CPSC 425: Computer Vision

Instructor: Jim Little
little@cs.ubc.ca

Department of Computer Science
University of British Columbia

Lecture Notes 2016/2017 Term 2
Menu January 24, 2017

Topics:
  Template Matching
  Scaled Representations
  Estimating Derivatives

Reading:
  Today: Forsyth & Ponce (2nd ed.) 4.6–4.7, 5.1
  Next: Forsyth & Ponce (2nd ed.) 5.2

Handouts:
  Assignment 3: Face Detection in a Scaled Representation due Thursday, February 1

Reminders:
  www: http://www.cs.ubc.ca/~little/cpsc425/
  piazza: https://piazza.com/ubc.ca/winterterm2015/cpsc425/
Today’s “Fun” Example

Original NASA Viking 1 “Face on Mars” image, July 25, 1976

Image Credit:
http://esamultimedia.esa.int/images/marsexpress/300-230906-3253-6-vkl-Cydonia_H.jpg
Today’s “Fun” Example (cont’d)

Perspective view of the same region, based on image data acquired July 22, 2006

Image Credit:

http://esamultimedia.esa.int/images/marsexpress/311-230906-3253-6-3d5-Cydonia_H.jpg
Lecture 5: Re-cap

- The bilateral filter considers both spatial distance and range (intensity) distance, and has edge-preserving properties.

- Images can be characterized mathematically.
  — In the continuous case, images are functions of two spatial variables, \( x \) and \( y \).
  — The discrete case is obtained from the continuous case via sampling (i.e. tessellation, quantisation).

- If a signal is bandlimited then it is possible to design a sampling strategy such that the sampled signal captures the underlying continuous signal exactly.

- Adequate sampling may not always be practical. In such cases there is a trade-off between “things missing” and “artifacts”.
  — Different applications make the trade-off differently.
Template Matching

Today we ask:
How can we find a part of one image that matches another?
or,
How can we find instances of a pattern in an image?
Template Matching

Today we ask:
How can we find a part of one image that matches another?
or,
How can we find instances of a pattern in an image?

Key Idea: Use the pattern as a template
Template Matching

A toy example

Figure credit: Kristen Grauman
Template Matching

We can think of convolution/correlation as comparing a template (the filter) with each local image patch.

- Consider the filter and image patch as vectors.

- Applying a filter at an image location can be interpreted as computing the dot product (recall: element-wise multiply and sum) between the filter and the local image patch.
Template Matching

We can think of convolution/correlation as comparing a template (the filter) with each local image patch.

- Consider the filter and image patch as vectors.
- Applying a filter at an image location can be interpreted as computing the dot product (recall: element-wise multiply and sum) between the filter and the local image patch.

But... The dot product may be large simply because the image region is bright. We need to normalize the result in some way.
Template Matching

Let $a$ and $b$ be vectors. Let $\theta$ be the angle between them.

We know

$$\cos \theta = \frac{a \cdot b}{|a| |b|} = \frac{a \cdot b}{\sqrt{(a \cdot a)(b \cdot b)}}$$

where $\cdot$ is dot product and $| |$ is vector magnitude.

Correlation is a dot product.

Correlation measures similarity between the filter and each local image region.

Normalized correlation varies between $-1$ and $1$.

Normalized correlation attains the value $1$ when the filter and image region are identical (up to a scale factor).
Template Matching

Detected template

Correlation map

Figure credit: Kristen Grauman
Template Matching

Linear filtering the entire image computes the entire set of dot products, one for each possible alignment of filter and image.

Important Insight:
- filters look like the pattern they are intended to find
- filters find patterns they look like

Linear filtering is sometimes referred to as template matching.
Example 1: TV Remote Control

Example 1 (cont’d):

[Image of a person waving and an interactive interface panel]
Example 1 (cont’d):

Example 1 (cont'd):
Example 1 (cont’d):
Example 1 (cont’d): Normalized Correlation

Template (left), image (middle), normalized correlation (right)

Note peak value at the true position of the hand

When might template matching fail?
Template Matching (Re-cap)

- Good News:
  - works well in presence of noise
  - relatively easy to compute

- Bad News:
  - sensitive to (spatial) scale change
  - sensitive to 2D rotation

- More Bad News:
  When imaging 3D worlds:
  - sensitive to viewing direction and pose
  - sensitive to conditions of illumination
Framework for Next Topic: Scaled Representations

Problem:
Make template matching robust to changes in 2D (spatial) scale.

