Computer Vision
Homework 07: Matching and Alignment
CS 670, Fall 2019

Name: Kunjal Panchal
Student ID: 32126469
Email: kpanchal@umass.edu

October 25, 2019

Contents

1 Problem 1: Gaussian Derivative Proportionality

1.1 Definitions

- GAUSSIAN FUNCTIONS: are widely used in statistics to describe the normal distributions, in signal processing to define Gaussian filters, in image processing where two-dimensional Gaussians are used for Gaussian blurs. [1]
  Gaussian functions are often used to represent the probability density function of a normally distributed random variable with expected value and variance σ^2. In this case, the Gaussian is of the form:

$$g = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$

- In scale space representation, Gaussian functions are used as smoothing kernels for generating multi-scale representations in computer vision and image processing.

- Specifically, derivatives of Gaussians (Hermite functions) are used as a basis for defining a large number of types of visual operations.

1.2 Derivative of Gaussian w.r.t x

1.2.1 First-order Derivative

$$\begin{aligned}
\frac{\partial g}{\partial x} &= \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \left(-\frac{1}{2\sigma^2}\right)(2x) \\
&= \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \left(-\frac{x}{\sigma^2}\right) \\
&= \frac{-x}{\sqrt{2\pi}\sigma^3} e^{-\frac{x^2}{2\sigma^2}}
\end{aligned}$$

- In first step, we took derivative of exponent term e as the fractional part before it is constant. We applied

$$\frac{d}{dx}e^{ax} = e^{ax}\frac{d}{dx}(ax)$$

  rule.

- Then in the next two steps, we just rearranged terms for better visual representation and understanding.

1.2.2 Second-order Derivative

$$
\begin{aligned}
\frac{\partial^2 g}{\partial x^2} &= \frac{-1}{\sqrt{2\pi}\sigma^3}\left[e^{-\frac{x^2}{2\sigma^2}} \cdot \frac{d}{dx}x + x \cdot \frac{d}{dx}e^{-\frac{x^2}{2\sigma^2}}\right] \\
&= \frac{-1}{\sqrt{2\pi}\sigma^3}\left[e^{-\frac{x^2}{2\sigma^2}} + x \cdot e^{-\frac{x^2}{2\sigma^2}}\frac{-2x}{2\sigma^2}\right] \\
&= \frac{-1}{\sqrt{2\pi}\sigma^3} \cdot e^{-\frac{x^2}{2\sigma^2}}\left[1 - \frac{x^2}{\sigma^2}\right] \\
&= \frac{-1}{\sqrt{2\pi}\sigma^3}\left[\frac{\sigma^2 - x^2}{\sigma^2}\right]e^{-\frac{x^2}{2\sigma^2}} \\
&= \frac{-1}{\sqrt{2\pi}\sigma^5}\left[\sigma^2 - x^2\right]e^{-\frac{x^2}{2\sigma^2}}
\end{aligned}
$$

- In first and second step, we took derivative by applying the Product Rule.
$$\frac{d}{dx}f(x)g(x) = f\frac{d}{dx}g(x) + g\frac{d}{dx}f(x)$$

- Rest, we just rearranged terms for better visual representation and understanding.

$$\boxed{\frac{\partial^2 g}{\partial x^2} = \frac{-1}{\sqrt{2\pi}\sigma^5}\left[\sigma^2 - x^2\right]e^{-\frac{x^2}{2\sigma^2}}} \tag{1}$$

1.3 Derivative of Gaussian w.r.t σ

$$
\begin{aligned}
\frac{\partial g}{\partial \sigma} &= \frac{1}{\sqrt{2\pi}}\left[\sigma^{-1} e^{-\frac{x^2}{2\sigma^2}}\right] \\
&= \frac{1}{\sqrt{2\pi}}\left[\sigma^{-1}\frac{d}{d\sigma} e^{-\frac{x^2}{2\sigma^2}} + e^{-\frac{x^2}{2\sigma^2}}\frac{d}{d\sigma}\sigma^{-1}\right] \\
&= \frac{1}{\sqrt{2\pi}}\left[\sigma^{-1} e^{-\frac{x^2}{2\sigma^2}}\left(-\frac{x^2}{2}\cdot -2\sigma^{-3}\right) + e^{-\frac{x^2}{2\sigma^2}}(-\sigma^{-2})\right] \\
&= \frac{1}{\sqrt{2\pi}}\left[e^{-\frac{x^2}{2\sigma^2}} x^2\sigma^{-4} - e^{-\frac{x^2}{2\sigma^2}}\sigma^{-2}\right] \\
&= \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}}\left[\frac{x^2}{\sigma^4} - \frac{1}{\sigma^2}\right] \\
&= \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}}\left[\frac{x^2-\sigma^2}{\sigma^4}\right] \\
&= \frac{1}{\sqrt{2\pi}\sigma^4}\left[x^2-\sigma^2\right] e^{-\frac{x^2}{2\sigma^2}} \\
&= \frac{-1}{\sqrt{2\pi}\sigma^4}\left[\sigma^2-x^2\right] e^{-\frac{x^2}{2\sigma^2}}
\end{aligned}
$$

- In first step, we represented σ in its inverse form so we don't have to deal with derivational division rule, and only with the product rule.
- Then we took derivative by applying the Product Rule.

$$\frac{d}{dx} f(x)g(x) = f\frac{d}{dx}g(x) + g\frac{d}{dx}f(x)$$

- Rest, we just rearranged terms for better visual representation and understanding. Took outside, whatever there was common.

