

## فیلتر میانگین (Mean Filter) :

یک فیلتر فطی ساده است که مقدار هر پیکسل در تصویر را با میانگین مقادیر پیکسل‌های اطرافش جایگزین می‌کند. این روش باعث می‌شود تصویر صاف‌تر و مات‌تر شود و نویزها و جزئیات ریز تصویر کاهش یابند. فیلتر میانگین در بسیاری از موارد برای حذف نویزهای تصادفی و ایجاد اثر مات شدن تصویر استفاده می‌شود.

### نمونه کار فیلتر میانگین

برای هر پیکسل در تصویر:

۱. یک پنجره (kernel) با اندازه مشخص، مثل  $3 \times 3$ ، در اطراف آن پیکسل در نظر می‌گیریم.
۲. مقدار تمام پیکسل‌های یک پنجره را با هم جمع کرده و میانگین می‌گیریم.
۳. مقدار میانگین را به جای مقدار اصلی پیکسل مرکزی قرار می‌دهیم.

### مثال عددی ساده

فرض کنیم یک تصویر کوچک با ماتریس زیر داریم و می‌فواهیم فیلتر میانگین  $3 \times 3$  را روی پیکسل مرکزی آن اعمال کنیم:

$$\begin{bmatrix} 100 & 150 & 100 \\ 50 & 80 & 90 \\ 70 & 120 & 110 \end{bmatrix}$$

برای محاسبه مقدار جدید پیکسل مرکزی (۸۰)، مجموع همه مقادیر در این پنجره را می‌گیریم و سپس آن را بر تعداد پیکسل‌ها (۹) تقسیم می‌کنیم:

$$\text{Mean} = (100 + 150 + 100 + 50 + 80 + 90 + 70 + 120 + 110) / 9 \approx 96.7$$

پس مقدار جدید پیکسل مرکزی ۹۷ خواهد بود.

پیاده‌سازی فیلتر میانگین در پایتون با OpenCV :

```
image = cv2.imread('path/to/image.jpg', 0)
mean_filter = cv2.blur(image, (3, 3))
```

### نتیجه استفاده از فیلتر میانگین:

در فیلتر میانگین، میانگین برای هر پنجره به صورت مستقل و جداگانه محاسبه می‌شود. به این معنا که فیلتر از مقدار پیکسل‌های اصلی تصویر استفاده می‌کند و به هیچ وجه از مقادیر فیلتر شده در پنجره‌های مجاور استفاده نمی‌کند. به عنوان مثال، در پنجره‌های  $9 \times 9$ ، با هر حرکت پنجره به اندازه یک پیکسل، در واقع یک ستون یا یک ردیف کامل شامل ۹ پیکسل جابجا می‌شود، و ۷۲ پیکسل از پنجره قبلی همپوشانی دارند که مقدار آن‌ها همگی از تصویر اصلی فوایده می‌شود.

---

پنجره‌ها (kernels) در فیلترهای تصویر، معمولاً همپوشانی دارند. در واقع، زمانی که فیلتر روی تصویر اعمال می‌شود، پنجره بر روی هر پیکسل حرکت می‌کند و هر بار یک مقدار جدید برای پیکسل مرکزی محاسبه می‌شود. این فرآیند با حرکت یک پیکسل به جلو در هر مرحله انجام می‌شود که باعث می‌شود پنجره با بخش قبلی تصویر همپوشانی داشته باشد. این همپوشانی در فیلترهای معمول مثل فیلتر میانگین، گوسی، میانه و لاپلاسیان دیده می‌شود و به فیلتر کمک می‌کند تا تغییرات کوچک و جزئیات در تصویر را هم شامل کند.

---

## فیلتر گوسی:

یکی از مهم‌ترین و پرستفاده‌ترین فیلترها در پردازش تصویر است، مخصوصاً برای حذف نویز و ایجاد مات‌شدگی طبیعی‌تر در تصاویر. این فیلتر از تابع توزیع گوسی برای تعیین وزن پیکسل‌ها استفاده می‌کند که باعث می‌شود محو شدن (بلور) در تصویر به صورت ملایم‌تر و طبیعی‌تر انجام شود.

### تابع گوسی

برای درک نحوه کار فیلتر گوسی، ابتدا باید با تابع گوسی آشنا شویم. این تابع برای یک تصویر دو بعدی به شکل زیر تعریف می‌شود:

$$\left( \frac{x^2 + y^2}{2\sigma^2} \right) \exp \frac{1}{2\pi\sigma^2} = G(x, y)$$

در اینجا:

- $x$  و  $y$  مختصات پیکسل‌های موجود در پنجره فیلتر هستند.
- $\sigma$  (سیگما) یک پارامتر است که میزان پراکندگی توزیع را تعیین می‌کند. هرچه سیگما بزرگ‌تر باشد، مات‌شدگی تصویر بیشتر خواهد بود.

---

سیگما، انحراف معیار توزیع گوسی است. هرچه سیگما بیشتر باشد، پراکندگی بیشتر، یعنی مقادیر پیکسل‌ها در فاصله‌های دورتر از مرکز هم اثر بیشتری خواهند داشت. یعنی پیکسل‌ها اطلاعات بیشتری از مناطق دورتر در تصویر رو ترکیب می‌کنند. این باعث می‌شود که تغییرات تند یا لبه‌های واضح در تصویر محو و نرم‌تر بشن.

