
University of Colorado
Boulder

Deep Learning Applications for Computer Vision

Lecture 9: Texture

Texture

- Difficult to define

Zebra, Leopard

- distinctive patterns
- repeated elements
- Textons
  - → spots
  - → stripes
  - → blades of grass
  - → strands of hair

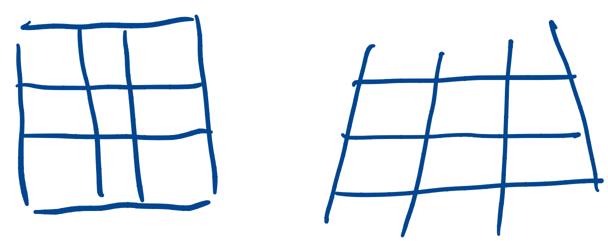


FIGURE 6 (Forsyth and Ponce, *Comp Vision - A modern Approach*)

Filters as Templates

These filters look poised to act as a template and discover vertical and horizontal edges.

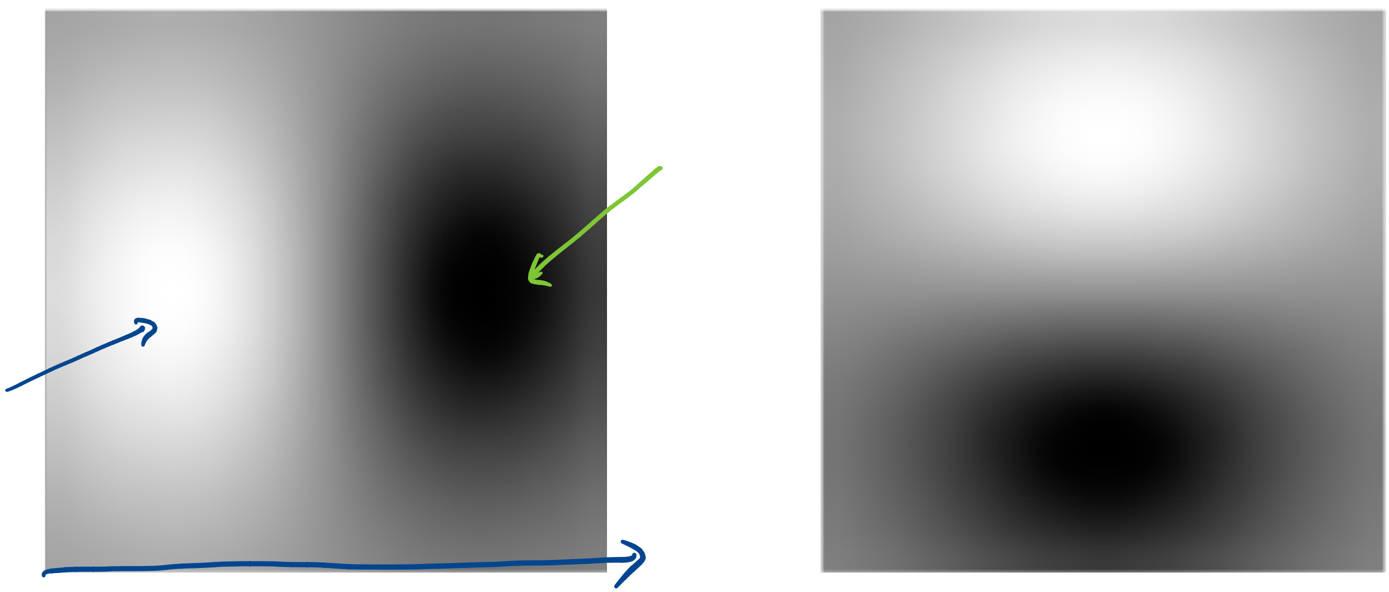
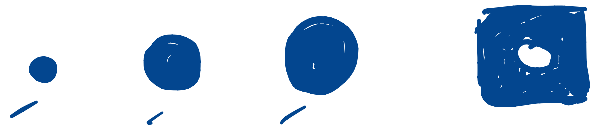
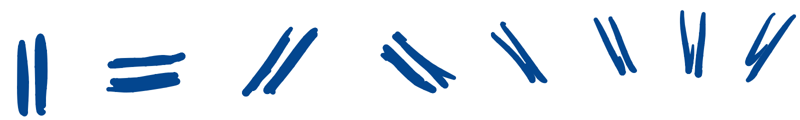


