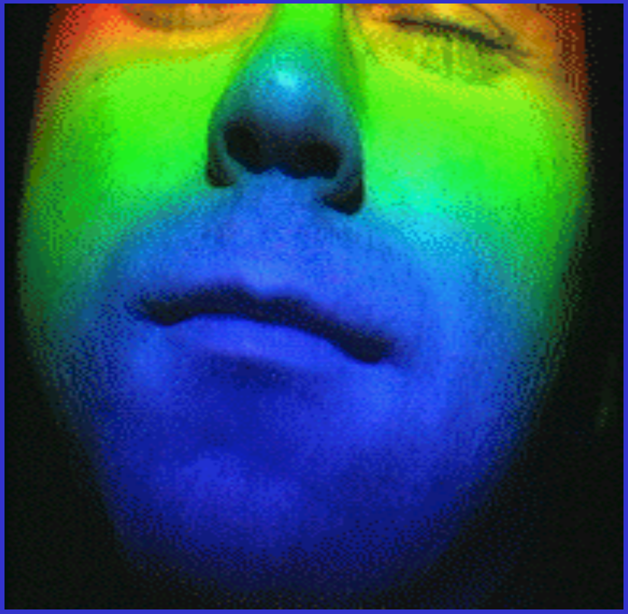
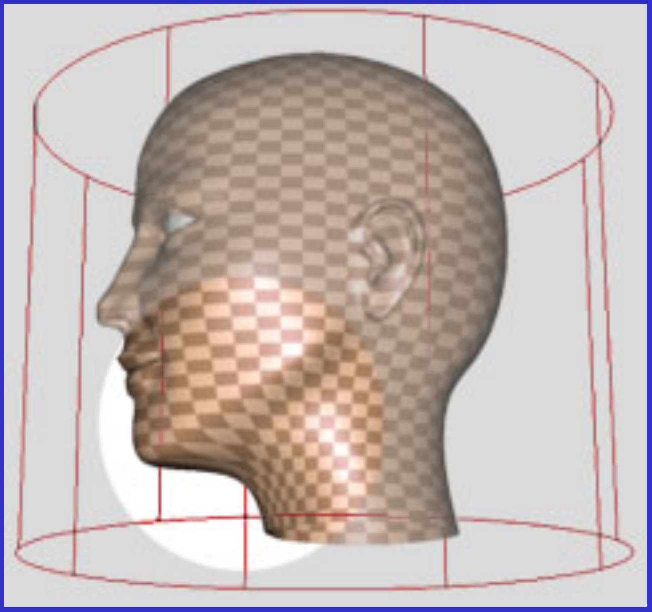
Surface Features: colour and texture





Introduction

colour
- color spaces
- colour constancy
- surface reflectance revisited
- illumination invariant colour features
- the holy grail: BRDFs

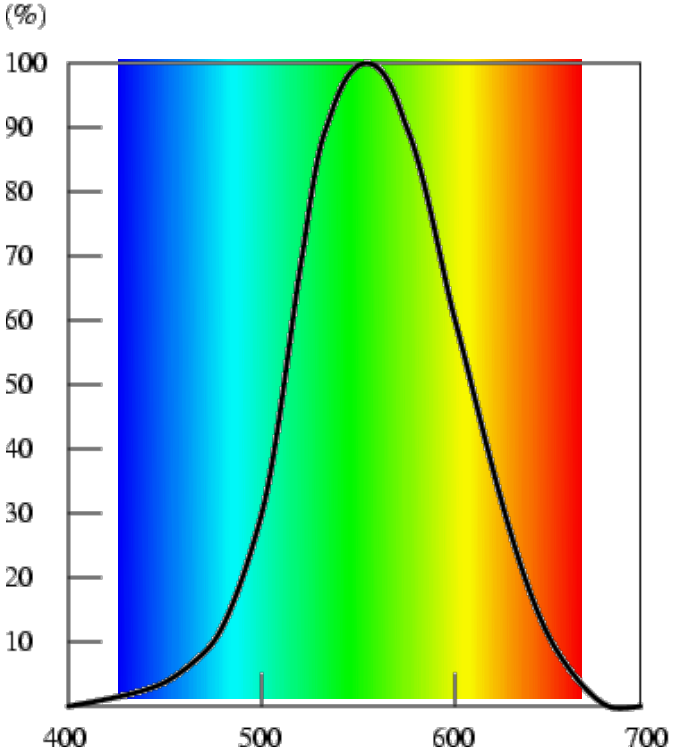
texture
- Fourier features
- cooccurrence matrices
- filter banks (Laws, Gabor, eigenfilters)
- stochastic model

Computer Vision

COLOUR

The perception of brightness

- Luminous efficiency function ($v(\lambda)$): relates radiometry & photometry



- C.I.E.(Commission Internationale de l’Eclairage) → standards

Link radiometry-photometry (Watt to lumen)