وقتی سیگما کمتر باشد، تاثیر پیکسل‌های نزدیک به مرکز پنجره بیشتر خواهد بود و پیکسل‌های دورتر تاثیر کمتری دارن. وقتی سیگما کوچک است، فیلتر بیشتر بر روی پیکسل‌های نزدیک به مرکز تمرکز می‌کند، و لبه‌ها و جزئیات تصویر بهتر حفظ می‌شوند.

---

## نمونه اعمال فیلتر گاوسی

۱. **ایجاد پنجره:** (Kernel) برای اعمال فیلتر گاوسی، ابتدا یک پنجره مربعی با اندازه مشخص ایجاد می‌کنیم (مانند  $3 \times 3$ ,  $5 \times 5$ ). اندازه این پنجره باید به گونه‌ای انتخاب شود که تمام تأثیرات تابع گاوسی پوشش داده شود. معمولاً اندازه پنجره تقریباً سه برابر سیگما است.
۲. **مماسبه وزن‌ها:** هر عنصر در پنجره فیلتر گاوسی با استفاده از تابع گاوسی مماسبه می‌شود. این باعث می‌شود که پیکسل‌های نزدیک به مرکز پنجره وزن بیشتری داشته باشند، در حالی که پیکسل‌های دورتر وزن کمتری دارند. این ویژگی باعث ایجاد مموشدگی طبیعی‌تر در تصویر می‌شود.
۳. **اعمال فیلتر:** پنجره فیلتر گاوسی روی هر پیکسل در تصویر حرکت می‌کند. مقدار هر پیکسل جدید به صورت میانگین وزنی مقادیر پیکسل‌های داخل پنجره مماسبه می‌شود.

## مثال عددی

فرض کنیم یک پنجره گاوسی  $3 \times 3$  با  $\sigma=1$  داریم. در این صورت وزن‌ها برای این پنجره به صورت زیر خواهند بود:

0.0625	0.125	0.0625
0.125	0.25	0.125
0.0625	0.125	0.0625

در این مثال:

- مقدار پیکسل مرکزی، که بیشترین تأثیر را دارد، دارای وزن 0.25 است.
- پیکسل‌های نزدیک به مرکز وزن‌های بیشتری دارند و پیکسل‌های دورتر وزن‌های کمتری.

## مماسبه پیکسل جدید با پنجره گاوسی

اگر تصویر اصلی ما به صورت زیر باشد:

100	150	100
90	80	50
110	120	70

برای مماسبه مقدار پیکسل مرکزی جدید، میانگین وزنی زیر را به دست می‌آوریم:

$$(100 \times 0.0625) + (150 \times 0.125) + (100 \times 0.0625) + (50 \times 0.125) + (80 \times 0.25) + (90 \times 0.125) + (70 \times 0.0625) + (120 \times 0.125) + (110 \times 0.0625) = 85.625$$

بنابراین مقدار جدید پیکسل مرکزی برابر با مدود ۸۶ خواهد بود.

## پیاده‌سازی فیلتر گاوسی در پایتون با OpenCV :

```
image = cv2.imread('path/to/image.jpg', 0)
gaussian_blur = cv2.GaussianBlur(image, (3, 3), 1)
```

## مزایا و معایب فیلتر گاوسی

- **مزایا:** فیلتر گاوسی به دلیل وزن دهی ملایم به پیکسل‌های نزدیک به مرکز، مموشدگی طبیعی‌تری نسبت به فیلتر میانگین ایجاد می‌کند. این ویژگی باعث می‌شود که لبه‌های نافه‌استه کمتر آسیب ببینند.
- **معایب:** اگر سیگما یا اندازه پنجره بزرگ انتخاب شود، لبه‌ها نیز مموش می‌شوند و این موضوع ممکن است در برخی کاربردها نامطلوب باشد.

## فیلتر میانه (Median Filter) :

یکی از فیلترهای غیرخطی در پردازش تصویر است که برای کاهش نویز، به ویژه نویزهای نمک و فلفل، بسیار مؤثر است. برخلاف فیلترهای میانگین و گاوسی که میانگین یا ترکیب وزنی از پیکسل‌ها را مناسبه می‌کنند، فیلتر میانه مقدار میانه پیکسل‌های موجود در پنجره را به عنوان مقدار جدید پیکسل مرکزی تعیین می‌کند.

### مراحل کار فیلتر میانه

۱. **انتخاب پنجره (Kernel):** یک پنجره مربعی (مثلاً  $3 \times 3$ ،  $5 \times 5$ ) روی تصویر تعریف می‌شود.
۲. **مرتب‌سازی مقادیر در پنجره:** مقادیر پیکسل‌های درون پنجره را به ترتیب صعودی مرتب می‌کنیم.
۳. **انتخاب میانه:** مقدار میانه پیکسل‌ها را پیدا کرده و آن را به عنوان مقدار جدید برای پیکسل مرکزی قرار می‌دهیم.

