1 Introduction

In Lecture 2 we have learned about matched filtering. We made a number of hypotheses, including that the ‘signal’ is constant, even if it is possibly corrupted by noise.

Two questions arise: how do we estimate the signal from noisy examples? (Simple answer: average the examples). What do we do when the signal is not constant? The answer to this second question is more complicated and depends on the source of variability.

There are many reasons why the signal may not be constant:

1. The object is rotated/translated in space so that its image appears rotated and scaled in the image plane.
2. The object is in a different position in the image.
3. The object (class) is inherently non-constant (e.g. cars, human faces).
4. The lighting conditions change.
5. There is occlusion.

In this lecture we learn a technique, principal component analysis (PCA), that allows us to account for changes in the object appearance that belong to a linear subspace, e.g. those that are due to changes in lighting. The technique may be stretched to approximate some amount of intrinsic variation (e.g. faces).

In future lectures we will learn how to model large amounts of intrinsic variation, occlusion and rotations in depth (i.e. not around the optical axis of the camera) using deformable models.

Variations in orientation around the optical axis of the camera (i.e. causing a rotation of the image of the object in the image plane), translation and scale will be handled ‘brute force’, i.e. the training images will have to be normalized (translated, rotated, stretched) by hand so that the training examples are aligned one to the other and the process of object detection and recognition will have to be performed on different copies of the images: rotated and scaled.
versions of the original image. We have encountered one example of such ‘brute force’ detection in Lecture 2 when we used convolution to detect the presence of an object in an unknown location.

2 Example 1: Signal variability as an effect of illumination

The Lambertian reflectance model, which is valid for certain matte surfaces (e.g. chalk) says that the image $I(x, y)$ of a surface $z(x, y)$ is given by:

$$I(x, y) = <\vec{n}(x, y), \vec{l}>$$ (1)

$$\vec{n}(x, y)^T = \left[\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}, 1\right]$$ (2)

Where $x$ and $y$ are the coordinates in the image plane, also corresponding to a point on the visible surface of the object. $n(x, y)$ is the normal vector to the surface of the object in the location $x, y$, and $\vec{l}$ is the vector that from the same point $x, y$ of the surface points towards the light source (the norm of $\vec{l}$ encodes for the product of the energy emitted by the light source and the albedo of the surface).

If the light source is a point far from the surface of the object, then $\vec{l}$ is approximately constant. We will make this approximation.

So: depending on the direction of the light source $l$ we will obtain different images even if the scene (i.e. $\vec{n}(x, y)$) does not change. We may write this dependency of $I$ from $\vec{l}$ using the notation $I(x, y; \vec{l})$. In principle, since there are infinite positions for the light source we may obtain infinite different images. However, it is possible to show that all these images in the Lambertian case take a 3-dimensional space.

In fact: $\vec{l}$ is a 3-vector (it has $x, y, z$ components) and therefore it may be written as the linear combination of three basis vectors: $\vec{l} = \alpha_1 \vec{l}_1 + \alpha_2 \vec{l}_2 + \alpha_3 \vec{l}_3$. Therefore $I$ may be written as the linear combination of three ‘basis’ images:

$$I(x, y; l) = \sum_i 1^3 \alpha_i <\vec{n}(x, y), l_i> = \sum_i 1^3 \alpha_i I_i(x, y)$$ (3)

$$I_i(x, y) = <\vec{n}(x, y), l_i>$$ (4)

where the basis images $I_i$ are images obtained under the lighting $l_i$. (See A. Shashua’s 1997 article for more details).

If the signal may be represented in a 3-dimensional space we may hope to be able to extend the matched filtering idea by using 3 filters instead of one. More on this later.

Is the Lambertian approximation reasonable? There are obvious situations where the approximation is clearly not valid. The most obvious ones are:
1. **Cast shadows and self-shadows** are areas of the surface that cannot ‘see’ the light source. In other terms, the norm of $\vec{l}$ goes to zero and therefore $l$ is not constant any longer.

2. **Specularities** are regions where the surface behaves like a mirror, not like a matte surface. In those places the brightness of the image $I$ depends on the position of the camera (or the eye) and not only on the position of the light source.

3. **Close-by light source**: in this case $\vec{l}$ is not constant on the surface.

### 3 Principal component analysis

How can we find out whether a given object produces images that are approximately in 3-space, or whether many more dimensions are necessary?

Suppose that one could only use one kernel, then one would pick the kernel that maximizes the average product with the observed signals $s_i$:

$$k = \arg \max_k \sum_i < k, s_i >^2 = k^T S S^T k$$

where $S = [s_1 \ldots s_N]$ (here, for simplicity of notation, we think of both $k$ and the $s_i$ as column vectors).

Now notice that $A = SS^T$ is a positive semidefinite symmetric matrix. Therefore its eigenvalues are real and semipositive and its eigenvectors are an orthonormal set. Call $u_i$ the set of eigenvectors of $A$ (equivalently, they are the left-eigenvectors of $S$, and call $\sigma_i$ the corresponding eigenvalues. Suppose that they are sorted in descending order of magnitude. Then $A$ may be expressed in terms of the basis formed by the $u_i$, and the maximization has a simple solution:

$$A = SS^T = \sum_i \sigma_i u_i u_i^T$$

$$k = \arg \max_k k^T A k = u_1$$

and

$$\max_k k^T A k = \sigma_1 \|k\|^2 = \sigma_1$$

Suppose that we allowed $k$ to belong to a $r$-dimensional space with $r << N$. Then the solution would be $k = u_i$.

This procedure is called **principal component analysis** (or, equivalently, singular value decomposition, check out the `svd()` function in Matlab) to select suitable subspace.

### 4 Representation error and classification error

Let’s briefly go back to the principal components analysis. The representation of the signal is as good as the norm of the principal values used in the representation (see calculations in class).
It has to be remarked that the classification error is not necessarily proportional to the representation error - this will be further explored in Lecture 4.

5 When multidimensional filtering is not sufficient

Consider the example shown in class (see lecture3.m). The signal is composed of two ‘blips’ with variable height and variable mutual position. Observe that when the mutual position of the two blips is highly variable the principal components (the singular values of the matrix composed of examples) decrease very slowly. Correspondingly observe that the first 16 principal vectors are ‘meaningful’, as opposed to the case where the mutual position of the ‘blips’ varies little.

It is quite clear that, as the signal deforms more and more, matched filtering works less and less well. There are two problems:

1. The number of principal components required for representing the signal increases more and more.

2. The filters are broad, thus capturing lots of noise, while the useful part of the signal is small.