



S A I R

Spatial AI & Robotics Lab

# CSE 473/573-A

## L17: TEXTURE

Chen Wang

Spatial AI & Robotics Lab

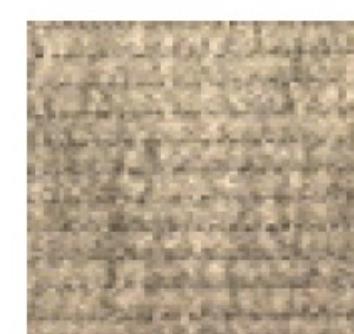
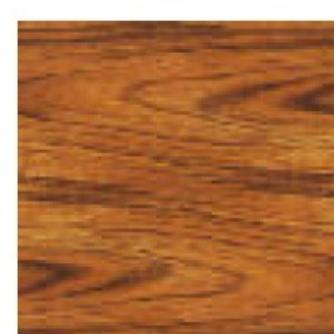
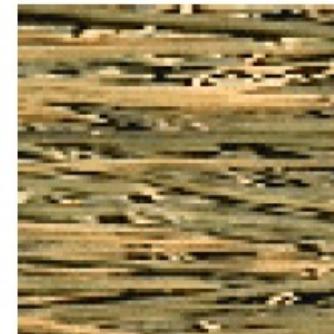
Department of Computer Science and Engineering



University at Buffalo The State University of New York

# Texture

- What defines a texture?



# What is Texture?

---

- No Formal Definition
  - There will be **significant variation** in intensity levels between pixels
  - these variations perform **repetitive** patterns- homogeneous at some scale
  - **local statistics** are constant and slowly varying
- Human visual systems perceived textures as **homogeneous regions** even though they don't have the same intensity

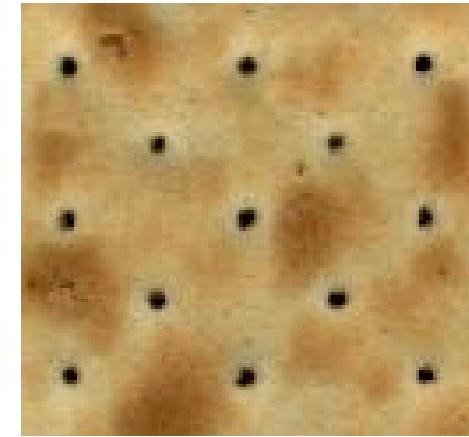
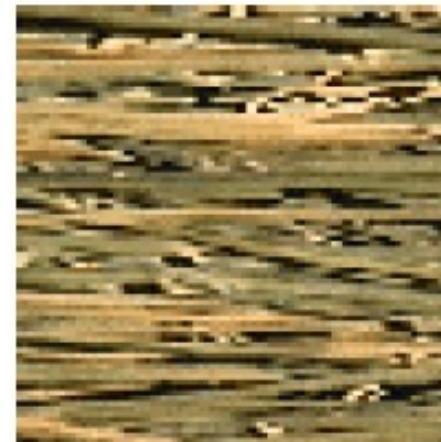
# Regular (Repetitive) patterns

- We interpret this first image as a brick wall



# Random patterns

---



# Texture

---

- Is a property of a "group of pixels" or Area
  - a single pixel does not have texture
- It is **scale dependent**
  - at different scales textures have different properties
- It contains many possibly countless **primitive objects**
- It involves the **spatial distribution** of intensities
  - 2D histograms
  - Co-occurrence matrices

# Scale

---

- Scale is important – consider sand

- Close up

- “small rocks, sharp edges”
- “rough looking surface”
- “smoother”



- Far Away

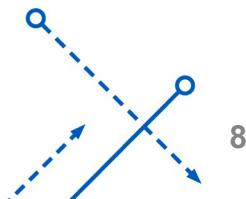
- “one object
- ⇒ brown/tan color”



# How would you describe a Texture?

---

- Coarseness
  - Roughness
  - Direction
  - Frequency
  - Uniformity
  - Density
- 
- How do you describe
    - dog fur, cat fur, wood grain, or cloth?



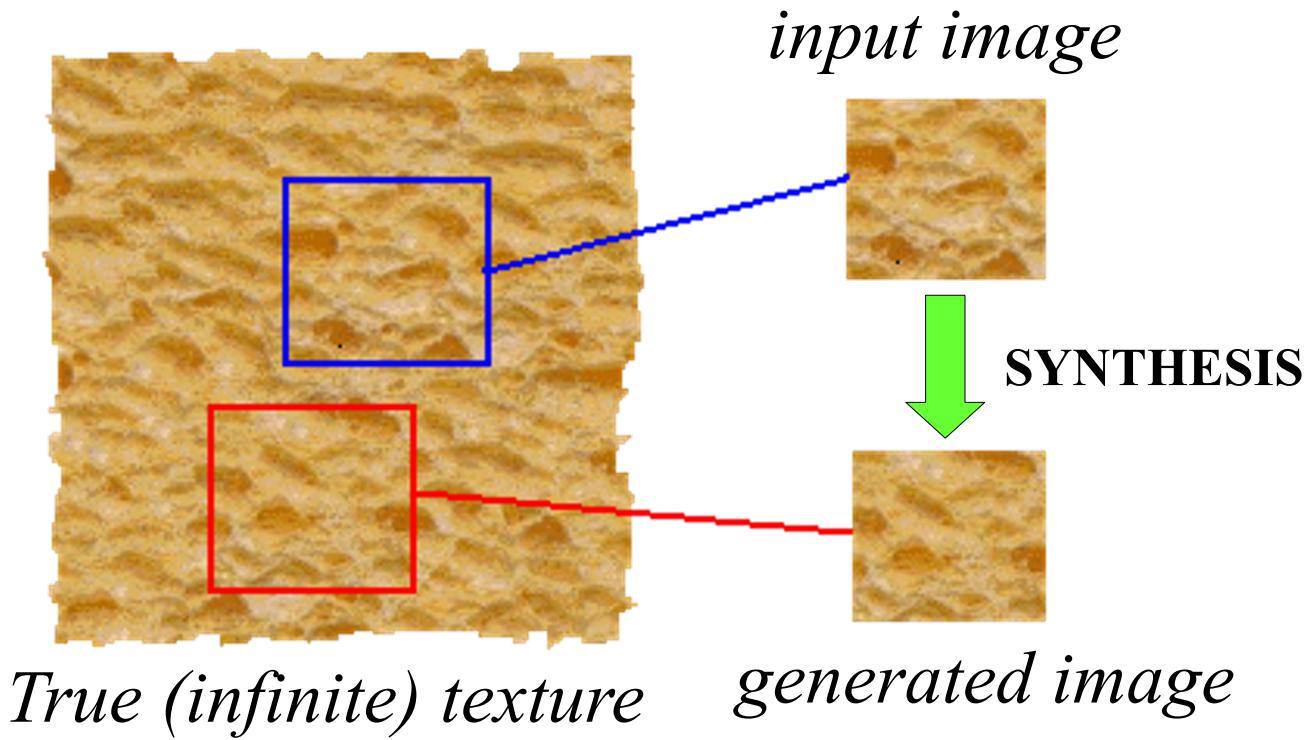
# Textures has different instances.

- All images are different instances of the same texture
- We can differentiate between them, but they seem generated by the same process.



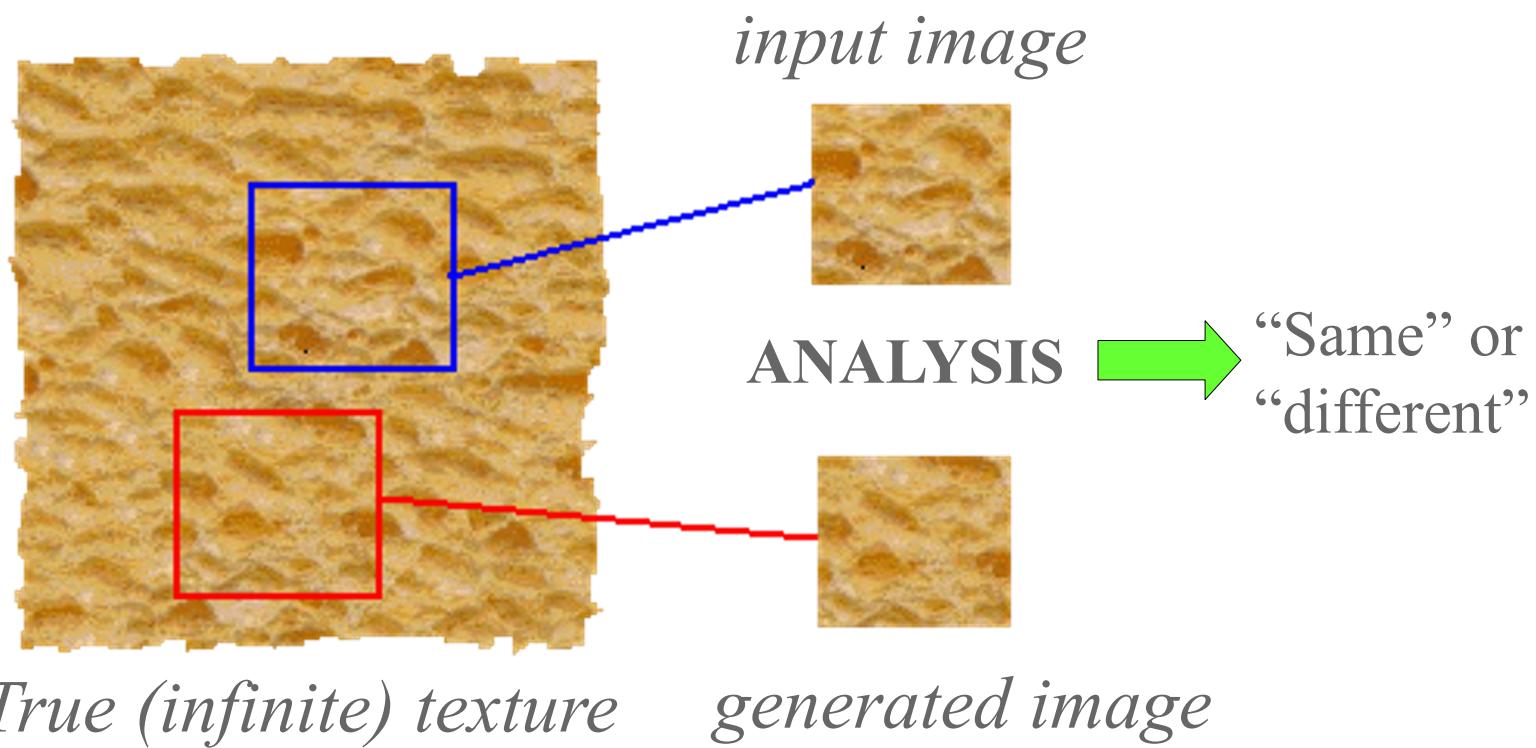
# The Goal of Texture Synthesis

- Given a finite sample of some texture, the goal is to synthesize other samples from that same texture
- The sample needs to be "large enough"



# The Goal of Texture Analysis

- Compare textures and decide if they're made of the same “stuff”.



# Scale: objects vs. texture

---

- The same thing can occur as texture or an object
  - depending on the **scale** we are considering.