Photometry: subjective impressions
Radiometry: objective, physical measurements

at 555 nm : 1lm = 1/683 W = 1.46 mW

for light with spectral composition $c(\lambda)$ (radiant flux)

$$l = k\int_{\lambda=0}^{\infty} c(\lambda)v(\lambda)d\lambda$$

with k is 683 lumens/watt

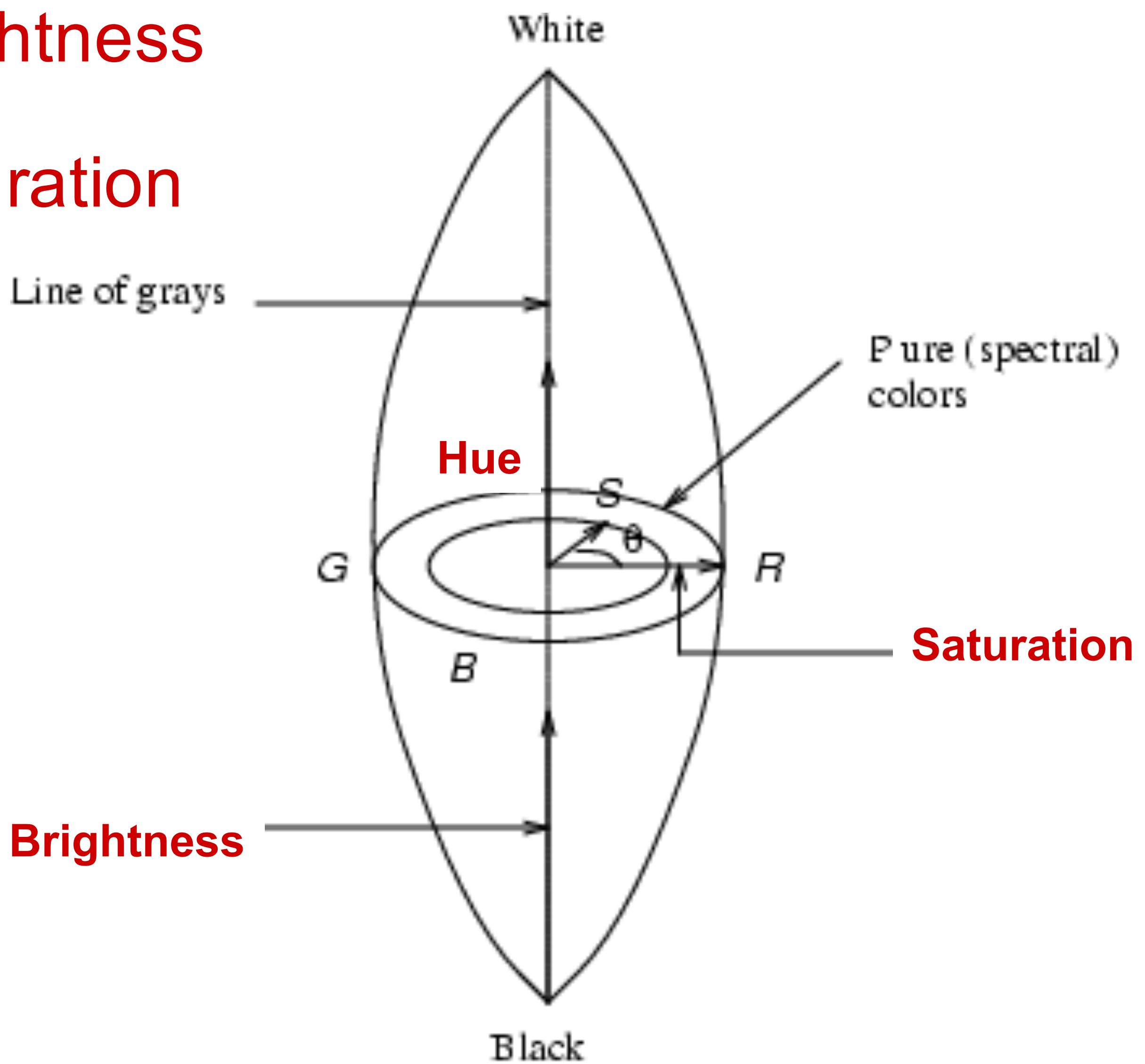
The study of colour…

Use :

- pleasing to the eye (visualisation of results)
- characterising colours (features e.g. for recognition)
- generating colours (displays, light for inspection)
- understanding human vision

