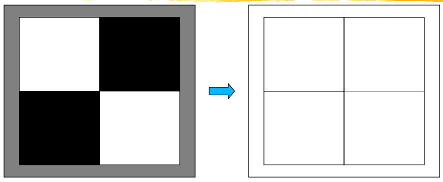
TEXTURE

- What is texture?
- Texture analysis
- Texture synthesis
- Shape from texture

EDGES AND REGIONS

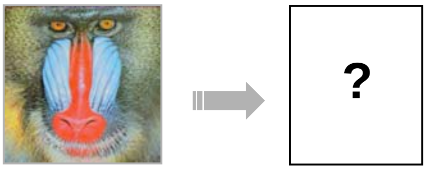


Edges: Boundary between bland image regions

Regions: Areas without edge

→ Duality Edge/Regions.

COLOR SEGMENTATION

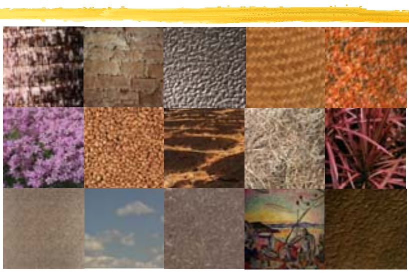


In general, regions can be formed from the original image data or from 'derived' images:

- color images from R, G, B
- textural images
- displacement images from motion analysis
- 3D depth images

TEXTURE EDGES

WHAT IS TEXTURE?

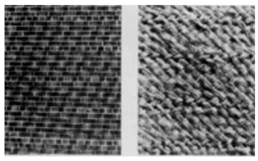


Repetition of a basic pattern:

- Structural
- Statistical

→ Non local property, subject to distortions.

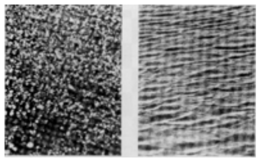
STRUCTURAL TEXTURES



Repetitive Texture Elements (Texels)

TEXTURE GRADIENT

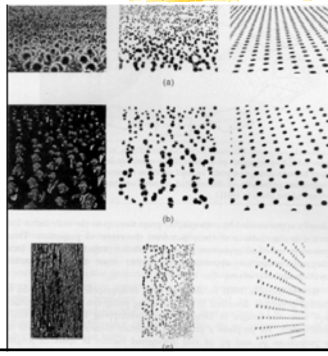
STATISTICAL TEXTURES



Homogeneous Statistical Properties

TEXTURE GRADIENT

SHAPE FROM TEXTURE



TEXTURED vs SMOOTH

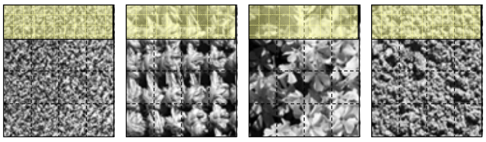
A "featureless" surface can be regarded as the most elementary spatial texture:

- Microstructures define reflectance properties.
- These properties are uniform or smoothly varying.

→ Texture is a scale dependent phenomenon

SCALE DEPENDENCE

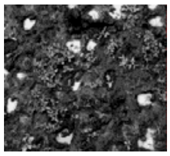
TEXTURE CLASSIFICATION



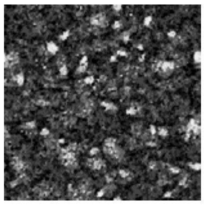
Given a set of training samples, determine the texture type for a new test sample.
→ Statistical classification techniques.

TEXTURE SYNTHESIS

Given a sample texture, synthesize other textures that are similar in appearance or some quantitative similarity measure.

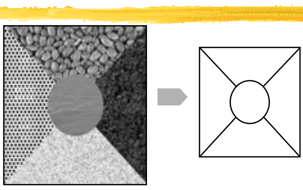


Original image



Synthesized image

TEXTURE SEGMENTATION



Goal: Partition an image into roughly homogeneous texture regions
- How is the texture to be represented?
- How to use this representation to find regions?