**مثال با پنجره  $3 \times 3$ :** فرض کنیم بخش کوچکی از یک تصویر با نویز به صورت زیر باشد:

10	150	100
255	5	200
90	120	180

برای مناسبه مقدار جدید پیکسل مرکزی (که مقدار اصلی آن ۵ است):

۱. تمام مقادیر درون پنجره را مرتب می‌کنیم:

[5,10,90,100,120,150,180,200,255]

۲. میانه این لیست مقدار 120 است (پنجمین عنصر در لیست مرتب‌شده).

بنابراین مقدار جدید برای پیکسل مرکزی برابر با ۱۲۰ خواهد بود.

### پیاده‌سازی فیلتر میانه در پایتون با OpenCV

در OpenCV، فیلتر میانه را به سادگی می‌توان با تابع `medianBlur` اعمال کرد. به مثال زیر توجه کنید:

```
image = cv2.imread('path/to/image.jpg', 0)
median_filter = cv2.medianBlur(image, 3)
```

## مزایا و معایب فیلتر میانه

**مزایا:** فیلتر میانه به طور مؤثری نویزهای شدید مانند نویز نمک و فلفل را حذف می‌کند، بدون این‌که مانند فیلترهای میانگین و گاوسی تصویر را مات کند. این ویژگی آن را برای تصاویر با نویزهای شدید بسیار مناسب می‌کند.

**معایب:** برای نویزهای تصادفی‌تر و کمتر مشخص، فیلتر میانه ممکن است به اندازه فیلترهای دیگر مؤثر نباشد. همچنین، برای لبه‌های دقیق، ممکن است جزئیات تصویر به میزان اندکی تغییر کند.

## نکته در مورد همپوشانی

مانند سایر فیلترها، پنجره فیلتر میانه هم با هر حرکت یک پیکسل جابجا می‌شود و همپوشانی زیادی دارد. برای مثال، در یک پنجره  $5 \times 5$ ، با هر حرکت، ۲۰ پیکسل همپوشانی دارند، و فقط ۵ پیکسل جدید وارد پنجره می‌شوند.

## فیلتر دوطرفه:

فیلتر دوطرفه از دو نوع وزن‌دهی همزمان استفاده می‌کند:

۱. **وزن‌دهی بر اساس فاصله مکانی:** مشابه فیلتر گاوسی، پیکسل‌های نزدیک به مرکز پنجره وزن بیشتری نسبت به پیکسل‌های دورتر دارند. این ویژگی به خاطر کاهش اثر پیکسل‌های دورتر از پیکسل مرکزی به کار می‌رود.
۲. **وزن‌دهی بر اساس تفاوت شدت (روشنایی) (رنگ یا شدت نور):** پیکسل‌هایی که از نظر شدت نور مشابه پیکسل مرکزی هستند وزن بیشتری می‌گیرند و پیکسل‌های با شدت‌های متفاوت وزن کمتری دارند. این ویژگی به حفظ لبه‌ها کمک می‌کند. به عبارت دیگر، فیلتر دوطرفه به جای اینکه مانند فیلترهای معمول تمام پیکسل‌های داخل پنجره را با یک وزن اعمال کند، به پیکسل‌های مشابه با مرکز پنجره (از لحاظ شدت نور) اجازه می‌دهد بیشتر در محاسبه تأثیر بگذارند.

در OpenCV، فیلتر دوطرفه به راحتی قابل اعمال است. به مثال زیر توجه کنید:

```
image = cv2.imread('path/to/image.jpg', 0)
bilateral_filtered = cv2.bilateralFilter(image, d=9, sigmaColor=75, sigmaSpace=75)
```

در این مثال:

- $d$  اندازه پنجره (قطر فیلتر) است. مقدار بالاتر برای پنجره بزرگ‌تر و هموارتر استفاده می‌شود.
- $\sigma_{\text{Color}}$  میزان حساسیت به تفاوت شدت نور. مقدار بالاتر باعث کاهش اثر نویز می‌شود، ولی ممکن است برخی از جزئیات را هم از بین ببرد.
- $\sigma_{\text{Space}}$  میزان حساسیت به فاصله مکانی. مقدار بالاتر باعث می‌شود اثر فیلتر روی مناطق بزرگ‌تر پخش شود.

## مزایا و معایب فیلتر دوطرفه

- **مزایا:** فیلتر دوطرفه به فوبی نویز را از بین می‌برد و در عین حال، لبه‌ها و جزئیات تصویر را حفظ می‌کند.
  - **معایب:** از نظر محاسباتی، این فیلتر سنگین است و برای تصاویر بزرگ، زمان اجرای بیشتری نسبت به فیلترهای دیگر دارد. همچنین، تنظیم پارامترها برای بهینه‌سازی نتایج ممکن است نیاز به آزمون و خطا داشته باشد.
- فیلتر دوطرفه معمولاً در کاربردهایی مانند آماده‌سازی تصاویر قبل از پردازش لبه‌ها یا در کاربردهایی که نیاز به حفظ جزئیات همراه با حذف نویز است، بسیار مفید است.