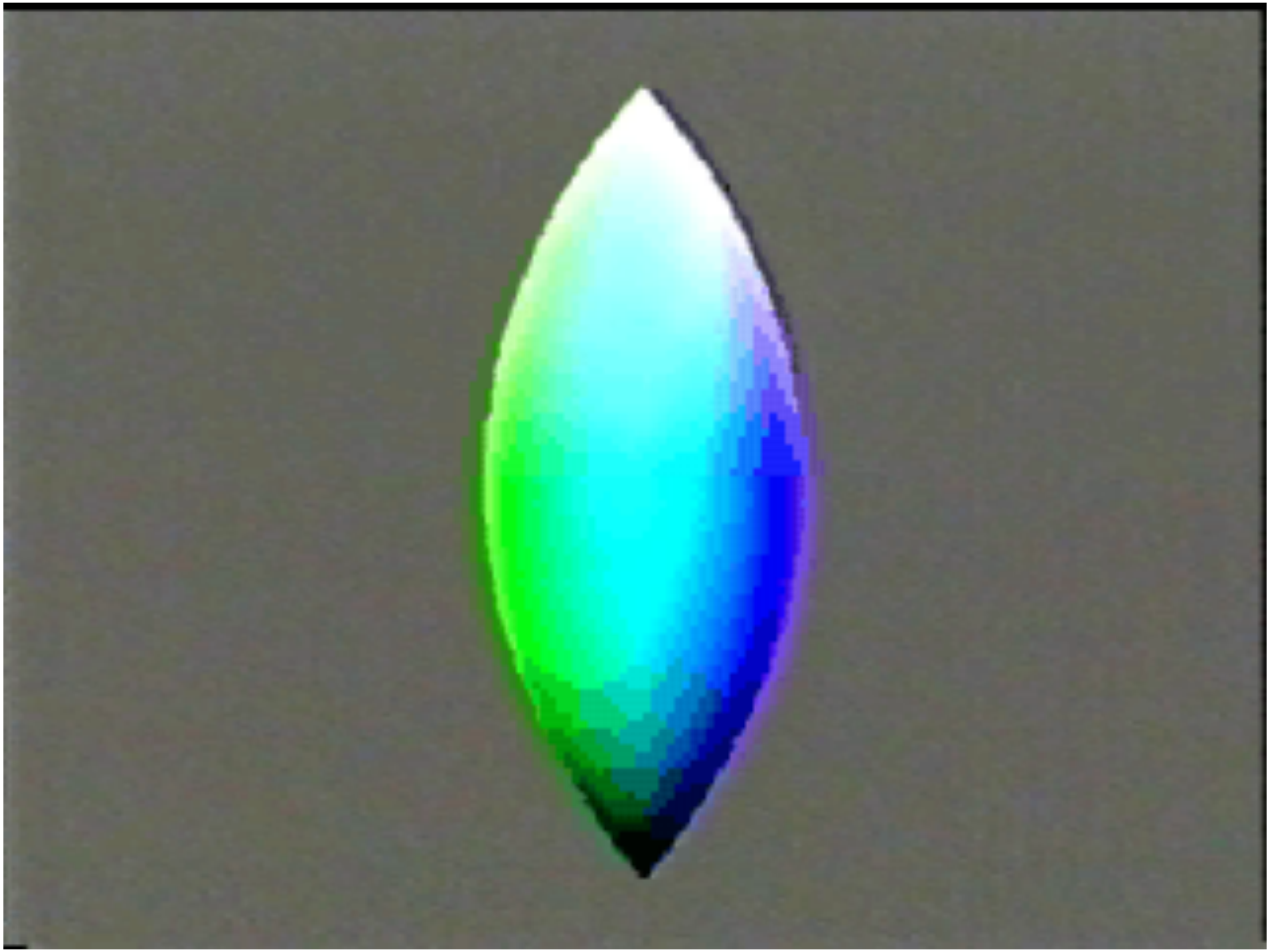
The perceptual attributes of light (humans)

