

Filtro Mediano:

O filtro mediano é uma técnica que permite a redução de ruído em uma imagem. Considerando uma dada vizinhança nbh, o método consiste em colocar em ordem crescente todos os pontos dentro desta vizinhança e indicar o valor mediano para cada pixel da imagem original. Considerando $nbh = 1$, uma imagem de entrada src e uma imagem de retorno dst., conforme figura 1.

S ₁₁	S ₁₂	S ₁₃	S ₁₄
S ₂₁	S ₂₂	S ₂₃	S ₂₄
S ₃₁	S ₃₂	S ₃₃	S ₃₄
S ₄₁	S ₄₂	S ₄₃	S ₄₄

(a)

D ₁₁	D ₁₂	D ₁₃	D ₁₄
D ₂₁	D ₂₂	D ₂₃	D ₂₄
D ₃₁	D ₃₂	D ₃₃	D ₃₄
D ₄₁	D ₄₂	D ₄₃	D ₄₄

(b)

Figura 1. Imagens de (a) entrada e (b) saída do filtro.

A fim de se realizar o ordenamento, todos os pontos da vizinhança indicada que estejam dentro da imagem são avaliados. Uma vez colocados em ordem crescente, o ponto central deste ordenamento é colocado no ponto de saída.

Considerando um exemplo simples, onde a imagem de entrada (src) é apresentada na figura 2(a).

A função cria uma nova imagem (dst), originalmente em branco, do mesmo tamanho da original, conforme figura 2(b). Em seguida, inicia o cálculo os valores dos pixels a partir dos valores da imagem de entrada.

50	125	127	35
45	130	143	44
56	137	139	50
44	153	167	81

D ₁₁	D ₁₂	D ₁₃	D ₁₄
D ₂₁	D ₂₂	D ₂₃	D ₂₄
D ₃₁	D ₃₂	D ₃₃	D ₃₄
D ₄₁	D ₄₂	D ₄₃	D ₄₄

Figura 2. (a) Imagem de entrada no exemplo e (b) imagem de saída, ainda em branco.

Para calcular o valor de D₁₁, deve-se ordenar os quatro pontos da imagem de entrada, uma vez que estes são os únicos pontos da vizinhança que estão dentro da imagem. Uma vez ordenados: 45, 50, 125, 130. A escolha recai sobre o pixel mediano do vetor ordenado, ou seja, 125 (com arredondamento para cima).

$$D_{11} = 125$$

Após a avaliação de D₁₁, a imagem de saída (dst) fica conforme a figura 3.

125	D ₁₂	D ₁₃	D ₁₄
D ₂₁	D ₂₂	D ₂₃	D ₂₄
D ₃₁	D ₃₂	D ₃₃	D ₃₄
D ₄₁	D ₄₂	D ₄₃	D ₄₄

Figura 3. Imagem de saída após cálculo de D₁₁.

Em seguida, a função calcula o valor de D_{12} . Por este ser um pixel da lateral da imagem, com apenas 6 vizinhos, o ordenamento fica como segue: 45, 50, 125, 127, 130, 143. A escolha recai sobre o pixel mediano do vetor ordenado, ou seja, 127 (com arredondamento para cima).

$$D_{12} = 127$$

Os cálculos se seguem e o resultado final é apresentado na figura 4.

125	127	127	127
125	130	127	127
130	137	137	139
137	139	139	139

Figura 4. Imagem de saída após cálculo de todos os pixels.

O filtro mediano é útil por ser uma técnica relativamente simples, porém mais complexa que o filtro da média. Contudo, em geral apresenta resultados melhores.

O filtro mediano é mais indicado para ruído do tipo “Salt and Pepper”, pois pode retirar pixels espúrios de uma imagem. Se comparado com o filtro da média, pode-se observar que ele preserva as regiões de borda, embora tenha o mesmo efeito de remoção de pixels espúrios. Veja o exemplo de um pixel preto em uma região branca, na figura 5(a). Como se pode observar na saída (figura 5(b)), a região fica perfeitamente uniforme, sem nem ao menos diluir o ruído. O que acontece de fato é que o ruído é subtraído da imagem.

255	255	255	255
255	0	255	255
255	255	255	255
255	255	255	255

255	255	255	255
255	255	255	255
255	255	255	255
255	255	255	255

Figura 5. (a) Imagem com um pixel preto no centro de uma região branca e (b) o resultado da aplicação do filtro mediano.

Além disso, o filtro mediano não apresenta uma perda significativa nos detalhes de borda da imagem, reduzindo o efeito de borramento apresentado pelo filtro da média. Considerando a imagem de entrada com duas regiões, uma branca à esquerda e outra preta à direita, conforme figura 6(a). Após a aplicação do filtro mediano (figura 6(b)), o limite entre as duas regiões não perde nitidez, preservando o degrau de 255, exatamente devido a sua característica de não criar novos valores de pixel para a imagem processada.

255	255	0	0
255	255	0	0
255	255	0	0
255	255	0	0

255	255	0	0
255	255	0	0
255	255	0	0
255	255	0	0

Figura 5. (a) Imagem com duas regiões bem definidas, uma branca e outra preta e (b) o resultado da aplicação do filtro mediano.

A biblioteca *lil* possui duas funções para aplicação do filtro mediano, uma para imagens em tons de cinza e outra para imagens coloridas, no formato RGB. São elas:

```
limage* lmedian_filter_gray(limage* src, int nbh, int mode);  
limage* lmedian_filter_rgb(limage* src, int nbh, int mode);
```

Onde o retorno da função é a imagem filtrada. A imagem *src* é a imagem de entrada, *nbh* é o tamanho da vizinhança (1 para vizinhança 3x3, 2 para vizinhança 5x5 e assim por diante) e *mode* indica como a vizinhança será avaliada (0 se todos os pontos da vizinhança serão avaliados, 1 se apenas a vizinhança em cruz e 2 se apenas a vizinhança em xis).

Abaixo seguem alguns exemplos da aplicação do filtro da média em imagens reais.



Tanto para uma imagem em tons de cinza quanto para uma imagem colorida, sem ruído, o efeito de borramento é pequeno. Pode-se observar a perda de alguns detalhes, porém muito atenuada se comparada com a perda após a aplicação de um filtro da média.



O ruído é praticamente removido da imagem, enquanto a perda de nitidez é pouco significativa. Vale lembrar que o ruído apresentado nestas imagens é do tipo Salt and Pepper. Outros tipos de ruído podem não apresentar resultados tão vantajosos.