# Why analyze texture?

---

Importance to perception:

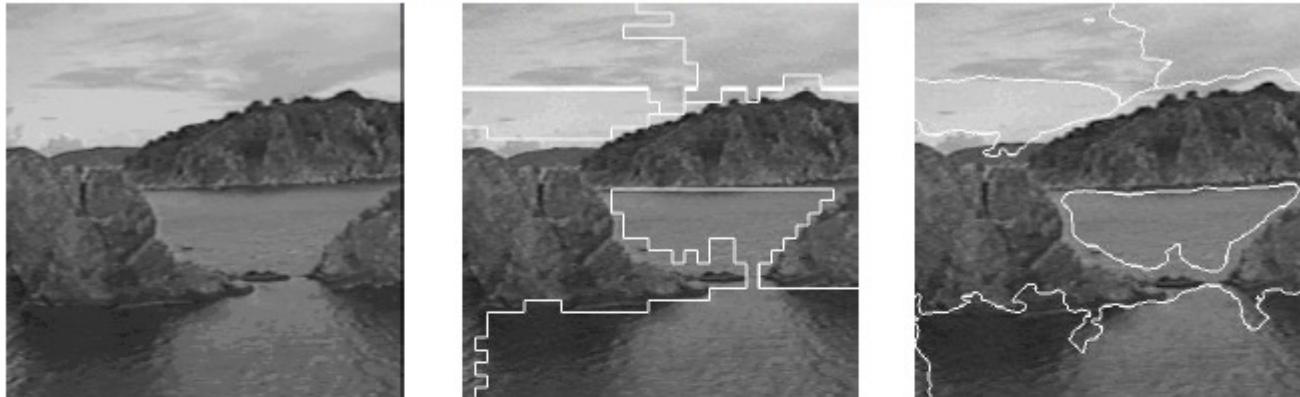
- Often indicative of a material's **properties**
- Can be important **appearance cue**, especially if shape is similar across objects
- Distinguish between shape, boundaries, and texture

Technically:

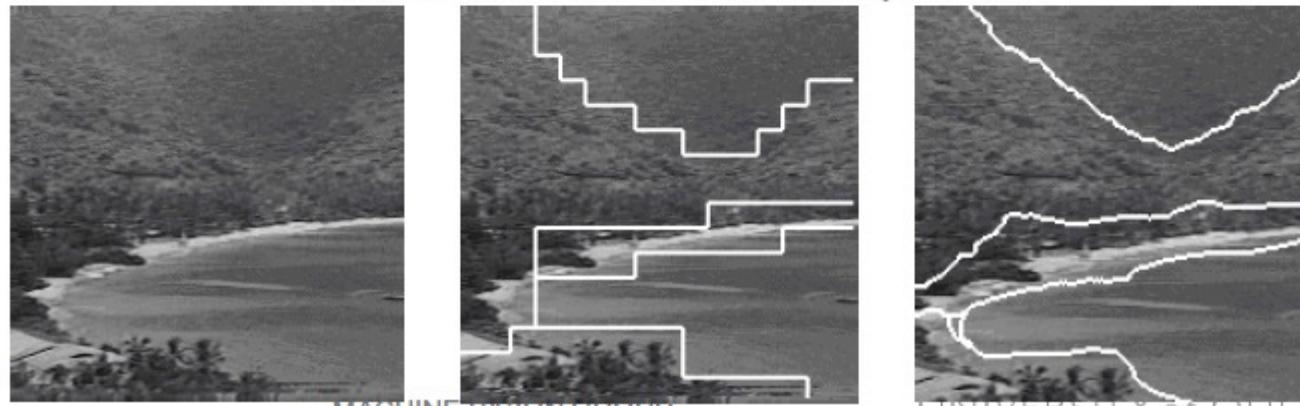
- Representation-wise, we want a feature one step above "building blocks" of filters, edges.

# Why analyze texture?

Natural scene #1: 384x384 pixels



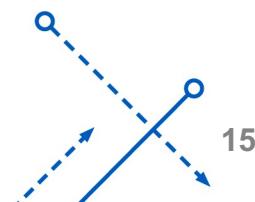
Natural scene #2: 192x192 pixels



# Texture-related tasks

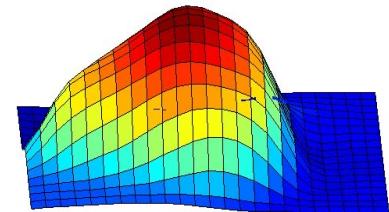
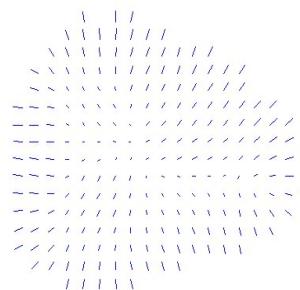
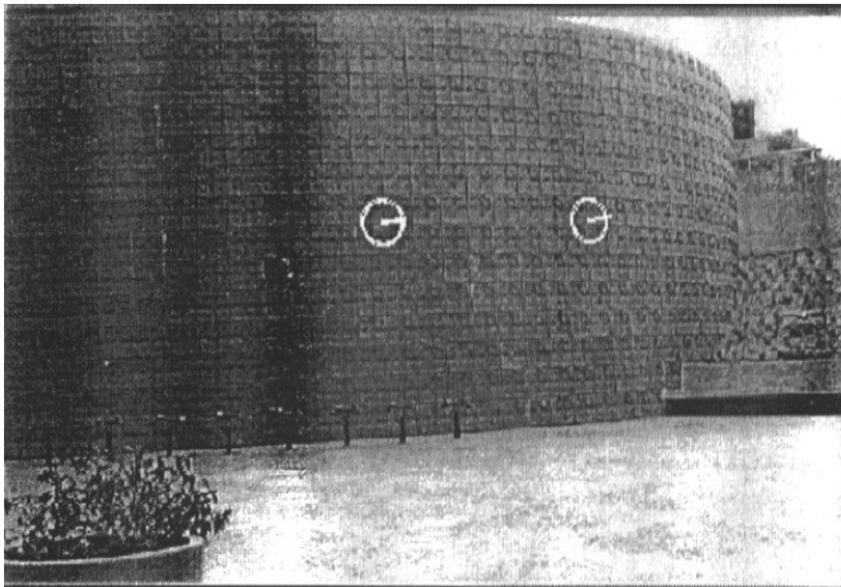
---

- **Shape from texture**
  - Estimate surface orientation or shape from image texture
- **Segmentation/classification** from texture cues
  - Analyze, represent texture
  - Group image regions with consistent texture
- **Synthesis**
  - Generate new texture patches/images given some examples



# Shape from texture

- Use deformation of texture from point to point to estimate surface shape

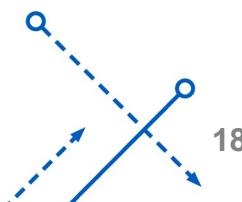




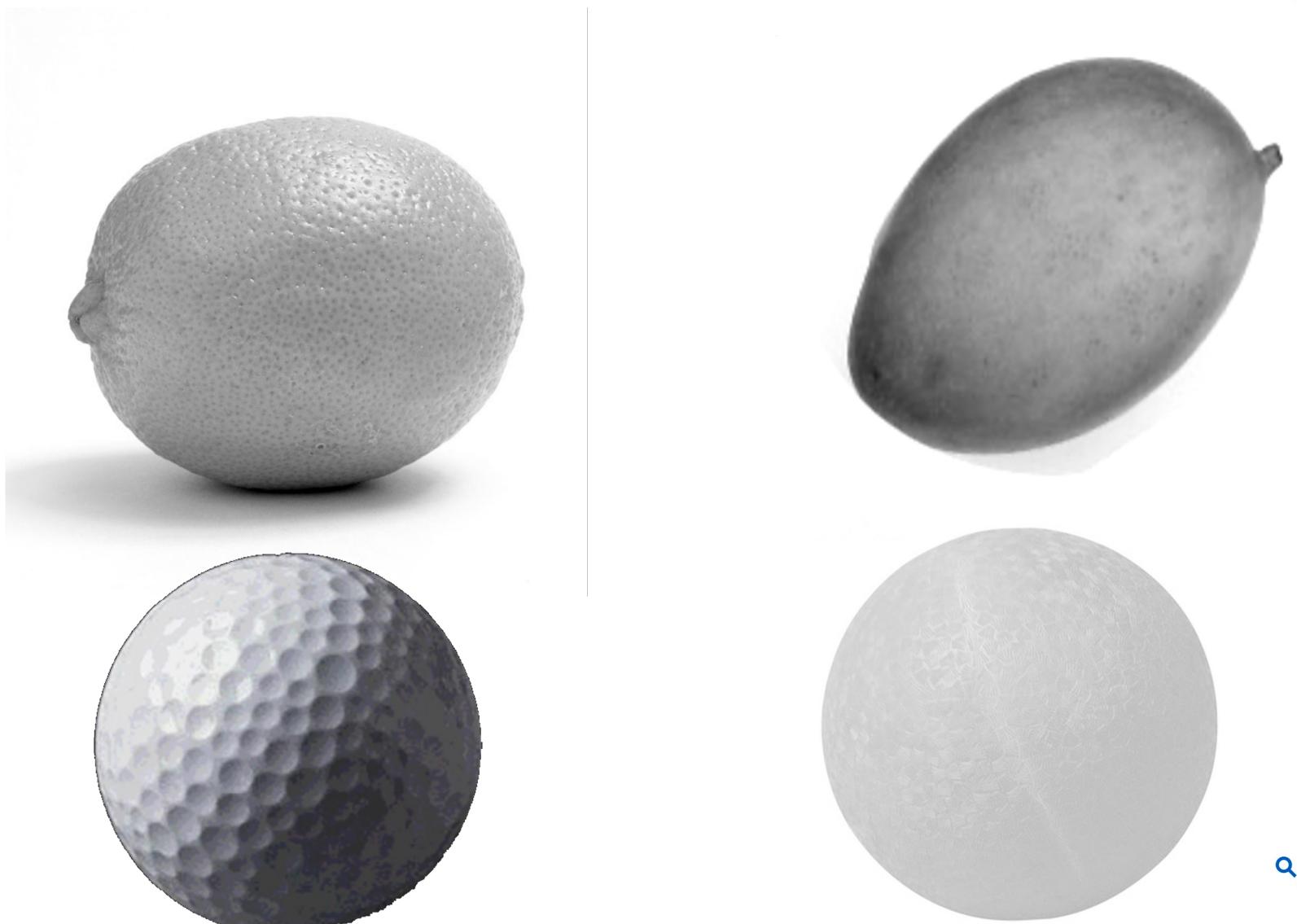
# Texture-related tasks

---

- **Shape from texture**
  - Estimate surface orientation or shape from image texture
- **Segmentation/classification** from texture cues
  - Analyze, represent texture
  - Group image regions with consistent texture
- **Synthesis**
  - Generate new texture patches/images given some examples



# Texture-related tasks



# Texture-related tasks

---



# Color vs. texture

- Recall: These looked very similar in terms of their color distributions (when our features were R-G-B)
- But how would their texture distributions compare?

query



query



# Psychophysics of texture

---

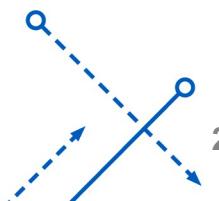
- Some textures distinguishable with *pre-attentive* perception— without scrutiny, eye movements [Julesz 1975]

Watch the next screen very carefully.

