CSCI 1290: Comp Photo

Fall 2018 @ Brown University
James Tompkin

Many slides thanks to James Hays’ old CS 129 course, along with all of its acknowledgements.
Questions from Tuesday

• Aaron:
  • Crepuscular rays during an eclipse

• Keiichiro:
  • Dark current sensor noise

• Eric
  • How can we model noise in general?

• Leslie:
  • Why is $I = D \times D$?
Aaron – Crepuscular Rays (shadowing)
Aaron – eclipse pinhole cameras!

Keiichiro – Dark Current

• Residual current (rate of flow of charge) within a sensor when there is no illumination -> NOISE!
  • How to measure? *Put the lens cap on!*
  • We will do this in lab 3...

---

ISO 51200
30 sec exposure
25 deg celsius

[Canon; Shigemi Numazawa]
Keiichiro – Dark Current

- Camera operation itself
  - Dark noise varies per pixel!

[Dunlap et al., Correction of Dark Current in Consumer Cameras, 2010; https://www.photonics.com/Articles/Dark_Current_in_Consumer_Cameras_Not_What_you/a44298]
Keiichiro – Dark Current

- Ambient temperature

Very important to astronomical imaging community – e.g., many forum posts:

[Dunlap et al., Correction of Dark Current in Consumer Cameras, 2010; https://www.photonics.com/Articles/Dark_Current_in_Consumer_Cameras_Not_What_you/a44298
https://www.cloudynights.com/topic/483729-new-canon-7dii-has-one-tenth-the-dark-current/]
Sensor cooling

‘Dark frame’; ISO 6400; *brightened* 40%

Left-hand side: -2 deg Celsius
Right-hand side: 20 deg Celsius

[PrismaLuceLab; https://petapixel.com/2016/10/11/cooled-nikon-d5500a-chills-sensor-clearer-star-photos/]
Eric – How to model noise?

• Dark current noise -> residual noise with no illuminant
  • Thermal noise -> ‘white’ noise, i.e., Gaussian iid per pixel

• Shot noise -> variation in the number of photons sensed at a given exposure level
  • Poisson distribution; models discrete photons arriving at sensor pixels
  • Proportional to $\sqrt{\text{intensity}}$

• Quantization noise -> analog to digital conversion noise
  • Uniform; signal independent

• Noise floor -> sum of all noise (non-signal) across the imaging pipeline
  • Below this, no signal can be detected
  • E.G., defines limit on signal to noise ratio.
Eric – How to model noise?

Dead pixels -> sensor pixel always outputs 0 / receives no power
Stuck pixels -> sensor pixel always outputs the same value
   Hot pixels -> sensor pixel over outputs as gain / temperature increase

Not stars; hot pixels!

ISO 3200
(Brightness boosted 40%)

[Canon; Shigemi Numazawa]
Leslie – $I = D \times D$

Mae – “...to do with lack of content (black) at edges”
Leslie – $I = D \ast D$

$$D = \text{im2double}(\text{imread}('\text{convexample.png'}))$$

$$I = \text{conv2}(D, D)$$

$$\text{max} (\text{max}(I))$$

ans =
1.1021e+04

$$\text{l}_\text{norm} = (I - \text{min}(\text{min}(I))) / (\text{max}(\text{max}(I)) - \text{min}(\text{min}(I)))$$

$$\text{imshow} (\text{l}_\text{norm})$$
Leslie – I = D * D

For x: 275 + (275-1)/2 + (275-1)/2
= 549
Leslie \(- I = D \ast D \)

>> \texttt{I = conv2( D, D, 'full' );} (Default; pad with zeros)

>> \texttt{I = conv2( D, D, 'same' );} (Return same size as A)

>> \texttt{I = conv2( D, D, 'valid' );} (No padding)
Practical matters

• What about near the edge?
  – the filter window falls off the edge of the image
  – need to extrapolate
  – methods:
    • clip filter (black)
    • wrap around
    • copy edge
    • reflect across edge
Filtering as template matching

Then I got ahead of myself and started talking about when the filter ‘looks like’ the image... ‘template matching’...

Filtering viewed as comparing an image of what you want to find against all image regions
Earlier on, we had this example:

\[ b) \ A = \_ \ast \_ \]

In class, the response was:
\[ A = B \ast C \]
Intuitively, “because it kind of looks like it.”

C is a Gaussian filter
(or something close to it it),
and we know that it ‘blurs’.
Filtering: Correlation and Convolution

- 2d correlation
  \[ h[m,n] = \sum_{k,l} f[k,l] I[m+k,n+l] \]

  \( h=\text{filter2}(f,I); \) or \( h=\text{imfilter}(I,f); \)

- 2d convolution
  \[ h[m,n] = \sum_{k,l} f[k,l] I[m-k,n-l] \]

  \( h=\text{conv2}(f,I); \) or \( h=\text{imfilter}(I,f,'\text{conv}'); \)

  \( \text{conv2}(I,f) \) is the same as \( \text{filter2}(\text{rot90}(f,2),I) \)

Correlation and convolution are identical when the filter is symmetric.
OK, so let’s test this idea. Let’s see if we can use correlation to ‘find’ the parts of the image that look like the filter.

```matlab
>> f = D( 57:117, 107:167 )
```

*Expect response ‘peak’ in middle of I*

```matlab
>> I = filter2( D, f, ‘same’ );
```

Hmm…
That didn’t work – why not?

[Thanks to Robert Collins @ Penn State]
Correlation

\[ h[m,n] = \sum_{k,l} f[k,l] I[m+k,n+l] \]

\[ h=\text{filter2}(f,I); \text{ or } h=\text{imfilter}(I,f); \]

As brightness in \( I \) increases, the response in \( h \) will increase, as long as \( f \) is positive.
OK, so let’s subtract the mean

```matlab
>> f = D( 57:117, 107:167 );
>> f2 = f – mean(mean(f));
>> D2 = D – mean(mean(D));
```

Score is higher only when dark parts match and when light parts match.

```matlab
>> I2 = filter2( Dm, f2, ‘same’ );
```
Or even

\[
\text{>> } I_3 = \text{filter2}( D_2, D_2, \text{`full' });
\]