$$\boxed{\frac{\partial g}{\partial \sigma} = \frac{-1}{\sqrt{2\pi}\sigma^4}\left[\sigma^2-x^2\right] e^{-\frac{x^2}{2\sigma^2}}} \tag{2}$$

From Equations 1 and 2, we see that

$$\frac{\partial^2 g}{\partial x^2} = \frac{1}{\sigma}\cdot\frac{\partial g}{\partial \sigma}$$

Hence, we can say that the second derivative of this Gaussian w.r.t. x is proportional to its derivative w.r.t. scale parameter σ

$$\boxed{\frac{\partial^2 g}{\partial x^2} \propto \frac{\partial g}{\partial \sigma}} \tag{3}$$

2 Problem 2: Normalized Correlation over SSD

2.1 Definitions

- INVARIANCE : can recognize an object as an object, even when its appearance varies in some way.
  It preserves the object's identity, category, (etc) across changes in the specifics of the visual input, like relative positions of the viewer/camera and the object.

- FEATURE DESCRIPTOR is an algorithm which takes an image and outputs feature descriptors/feature vectors. Feature descriptors encode interesting information into a series of numbers and act as a sort of numerical "fingerprint" that can be used to differentiate one feature from another.
  E.g., SIFT, which encodes information about the local neighborhood image gradients the numbers of the feature vector

- SIMPLEST DESCRIPTOR is a vector of raw intensity values

2.2 How to Compare Two such vectors/ descriptors?

SSD - SUM of SQUARED DIFFERENCES is a distance measure.[2]

$$SSD(u, v) = \sum_i (u_i - v_i)^2$$

But as one can infer, this is **INVARIANT** to intensity changes.

If one vector is as it was before, but now the other vector is under, say, a strong monochromatic light, the distance between two values at same vector indices will be changed conspicuously.

Even when two vectors are representing very similar objects, the SSD relies on their pixel intensities and changing their value will determine how "similar" the two objects are.

Thus, under different intensities, the raw pixel value difference is calculated for SSD, which is not the ideal candidate, we need some preprocessing on raw intensities before we measure the similarity.

NORMALIZED CORRELATION is a similarity measure.

It "normalizing" the pixel values with respect to all the other pixel values of that image. So, even if there is a tint or general overall effect of particular lights, it will be on every pixel and thus, after the normalizing, that effect will be removed.

It is calculated as shown below:

$$\rho(u, v) = \frac{(u - \bar{u})}{||u - \bar{u}||} \cdot \frac{(v - \bar{v})}{||v - \bar{v}||}$$

As we can see, pixel values are pre-processed first, as a bias due to light change on all pixels will be removed if we centralize them on the mean, making it 0.

In conclusion, Normalized Correlation is preferred over SSD because Normalized Correlation is invariant to affine (translation + scaling) intensity changes; while SSD is not invariant. Which might result in highly fluctuating and incorrect descriptors between two similar images.

3 Problem 3: RANSAC

- Fraction of points that are outliers: p
- Fraction of points that are inliers: $1-p$
- Probability of Success: s
- Probability of Failure: $1-s$
- Number of points needed for estimating a model: $n(=2, forthisproblem)$
- Number of Iterations: T

If p is the fraction of points that are outliers, the probability that choosing **one** point yields an **inlier** will be

$$1-p$$

Now, the probability of choosing n inliers in a row (case when sample only contains inliers) will be

$$(1-p)^n$$

For our case, we have $n=2$;

$$\therefore (1-p)^2$$

So, the probability that one or more points in the sample were outliers (sample is contaminated) will be

$$1-(1-p)^2$$

And the probability that one or more points in the sample were outliers (sample is contaminated) EVEN AFTER T iterations will be

$$\boxed{(1-(1-p)^2)^T} \tag{4}$$

3.1 (a) Terminate Without Finding the Correct Solution

From Equation 4, we can determine that the probability that RANSAC will terminate without finding the correct solution after T iterations will be, Probability of failure =

$$(1-(1-p)^2)^T$$

3.2 (b) Iterations T needed to Find the Correct Solution

If $p = 0.5$ i.e., 50% of the points are outliers;
From Equation 4; Iterations T are needed to find the correct solution with probability > 0.99 will be,

$$
\begin{aligned}
s &= 1 - (1 - (1 - p)^2)^T \\
0.99 &= 1 - (1 - (1 - 0.5)^2)^T \\
T &= \frac{log(1 - 0.99)}{log(1 - (1 - 0.5)^2)} \\
T &= \frac{log(0.01)}{log(1 - 0.25)} \\
T &= \frac{log(0.01)}{log(0.75)} \\
T &= 16.007846
\end{aligned}
$$

$$\boxed{\therefore T \approx 17} \tag{5}$$

3.3 (c) k points needed for estimating a model

We just have to change the value of $n = 2$ and make it $n = k$ in the Equation 4, as this question just says that instead having both points right, we need all k points to fit the model.

So, the probability that one or more points in the sample were outliers (sample is contaminated) even after T iterations will be now

$$\boxed{(1 - (1 - p)^k)^T} \tag{6}$$

Because, all the cases where $n = k$ inliers in a row (case when sample only contains inliers) will be chosen, will simply be $(1 - p)^k$ and hence, that being not the case is $(1 - (1 - p)^k)$ and that being not the case for T iterations will be $(1 - (1 - p)^k)^T$.

References

[1] *Gaussian function* `https://en.wikipedia.org/wiki/Gaussian_function`

[2] Subhransu Maji, *Image Alignment* `https://www.dropbox.com/s/5flemuwc73wv7o9/lec11\%2B12\%2B13_image_alignment.pdf?dl=0`