→**Difficult problem:**
- Feature statistics/models not known a priori.
- Sample regions needed to compute statistics.

TEXTURE REPRESENTATION

Structural Metrics:
- Texture is a set of primitive *texels* in some regular or repeated relationship.
- Description of the generative process and placement rules for sub-patterns occurring repeatedly within an overall pattern.

Statistical Metrics:
- Texture is a quantitative measure of the arrangement of intensities in a region.
- Model as a Markov process.

Spectral metrics:
- Texture characterized by properties of its Fourier transform.

STRUCTURAL METRICS

- A structural texture can be thought of as a set of primitive texels in a particular spatial relationship.
- A structural description of the texture would then include a description of the texels and a specification of the spatial relationship.

→ Texels must be segmentable and their relationships computable.

STRUCTURAL vs STATISTICAL

Segmenting out texels is difficult or impossible in real images.

What are the fundamental texture primitives in this image?

Numeric quantities or statistics that describe a texture can be computed from the gray levels or colors alone.
→ This approach is less intuitive, but is computationally efficient.

DISCRETE FOURIER TRANSFORM

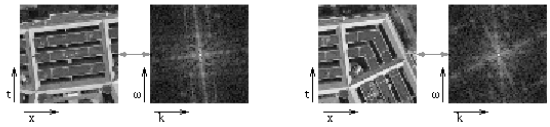
$$F(\mu,\nu) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-i2\pi(\mu x/M + \nu y/N)}$$

$$f(x,y) = \frac{1}{\sqrt{MN}} \sum_{\mu=0}^{M-1} \sum_{\nu=0}^{N-1} F(\mu,\nu) e^{+i2\pi(\mu x/M + \nu y/N)}$$

- The DFT of f ** g is the product of the DFT of f with the DFT of g.
- The DFT of a symmetric function is real.

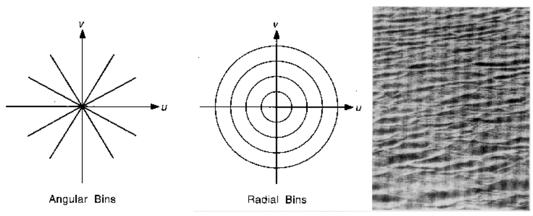
Bracewell, chap. 11

SPECTRAL ANALYSIS



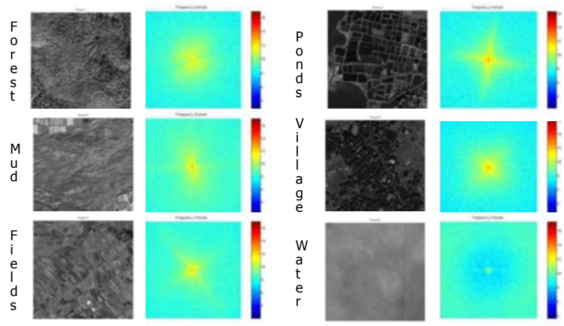
The magnitude of the DFT captures the main orientations in the image.

TEXTURE ANALYSIS



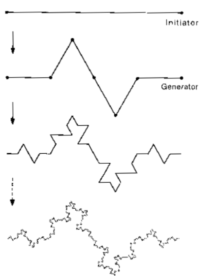
Angular and radial bins in the Fourier domain capture the directionality and rapidity of fluctuation of an image texture.

TEXTURE CLASSIFICATION



FRACTALS

Recursive "self-similar" shapes.

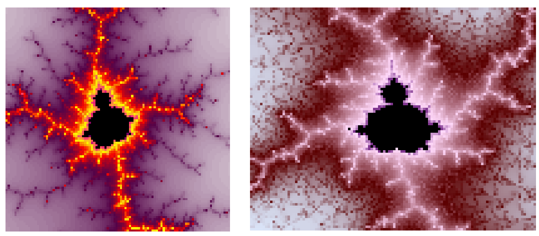


Shape subdivided into N self-similar shapes scaled by a factor r.

