Chapter 8

Lecture 8: Textures

8.1 Introduction

Texture is a region of the image made of elements with similar properties that is perceived as a being homogenous. Despite that textured image regions are made by elements with boundaries, one can trace the boundaries of the region containing the homogenous set of items. It is deliberated that I have included the observer in this description. Similar to color, texture is perceptual a concept. We could build a machine insensitive to texture and that only cares about the details. But humans prefer to create an abstraction and ignore the details and focus on some of the statistical properties of the elements inside a texture. Only when *attention* is focused we can extract detailed information about specific items in the texture.

8.2 Experiencing texture perception

Let's start with some exercises of introspection that might reveal some of the properties of our own mechanisms for texture processing. In the beginning of this lecture I will review three main tasks:

- Discrimination of image regions of homogeneous texture
- Perception of statistical properties of sets of elements
- Object recognition and crowding effects

8.2.1 Pre-attentive texture discrimination

The first task is texture segmentation. The displays created by Bela Julesz contain two textures and the goal is to find the boundary between the two regions. It is interesting to study which texture pairs can be easily discriminated effortlessly. If the two textures can easily be segmented from each other, then it means that the properties that define those textures are well matched to perceptual mechanisms. In particular it is interesting to study situations in which the textures vary only along one dimension (e.g., size of the constituent elements, orientation, mirror symmetry, etc.). However, not all textures are easy to segment. For instance, two textures in which one is composed of letters **R** in random orientations, and the second texture is composed of randomly oriented mirror versions of the letter **R** are extremely hard to segment. We need to carefully pay attention in order to be able to find the boundary between the two textured regions.

8.2.2 Perception of sets

When perceiving sets of similar items (e.g, a crowd of people, ...) we could store in memory a detailed account of each individual item. However, in most cases, such a detailed representation would be unnecessary costly. More useful will be to extract more general statistical properties such as the expected size of the elements, the average distance between items, etc. Those statistical properties of the set might be important abstractions to know properties such as what is the overall behavior of the set, are there any causes (forces, motivations, etc.) that apply to the set?

8.2.3 Crowding

Crowding occurs when objects are very close to each other and their *features* get mixed producing a percept reminiscent of a texture. We will talk more about this phenomenon when talking about bag of words models for object recognition. However, it is interesting within the framework of texture perception as allows experiencing a percept of a collection of features. Even though the image contains well formed objects, when looking at them on the periphery of our visual field we can not see well separated objects, instead we only see the mixture of their features. Crowding turns a set of objects in the periphery into a soup of features, into something that *feels* like a texture.

It is important to note that crowding effects are not due to the poor resolution of the periphery. Crowding takes place at a resolution clearly away from the limits of the retinal sampling.

8.3 Representation of textures

Let's first review some of the theories on texture representation. How is that a texture region is processed so that we can detect the boundary between two textured regions? one challenge is that textures contain lots of internal boundaries and are composed by elements that are not perfectly identical. The delineation of the precise boundaries between texture regions will be described in more detail in the next lecture when we study the problem of image segmentation. In this lecture we will focus on the type of image representation needed to discriminate between different texture regions.

As textures are perceptual, a successful texture representation should:

- discriminate between regions of different textures as humans do,
- account for the perceptual similarity between textures by humans,
- explain why some textures can not be discriminated from each other.

8.3.1 Textons

Textons were introduced by Bela Julesz (1981) as a way of representing textures. In this view, textures are formed by simple elementary components called textons.

Not all properties from the textons are analyzed. Julesz argued that the characteristics of textons that are represented are: junctions, terminations, orientation. Other properties are not encoded such as precise phase information (the exact location of the textons is not important). In particular, he argued that only first-order and second-order statistics of the textons had perceptual significance. However, most of the studies were perform on simple binary texture patterns.

8.3.2 Filter banks

As pointed out by Bergen and Adelson (1988), one limitation of the textons approach to represent textures is that it relies on the detection of intermediate image structures. The detection of junctions and terminations that can be very challenging and there are no reliable detectors for those image features yet.