FIGURE 4.15: Filter kernels look like the effects they are intended to detect. On the left, a smoothed derivative of Gaussian filter that looks for large changes in the x-direction (such as a dark blob next to a light blob); on the right, a smoothed derivative of Gaussian filter that looks for large changes in the y-direction.

Filters for detecting Texture?

- Can we detect repeated elements (*textons*) by using template filters?
- Each filter is a detector for a sub-element

- What would the filters look like?
  - edges 
  - spots 
  - multiple scales
  - multiple orientations 

Filters for detecting Texture?

- Can we detect repeated elements (*textons*) by using template filters?
- Each filter is a detector for a sub-element

- What would the filters look like?
  – bar filters (for edges and stripes)
  – spot filters (for .. spots or blobs)
  – multiple scales (for all) and multiple orientations (for edges and stripes)

Spots and Bars

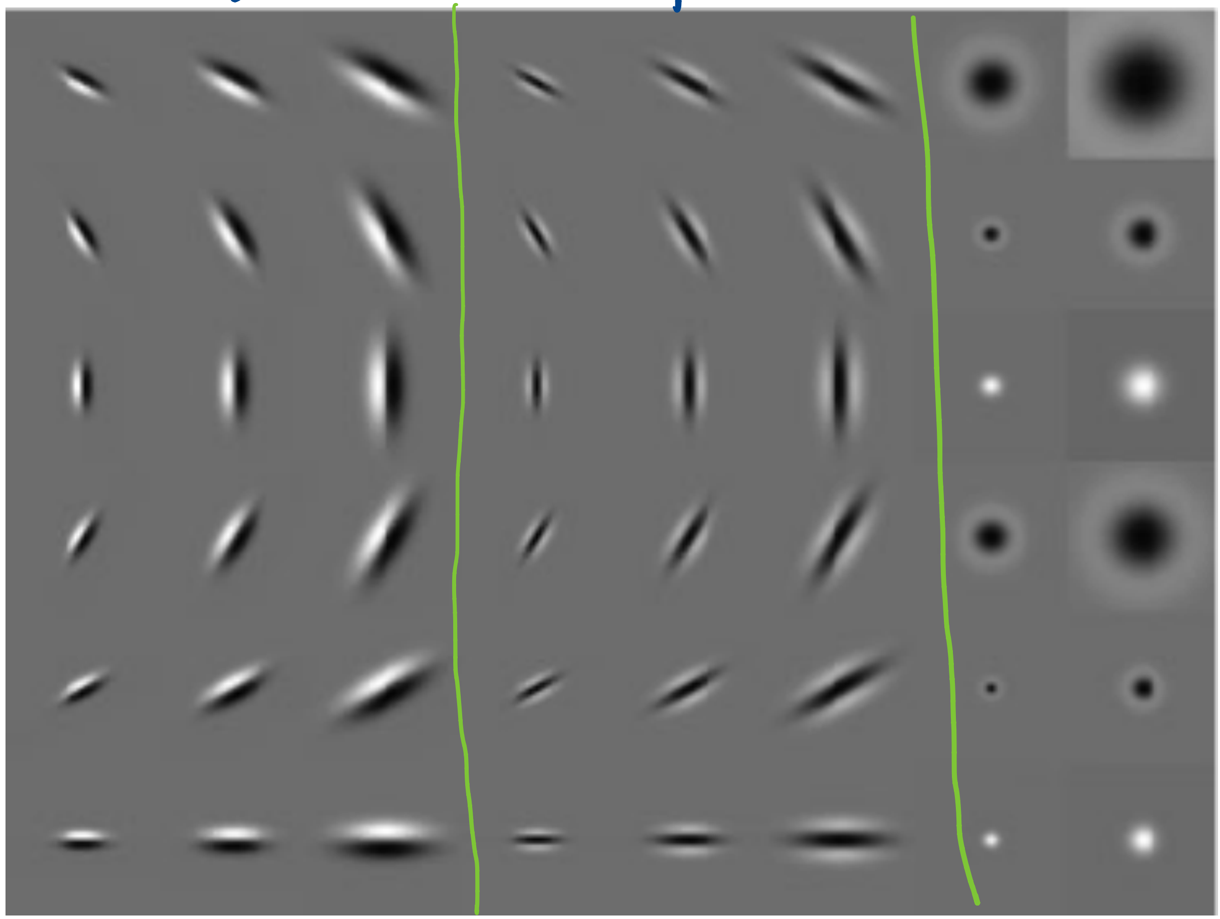
Edges stripes Blobs / spots

Filter examples:

FIGURE 6.4 (Left)

- 48 filters
- oriented filters

gray → 0
white → +
black → −



T. Leung, J. Malik. *Representing and recognizing the visual appearance of materials using three-dimensional textons*. IJCV, 2001.

Texture maps

Texture as a point in the image:

- A summary of all the texture maps in that neighborhood
- Detecting repeated textons which might be slightly different

Strategies:

- Each point represented as a vector of filter responses
- Clustering

↳ distance measures

1. Use **Training Data** to build a Learning Dictionary
2. Extract patches from a new images and see which cluster centers they are closest to

