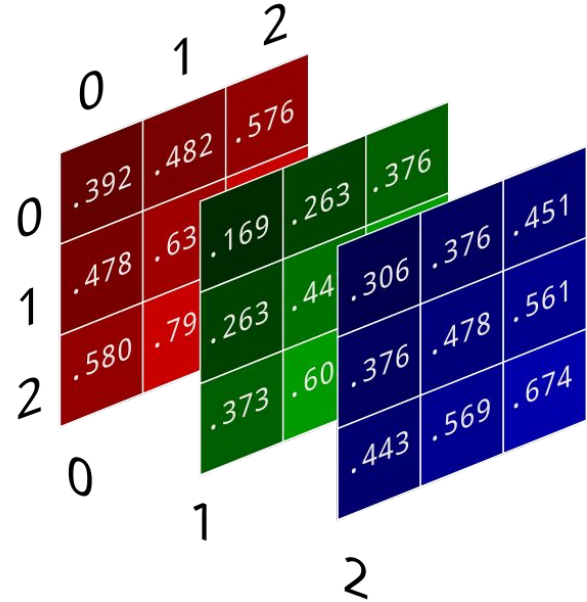
- Lena Söderberg (1972)

**Como o sistema visualiza?**

# Imagem em formato RGB

Entregamos uma imagem RGB que o sistema separa em matrizes 2D.

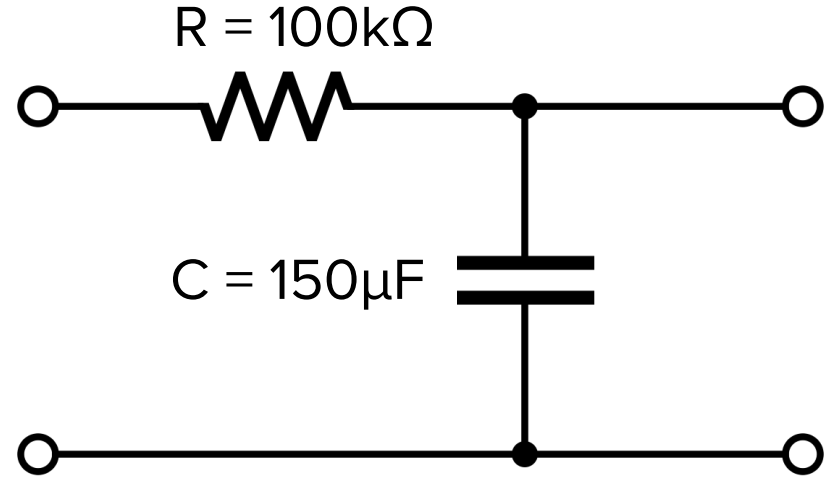
- Para preto e branco temos uma situação análoga.



# Implementação: Passa-baixa

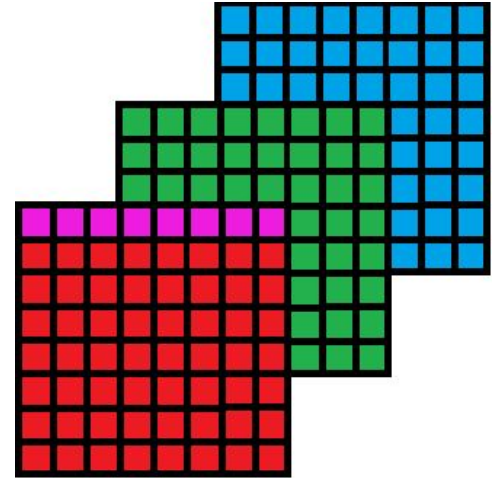
Aplicamos um filtro passa-baixa como o ilustrado ao lado.

- $R = 100\text{k}\Omega$ ;
- $C = 150\mu\text{F}$ .