1. brightness
2. hue
3. saturation



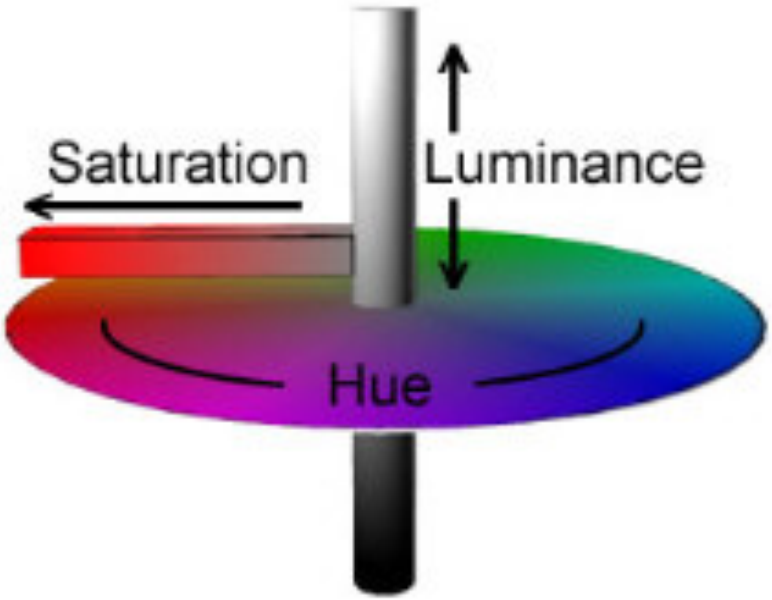
Computer
Vision

The C.I.E. color space



Computer Vision

Human perception of light = 3-dimensional



The history of colour

Newton → spectrum

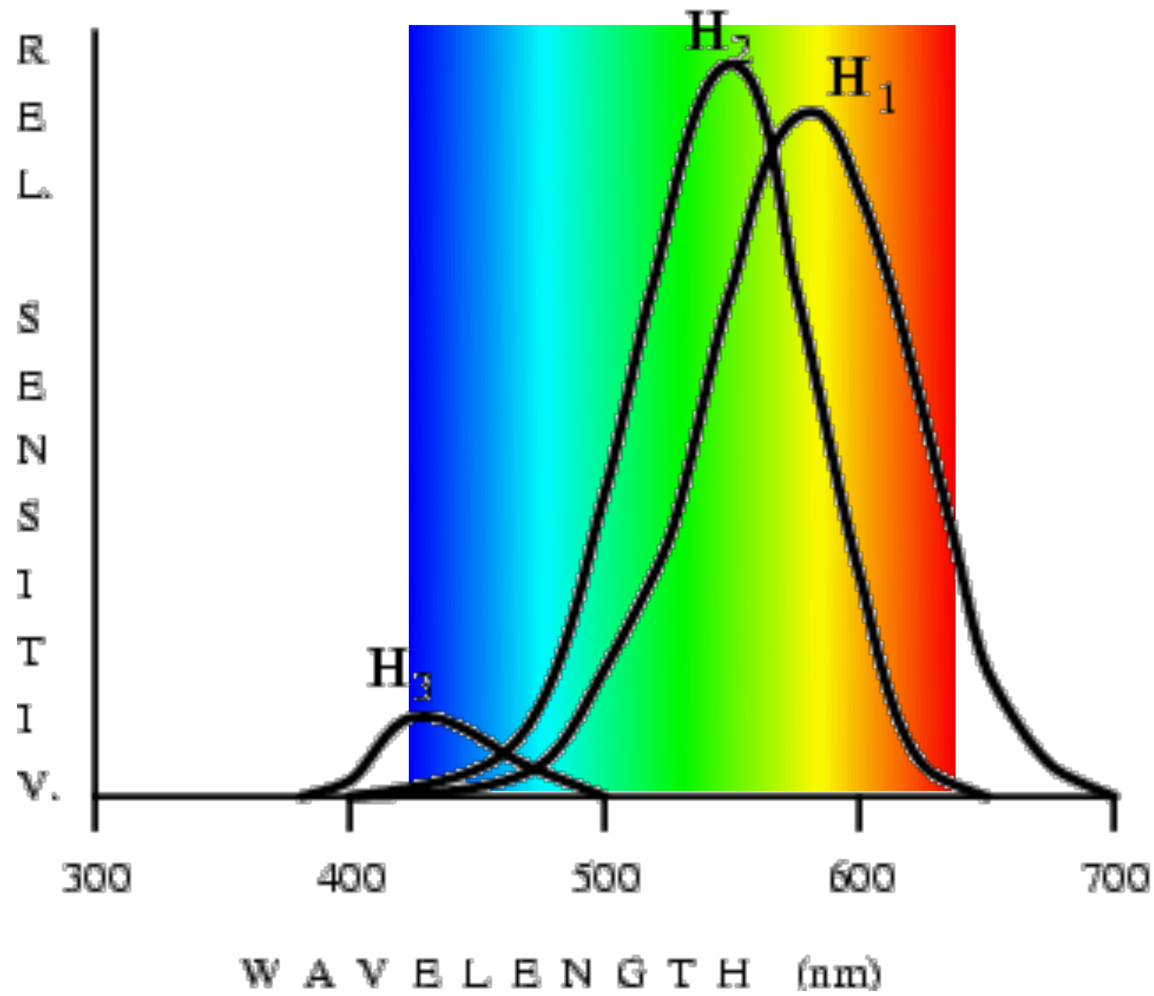
Young → tristimulus model

later : physiological underpinning :

