Filtering Images

April 25, 2019
Filtering Images

Which of the following space-domain images can be constructed by filtering one of the other images by the DFT of another of them?
Filtering Images

Filter model:

Space-domain interpretation:

\[ x[r, c] \rightarrow h[r, c] \rightarrow y[r, c] = (x \ast h)[r, c] \]

Frequency-domain interpretation:

\[ X[k_r, k_c] \rightarrow H[k_r, k_c] \rightarrow Y[k_r, k_c] = X[k_r, k_c]H[k_r, k_c] \]

We should be able to understand the previous problem both ways.
Filtering Images

Which of the following images can be constructed by

- circularly convolving two of the other images
- inverse transforming the product of the DFTs of two images
Filtering Images

Which of the following images can be constructed by

- circularly convolving two of the other images
- inverse transforming the product of the DFTs of two images

\[ D \propto A \ast B. \]

\[ D \propto A \ast C. \]

\[ D \propto B \ast C. \]
Filtering Images

Try the transform method.
Filtering Images

Try the transform method.
Filtering Images

Try the transform method.

\[
\text{DFT}(D) \propto \text{DFT}(A) \times \text{DFT}(B) \\
\text{DFT}(D) \propto \text{DFT}(A) \times \text{DFT}(C) \\
\text{DFT}(D) \propto \text{DFT}(B) \times \text{DFT}(C)
\]
Application of Filtering: Motion Blur

Remove motion blur from the following image.
Reducing Motion Blur

How can we remove the motion blur?

There are multiple streaks in the blurred image that are about 20 pixels long and angled down at about 30 degrees. Assume that these streaks resulted from the blurring.

There is an isolated streak near the point \( r = 120, c = 250 \). It is approximately 19 pixels wide and 6 pixels high. We can use this information to make a model of the blurring in this image.
Reducing Motion Blur

Let $X$ represent a perfect image of the car with no blur.

Let $H$ represent a blurring filter.

Then our blurred image $Y$ can be represented by applying $H$ to $X$.

The idea behind inverse filtering is that we can get back the original $X$ by applying a filter $G = \frac{1}{H}$ to $Y$.
Inverse Filtering

Make an image $h$ to represent the presumed blurring function.

Let $H$ represent the DFT of $h$, and filter the blurred image with $\frac{1}{H}$. 
Inverse Filtering

Here is the resulting inverse filtered image – not at all what we want.

What went wrong?
Inverse Filtering

This image shows the magnitude of $H$ (DFT of blur function).
Inverse Filtering

This image shows the magnitude of $\frac{1}{H}$.

What causes the bright spots? Why are they a problem?
The bright spots in $\frac{1}{H}$ come from points in $H$ with values near zero.

\[ G = \frac{1}{H} \]

Such bright spots dominate the result. Try limiting their magnitudes.

**Method 1:**

Start with $G = \frac{1}{H}$, but limit the magnitude of every point in $G$ to 4:

```python
for kr in range(R):
    for kc in range(C):
        if abs(G[kr,kc]) > 4:
            G[kr,kc] *= 4/abs(G[kr,kc])
```
Deblurring

This deblurring filter works better: easy to read license number.

But there are many artifacts.
Deblurring

The form of the previous deblurring function is a bit arbitrary.

\[ H G = 1 \]

Method 2:

Here is a frequently used alternative (a “Weiner filter”):  

\[ G = \frac{1}{H} \frac{|H|^2}{|H|^2 + C} \]

where \( C = 0.004 \) (chosen by trial and error).
Deblurring

Alternative deblurring function.

But there are still artifacts.
Edge Effects

Much of the ringing results from circular convolution. Window edges in original image to reduce step change due to periodic extension.
Comparison

Method 1 with and without windowing.
Comparison

Method 2 with and without windowing.
Conclusions

Problems with inverse filtering. In today’s deblurring problem, there were points \((k_r, k_c)\) in the frequency representation of motion blur whose values \(H[k_r, k_c]\) were near zero.

Such points are not recoverable with inverse filtering. In fact, they corrupt the overall process of deblurring and make the inverse-filtering approach useless.

However, there were only a few such points. Arbitrarily limiting the values of such points results in useful deblurring.

Problems with circular convolution. Circular convolution introduces enormous artifacts if the left and right (or top and bottom) edges differ in brightness. These artifacts can be reduced by windowing.

Remaining problems. The resulting images still suffer from ringing – presumably because of sharp discontinuities in the frequency representation of blurring.