“Learning” Texture - Statistics

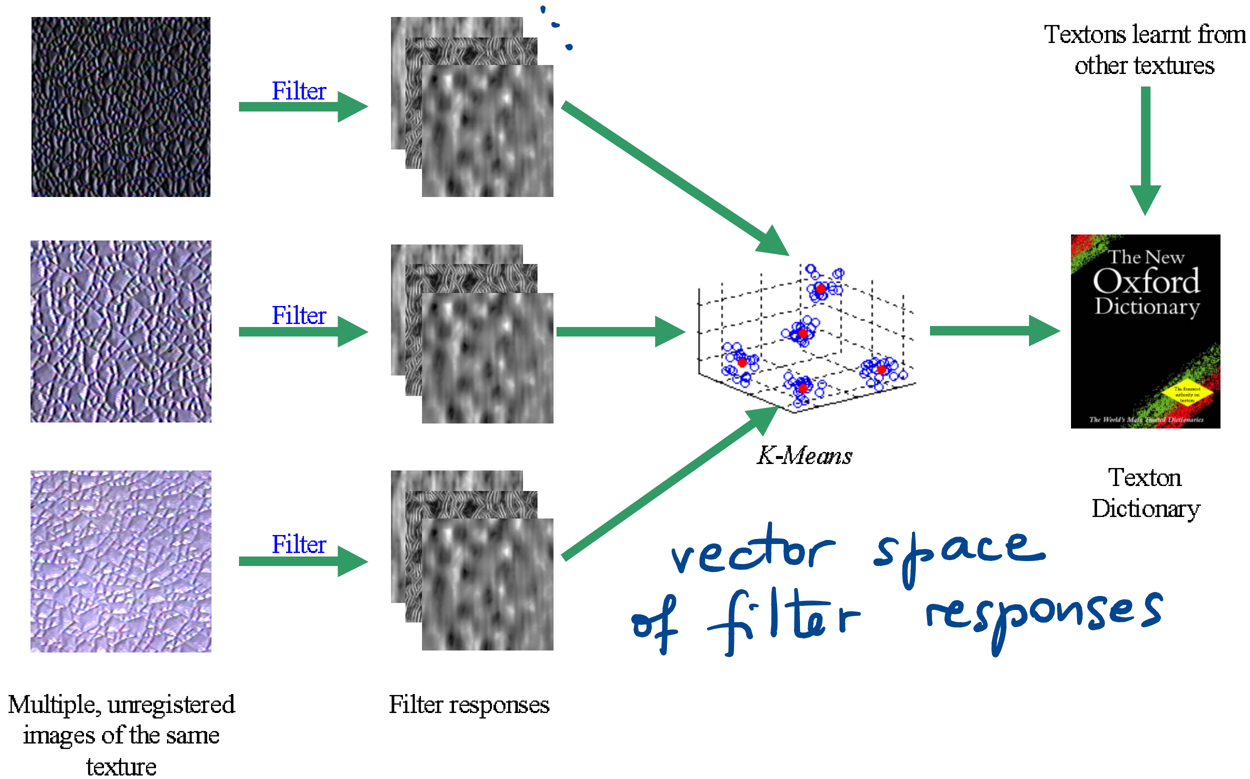


Figure 4. **Learning stage I: Generating the texton dictionary**. Multiple, unregistered images from the training set of a particular texture class are convolved with a filter bank. The resultant filter responses are aggregated and clustered into textons using the *K-Means* algorithm. Textons from different texture classes are combined to form the texton dictionary.

Varma, Zisserman, *A Statistical Approach to Texture Classification from Single Images* , IJCV 2005

