

Chapter 11

Image Restoration

- In general, images degraded by
 - blurring (optical system, motion)
 - noise (from electric system)
- Can we compensate these effects in order to restore the un-degraded image?

11.1 Statistical properties of an image

We need a few concepts from probability theory (cf. Section 8.1)

- random variable X : a function that maps an event to a real number x ("measurement")
- probability density function (PDF) $p_x(x)$: Probability that X takes a value between x_1 and x_2 :

$$P(x_1 \leq X \leq x_2) = \int_{x_1}^{x_2} p_x(x) dx \quad (11.1)$$

PDF can only be used in an integral

- Expectation of random variable X :

$$E[X] = \mu_x = \int_{-\infty}^{+\infty} x p_x(x) dx \quad (11.2)$$

- Expectation of a function:

$$E[f(X)] = \int_{-\infty}^{+\infty} f(x) p_x(x) dx \quad (11.3)$$

- Variance (mean square deviation from mean):

$$Var[X] = \sigma_x^2 = E[(X - E[X])^2] \quad (11.4)$$

- Covariance of two random variables

$$Cov[X, Y] = E[XY] \quad (11.5)$$

Now, an image is viewed as a two dimensional random field
 = random variable depending on two spatial coordinates
 = “measurement” $f(x, y)$ at each position (x, y)

- mean:

$$\mu_f = E[f(x, y)] \quad (11.6)$$

- variance:

$$\sigma_f^2(x, y) = E [(f(x, y) - \mu_f(x, y))^2] \quad (11.7)$$

- autocorrelation

$$R_f(x, y, x', y') = E [f(x, y)f(x', y')] \quad (11.8)$$

tells how the value of f at (x, y) is correlated with the value at (x', y')

- stationary process:

$$\Rightarrow R_f(x, y, x', y') = R_f(x - x', y - y') \quad (11.9)$$

- Ergodic assumption: $E[.]$ can be replaced by temporal or spatial average

$$\Rightarrow \mu_f = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x, y) dx dy \quad (11.10)$$

no PDF needed

- The autocorrelation function can be rewritten as a convolution:

$$R_f(\Delta x, \Delta y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(\alpha, \beta) f(\alpha + \Delta x, \beta + \Delta y) d\alpha d\beta = f(x, y) * f(-x, -y) \quad (11.11)$$

- Power spectrum of the image = absolute square of its Fourier transform

$$S_f(u, v) = \mathcal{F} \mathcal{T}[R_f] = \dots = |F(u, v)|^2 \quad (11.12)$$

where $F(u, v) = \mathcal{F} \mathcal{T}[f(x, y)]$

11.2 Noise, Degradation

- noise can be seen as some random signal $n(x, y)$ that is added to the image and that has zero mean μ_n

$$\mu_n = 0 \quad (11.13)$$

- in the simplest case, we have “white noise”:
the autocorrelation function then is

$$R_n(x, y) = \sigma_n^2 \delta(x, y) \quad \text{i.e., no spatial correlation} \quad (11.14)$$

and hence its power spectrum (absolute square of its Fourier transform $N(u, v)$)

$$S_n(u, v) = \sigma_n^2 \quad \text{i.e., all frequencies contribute} \rightarrow \text{"white"} \quad (11.15)$$

To estimate the noise variance, calculate the variance of a homogeneous image area

- Image degradation = blurring + noise:

$$\underbrace{g(x, y)}_{\text{degraded im.}} = \underbrace{h(x, y)}_{\text{blurring PSF}} * \underbrace{f(x, y)}_{\text{orig. im.}} + \underbrace{n(x, y)}_{\text{noise}} \quad (11.16)$$

where the blurring point spread function (PSF), can, e.g., be Gaussian

- So the Fourier transform $G(u, v)$ of the degraded image is

$$G(u, v) = H(u, v)F(u, v) + N(u, v) \quad (11.17)$$

11.3 Image Restoration Filters

- Image restoration means:
try to reconstruct a good estimate $\hat{f}(x, y)$ of the original image $f(x, y)$ from the degraded image $g(x, y)$
- Often called *deconvolution* because we want to revert the effect of the convolution with the blurring PSF $h(x, y)$
- We look for a restoration filter in the frequency domain (easier because there, the blurring is a multiplication, not a convolution, see Eq. s 11.16 and 11.17)
- Wanted: filter $H_R(u, v)$ that gives a good estimate $\hat{F}(u, v)$ of $F(u, v)$:

$$\hat{F}(u, v) = H_R(u, v)G(u, v) \Rightarrow \hat{f}(x, y) = h_R(x, y) * g(x, y) \quad (11.18)$$

so

$$\hat{F}(u, v) = H_R(u, v)H(u, v)F(u, v) + H_R(u, v)N(u, v) \quad (11.19)$$

11.3.1 Inverse deconvolution filter

- if there is no noise, and blurring $h(x, y)$ or $H(u, v)$ known

$$H_R(u, v) = \frac{1}{H(u, v)} \quad (11.20)$$

this is the *inverse deconvolution filter*. Then

$$\hat{F}(u, v) = \frac{1}{H(u, v)} \underbrace{H(u, v)F(u, v)}_{G(u, v)} \quad (11.21)$$

Then

$$g(x,y) \xrightarrow{FT} G(u,v) \xrightarrow{\frac{1}{H(u,v)}} \frac{G(u,v)}{H(u,v)} \xrightarrow{FT^{-1}} \hat{f}(x,y) \quad (11.22)$$

Problems

1. $H(u,v)$ maybe = 0 for some (u,v)
2. if there is noise:

$$\hat{F}(u,v) = F(u,v) + \underbrace{\frac{N(u,v)}{H(u,v)}}_{\text{large where } H(u,v) \text{ is small}} \quad (11.23)$$

“ill-conditioned problem” \rightarrow Fig. 11.1

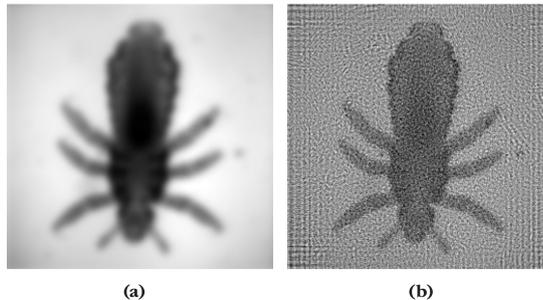


Fig. 11.1: Effect of noise on deconvolution: (a) severely blurred image with added random noise; (b) deconvolution with inverse filter (Fig. 6.39 from Russ, 2011).

- Solution: *Wiener deconvolution filter*:

11.3.2 Wiener deconvolution filter

- Idea: try to minimise the mean square difference between f and \hat{f}

$$\hat{f}(x,y) = h_R(x,y) * g(x,y) \quad (11.24)$$

using the abbreviation

$$S(u,v) = H(u,v)F(u,v) \quad (11.25)$$

and writing the new filter as the simple inverse filter $1/H(u,v)$ with real adjustment factor $\Phi(u,v)$:

$$\hat{F}(u,v) = \frac{\Phi(u,v)}{\underbrace{H(u,v)}_{H_R(u,v)}} G(u,v) \quad (11.26)$$

we write the mean square difference of \hat{f} and f

$$\begin{aligned}
 e^2 &= \iint |\hat{f}(x,y) - f(x,y)|^2 dx dy & (11.27) \\
 &= \iint |\hat{F}(u,v) - F(u,v)|^2 du dv \\
 &= \iint \left| \frac{\Phi(u,v)G(u,v)}{H(u,v)} - \frac{S(u,v)}{H(u,v)} \right|^2 du dv \quad (\text{used: } F = S/H) \\
 &= \iint \frac{1}{|H(u,v)|^2} |\Phi(u,v)(S(u,v) + N(u,v)) - S(u,v)|^2 du dv \quad (\text{used: } G = S + N) \\
 &= \iint \frac{1}{|H(u,v)|^2} \left\{ |S(u,v)|^2 |1 - \Phi(u,v)|^2 + |N(u,v)|^2 |\Phi(u,v)|^2 \right\} du dv \quad (\text{Exercise!})
 \end{aligned}$$

Using $|S|^2 = |H|^2 |F|^2 = |H|^2 S_f$, and $|N|^2 = S_n$, and the fact that Φ is real, we get

$$e^2 = \iint \frac{1}{|H(u,v)|^2} \left\{ |H(u,v)|^2 S_f(u,v) (1 - \Phi(u,v))^2 + S_n(u,v) \Phi(u,v)^2 \right\} du dv \quad (11.28)$$