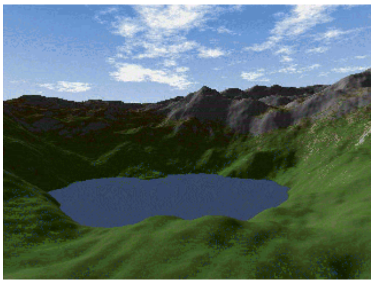
→ Fractal dimension

$$\log(N) / \log(1/r)$$

MANDELBROT SET



MOUNTAINSCAPE



The higher the fractal dimension, the more rugged the result.

FOURIER ESTIMATOR

If I(x,y) has a Fourier power spectrum F(f, θ) and has a fractal texture then:

$$\exists a,b \in \Re \text{ such that } \log(F(f,\theta)) = a\log(f) + b$$

FRACTAL DIMENSION

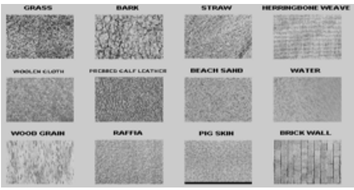


Image	Grass	Bark	Straw	Weave	Cloth	Leather
FD	2. 6571	2. 5494	2. 6881	2. 6323	2. 7170	2.6884
Image	Beach Sand	Water	Wood Grain	Raffia	Pigskin	Brick Wall
F D	2. 6432	2. 6827	2. 7256	2. 5665	2. 6142	2. 7039

STATISTICAL METRICS

Most natural textures are best modeled using such methods.

First order gray-level statistics:

- Statistics of single pixels in terms of histograms.
- Insensitive to neighborhood relationships.

Second order gray-level statistics:

$$P(l,m,\Delta i,\Delta j): P(i,j) = l \text{ and } P(i+\Delta i, j+\Delta j) = m$$

Given g gray-levels, for each $(\Delta i, \Delta j)$, P is represented as the gxg matrix H.

FIRST ORDER MEASURES

Edge Density and Direction

- Edge detection as a first step in texture analysis.
- The number of edge pixels in a fixed-size region tells us how busy that region is.
- The directions of the edges also help characterize the texture

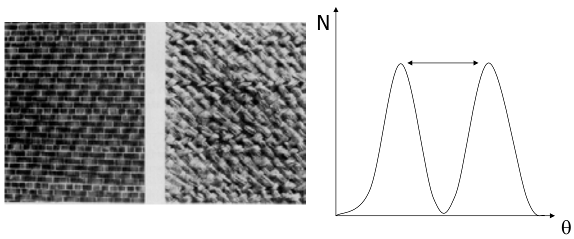
Edgeness per unit area

- { p | gradient_magnitude(p) ≥ threshold}| / N where N is the unit area or region.

Edge magnitude and direction histograms

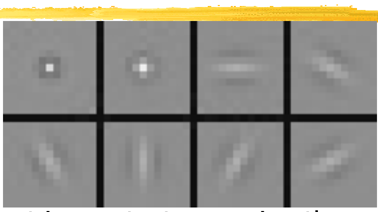
- Fmagdir = (Hmagnitude, Hdirection)

FIRST ORDER TEXTURE



Orientation histogram gives a clue to the orientation of the underlying plane.

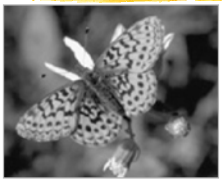
FILTER BANKS



Represent image textures using the responses of a collection of filters.