# Computacionalmente

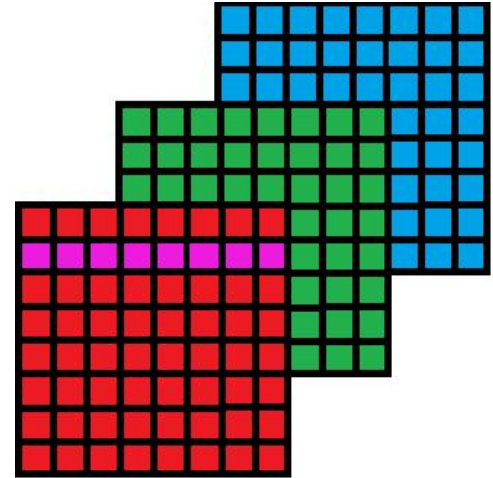
Temos um comportamento  
recursivo por parte do sistema.





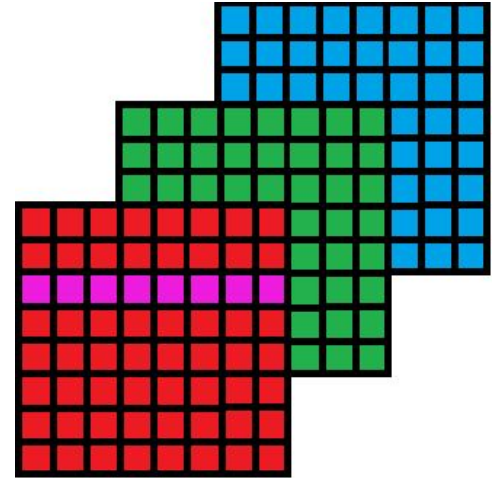
# Computacionalmente

Temos um comportamento  
recursivo por parte do sistema.



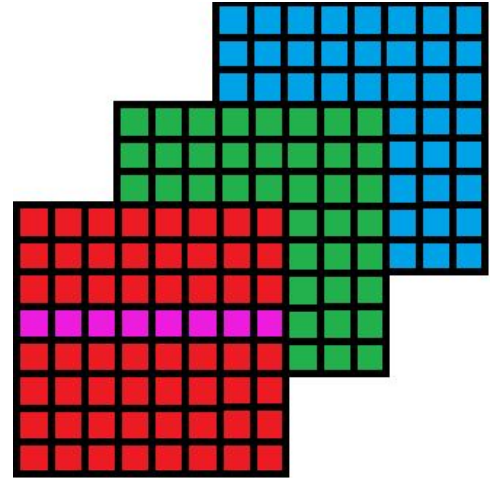
# Computacionalmente

Temos um comportamento  
recursivo por parte do sistema.



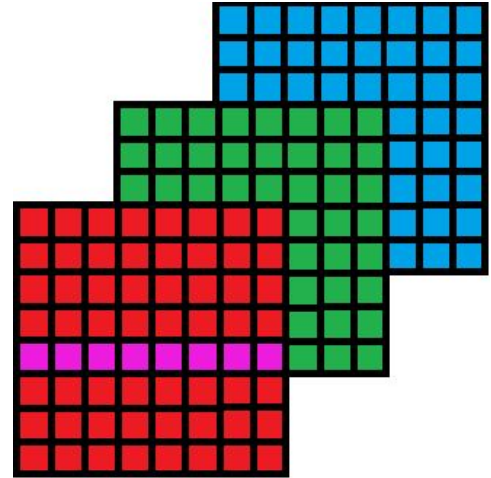
# Computacionalmente

Temos um comportamento  
recursivo por parte do sistema.



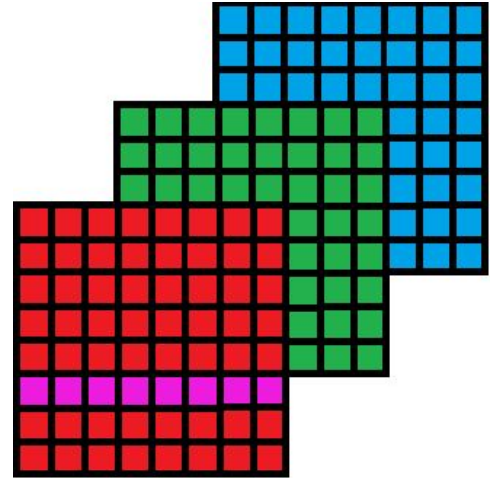
# Computacionalmente

Temos um comportamento  
recursivo por parte do sistema.



