Differential Equations in Image Processing and Computer Vision 2008
Example Solutions for Programming Assignments 1 (P1)

Problem 1

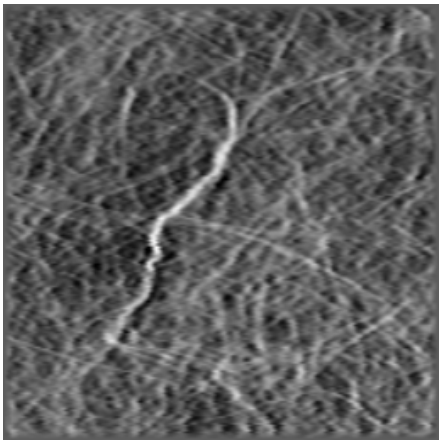
The complete file `lindiff.c` with supplemented code reads

```
/* ------------------------------------------------------ */
void lindiff

     (float    ht,        /* time step size, 0 < ht <= 0.25 */
      long     nx,        /* image dimension in x direction */
      long     ny,        /* image dimension in y direction */
      float    hx,        /* pixel size in x direction */
      float    hy,        /* pixel size in y direction */
      float    **u)       /* in: original image;  out: smoothed */


/*
 linear diffusion subroutine for du/dt = div(grad(u))
*/


{
long    i, j;                /* loop variables */
float   rx, ry;              /* mesh ratios */
float   **f;                 /* copy of input image u */


/* ---- allocate storage for f ---- */

alloc_matrix (&f, nx+2, ny+2);


/* ---- copy u into f ---- */

for (i=1; i<=nx; i++)
   for (j=1; j<=ny; j++)
       f[i][j] = u[i][j];
```

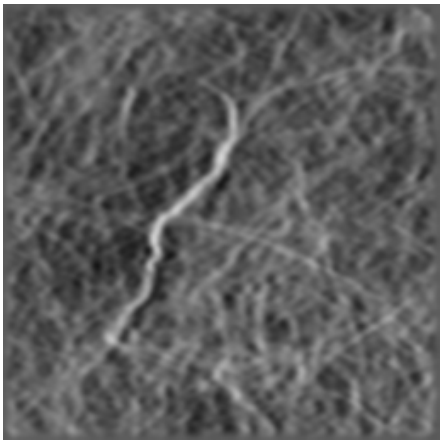
```
/* ---- create dummy boundaries for f by mirroring ---- */

dummies (f, nx, ny);


/* ---- diffusive averaging ---- */

rx = ht / (hx * hx);
ry = ht / (hy * hy);

for (i=1; i<=nx; i++)
   for (j=1; j<=ny; j++)
     {
       /* --- SUPPLEMENTED CODE --- */
       u[i][j]=  (1-2*rx-2*ry)*f[i  ][j  ]
                             +rx*f[i+1][j  ]
                             +rx*f[i-1][j  ]
                             +ry*f[i  ][j+1]
                             +ry*f[i  ][j-1];
       }

/* ---- disallocate storage for g ---- */

disalloc_matrix (f, nx+2, ny+2);

return;

} /* lindiff */

/* ------------------------------------------------------- */
```

Problem 2

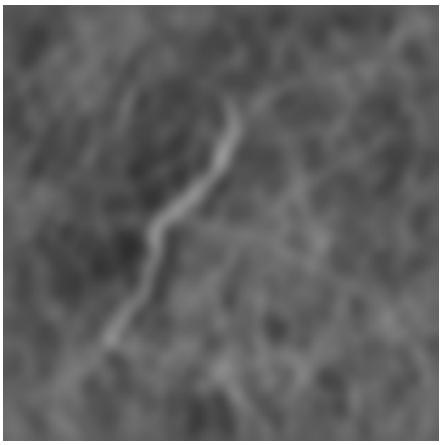
In order to visualise the cloudiness of the fleece at different scacles, one can for instance chose the following diffusion times



original



$t = 1$ (τ=0.2, n_{iter}=5)



$t = 8$ (τ=0.2, n_{iter}=40)

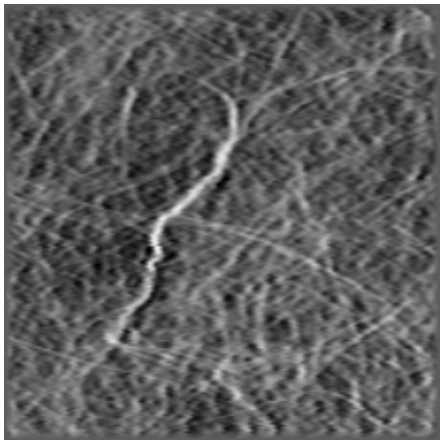
$t = 40$ (τ=0.2, n_{iter}=200)

Regarding the development of the minimum, the maximum ,the mean and the standard deviation during the iterations, one can observe the following: As to be expected the mean is preserved (mass preservation) and the minimum and the maximum become larger and smaller, respectively (min-max-stability). Moreover, the standard deviation decreases, since the image is simplified consecutively towards the mean value.