Are the **left** and **right** sides:

Same or different?

Pre-attentive processing is the subconscious accumulation of information from the environment. All available information is pre-attentively processed.

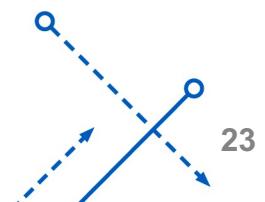
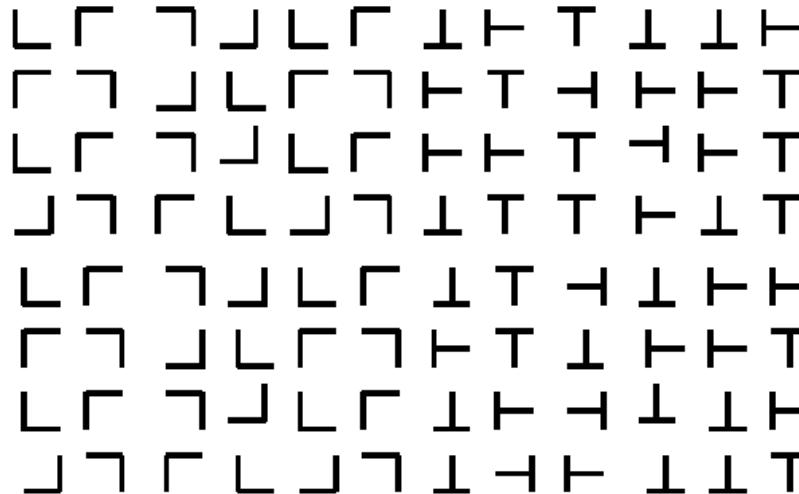


# Ready? First Example

---

**Left**

**Right**

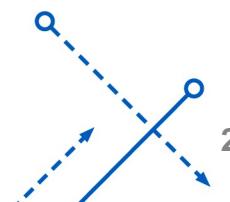
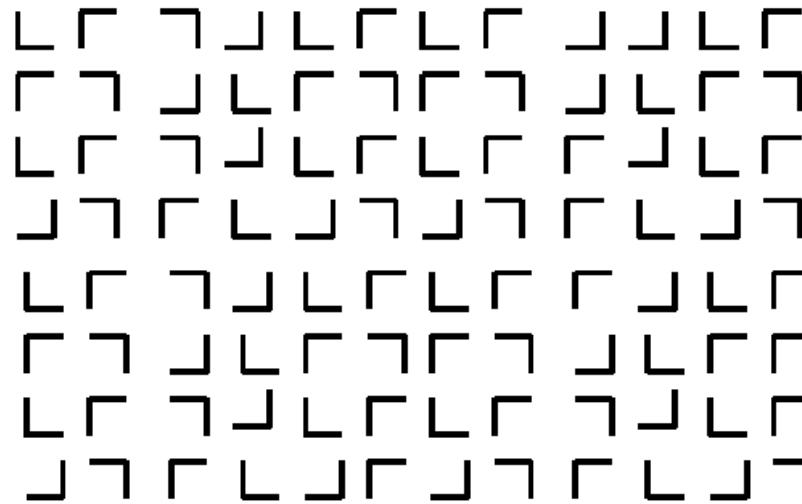


# Ready? Second Example

---

**Left**

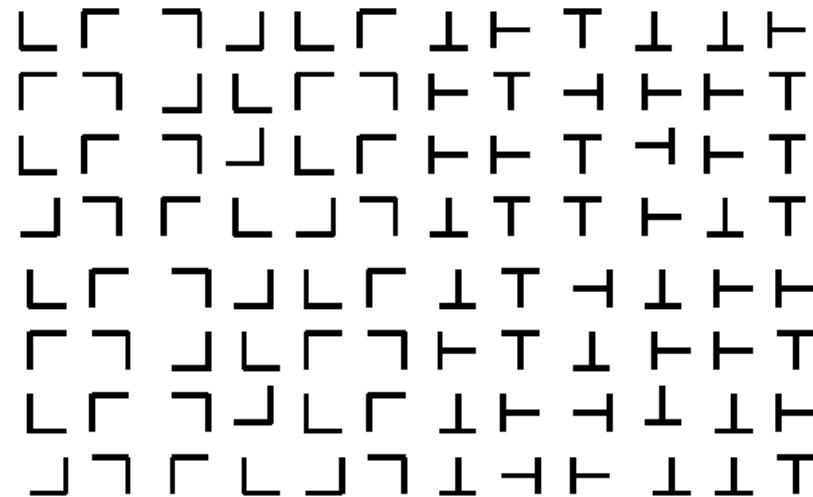
**Right**



# What you have seen

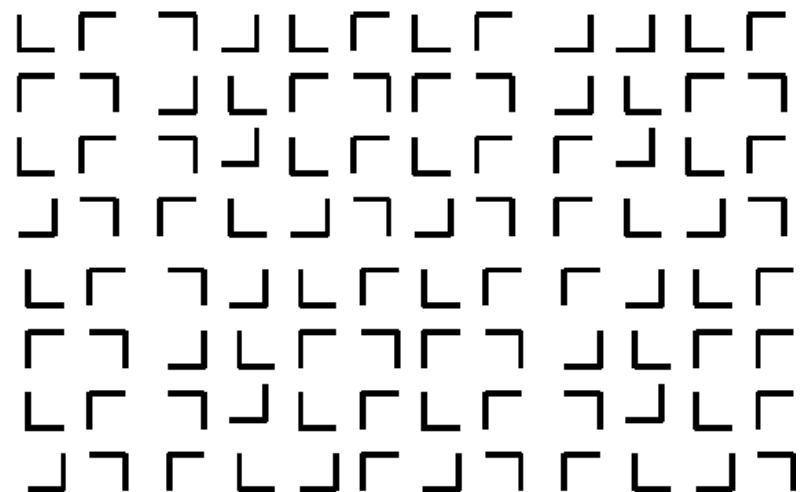
---

**Left**

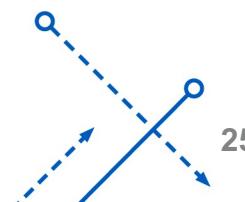


**Right**

**Left**

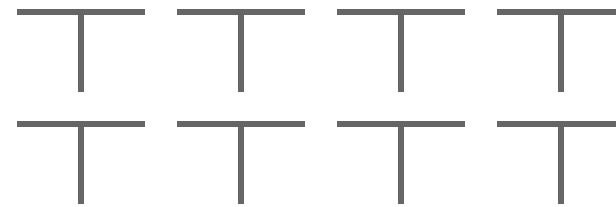
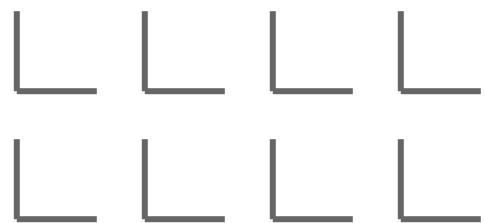


**Right**



# Texture Analysis

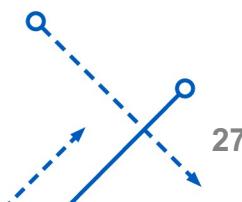
- Analyze the texture in terms of statistical relationships between fundamental texture elements
  - Called “textons”.
- It generally required a human to look at the texture in order to decide what those fundamental units were.



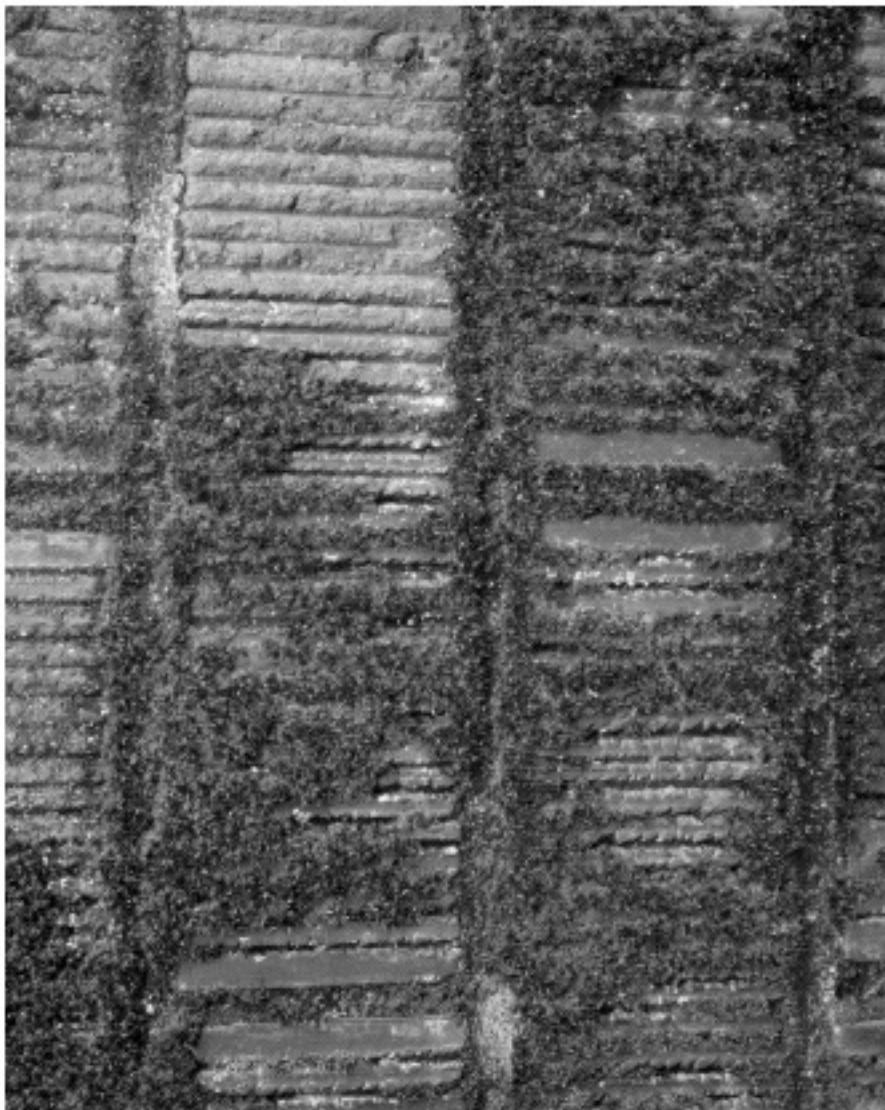
# Texture representation

---

- Textures are made up of repeated local patterns:
  - How to find the patterns
    - Use filters that look like patterns
      - Spots, bars, raw patches...
      - Consider magnitude of response
    - Describe their statistics within each local window
      - Mean, standard deviation, etc.
    - Histogram of “prototypical” feature occurrences