3 cone types

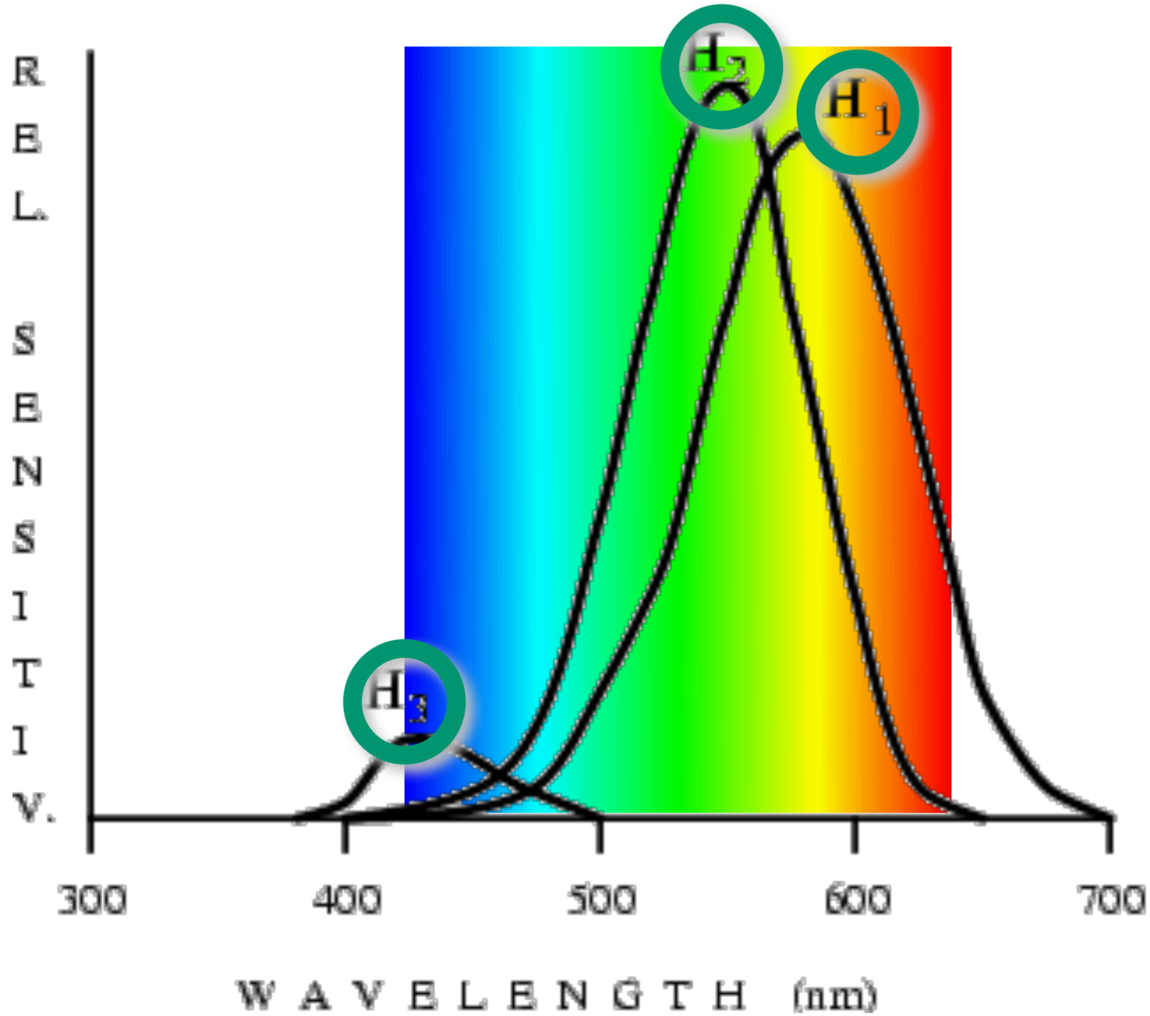
The retinal cones

3 types : blue, green, yellow-green



The retinal cones

3 types : blue, green, yellow-green



Prediction of colour sensation

source with spectral radiant flux $C(\lambda)$ produces responses R_i, with i = 1,2,3

$$R_i(c) = \int H_i(\lambda) C(\lambda) d\lambda, \; i = 1,2,3$$

Hence, our perception of multi-spectral sources is quite empoverished:

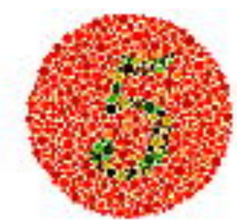
an entire distribution over λ is projected onto only 3 numbers R.

Prediction of colour sensation

source with spectral radiant flux $C(\lambda)$
produces responses R_i, with i = 1,2,3

$$R_i(c) = \int H_i(\lambda) C(\lambda) d\lambda, \; i = 1,2,3$$

- 2 sources with equal R_i's ⇒ observed as same colour!
- luminance ⊥ chrominance
- 10% of population have abnormal colour vision