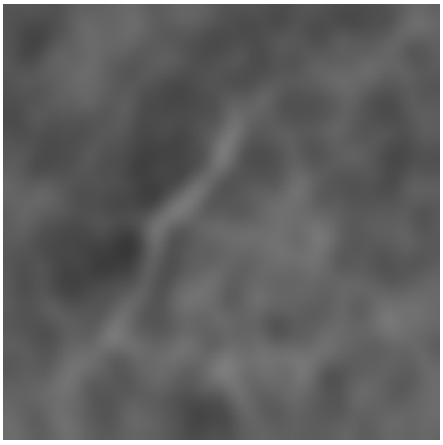
If the time step size τ is chosen to be larger than 0.25, one can observe that the min-max-stability no longer holds. The minimum may become negative and the maximum may increase. As a consequence, the process diverges (no stability). This, however, is not surprising, since for such a large time step size the central entry of the stencil of the explicit scheme becomes negative and thus does not fulfil the requirements for a discrete diffusion process any longer.

Problem 3

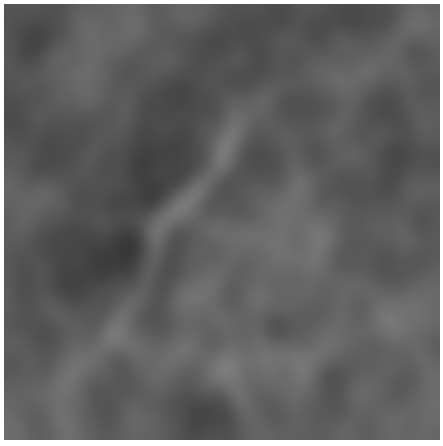
In oder to perform a cross validation between the diffusion process and the Gaussian convolution, we have to chose the result for a certain diffusion time, for instance $t = 20$ (τ=0.2, n_{iter}=100). The corresponding standard deviation of a Gaussian kernel is then given by $\sigma = \sqrt{2t} = 2\sqrt{10}$.



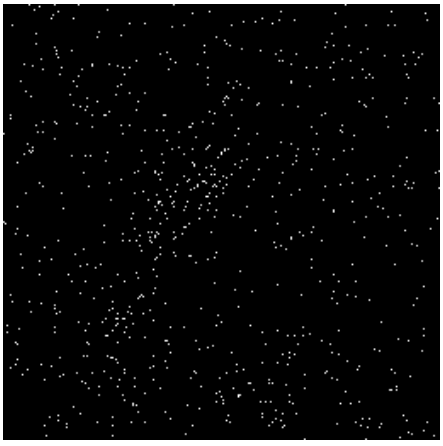
original



diffusion $t = 1$ (τ=0.2, n_{iter}=100)



convolution $\sigma = 2\sqrt{10}$

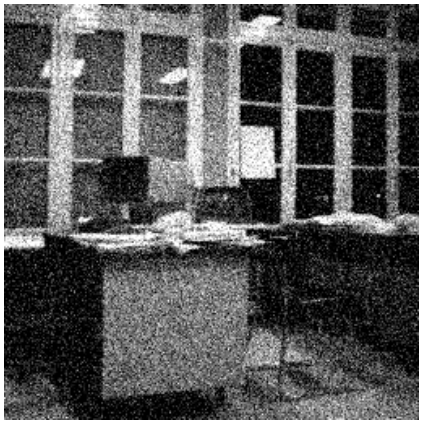


magnified difference

As one can see the result of the Gaussian diffusion differs only at a few pixels from the result of the linear diffusion. In this context, one should note that white pixels visualise a difference of only one grey value.

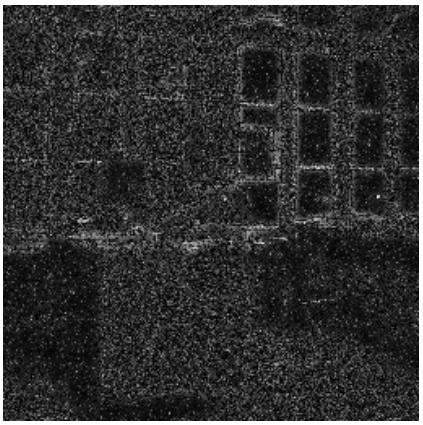
Problem 4

We show several denoising results and the corresponding method noise:



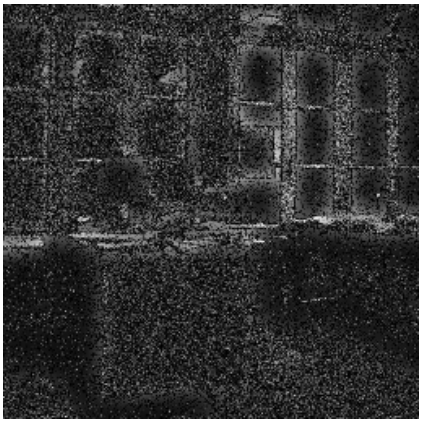
noisy input image

linear diffusion, $t = 2.5$



corresponding method noise

linear diffusion, $t = 10$



corresponding method noise

It is clearly visible that linear diffusion can reduce the noise in the images significantly. On the other hand, some important image features such as the edges are deteriorated, too. The difference images between in- and output of the linear diffusion filter make the most prominent edges of the image visible. With this respect, they show that the filter does not only remove noise, but also important image information.