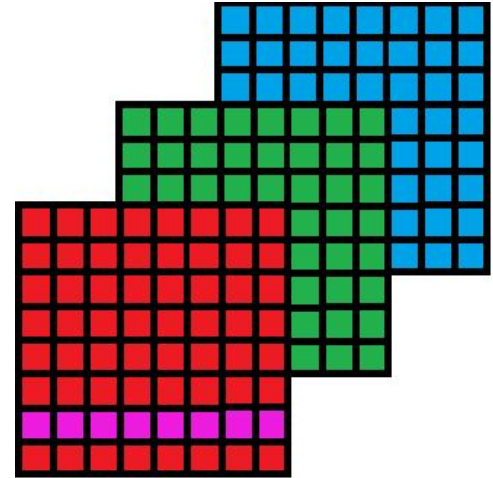
# Computacionalmente

Temos um comportamento  
recursivo por parte do sistema.



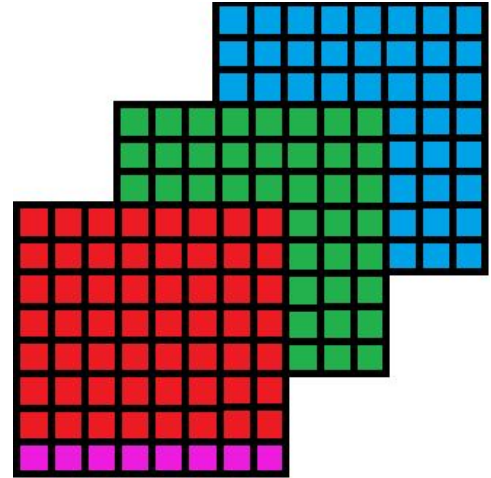
# Computacionalmente

Temos um comportamento  
recursivo por parte do sistema.



# Computacionalmente

Temos um comportamento  
recursivo por parte do sistema.



# Resultado

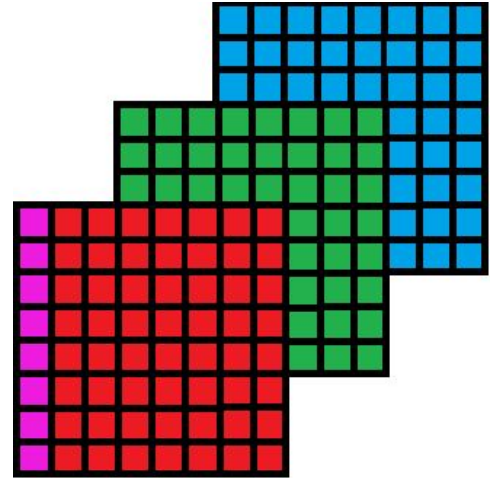


- Filtro passa-baixa no eixo X



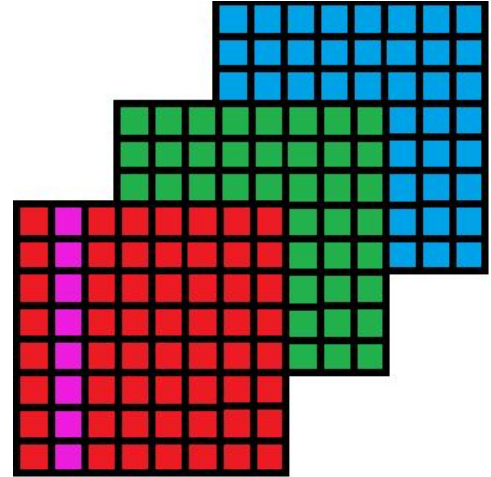
# Computacionalmente

Temos um comportamento  
recursivo por parte do sistema.