- several birds have 4 cone types (incl. UV)
- colour constancy

Tristimulus representation of colour

Camera $\Longrightarrow$ tristimulus values $\Longrightarrow$ display

3 primaries $P_j(\lambda)$, j = 1,2,3

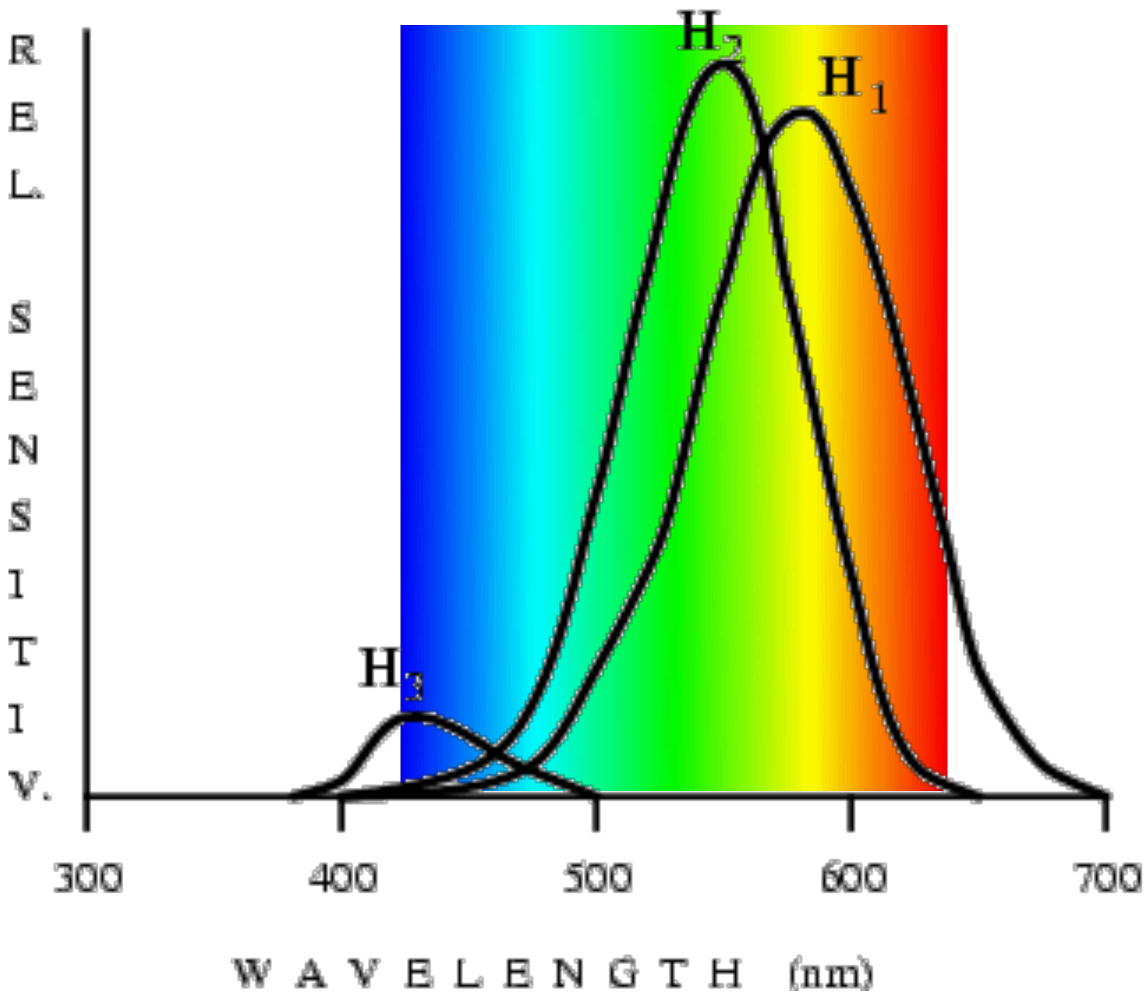
CIE primaries : $\lambda_1 = 700$
$\lambda_2 = 546.1$
$\lambda_3 = 435.8$

applications : practical primaries
e.g. TV : EBU and NTSC

Computer Vision

Reminder: 3 retinal cone types

3 types : blue, green, yellow-green



The matching of colour

source $C(\lambda)$ matched by primaries

$$\sum_{j=1}^{3} m_j P_j(\lambda)$$

$$R_i(C) = \int \quad C(\lambda) \qquad H_i(\lambda)d\lambda$$

The matching of colour

source $C(\lambda)$ matched by primaries

$$\sum_{j=1}^{3} m_j P_j(\lambda)$$

$$R_i(C) = \int \sum_{j=1}^{3} m_j P_j(\lambda)\ H_i(\lambda) d\lambda$$

The matching of colour

source $C(\lambda)$ matched by primaries

$$\sum_{j=1}^{3} m_j P_j(\lambda)$$

$$R_i(C) = \int \sum_{j=1}^{3} m_j P_j(\lambda) \; H_i(\lambda) d\lambda$$

$$= \sum_{j=1}^{3} m_j \int H_i(\lambda) P_j(\lambda) d\lambda$$

The matching of colour

source $C(\lambda)$ matched by primaries

$$\sum_{j=1}^{3} m_j P_j(\lambda)$$

$$R_i(C) = \int \sum_{j=1}^{3} m_j P_j(\lambda) H_i(\lambda) d\lambda$$

$$= \sum_{j=1}^{3} m_j \underbrace{\int H_i(\lambda) P_j(\lambda) d\lambda}_{l_{i,j}}$$

can be determined “off-line”

Computer Vision

The math

extremely simple : linear equations

$$\sum_{j=1}^{3} m_j \, l_{i,j} = R_i$$

implies inverting the matrix :
independent primaries!