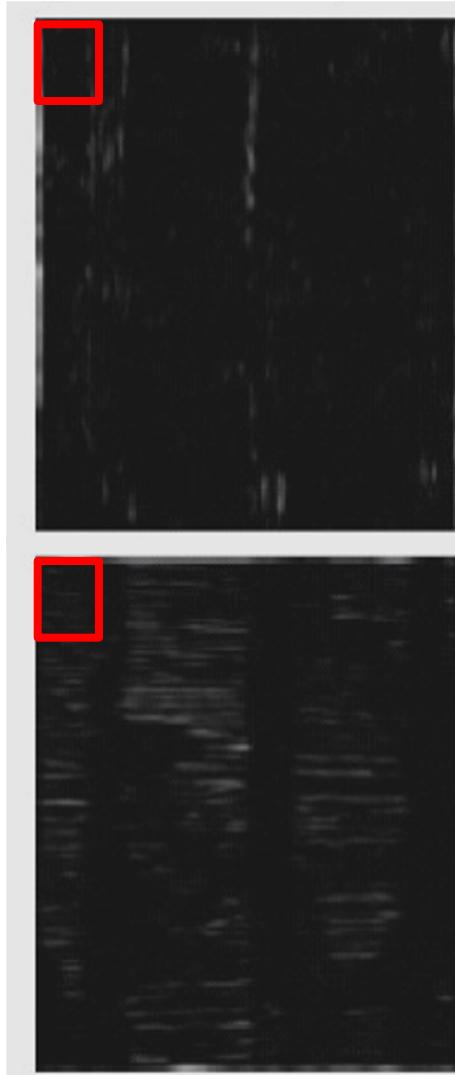
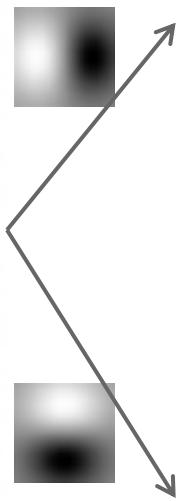
# Texture representation



# Texture representation



# original image



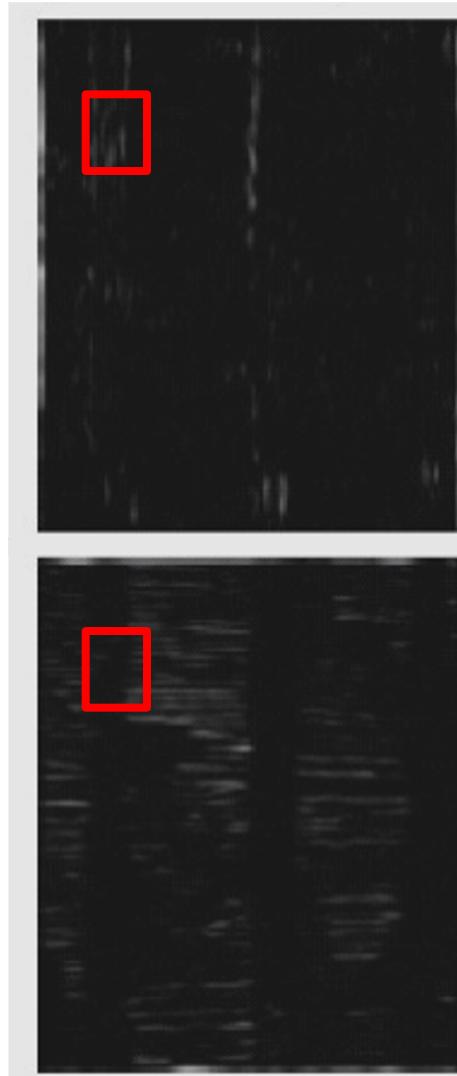
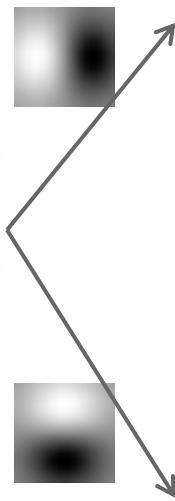
# derivative filter responses, squared

statistics to  
summarize patterns  
in small windows

# Texture representation



original image



derivative filter  
responses, squared

	<u>mean <math>d/dx</math> value</u>	<u>mean <math>d/dy</math> value</u>
Win. #1	4	10
Win.#2	18	7

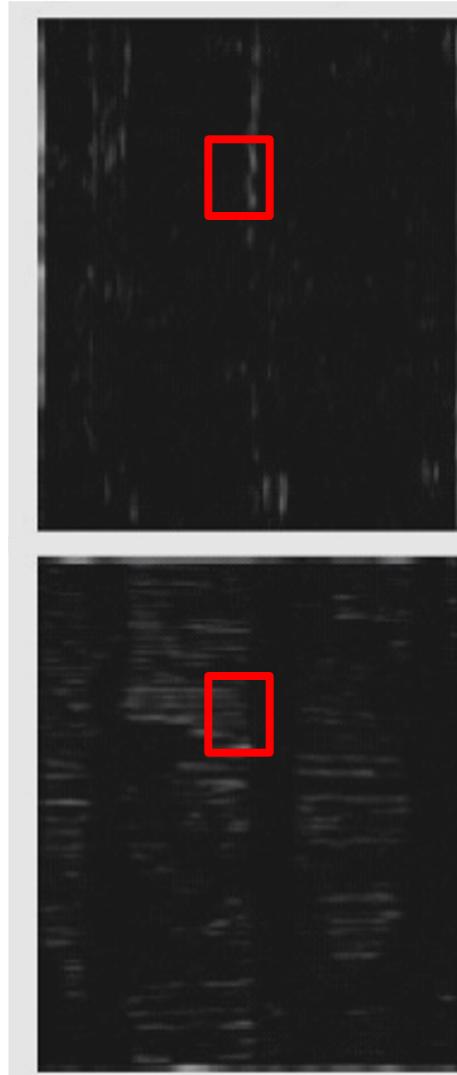
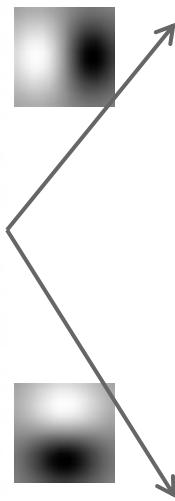
⋮

statistics to  
summarize patterns  
in small windows

# Texture representation



original image



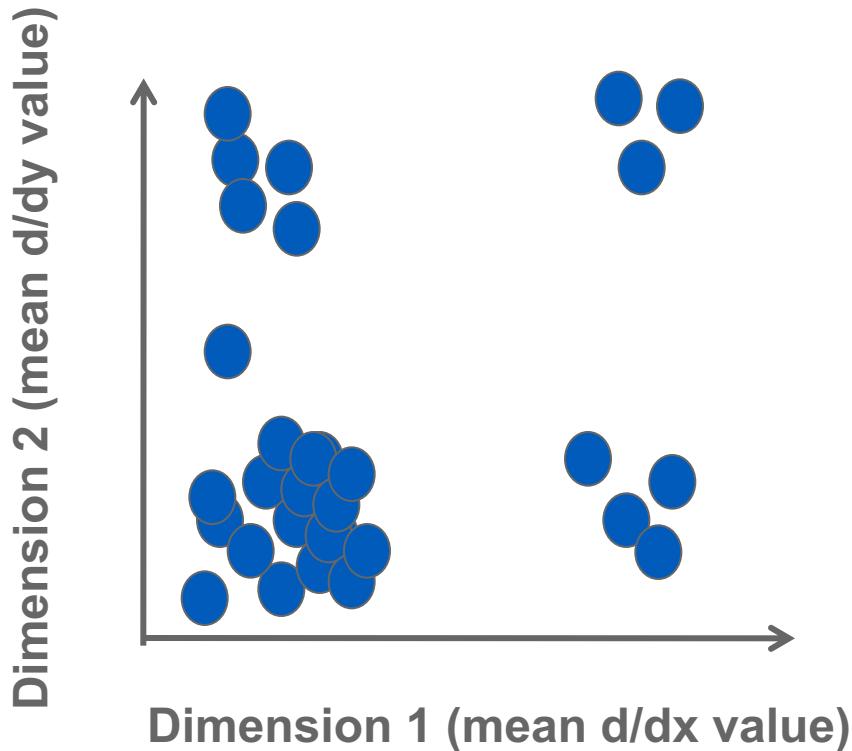
derivative filter  
responses, squared

	<u>mean <math>d/dx</math> value</u>	<u>mean <math>d/dy</math> value</u>
Win. #1	4	10
Win.#2	18	7
:		
Win.#9	20	20

⋮

statistics to  
summarize patterns  
in small windows

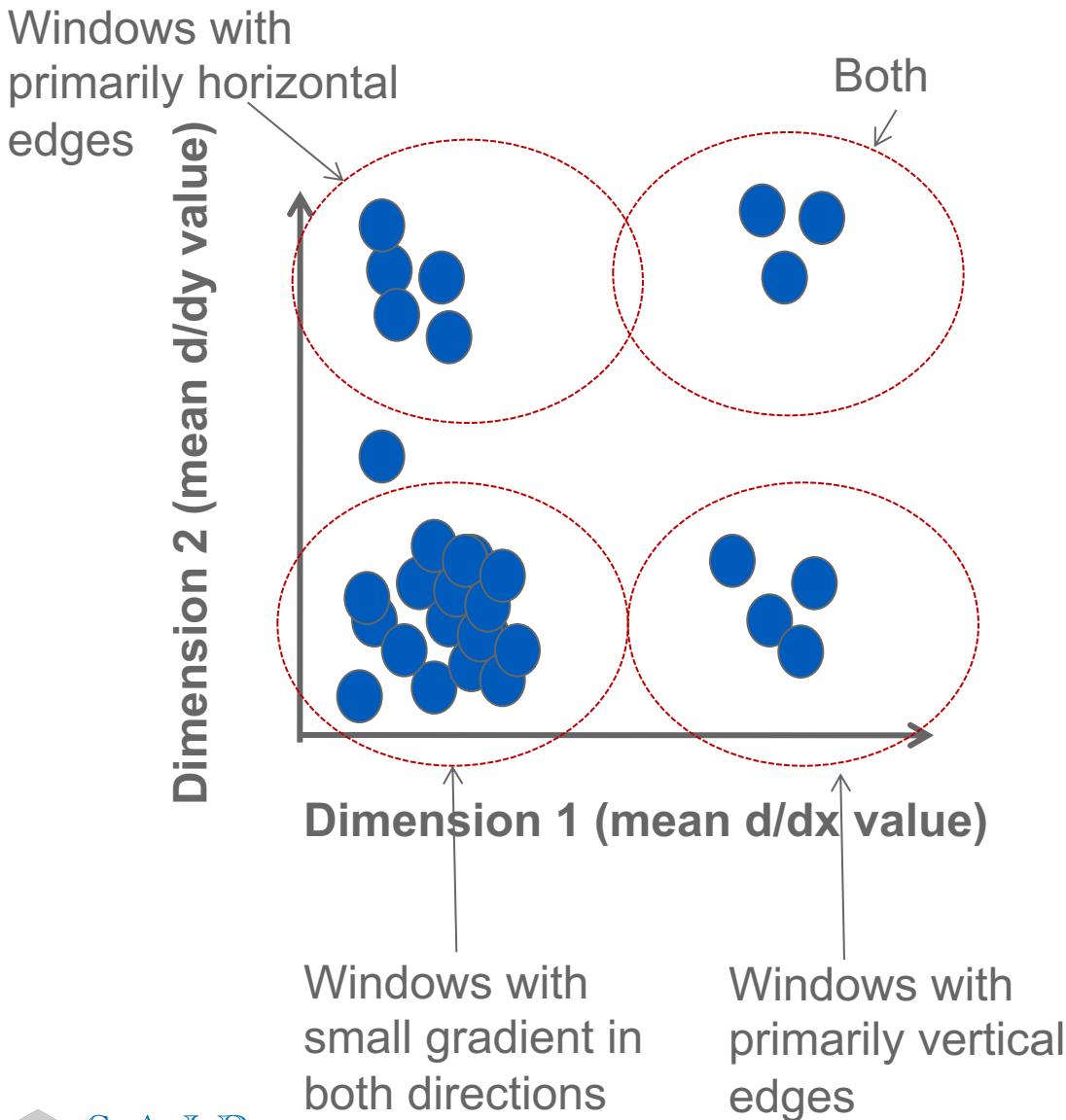
# Texture representation: example



	<u>mean</u> <u>d/dx</u> value	<u>mean</u> <u>d/dy</u> value
Win. #1	4	10
Win.#2	18	7
⋮	⋮	⋮
Win.#9	20	20
⋮	⋮	⋮

