

Simítás és szűrés képtérben MATLAB környezetben

Matlab-ban a legtöbb simító művelet elvégezhető az `fspecial(...)` függvény segítségével. Az `fspecial()` függvénynek meg kell adni a szűrő típusát és ha van, akkor a paramétereit is. Fontos megjegyezni, hogy az `fspecial(...)` függvény csupán a szűrő maszkot állítja elő, nem ez végzi el a szűrést a képen. A képműveletet az `imfilter(...)` függvény segítségével hajtjuk végre. Ezekhez az alábbiakban látunk részletesebb példákat.

Átlagoló szűrő

Állítsunk elő simított képeket a `szeder.jpg`-ből különböző méretű simító maszkokkal.

```
>> szeder = imread('szeder.jpg');
>> imshow(szeder);

>> H = fspecial('average', 3);
>> szeder3 = imfilter(szeder, H);
>> imshow(szeder3);

>> H = fspecial('average', 7);
>> szeder7 = imfilter(szeder, H);
>> imshow(szeder7);

>> H = fspecial('average', 15);
>> szeder15 = imfilter(szeder, H);
>> imshow(szeder15);
```



Átlagoló simítás kör alakú maszkkal

A `'disk'` paraméter meghívásával az `fspecial` függvény egy kör alakú maszkkal simítja el a képet. A kör sugara is megadható paraméterként, az így kapott maszk egy $2r+1$ átmérőjű kör lesz. A sugár alapértelmezett értéke $r=2$, tehát a szűrő átmérője 5.

```
>> C = fspecial('disk', 7);
>> szeder_disk = imfilter(szeder, C);
>> imshow(szeder_disk);

>> C = fspecial('disk', 15);
>> szeder_disk = imfilter(szeder, C);
>> imshow(szeder_disk);
```



Gauss-, Laplace- és LoG szűrők

A 'gaussian', 'laplacian' és 'log' sztringeket megadva paraméterként előállíthatjuk a kívánt szűrőket. A 'gaussian' paraméter hatására egy aluláteresztő szűrőt kapunk. A 'laplacian' paraméter esetében a szűrő a Laplace operátort közelíti, míg a 'log' paraméter esetén a Laplacian of Gaussian függvényt közelítést kapjuk operátorként.

```
>> G = fspecial('gaussian', 5, 0.3)
```

```
G =
```

```
      0      0.0000      0.0000      0.0000      0
0.0000      0.0000      0.0038      0.0000      0.0000
0.0000      0.0038      0.9847      0.0038      0.0000
0.0000      0.0000      0.0038      0.0000      0.0000
      0      0.0000      0.0000      0.0000      0
```

```
>> L = fspecial('laplacian', 0.2)
```

```
L =
```

```
  0.1667    0.6667    0.1667
  0.6667   -3.3333    0.6667
  0.1667    0.6667    0.1667
```

```
>> szeder_L = imfilter(szeder, L);
```

```
>> imshow(szeder_L);
```

```
>> LoG = fspecial('log', 5, 0.4)
```

```
LoG =
```

```
  0.2475    0.2475    0.2479    0.2475    0.2475
  0.2475    0.3545    1.2336    0.3545    0.2475
  0.2479    1.2336   -10.3145    1.2336    0.2479
  0.2475    0.3545    1.2336    0.3545    0.2475
  0.2475    0.2475    0.2479    0.2475    0.2475
```

```
>> szeder_Log = imfilter(szeder, LoG);
```

```
>> imshow(szeder_Log);
```

Elmozdulás-szűrő

A 'motion' opcióval elmozdulás effektet lehet készíteni. További két paraméterként meg kell adni az elmozdulás hosszát és irányát (szöggént).

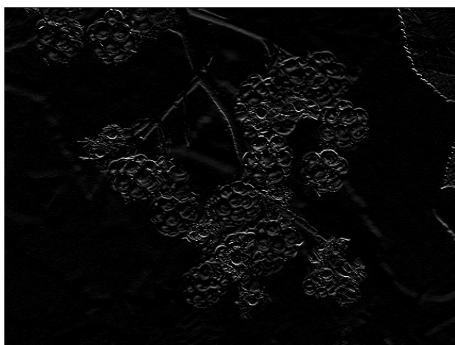
```
>> M = fspecial('motion', 50, 20);  
>> szeder_M = imfilter(szeder, M);  
>> imshow(szeder_M);
```



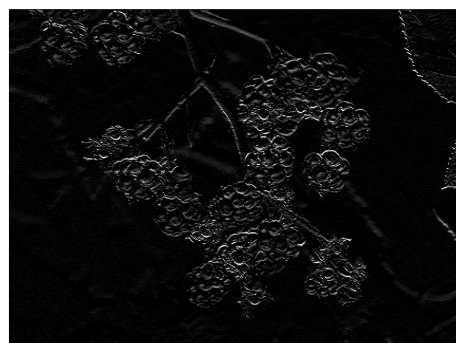
Gradiens operátorok

Az `fspecial(...)` függvényt a 'prewitt' és 'sobel' paraméterekkel meghívva gradiens operátorokat kapunk.

```
>> szeder_gray = rgb2gray(szeder);  
>> P = fspecial('prewitt');  
>> szeder_P = imfilter(szeder_gray, P);  
>> imshow(szeder_P);  
  
>> S = fspecial('sobel');  
>> szeder_S = imfilter(szeder_gray, S);  
>> imshow(szeder_S);
```



Prewitt



Sobel

Kép élesítése

A képet - az angol elnevezést követve - az `unsharp` maszkkal lehet élesíteni. A Matlab-ban használható élesítő maszk 3×3 -as méretű. A függvény második paraméterének egy $[0, \dots, 1]$ intervallumba eső értéket kell megadni, amely az élesítés mértékét jelenti. A maszk nem más, mint a Laplace operátor negatívja.

```
>> U = fspecial('unsharp', 0.6)
```

```
U =
```

```
   -0.3750   -0.2500   -0.3750  
   -0.2500    3.5000   -0.2500  
   -0.3750   -0.2500   -0.3750
```

```
>> szeder_U = imfilter(szeder, U);  
>> imshow(szeder_U);
```

```
>> U = fspecial('unsharp', 0.2)
```

```
U =
```

```
   -0.1667   -0.6667   -0.1667  
   -0.6667    4.3333   -0.6667  
   -0.1667   -0.6667   -0.1667
```

```
>> szeder_U = imfilter(szeder, U);  
>> imshow(szeder_U);
```

```
>> U = fspecial('unsharp', 0.8)
```

```
U =
```

```
   -0.4444   -0.1111   -0.4444  
   -0.1111    3.2222   -0.1111  
   -0.4444   -0.1111   -0.4444
```

```
>> szeder_U = imfilter(szeder, U);  
>> imshow(szeder_U);
```