also linear transform between the m_j' s
for different choices of primaries:

$$L\,m = R$$
$$L'\,m' = R$$

gives

$$m' = L'^{(-1)}\,L\,m$$

Tristimulus values

"white" considered a reference :
specify relative values w.r.t. m_js for white : w_j

Tristimulus values : $$T_j = \frac{m_j}{w_j}$$

The scaling preserves the linearity

CIE tristimulus values : **R**, **G**, **B**
(for CIE white = flat spectrum & $w_1 = w_2 = w_3$)

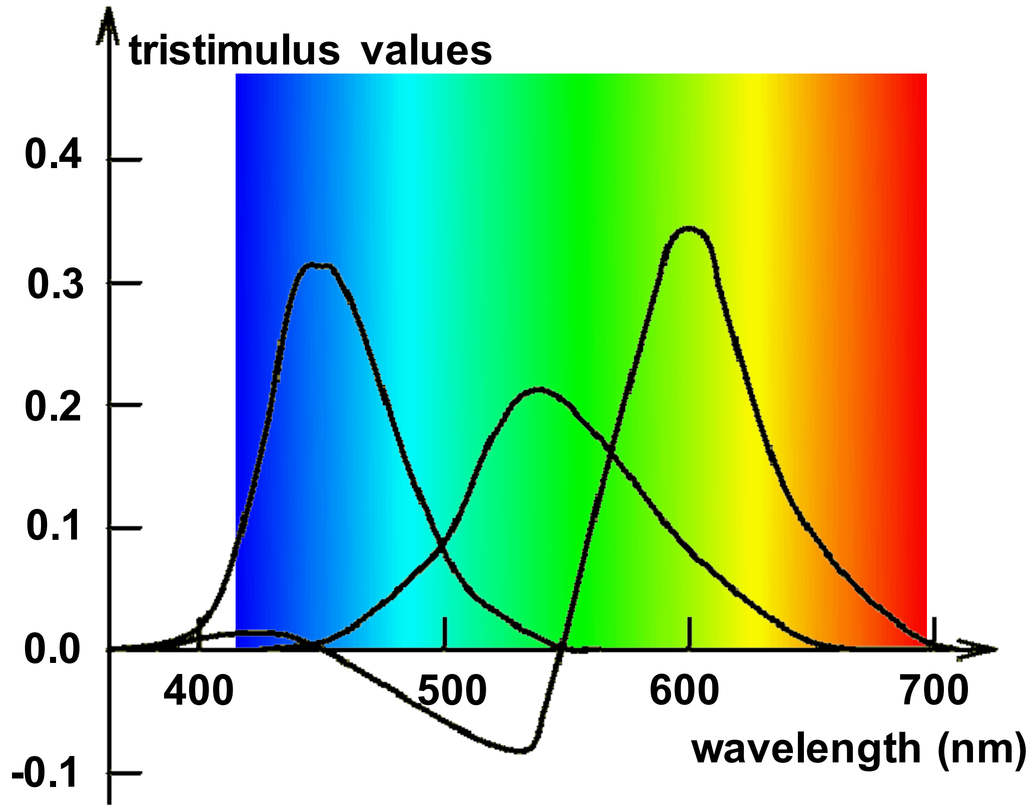
Note that for white $T_1 = T_2 = T_3 = 1$

Spectral matching curves

Spectral matching curves $T_j(\lambda)$:
values for monochromatic sources

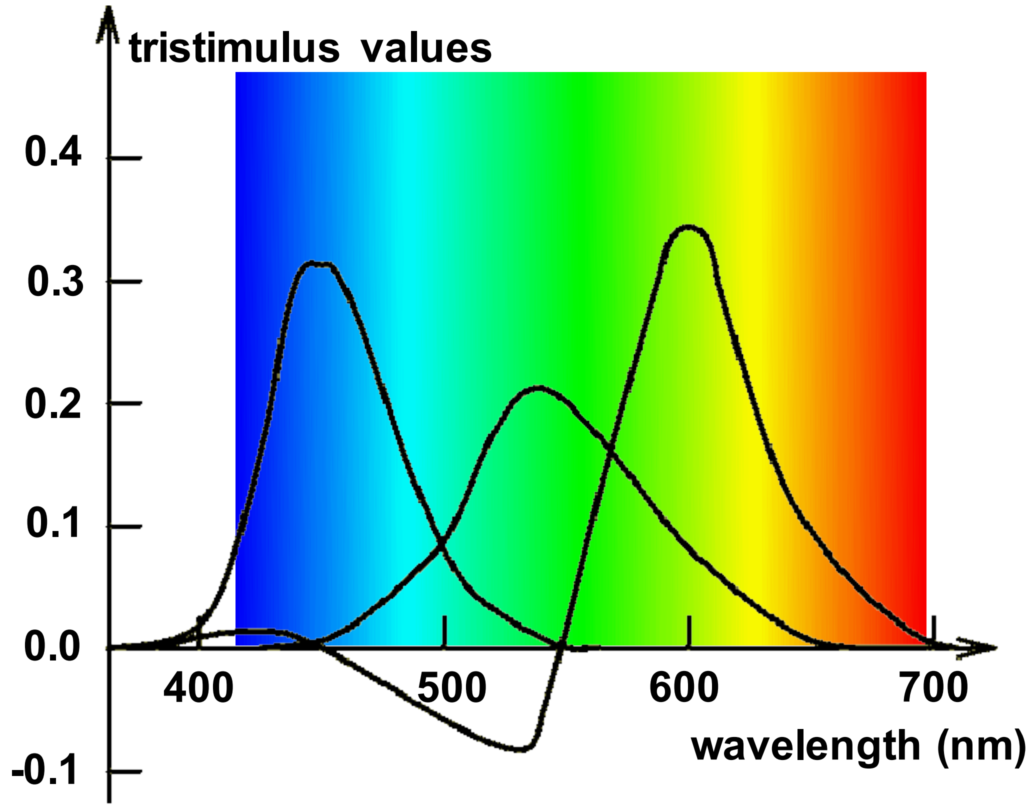
$$R_i(C_\lambda) = H_i(\lambda) = \sum_{j=1}^{3} \boxed{m_j}\, l_{i,j} = \sum_{j=1}^{3} \boxed{w_j}\, l_{i,j}\, \boxed{T_j(\lambda)}$$