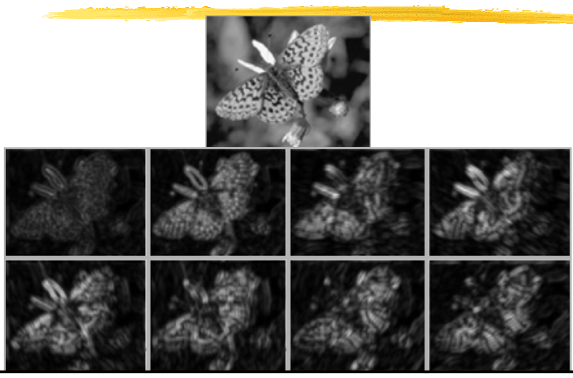
- An appropriate filter bank will extract useful information such as spots and edges
- Typically one or two spot filters plus several oriented bar filters

FILTER RESPONSES

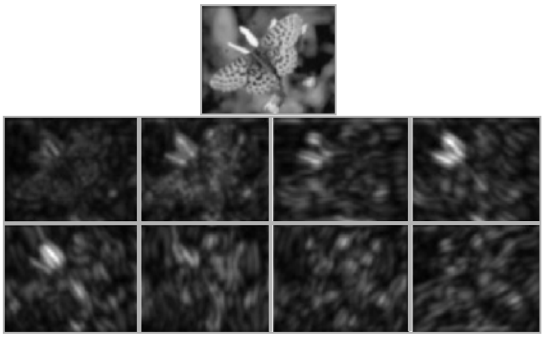


Based on the pixels with large magnitudes in the particular filter response, we can determine the presence of strong edges of certain orientation. We can also find spot patterns from the responses of the first two filters

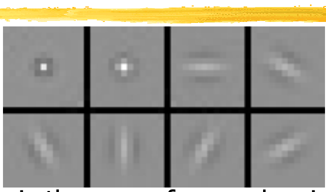
FILTER RESPONSES: HIGH RESOLUTION



FILTER RESPONSES: LOW RESOLUTION



FILTERS AS WEIGHTED SUMS



Each filter is the sum of several weighted Gaussian filters:

- The first spot filter is the sum of Gaussian filters with sigmas of 0.62, 1, and 1.6, and weights of 1, -2, 1.
- The second spot filter is the sum of Gaussian filters with sigmas of 0.71, 1.14, and weights of 1, and –1
- The six bar filters are rotated versions of a horizontal bar, which is the weighted sum of three Gaussian filters, each has sigma_x of 2, and sigma_y of 1, with centers at (0,1), (0,0), and (0,-1)

GABOR FILTERS

Gabor filters are the products of a Gaussian filter with oriented sinusoids. They come in pairs, each consisting of a symmetric filter and an anti-symmetric filter:

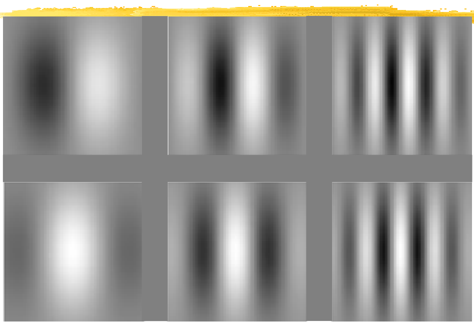
$$G_{symmetric}(x,y) = \cos(k_x x + k_y y)\exp\left\{-\frac{x^2+y^2}{2\sigma^2}\right\}$$

$$G_{antisymmetric}(x,y) = \sin(k_x x + k_y y)\exp\left\{-\frac{x^2+y^2}{2\sigma^2}\right\}$$

where k_x and k_y determine the spatial frequency and the orientation of the filter and s determines the scale.

→ A filter bank is formed by varying the frequency, the scale, and the filter orientation

GABOR FILTERS



GABOR FILTER CHARACTERISTICS

Respond strongly at points in an image where there are components that locally have a particular spatial frequency and orientation.

In theory, by applying a very large number of Gabor filters at different scales, orientations and spatial frequencies, one can analyze an image into a detailed local description.