statistics to  
summarize patterns  
in small windows

# Texture representation: example



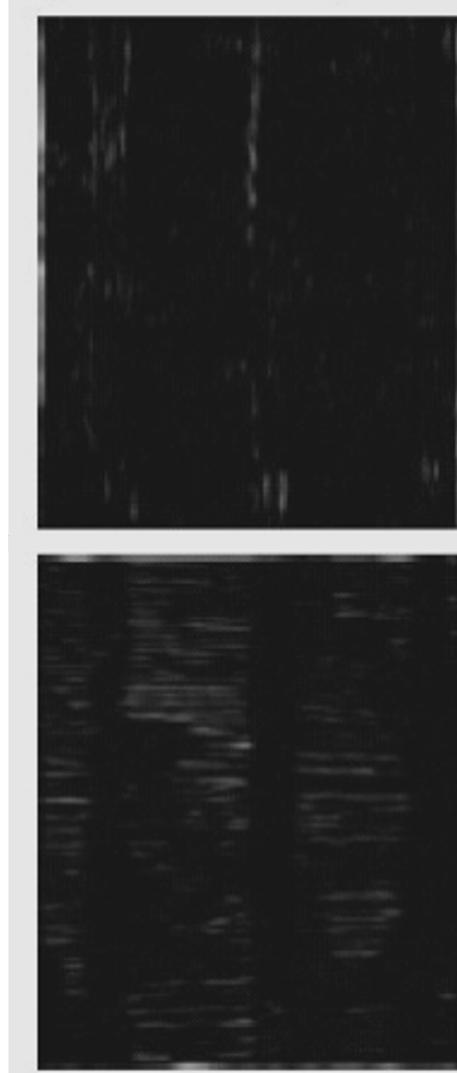
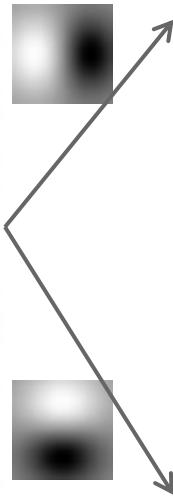
	<u>mean</u> <u><math>d/dx</math></u> <u>value</u>	<u>mean</u> <u><math>d/dy</math></u> <u>value</u>
Win. #1	4	10
Win.#2	18	7
⋮	⋮	⋮
Win.#9	20	20

statistics to  
summarize patterns  
in small windows

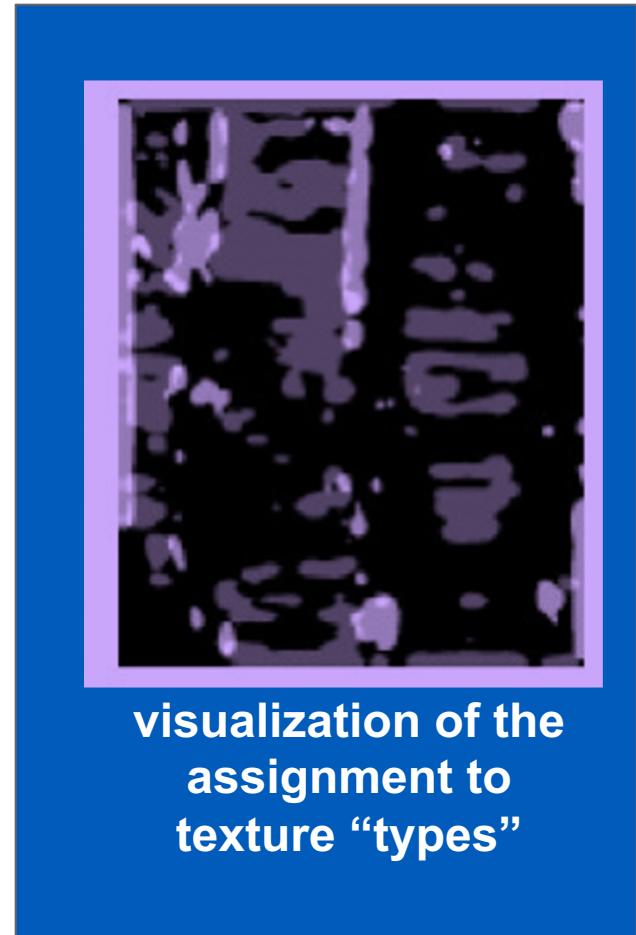
# Texture representation: example



original image

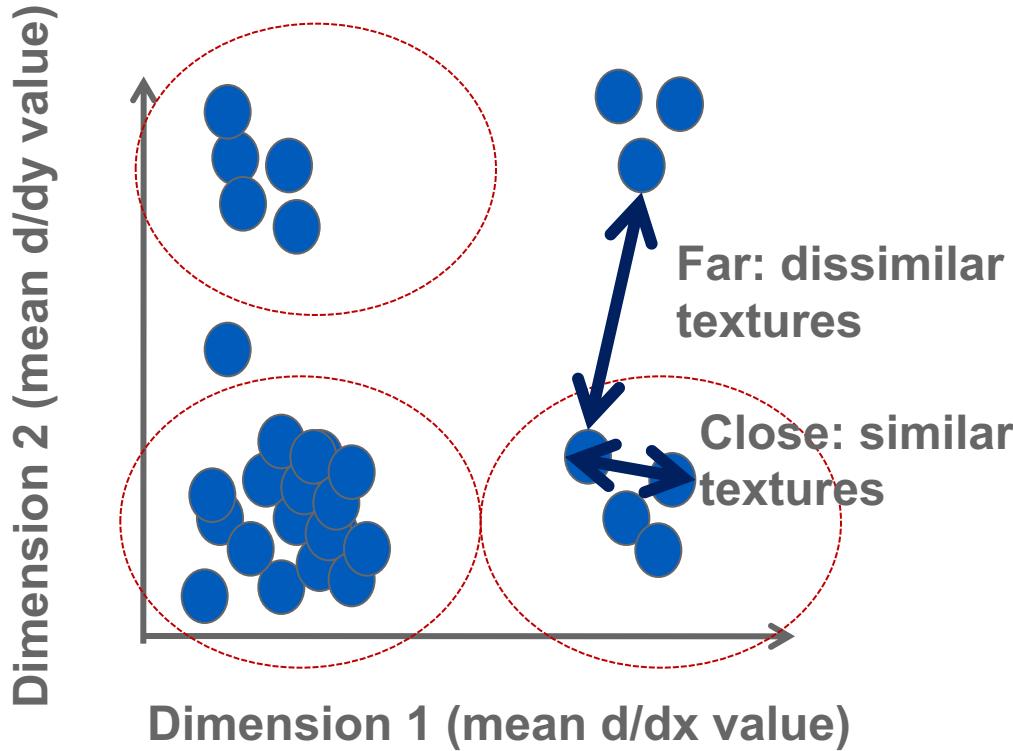


derivative filter  
responses, squared



visualization of the  
assignment to  
texture “types”

# Texture representation: example

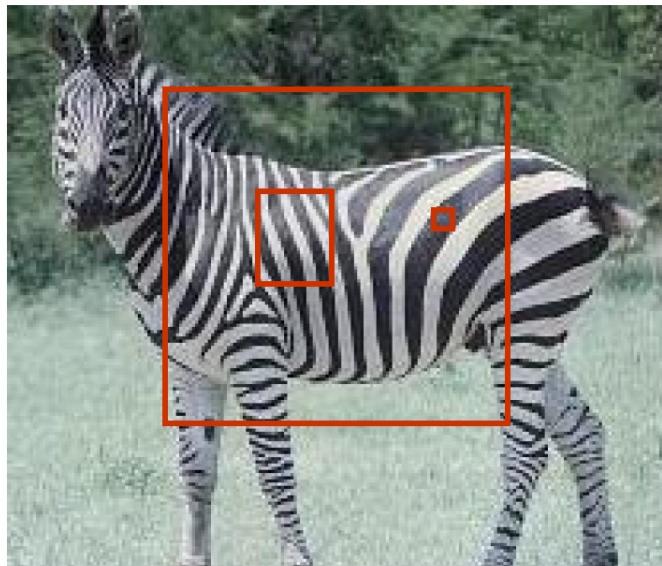


	<u>mean <math>d/dx</math> value</u>	<u>mean <math>d/dy</math> value</u>
Win. #1	4	10
Win.#2	18	7
:		
Win.#9	20	20

statistics to  
summarize patterns  
in small windows

# Texture representation: window scale

- We're assuming we know the relevant window size for which we collect these statistics.



Possible to perform scale selection by looking for window scale where texture description not changing.

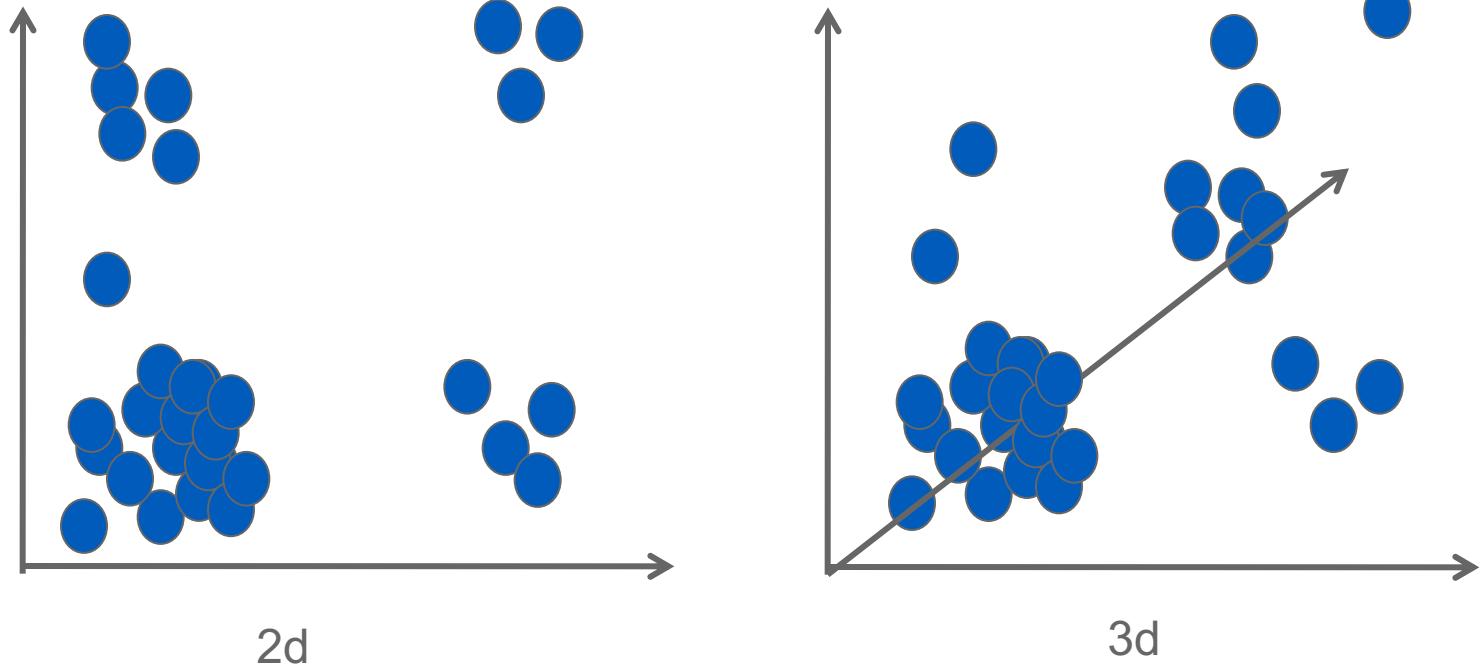
# Filter banks

---

- Our previous example used two filters, and resulted in a 2-dimensional feature vector to describe texture in a window.
  - x and y derivatives revealed something about local structure.
- We can generalize to apply a collection of multiple ( $d$ ) filters: a “filter bank”
  - Recall multi-channel convolution.
  - Then our feature vectors will be  $d$ -dimensional.
    - still can think of nearness, farness in feature space