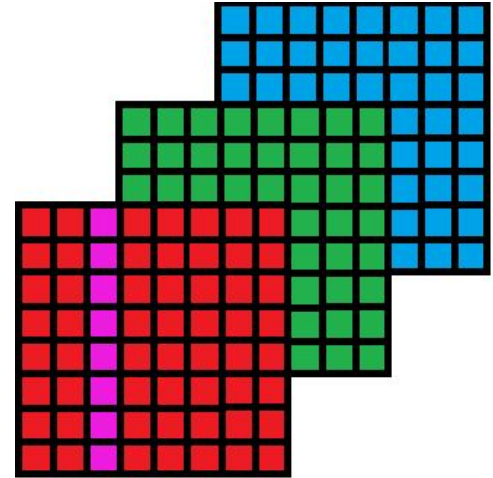
# Computacionalmente

Temos um comportamento  
recursivo por parte do sistema.



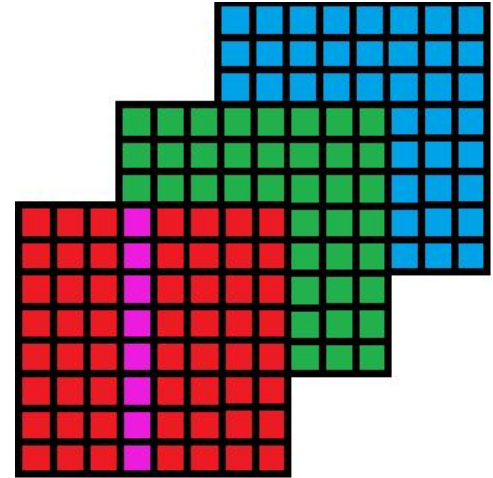
# Computacionalmente

Temos um comportamento  
recursivo por parte do sistema.



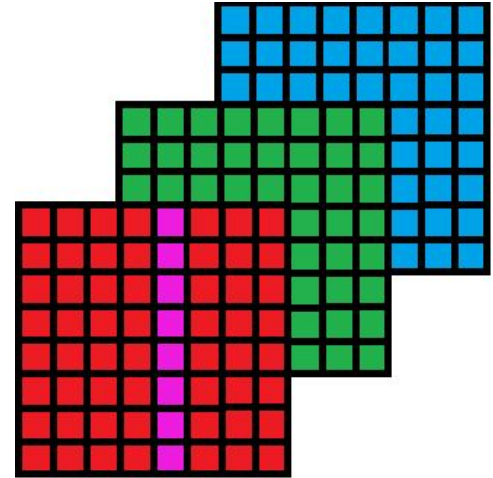
# Computacionalmente

Temos um comportamento  
recursivo por parte do sistema.



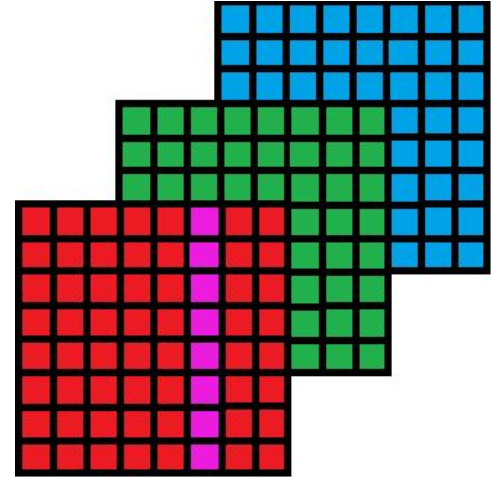
# Computacionalmente

Temos um comportamento  
recursivo por parte do sistema.



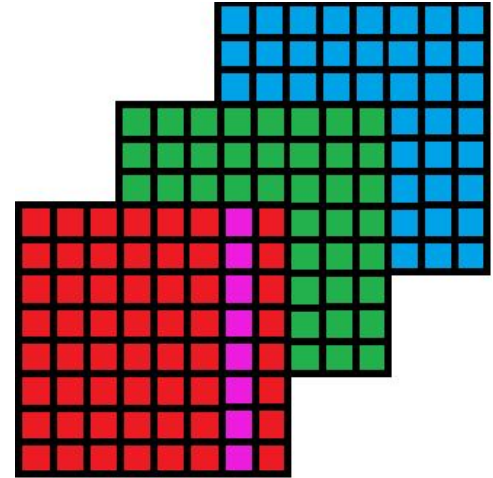
# Computacionalmente

Temos um comportamento  
recursivo por parte do sistema.