Key Idea(s):
Build a scaled representation: the Gaussian image pyramid

Alternatives:
— use multiple sizes for each given template
— ignore the issue of 2D (spatial) scale

Theory:
Sampling theory allows us to build image pyramids in a principled way

Practical Detail(s):
Building a Gaussian pyramid can be efficient in both space and time. Trade-offs remain between information lost and artifacts introduced

“Gotchas:”
— template matching remains sensitive to 2D orientation, 3D pose and illumination
Scaled Representations

Goals:

- to find template matches at all scales
  - template size constant, image scale varies
  - e.g., finding hands or faces when we don’t know what size they will be in the image
Scaled Representations

Goals:

- to find template matches at all scales
  - template size constant, image scale varies
  - e.g., finding hands or faces when we don’t know what size they will be in the image

- efficient search for image–to–image correspondences
  - look first at coarse scales, refine at finer scales
  - much less cost (but may miss best match)
Scaled Representations

Goals:

- **to find template matches at all scales**
  - template size constant, image scale varies
  - e.g., finding hands or faces when we don’t know what size they will be in the image

- **efficient search for image–to–image correspondences**
  - look first at coarse scales, refine at finer scales
  - much less cost (but may miss best match)

- **to examine all levels of detail**
  - find edges with different amounts of blur
  - find textures with different spatial frequencies (i.e., different levels of detail)
Shrinking an Image

We can’t shrink an image simply by taking every second pixel.

If we do, characteristic artifacts appear:
— small phenomena can look bigger
— fast phenomena can look slower

Common examples include:
— checkerboard patterns misrepresented in video games
— striped shirts look funny on colour television
— wagon wheels roll the wrong way in movies
Shrinking an Image

Forsyth & Ponce (2nd ed.) Figures 4.12–4.14 (top rows)
Template Matching

Subsampling with Gaussian pre-filtering

Figure credit: Steve Seitz and Richard Szeliski
Template Matching

Subsampling with Gaussian pre-filtering

Gaussian 1/2

G 1/4

G 1/8

Figure credit: Steve Seitz and Richard Szeliski
Template Matching

Subsampling without pre-filtering

Figure credit: Steve Seitz and Richard Szeliski
An image pyramid is a collection of representations of an image.

Typically, each layer of the pyramid is half the width and half the height of the previous layer.

In a Gaussian pyramid, each layer is smoothed by a Gaussian filter and resampled to get the next layer.
Gaussian Pyramid

Again, let $\otimes$ denote convolution

- Create each level from previous one — smooth and (re)sample

- Smooth with Gaussian, taking advantage of the fact that

$$G_{\sigma_1}(x) \otimes G_{\sigma_2}(x) \equiv G_{\sqrt{\sigma_1^2 + \sigma_2^2}}(x)$$
Example 2: A Gaussian Pyramid

Forsyth & Ponce (2nd ed.) Figure 4.17
We’ll now shift from global template matching to local feature detection.

Consider the problem of finding images of an elephant using a template.
We’ll now shift from global template matching to **local feature detection**.

Consider the problem of finding images of an elephant using a template.

An elephant looks different from different viewpoints

- from above (as in an aerial photograph or satellite image)
- head on
- sideways (i.e., in profile)
- rear on

What happens if parts of an elephant are obscured from view by trees, rocks, other elephants?
From Template Matching to Local Feature Detection

Find the chair in this image

Pretty much garbage
Simple template matching is not going to make it

Slide credit: Li Fei-Fei, Rob Fergus, and Antonio Torralba
Estimating Derivatives

Recall, for a 2D (continuous) function, \( f(x, y) \)

\[
\frac{\partial f}{\partial x} = \lim_{\varepsilon \to 0} \frac{f(x + \varepsilon, y) - f(x, y)}{\varepsilon}
\]

Differentiation is linear and shift invariant, and therefore can be implemented as a convolution

A (discrete) approximation is

\[
\frac{\partial f}{\partial x} \approx \frac{F(X + 1, Y) - F(X, Y)}{\Delta x}
\]
Estimating Derivatives

A similar definition (and approximation) holds for $\frac{\partial f}{\partial y}$.

Image noise tends to result in pixels not looking exactly like their neighbours, so simple “finite differences” are sensitive to noise.

The usual way to deal with this problem is to smooth the image prior to derivative estimation.
Example in 1D
Estimating Derivatives

Derivative in Y (i.e., vertical) direction

Forsyth & Ponce (1st ed.) Figure 7.4 (top left & top middle)
Estimating Derivatives

Derivative in X (i.e., horizontal) direction

Forsyth & Ponce (1st ed.) Figure 7.4 (top left & top right)
Summary

- Template matching as (normalized) correlation

- Template matching is not robust to changes in
  - 2D spatial scale and 2D orientation
  - 3D pose and viewing direction
  - illumination

- Scaled representations facilitate
  - template matching at multiple scales
  - efficient search for image–to–image correspondences
  - image analysis at multiple levels of detail

- A Gaussian pyramid reduces artifacts introduced when sub-sampling to coarser scales