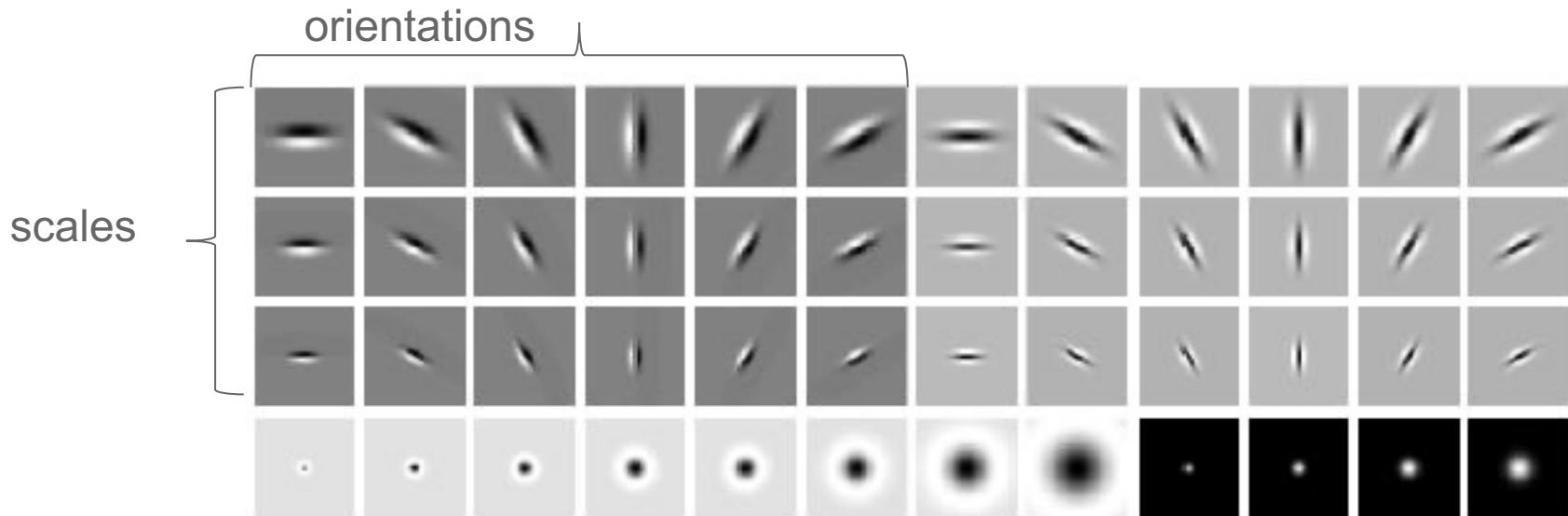


# *d*-dimensional features



# Filter banks

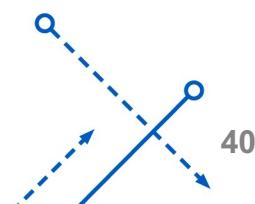
- What filters to put in the bank?
  - Typically we want a combination of scales and orientations, different types of patterns.



# Pixel Neighborhood-based Feature

---

- The most important for texture analysis is to describe the spatial behavior of intensity values in any neighborhood.
- Different methodologies have been proposed.
- **Local binary pattern (LBP)** is one of the most-widely used approach – mainly for face recognition.
- LBP is used for texture analysis too.



# Local Binary Pattern (LBP)

For each PIXEL of an image, a BINARY CODE is produced

$$LBP_{p,r}(N_c) = \sum_{p=0}^{P-1} g(N_p - N_c)2^p$$

where

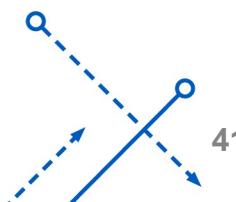
neighborhood pixels ( $N_p$ ) in each block → is thresholded by its center pixel ( $N_c$ )

$p \rightarrow$  sampling points (e.g.,  $p = 0, 1, \dots, 7$  for a  $3 \times 3$  cell, where  $P = 8$ )

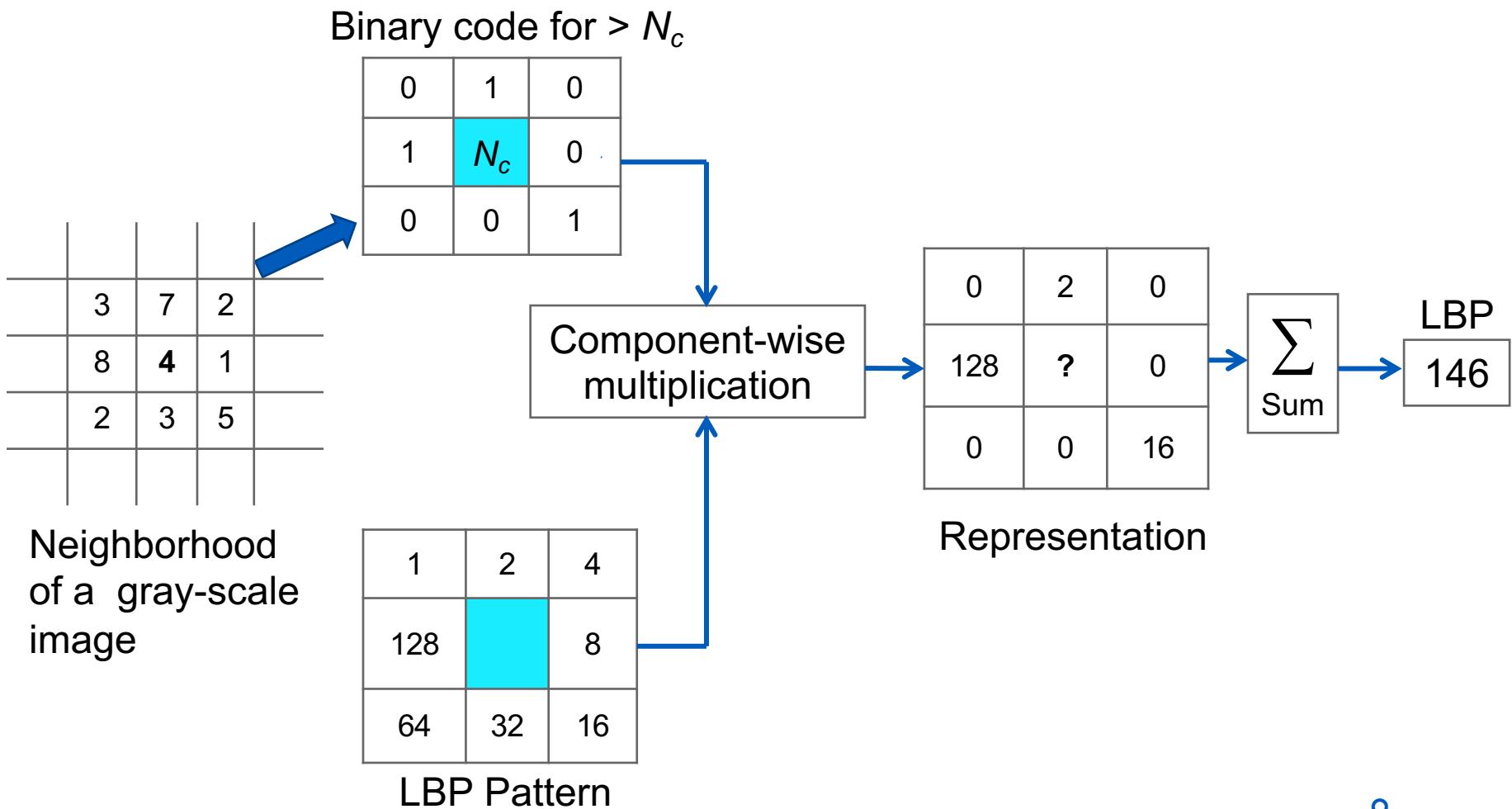
$r \rightarrow$  radius (for  $3 \times 3$  cell, it is 1).

Binary threshold function  $g(x)$  is,

$$g(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases}$$

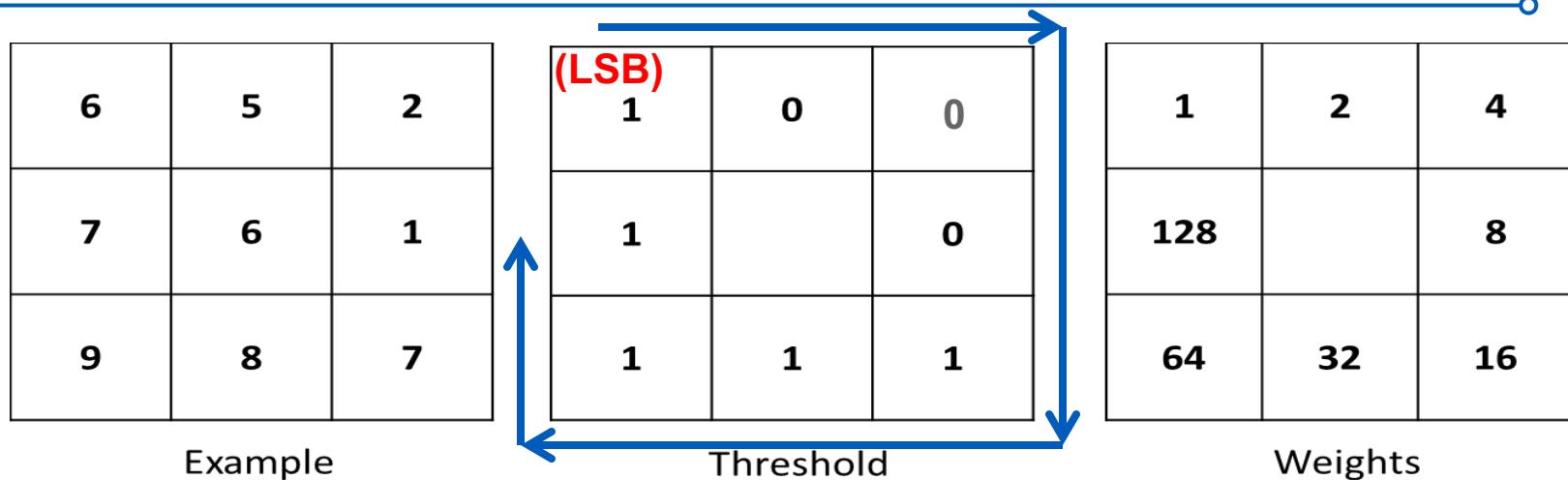


# Computation of Local Binary Pattern



Example of how the *LBP operator works*

# Computation of Local Binary Pattern



Binary Pattern:	1 (MSB)	1	1	1	0	0	0	1 (LSB)
-----------------	------------	---	---	---	---	---	---	------------