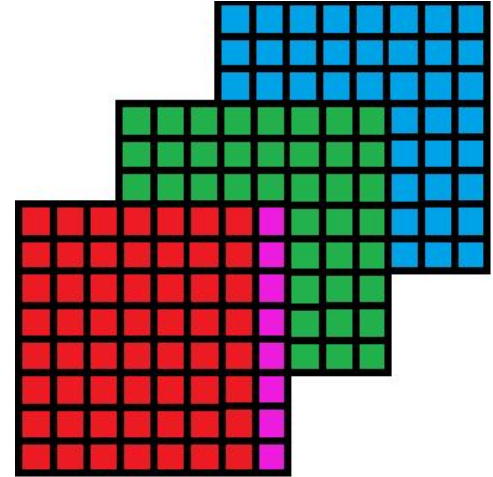
# Computacionalmente

Temos um comportamento  
recursivo por parte do sistema.



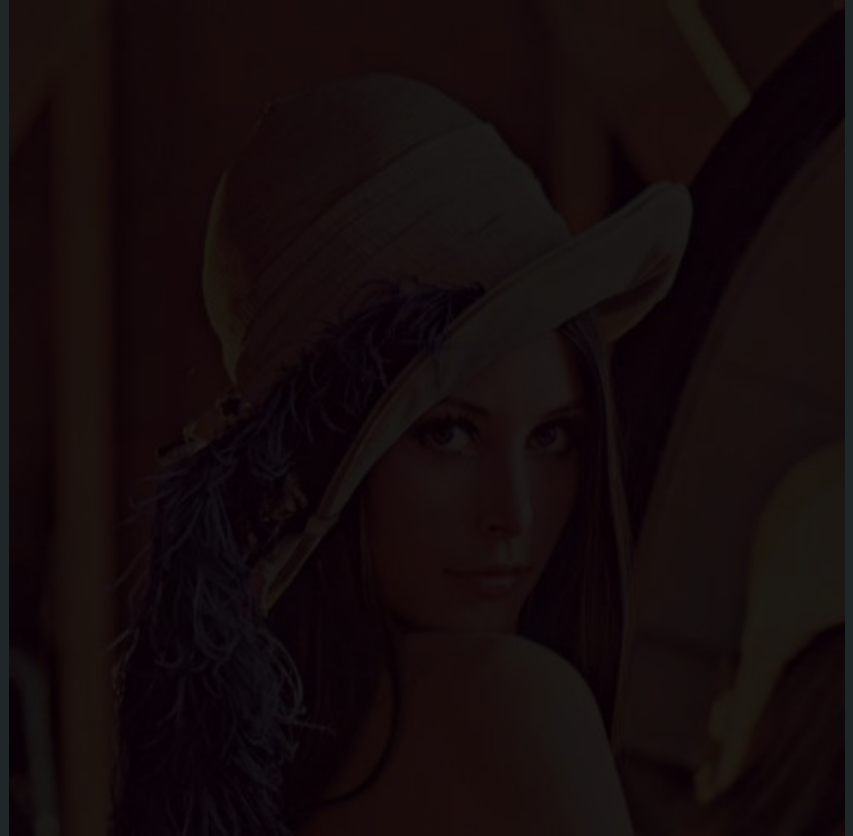
# Computacionalmente

Temos um comportamento  
recursivo por parte do sistema.





# Resultado



- Filtro passa-baixa no eixo Y

# Implementação: Derivada

Aplicamos o mesmo método recursivo apresentado anteriormente (eixo X e depois Y) e a derivada pela definição.