Looking for $\Phi(u,v)$ that minimises it: set derivative with respect to Φ to zero

$$\frac{de^2}{d\Phi(u,v)} = \frac{1}{|H(u,v)|^2} \left\{ |H(u,v)|^2 S_f(u,v) \cdot 2 \cdot (1 - \Phi(u,v)) \cdot (-1) + 2S_n(u,v) \Phi(u,v) \right\} = 0 \quad (11.29)$$

so

$$|H(u,v)|^2 S_f(u,v) (1 - \Phi(u,v)) = S_n(u,v) \Phi(u,v) \quad (11.30)$$

and thus

$$\Phi(u,v) = \frac{|H(u,v)|^2 S_f(u,v)}{|H(u,v)|^2 S_f(u,v) + S_n(u,v)} = \frac{|H(u,v)|^2}{|H(u,v)|^2 + \frac{S_n(u,v)}{S_f(u,v)}} \quad (11.31)$$

And so we finally get for the Wiener deconvolution filter:

$$H_R(u,v) = \frac{1}{H(u,v)} \frac{|H(u,v)|^2}{|H(u,v)|^2 + \frac{S_n(u,v)}{S_f(u,v)}} \quad (11.32)$$

Note:

1. if $S_n \rightarrow 0$ then $H_R = \frac{1}{H}$, i.e., when there is no noise, the Wiener filter is just the simple inverse filter
2. we need a good idea of the blurring point spread function $H(u,v)$
3. We also need to know the ratio of the noise power spectrum and the power spectrum of the original image, $S_n(u,v)/S_f(u,v)$, i.e., the inverse signal-to-noise ratio (as a function of frequency (u,v))
4. Wiener filter optimal in the average sense, so locally it might be not so good.

- Often, the inverse signal-to-noise ratio $S_n(u, v)/S_f(u, v)$, is not well known
- ⇒ replace it by empirical parameter K that can be adjusted, trade-off between sharpness and noise reduction

$$H_R(u, v) = \frac{1}{H(u, v)} \frac{|H(u, v)|^2}{|H(u, v)|^2 + K} \quad (11.33)$$

→Fig. 11.2

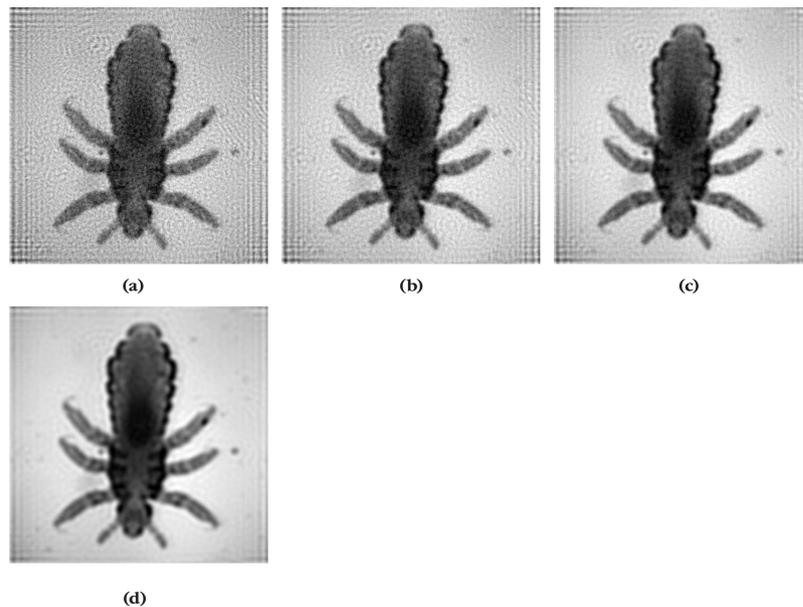


Fig. 11.2: Wiener deconvolution the image in Fig. 11.1a. Increasing the empirical parameter K ((a) to (d)) reduces noise at the expense of sharpness (Fig. 6.40 from Russ, 2011).

11.3.3 Other image restoration filters

- there are other types of image restoration and deconvolution filters
- usually adapted to special kinds of image degradation/blur and noise
- often iterative methods
- much depends on the degree of knowledge about the blurring and the noise
- see, e.g., Chap. 6, section in deconvolution in Russ (2011)

Bibliography

J. Russ. *The Image Processing Handbook*. CRC Press, 6th edition, 2011. ISBN 978-1-4398-4045-0.