Lecture 1

Related Fields

Image Processing - Take an image and produce another image.
Graphics - Take a description and convert it to an image.
Visualization - Take data and produce an image.
Computer Vision - Take image and produce a description.

Application

Computers can't see as well as humans. Vision has been successfully applied to domains with (1) specific constrained environments and, (2) high tolerance for error. Most ubiquitous application of vision is in *bar-code readers.* Zip-code recognition has humans in the loop. Industrial inspection and face detection in photo albums have high tolerance for error.

Why is vision hard ?

There is no well defined atomic unit or grammar. Images are 2-dimensional and many algorithms are NP-Hard in > 1-dimensions. Images don't have a well defined structure either. Some other challenges are

- Viewpoint variation
- Illumination changes
- Scale changes
- Intra-class variation
- Motion
- Background clutter
- Occlusion
- Role of high-level reasoning
- Need of prior information and context - perception is inherently ambiguous.

How do humans see ?

Optical illusions can be used to understand this. Humans use high-level reasoning and our understanding of the world. We try to make sense of images as 3-dimensional scenes. We concentrate on important details in a scene and ignore some smaller changes.

Deep Learning

There has been a lot of progress in last ~5 years. Deep learning beat all other techniques in the ImageNet Challenge 2012. Neural networks have been around for a long time. The difference this time was the use of large scale computation and data. Neural networks work really well on many vision tasks but they are vulnerable to adversarial attack. Since they are not fully understood, when neural nets fail, they are difficult to reason with. A lot of interesting works in vision use big data to give the illusion of intelligence.

Lecture 2

Image Acquisition

- Source of light
- Object (3D)
- Imaging System eg. eye or camera

Digital image is grid of pixel values. 0 - Low/Black, 255 - High/White. Grayscale = 1 mxn matrix (8 bits/pixel) and Color = 3 mxn matrices(24 bits/pixel).
Color Spaces: RGB, HSV (Hue, Saturation, Value/Brightness)

What is an image?

A 2D function f: R^2 -> R
f(x, y) gives intensity at (x, y).
A plot of f, x and y will give valleys for dark regions and peaks for bright regions.
A *digital* image is a discrete version of the function f. Domain is sampled since we only have f(x, y) values for fixed set of (x, y) range is quantized because the camera measures light intensity in a quantized way (0 to 255).

Image Transformations

Transforming f gives new image corresponding to the new function.
Noise Reduction - for a very still scene, take N pictures $f_1,\ldots,f_N$, define the transformed image as
g = 1/N * sum over i = 1 to N f_i
Assumptions -

- Scene is still.
- Noise is dynamic and random.
- Noise is not due to the scene.

Noise model-
(Additive gaussian noise)
$f_{Noisy}(x, y) = f_{Actual}(x, y) + e$
e ~ *N*(mean=0, stddev=sigma)

Image filtering

What if there is only one image ?
Assumptions-

- Visual world is kind of smooth.
- Neighbouring pixels tell a lot about the center pixel's value.

General framework to perform various tasks in image processing and computer vision.
Determine what the value for center pixel should have been on the basis of neighbouring pixels.
Mean filtering 3x3, k=1: [⅛ ⅛ ⅛; ⅛ ⅛ ⅛; ⅛ ⅛ ⅛].
Generally denom. = $(2k+1)^2$
The larger the k, the more blurry the resulting image will be.