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$

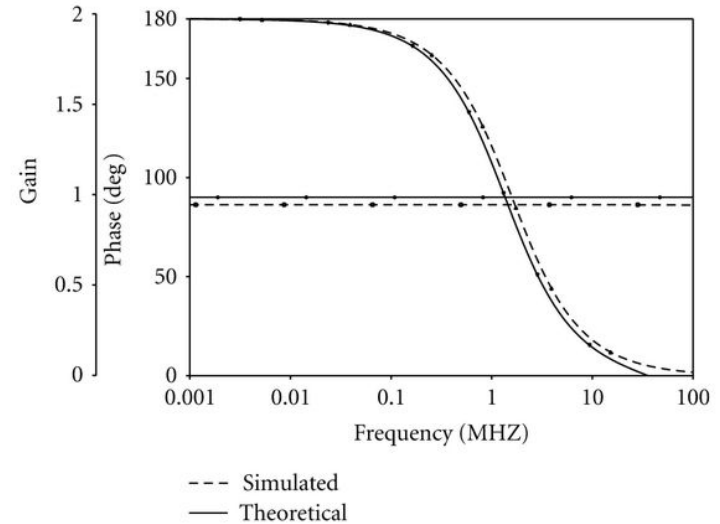
# Resultado



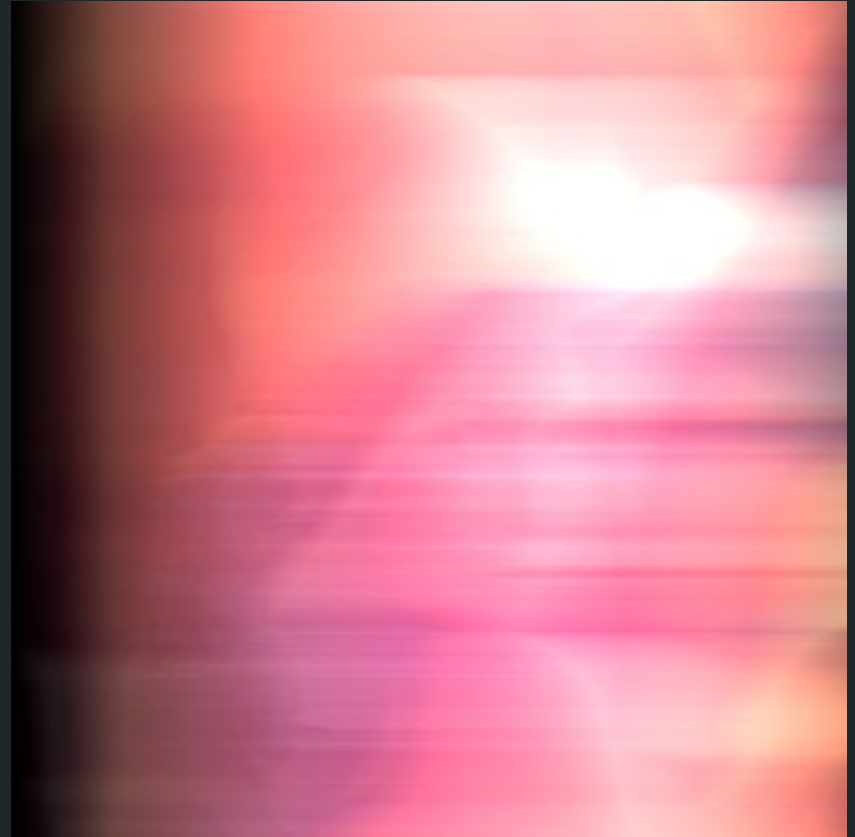
- Filtro derivador.

# Implementação: Passa-todas

Aplicamos o mesmo método recursivo apresentado anteriormente (eixo X e depois Y) e um filtro passa todas com polo em  $1/2$ .

