Wiener deconvolution

From left: Original image, blurred image, image deblurred using Wiener deconvolution.

In mathematics, **Wiener deconvolution** is an application of the Wiener filter to the noise problems inherent in deconvolution. It works in the frequency domain, attempting to minimize the impact of deconvolved noise at frequencies which have a poor signal-to-noise ratio.

The Wiener deconvolution method has widespread use in image deconvolution applications, as the frequency spectrum of most visual images is fairly well behaved and may be estimated easily.

Wiener deconvolution is named after Norbert Wiener.

1 Definition

Given a system:

$$y(t) = (h * x)(t) + n(t)$$

where $*$ denotes convolution and:

- $x(t)$ is some original signal (unknown) at time t.

- $h(t)$ is the known impulse response of a linear time-invariant system

- $n(t)$ is some unknown additive noise, independent of $x(t)$

- $y(t)$ is our observed signal

Our goal is to find some $g(t)$ so that we can estimate $x(t)$ as follows:

$$\hat{x}(t) = (g * y)(t)$$

where $\hat{x}(t)$ is an estimate of $x(t)$ that minimizes the mean square error.

The Wiener deconvolution filter provides such a $g(t)$. The filter is most easily described in the frequency domain:

$$G(f) = \frac{H^*(f)S(f)}{|H(f)|^2 S(f) + N(f)}$$

where:

- $G(f)$ and $H(f)$ are the Fourier transforms of g and h, respectively at frequency f.

- $S(f)$ is the mean power spectral density of the original signal $x(t)$

- $N(f)$ is the mean power spectral density of the noise $n(t)$

- the superscript * denotes complex conjugation.

The filtering operation may either be carried out in the time-domain, as above, or in the frequency domain:

$$\hat{X}(f) = G(f)Y(f)$$

(where $\hat{X}(f)$ and $Y(f)$ are the Fourier transforms of $\hat{x}(t)$ and $y(t)$, respectively) and then performing an inverse Fourier transform on $\hat{X}(f)$ to obtain $\hat{x}(t)$.

Note that in the case of images, the arguments t and f above become two-dimensional; however the result is the same.

2 Interpretation

The operation of the Wiener filter becomes apparent when the filter equation above is rewritten:

$$\begin{aligned}
G(f) &= \frac{1}{H(f)} \left[\frac{|H(f)|^2}{|H(f)|^2 + \frac{N(f)}{S(f)}} \right] \\
&= \frac{1}{H(f)} \left[\frac{|H(f)|^2}{|H(f)|^2 + \frac{1}{\text{SNR}(f)}} \right]
\end{aligned}$$

Here, $1/H(f)$ is the inverse of the original system, and $\text{SNR}(f) = S(f)/N(f)$ is the signal-to-noise ratio. When there is zero noise (i.e. infinite signal-to-noise), the

term inside the square brackets equals 1, which means that the Wiener filter is simply the inverse of the system, as we might expect. However, as the noise at certain frequencies increases, the signal-to-noise ratio drops, so the term inside the square brackets also drops. This means that the Wiener filter attenuates frequencies dependent on their signal-to-noise ratio.

The Wiener filter equation above requires us to know the spectral content of a typical image, and also that of the noise. Often, we do not have access to these exact quantities, but we may be in a situation where good estimates can be made. For instance, in the case of photographic images, the signal (the original image) typically has strong low frequencies and weak high frequencies, and in many cases the noise content will be relatively flat with frequency.

3   Derivation

As mentioned above, we want to produce an estimate of the original signal that minimizes the mean square error, which may be expressed:

$$\epsilon(f) = \mathbb{E}\left|X(f) - \hat{X}(f)\right|^2$$

where $\mathbb{E}$ denotes expectation.

If we substitute in the expression for $\hat{X}(f)$, the above can be rearranged to

$$\begin{aligned}\epsilon(f) &= \mathbb{E}\left|X(f) - G(f)Y(f)\right|^2 \\ &= \mathbb{E}\left|X(f) - G(f)\left[H(f)X(f) + V(f)\right]\right|^2 \\ &= \mathbb{E}\left|\left[1 - G(f)H(f)\right]X(f) - G(f)V(f)\right|^2\end{aligned}$$

If we expand the quadratic, we get the following:

$$\begin{aligned}\epsilon(f) = &\left[1 - G(f)H(f)\right]\left[1 - G(f)H(f)\right]^*\mathbb{E}|X(f)|^2 \\ &- \left[1 - G(f)H(f)\right]G^*(f)\,\mathbb{E}\left\{X(f)V^*(f)\right\} \\ &- G(f)\left[1 - G(f)H(f)\right]^*\mathbb{E}\left\{V(f)X^*(f)\right\} \\ &+ G(f)G^*(f)\,\mathbb{E}|V(f)|^2\end{aligned}$$

However, we are assuming that the noise is independent of the signal, therefore:

$$\mathbb{E}\left\{X(f)V^*(f)\right\} = \mathbb{E}\left\{V(f)X^*(f)\right\} = 0$$

Also, we are defining the power spectral densities as follows:

$$S(f) = \mathbb{E}|X(f)|^2$$

$$N(f) = \mathbb{E}|V(f)|^2$$

Therefore, we have:

$$\epsilon(f) = \left[1 - G(f)H(f)\right]\left[1 - G(f)H(f)\right]^*S(f) + G(f)G^*(f)N(f)$$

To find the minimum error value, we calculate the Wirtinger derivative with respect to $G(f)$ and set it equal to zero.

$$\frac{d\epsilon(f)}{dG(f)} = G^*(f)N(f) - H(f)\left[1 - G(f)H(f)\right]^*S(f) = 0$$

This final equality can be rearranged to give the Wiener filter.

4   See also

- Deconvolution
- Wiener filter
- Point spread function
- Blind deconvolution
- Fourier transform

5   References

- Rafael Gonzalez, Richard Woods, and Steven Eddins. *Digital Image Processing Using Matlab*. Prentice Hall, 2003.

6   External links

- Comparison of different deconvolution methods.
- Deconvolution with a Wiener filter

7 Text and image sources, contributors, and licenses

7.1 Text

- **Wiener deconvolution** *Source:* https://en.wikipedia.org/wiki/Wiener_deconvolution?oldid=750255354 *Contributors:* The Anome, Michael Hardy, Dhwoow, Oli Filth, Berland, Rainwarrior, VolkovBot, DaBler, Dekart, Addbot, Yobot, Beao, Ukkar, ChuispastonBot, NCOH, Waelhalbawi, Marcocapelle, Brunobeltran and Anonymous: 15

7.2 Images

- **File:Commons-logo.svg** *Source:* https://upload.wikimedia.org/wikipedia/en/4/4a/Commons-logo.svg *License:* PD *Contributors:* ? *Original artist:* ?
- **File:Image_restoration_(motion_blur,_Wiener_filtering).png** *Source:* https://upload.wikimedia.org/wikipedia/commons/e/e3/Image_restoration_%28motion_blur%2C_Wiener_filtering%29.png *License:* Public domain *Contributors:* Own work *Original artist:* DaBler

7.3 Content license

- Creative Commons Attribution-Share Alike 3.0