They proposed a simpler mechanism able to account for the discrimination of texture regions that did not rely on solving an intermediate, apparently *harder*, task.

They showed that by using simple filters with a center-surround form (like a laplacian filter) followed by a non-linearity it is possible to explain the easiness and difficulty to separate to some pairs of textures. They suggested that using filter banks followed of a rectifying non-linearity provides a simple mechanism able to discriminate among different types of textures. This simple approach does not requires intermediate decisions (like detecting where corners are), and is compatible with mechanisms believed to be part of early visual areas of the brain.

Malik and Perona (1990) proposed a full model of texture analysis based on this idea. The approach was able to deal with gray-scale real images, and was able to explain a number of perceptual phenomenon.

8.4 Texture analysis and synthesis

The textures we will describe in this lecture can be described as stationary stochastic processes. In this lecture we will study two big families of texture models.

There are two tasks that will be on interest:

- Texture analysis: given a texture the goal is to build a representation of the texture
- Texture synthesis: the goal is, given an example of a texture, to generate new instances that look as belonging to the same texture.

8.4.1 Statistical models of textures

In these models, a texture is defined by a distribution $P(\mathbf{I})$, where $\mathbf{I}$ belongs to a particular texture type. The goal is to build the density P so that samples from this density look like all being different instances of the same texture.

There are several models that belong to this family. In this lecture we will describe the method proposed by Heeger and Bergen (1994). Their approach is inspired by the following observation: "If matching the averaged squared filter values is a good way to match a given texture, then maybe matching the entire marginal distribution (eg, the histogram) of a filters response would be even better".

Their algorithm has to components:

- Texture analysis: The texture is represented by the marginal distributions of the outputs of a filter bank (they use the steerable pyramid).
- Texture synthesis: a new sample from the texture can be generated by sampling new filter outputs using the marginals and then reconstructing a new image.

The algorithm proposed by Heeger and Bergen is an iterative approach.

A texture image is first decomposed using the steerable pyramid. The texture is represented by the histogram of the image and the histogram of the outputs of all the subbands (including the low-pass and high-pass residuals) from the steerable pyramid decomposition.

In order to generate new samples of the texture, we start with an image of white noise (the size of the image should match the size of the image that we want to generate). First we force the histogram of the noise to match the histogram of the texture image. Then, the result is decomposed using the steerable pyramid. After this, all the subbands are modified so match the histograms of the subbands outputs computed from the texture. The modified subbands are recombined to reconstruct and image. This process needs to be iterated several times using as input the output. This is necessary as the decomposition is non-orthogonal an modifying the histograms of the subbands does not guarantee that the subbands of the reconstructed image matches the desired histograms.

This method, despite its simplicity, motivated numerous new models of increasing complexity and able to produce higher quality textures. One important model is the one proposed by Portilla and Simoncelli (2000) that incorporates richer statistical descriptors.

8.4.2 Non-parametric models

Efros and Leung (1999) proposed a very different approach for texture analysis and synthesis. The process uses as representation the input texture sample itself. In order to do synthesis the algorithm starts with a seed. It proceeds one pixel at a time by looking in the original texture for neighborhoods in the input texture similar to the neighborhood of the pixel to be filled. It collects multiple candidates and it randomly selects one as the value that will be used to render the new pixel. Then it moves to another non-rendered pixel and repeats the same procedure.

Pseudo code for the algorithm can be found here:

http://graphics.cs.cmu.edu/people/efros/research/NPS/alg.html

This method for texture synthesis is able to generate high quality textures

A hybrid method between that uses non-parametric models combined with filter banks was proposed by jeremy De Bonet (1997): In a two-phase process, the input texture is first analyzed by measuring the joint occurrence of texture discrimination features at multiple resolutions. In the second phase, a new texture is synthesized by sampling successive spatial frequency bands from the input texture, conditioned on the similar joint occurrence of features at lower spatial frequencies.