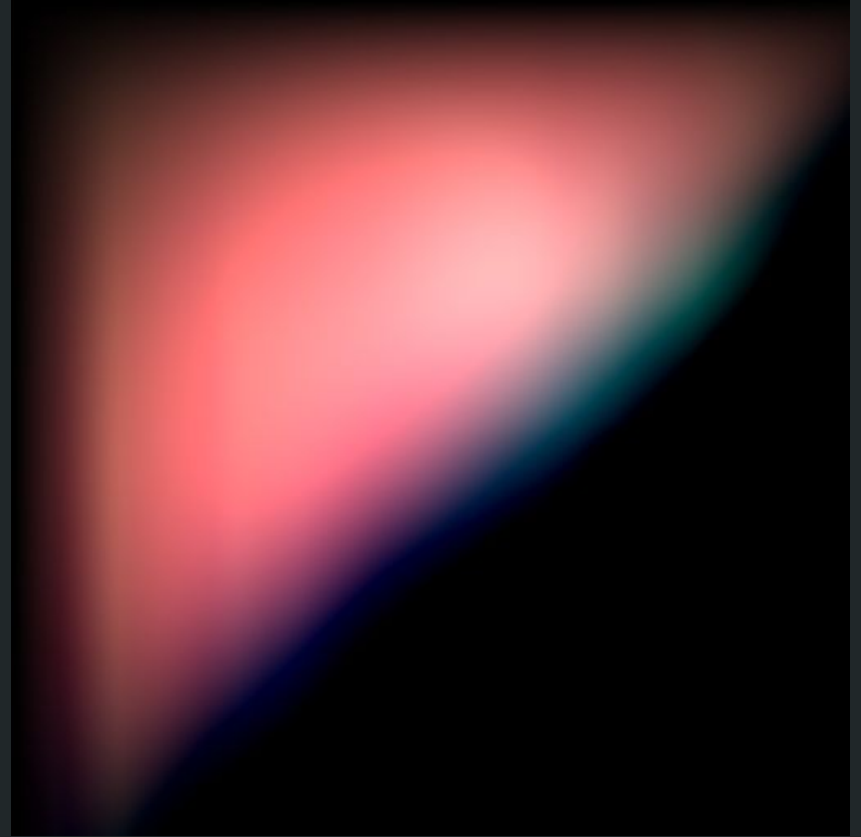


# Resultado



- Filtro passa-todas no eixo X.

# Resultado



- Filtro passa-todas no eixo Y.

# Implementação: Detecção de bordas

Mudamos um pouco o método de aplicação. Utilizamos aqui um kernel gaussiano seguido de um filtro de píxeis fortes/fracos.

7	23	50	64	14
15	13	31	46	8
42	25	92	31	32
71	44	74	94	92
2	43	51	35	4

X

0	2	0
0	0	0
0	0	0

# Resultado



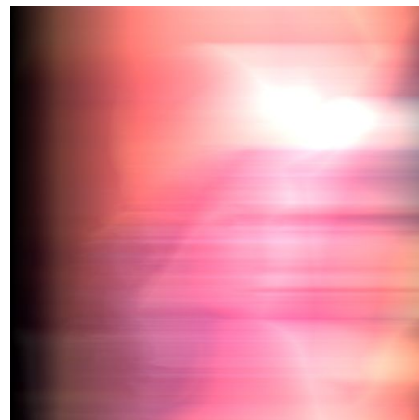
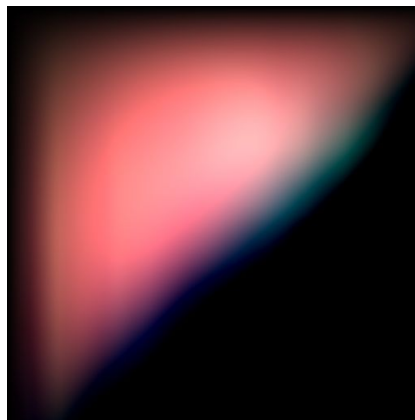
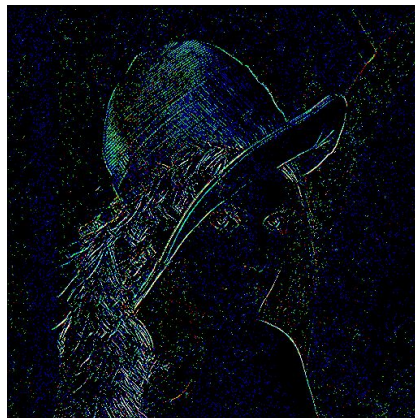
- Detecção de bordas.



Aha!

# Minhas descobertas

1. Aplicação de filtros por convolução x por linhas, colunas;
2. Mudança de fase em imagens;
3. Detecção de bordas por kernel gaussiano.



# O que poderia ser melhorado?

1. Implementação de multithreading, imagens em resoluções acima de 1080p tem tempo de execução absurdo.
2. Design melhor do filtro passa-baixa, de forma a obter resultados melhores no edge detection.