In practice, it is not known how many filters, at what scale, frequencies, and orientations, to use

- Number of orientations is application dependent
- Six orientations appears to be a reasonable choice

WHAT FILTERS TO USE

Filter Selection:

- Typically a couple of spot filters plus some oriented bar filters at different orientations and scales

Form of filter

- Weighted sums of Gaussians
- Gabor filters

->Very little reason to believe that optimizing the choice of filters will result in any major advantage

SECOND ORDER MEASURES

Histogram of the co-occurrence of particular intensity values in the image.

- Specified in terms of the geometric relationships between pixel pairs:
  - Distance
  - Orientation
- P(i,j,d,θ) Frequency with which a pixel with value j occurs at distance d and orientation θ from a pixel with value i.

SIMPLE EXAMPLE

If $$I = \begin{bmatrix} 0 & 1 & 2 & 3 & 1 \\ 2 & 1 & 3 & 1 & 1 \\ 0 & 0 & 2 & 2 & 1 \\ 1 & 2 & 0 & 3 & 1 \\ 0 & 0 & 0 & 0 & 3 \end{bmatrix},$$

then $$H = \begin{bmatrix} 4 & 1 & 1 & 2 \\ 0 & 1 & 2 & 1 \\ 1 & 2 & 1 & 1 \\ 0 & 3 & 0 & 0 \end{bmatrix},$$

and $$P(l,m,1,0) = \frac{H(l,m)}{20}.$$

CO-OCURRENCE MATRIX

No need to distinguish between

$$P(m,l,\Delta i,\Delta j)$$

and

$$P(l,m,\Delta i,\Delta j)$$

→ Co-Occurrence matrix C:

$$C = H + H^T$$

2ND ORDER TEXTURE MEASURES

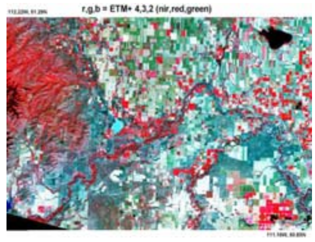
Uniformity of energy $: \sum_{ij} P_{ij}^2$

Entropy $: -\sum_{ij} P_{ij} \log(P_{ij})$

Contrast $: \sum_{ij} |i - j| P_{ij}$

and many more ....

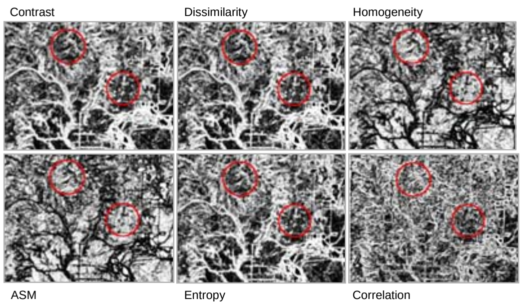
LANDSAT IMAGE



The image is excerpted from Path 41, Row 25 of Landsat 7 ETM+, dated 4 September 1999. This is an area in the Rocky Mountain Foothills near Waterton National Park, Alberta. The western edge of the image contains steep slopes and deep valleys. To the east is both grassland and annual crops, mostly grains. The eastern area is bisected by numerous small streams.

FULL RESOLUTION

COMPARISON



CLASSIFICATION

Used on aerial images to identify eight terrain classes:

- Old residential
- New residential
- Urban
- Lake
- Swamp
- Scrub
- Wood

AERIAL TEXTURES

PARAMETER CHOICES

Using co-occurence matrices requires us to choose:

- window size,
- direction of offset,
- offset distance,
- which channel(s) to use,
- which measure to use.

How do we choose these parameters?

- This is actually a critical question with **all** the statistical texture methods.
- Are the "texels" tiny, medium, large, all three …?

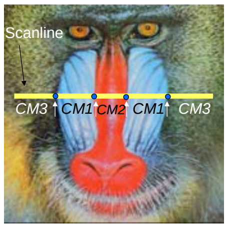
→ Yet another open problem !

TEXTURE BOUNDARY DETECTION

A Markovian model of order 1 defines the probability of each pixel intensity given the previous pixel intensity along a line.