“Learning” Texture - Statistics

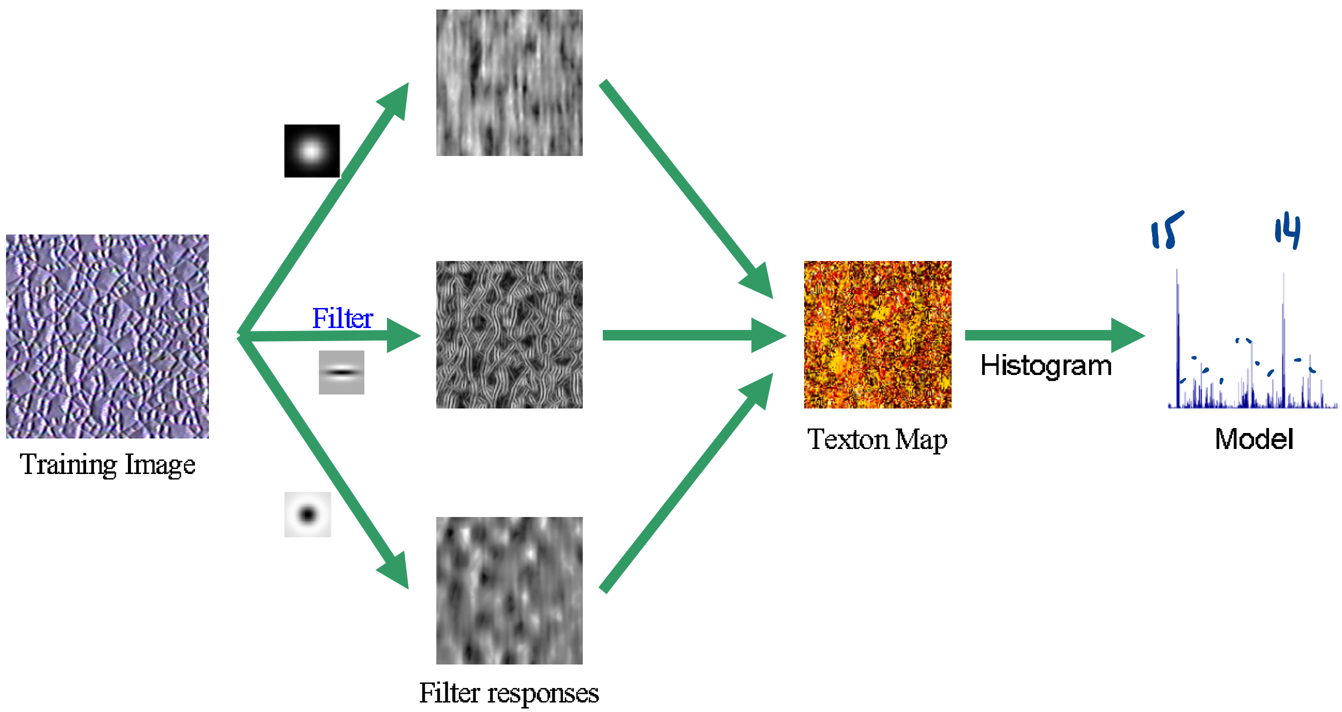


Figure 5. **Learning stage II: Model generation**. Given a training image, its corresponding model is generated by first convolving it with a filter bank and then labelling each filter response with the texton which lies closest to it in filter response space. The histogram of textons, i.e. the frequency with which each texton occurs in the labelling, forms the model corresponding to the training image.