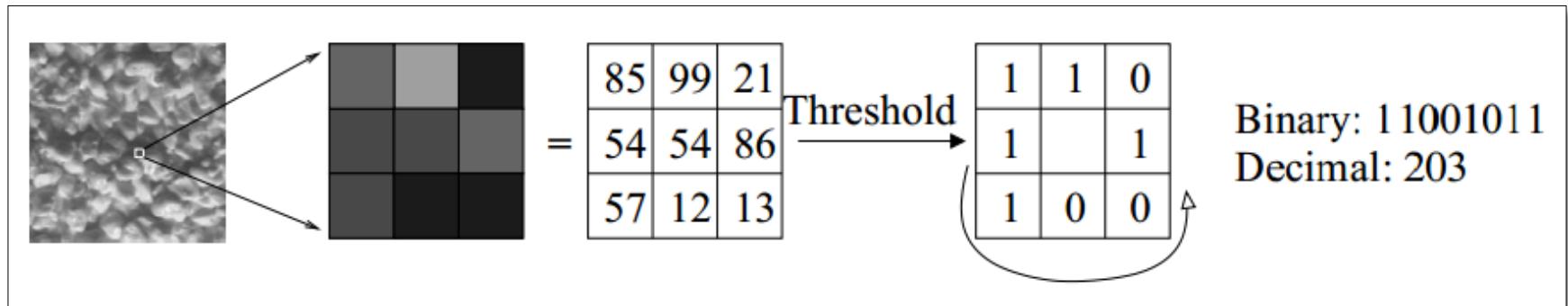
Code/Weight ( $2^p$ ):	$1 \times 2^7$	$1 \times 2^6$	$1 \times 2^5$	$1 \times 2^4$	$0 \times 2^3$	$0 \times 2^2$	$0 \times 2^1$	$1 \times 2^0$
	= 128	= 64	= 32	= 16	= 0	= 0	= 0	= 1

LBP:	$1 + 0 + 0 + 0 + 16 + 32 + 64 + 128 = 241$
------	--

MSB stands for Most Significant Bit, while LSB is Least Significant Bit.

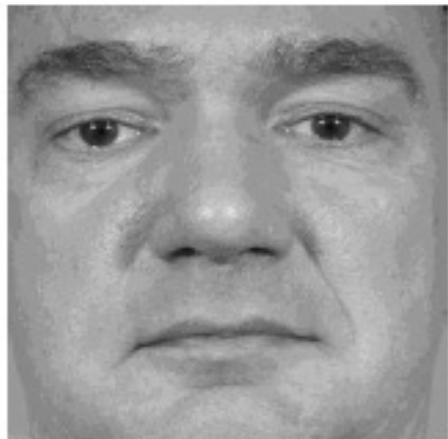
# LBP

- One of the best performing texture descriptors
- A label is assigned to every pixel
- Use center pixel to threshold the 3x3 neighborhood
  - Result in binary number
- Histogram of the labels is used as a texture descriptor



The Basic LBP operator

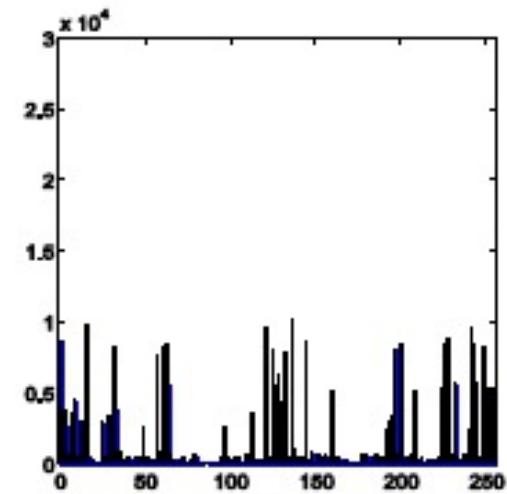
# LBP histogram



**Input image**



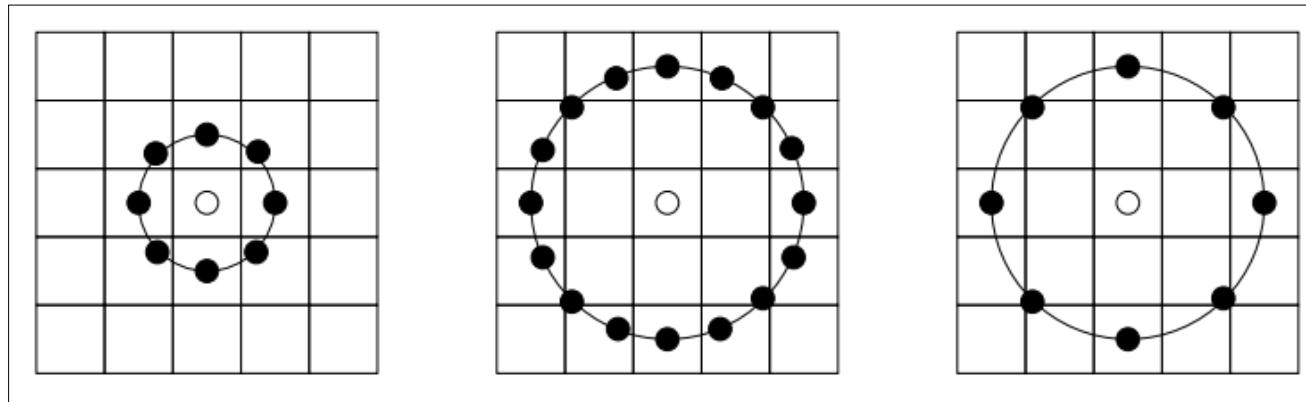
**LBP image**



**LBP histogram**

# LBP

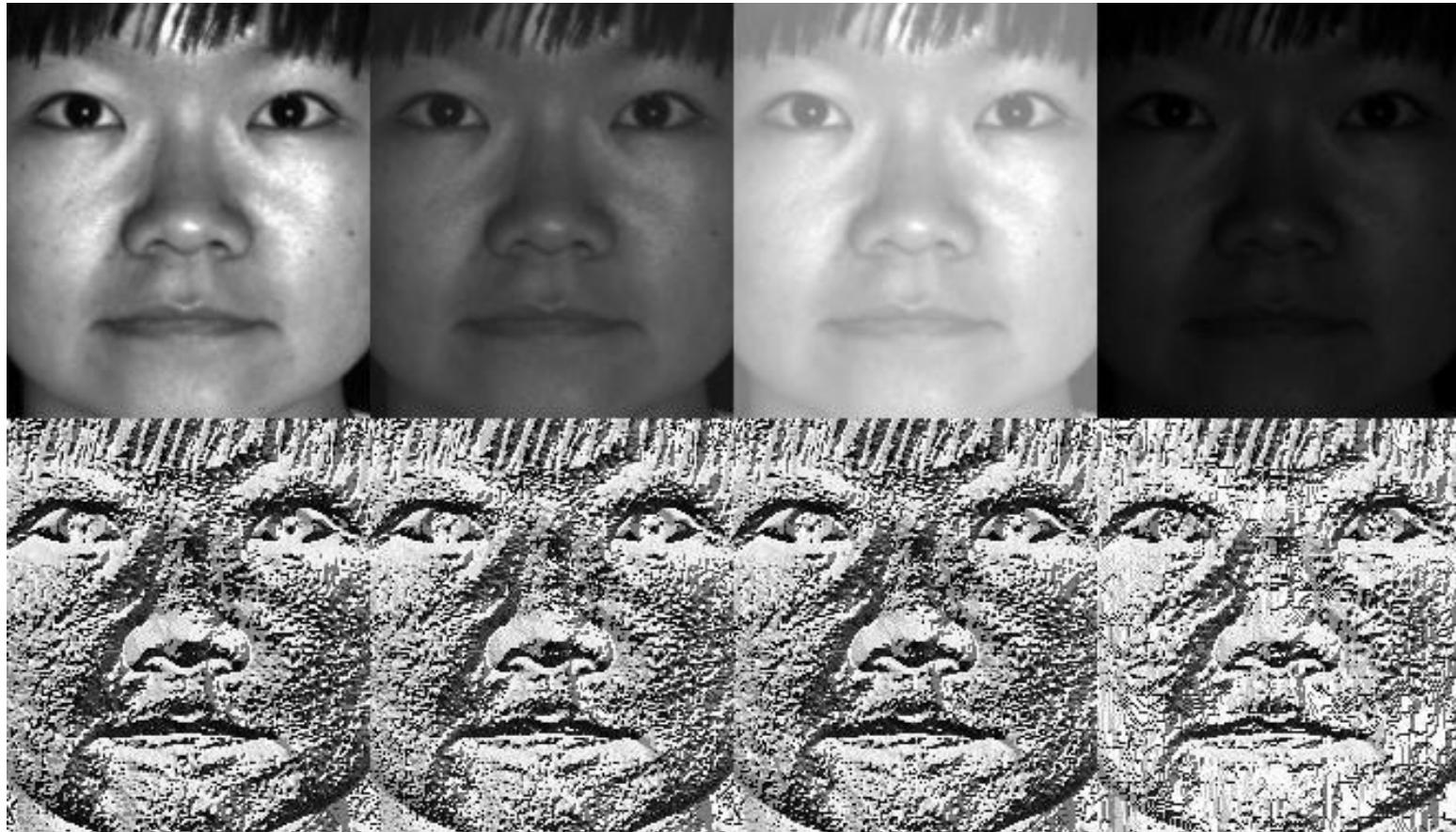
- LBP is extended to use different sizes of neighborhoods.
- Local neighborhoods is defined as a set of **sampling points**.
  - points evenly spaced on a circle centered at the labeled pixel.
- **(P, R)** , P = number of sampling points, R = radius
- **Bilinear interpolation** is used
  - If sampling point is not in the center of the pixel.



The circular (8,1), (16, 2) and (8, 2) neighborhoods.