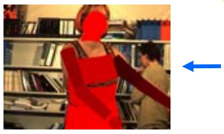
These probabilities can be represented by a *coocurrence matrix* for which each element *CM(i,j)* gives the probability of having adjacent pixels of intensity *i* and *j*.

Therefore, *coocurrence matrices* can be used to detect the *intersection of two different textures*

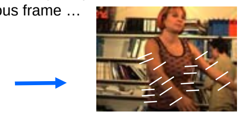


OUTLINE DETECTION

BODY TRACKING



Approximate projection of the model using the pose in previous frame …



It is used to define scanlines to search for body/background boundaries …

using coocurrence matrices or traditional gradient maxima

REAL TIME TRACKING

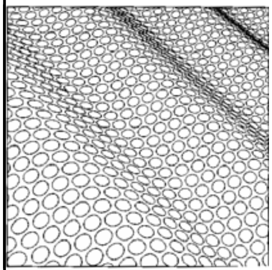
SHAPE FROM TEXTURE

SHAPE FROM TEXTURE

Recover surface orientation or surface shape from image texture.

- Assume texture 'looks the same' at different points on the surface
- This means that the deformation of the texture is due to the surface curvature

STRUCTURAL SHAPE RECOVERY

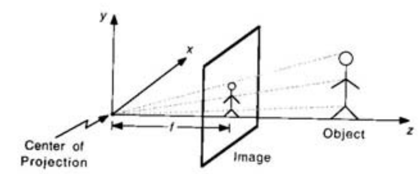


Basic hypothesis: Texture resides on the surface and has no thickness.

--> Computation under:

- Perspective projection
- Paraperspective projection
- Orthographic projection

PERSPECTIVE PROJECTION

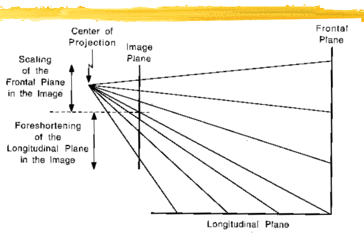


$$u = f\,\frac{x}{z}$$

$$v = f\,\frac{y}{z}$$

Pinhole geometry without image reversal

PERSPECTIVE DISTORTION



Perspective projection distortion of the texture

- depends on both depth and surface orientation,
- is anisotropic.

FORESHORTENING

Depth vs Orientation:

Infinitesimal vector [Δx,Δy,Δz] at location [x,y,z]. The image of this vector is

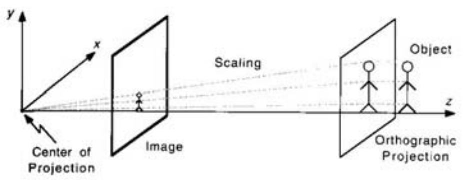
$$\frac{f}{z}[\Delta x - \frac{x}{z}\Delta z, \Delta y - \frac{y}{z}\Delta z]$$

Two special cases:

Δz=0 : The object is scaled

Δx=Δy=0 : The object is foreshortened

ORTHOGRAPHIC PROJECTION



$$u = sx$$

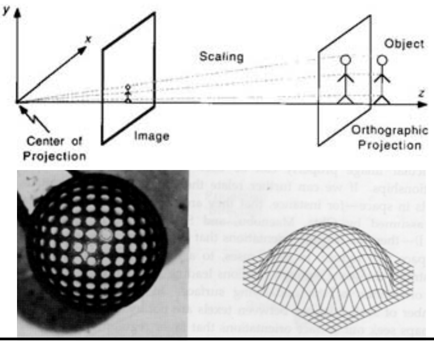
$$v = sy$$

Special case of perspective projection:

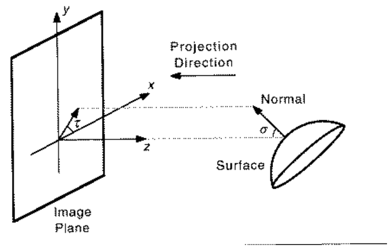
- Large f
- Objects close to the optical axis

→Parallel lines mapped into parallel lines.