Varma, Zisserman, *A Statistical Approach to Texture Classification from Single Images* , IJCV 2005

New Texture Classification

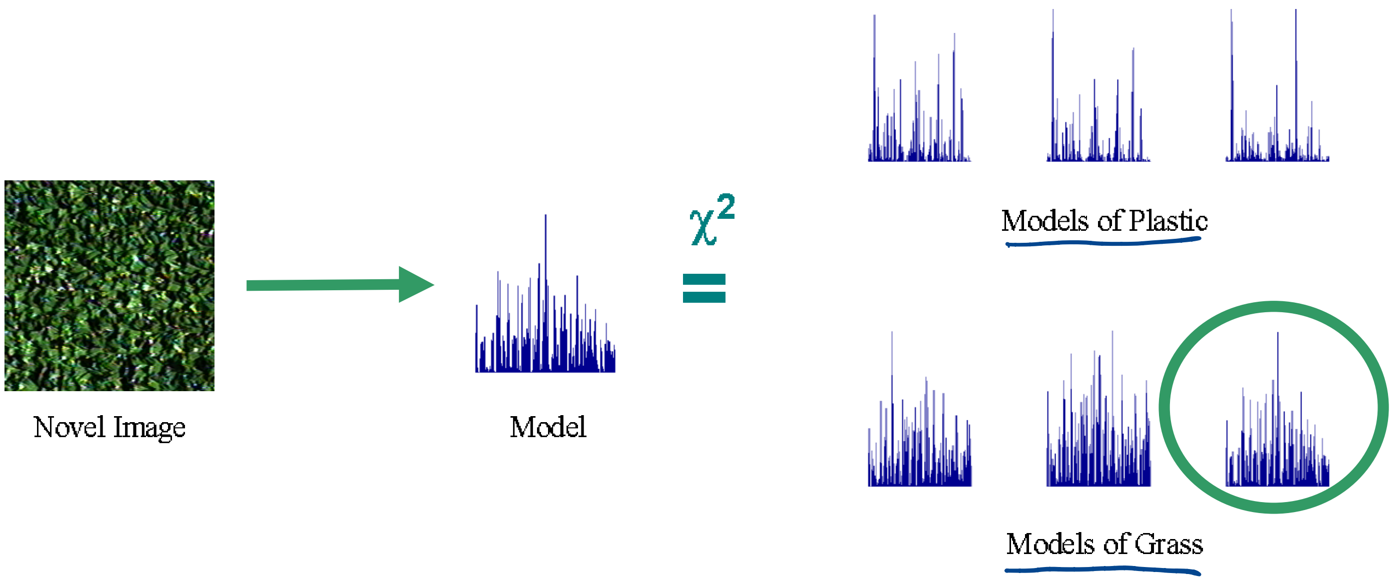


Figure 6. **Classification stage.** A novel image is classified by forming its histogram and then using a nearest neighbour classifier to pick the closest model to it (in the χ^2 sense). The novel image is declared as belonging to the texture class of the closest model.

Varma, Zisserman, *A Statistical Approach to Texture Classification from Single Images* , IJCV 2005

Classifying Images

- Detecting texture is challenging – all possibilities
- Same class object - made of different materials
- Classifying images replies on choosing good image features

Goal: build features that *expose variations between classes*, and *suppress within-class variation*

Liu et all, Exploring Features in a Bayesian Framework for Material Recognition, IEEE-TPAMI, 2010