# LBP Example

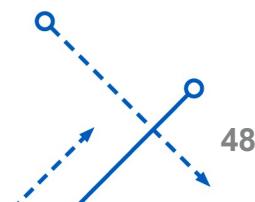
- Notice how LBP feature are illumination invariant



# Uniform LBP

---

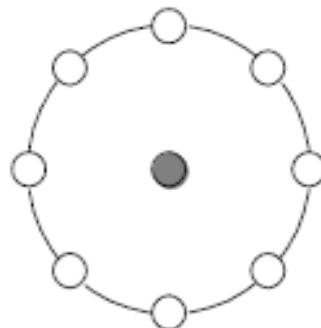
- Uniform patterns to further improve LBP.
- Has at most 2 bitwise transitions in binary pattern.
- Histogram
  - assigns separate bins for every uniform pattern.
  - assigns a single bin for all non-uniform pattern.



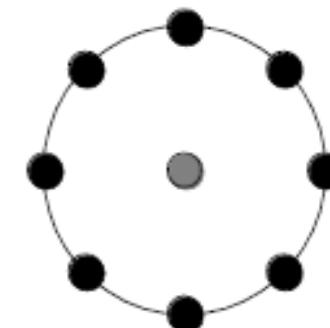
# Uniform LBP

Texture primitives (“micro-textons”) detected by the uniform patterns of LBP

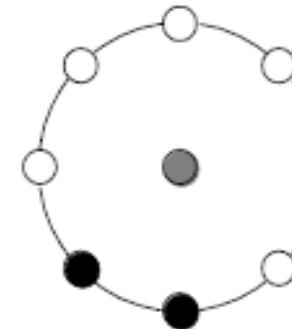
1 = black  
0 = white



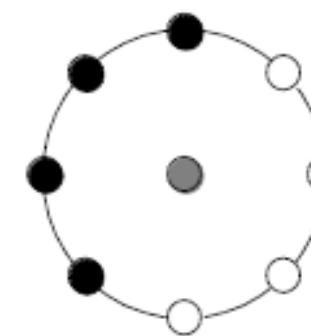
**Spot**



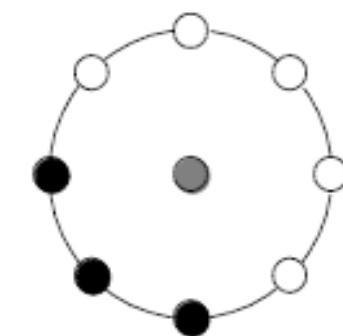
**Spot / flat**



**Line end**



**Edge**



**Corner**

# Uniform LBP

- Uniform patterns examples
  - 00000000 (0 transitions)
  - 01110000 (2 transitions)
  - 11001111 (2 transitions)
- Non-uniform patterns examples
  - 11001001 (4 transitions)
  - 01010011 (5 transitions)
- 59 bins histogram
  - 58 uniform bins →
  - 1 non-uniform bin

Length of feature vector

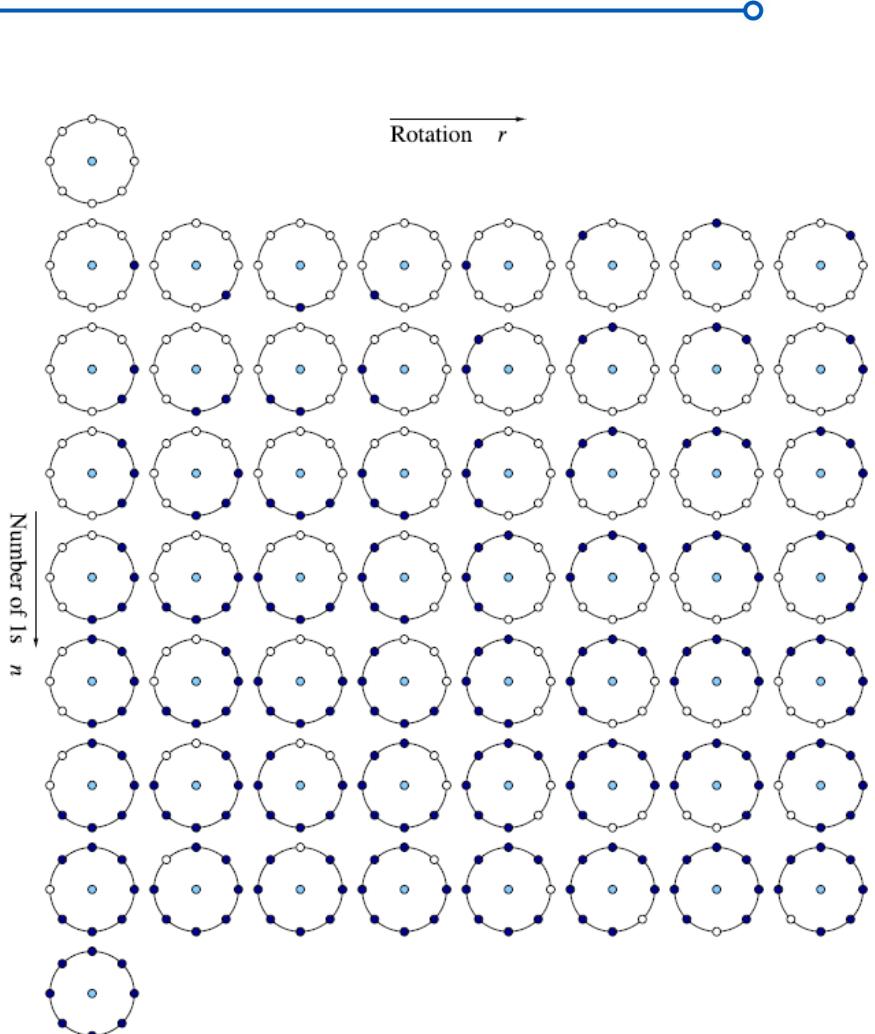
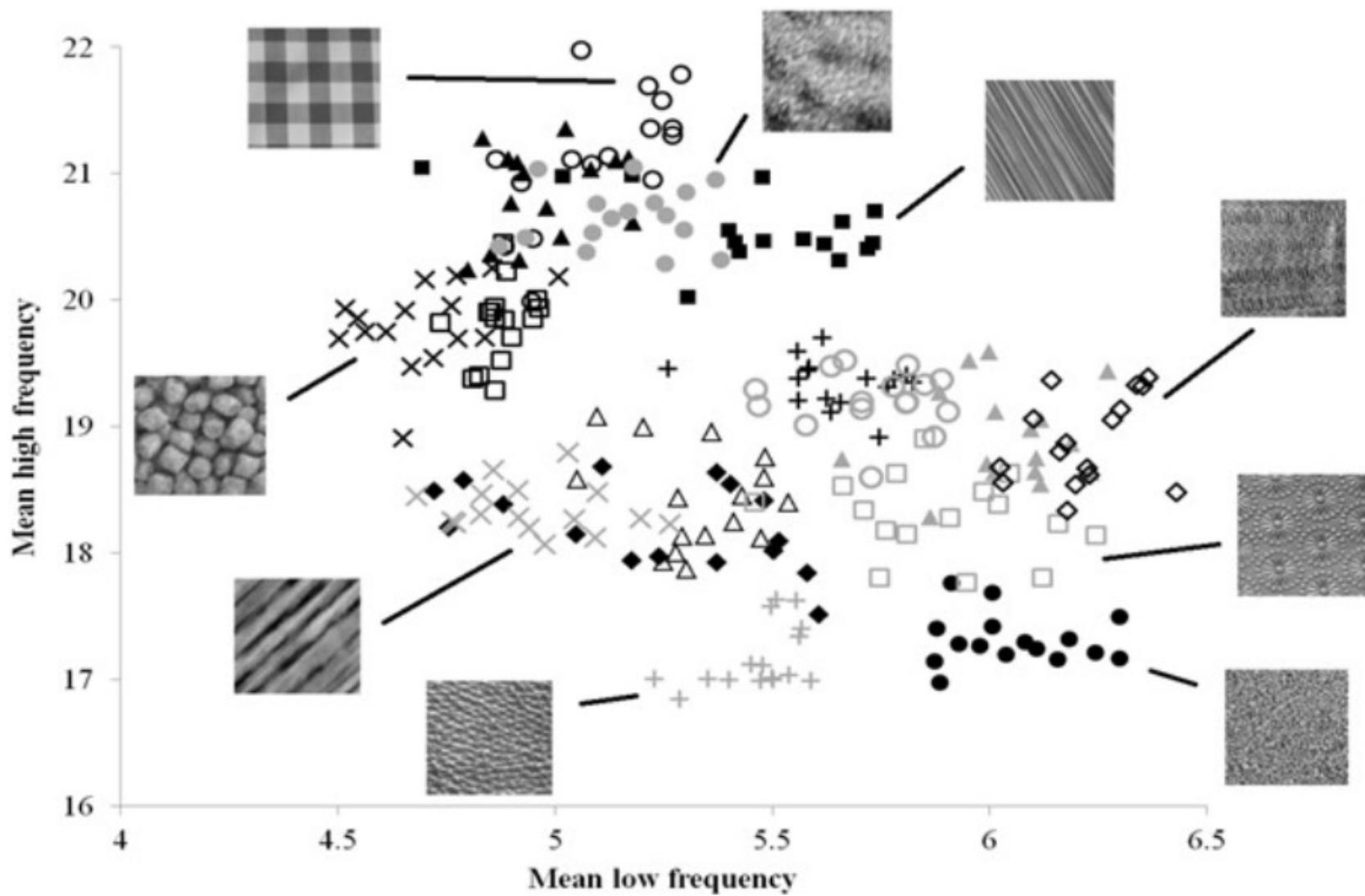


Fig. 2.4 The 58 different uniform patterns in  $(8, R)$  neighborhood

# Other Measures of Texture (frequency)



# Other Measures of Texture

Texture measure	Description	Equation*
Mean	Mean is the average grey level for each sample	$\sum_{i,j=0}^{N-1} i(P_{i,j})$
Variance	Variance is a measure of heterogeneity; it increases when the grey level values differ from their mean. In general, coarse-textured features are associated with higher variances	$\sum_{i,j=0}^{N-1} P_{i,j}(i-\mu_i)^2$
Homogeneity	This parameter measures image homogeneity as it assumes larger values for smaller grey tone differences in pair elements	$\sum_{i,j=0}^{N-1} \frac{P_{i,j}}{1+(i-j)^2}$
Contrast	Contrast is a measure of spatial frequency, the difference between the highest and the lowest values of a contiguous set of pixels. A high contrast implies high coarse texture	$\sum_{i,j=0}^{N-1} P_{i,j}(i-j)^2$
Dissimilarity	Dissimilarity, akin to contrast, describes the heterogeneity of the grey levels. Higher values of dissimilarity in the GLC matrix indicate coarser textures	$\sum_{i,j=0}^{N-1} P_{i,j} i-j $
Entropy	Entropy measures the disorder of an image. When the image is not texturally uniform, many GLC matrix elements have very small values, which imply that entropy is very large	$\sum_{i,j=0}^{N-1} P_{i,j}(-\ln p_{i,j})^2$
Angular second moment	This parameter measures textural uniformity. Thus, high angular second moment values occur when the grey level distribution over the window has either a constant or a periodic form	$\sum_{i,j=0}^{N-1} P_{i,j}^2$
Correlation	Correlation is a measure of grey-tone linear dependencies in the image. High correlation values imply a linear relationship between the grey levels of pixel pairs	$\sum_{i,j=0}^{N-1} P_{i,j} \left[ \frac{(i-\mu_i)(j-\mu_j)}{\sqrt{(\text{SD}_i)^2 (\text{SD}_j)^2}} \right]$