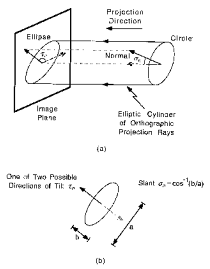
ORTHOGRAPHIC PROJECTION



TILT AND SLANT



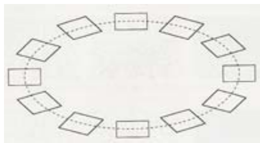
ORTHOGRAPHIC PROJECTION



Tilt: Derived from the image direction in which the surface element undergoes maximum compression.

Slant: Derived from the extent of this compression.

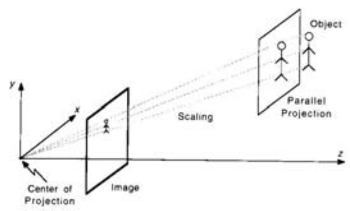
PERPENDICULAR LINES



Orthographic projections of squares are rotated with respect to one another in a plane inclined at ω=60° to the image plane.

$$\frac{\left\|(\vec{p}_1/l_1)\times(\vec{p}_2/l_2)\right\|}{\left\|\vec{p}_1/l_1\right\|^2+\left\|\vec{p}_2/l_2\right\|^2}=\frac{\cos(\omega)}{1+\cos^2(\omega)}$$

PARAPESPECTIVE PROJECTION

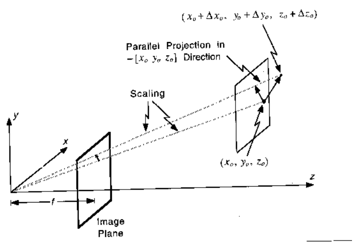


Generalization of the orthographic projection:

- Object dimensions small wrt distance to the center of projection.

→Parallel projection followed by scaling

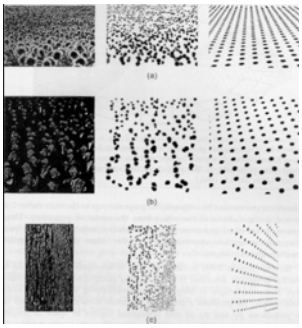
PARAPERSPECTIVE PROJECTION



For planar texels:

$$A' = -\frac{f^2}{z_0^3}\vec{n}\bullet[x_0\, y_0\, z_0]A$$

PARAPERSPECTIVE DISTORTION



Texels:

- Image regions that are brighter or darker than their surroundings.
- Assumed to have the same area in space.

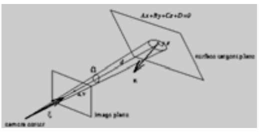
→ Given enough texels, it becomes possible to estimate the normal.

STATISTICAL SHAPE RECOVERY

Measure "texture density" as opposed to texel area: number of textural primitives per unit surface

Basic Hypotheses:
Textural isotropy
Textural homogeneity

TEXTURE DENSITY



$$\gamma = k \frac{A_s}{A_i} = k \frac{-fD^2}{(Au + Bv + Cf)^3}$$

$$[u, v, f]\vec{n} = \lambda\beta$$

where $\lambda = (kfD)^2$

$$\beta = 1 / \sqrt[3]{\gamma}$$

TEXTURE DENSITY

For several texture elements: $\psi\vec{n} = \lambda\vec{b}$

$$\psi = \begin{bmatrix} u_1 & v_1 & f \\ \dots & \dots & \dots \\ u_n & v_n & f \end{bmatrix}$$

$$b = \begin{bmatrix} \beta_1 \\ \dots \\ \beta_n \end{bmatrix}$$

Therefore: $\vec{n} = \frac{\psi^t \vec{b}}{\|\psi^t \vec{b}\|}$

IN SHORT

Texture is a key property of objects which is

- Non local
- Non trivial to measure
- Subject to deformations

→ Useful in theory to segment and capture shape, but hard to use in practice.