for the CIE primaries:



Computer Vision

Interpretation of these curves



negative values : colours that cannot be produced

In such case:
mix(target , neg. primary) = mix(pos. primaries)

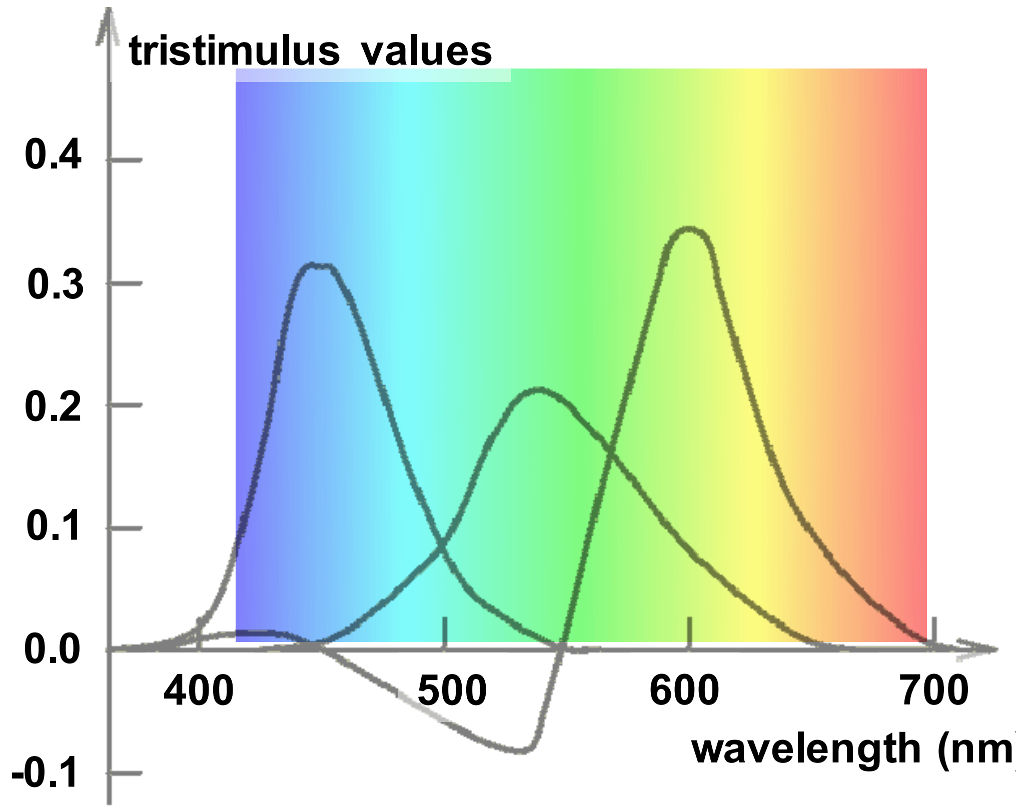
for any primary triple some colours cannot be produced

Computer Vision

Interpretation of these curves

for arbitrary source $C(\lambda)$:

$$T_j(C) = \int C(\lambda) T_j(\lambda) d\lambda$$



Chromaticity coordinates

tristimulus values still contain brightness info

chrominance info pure : normalising the tristimulus values

↓

chromaticity coordinates :

$$t_j = \frac{T_j}{T_1 + T_2 + T_3}$$

$t_1 + t_2 + t_3 = 1$ allows to eliminate one

2 chromaticity coordinates specify saturation and hue

Chromaticity coordinates

tristimulus values still contain brightness info

chrominance info pure : normalising the tristimulus values

↓

chromaticity coordinates :

$$t_j = \frac{T_j}{T_1 + T_2 + T_3}$$

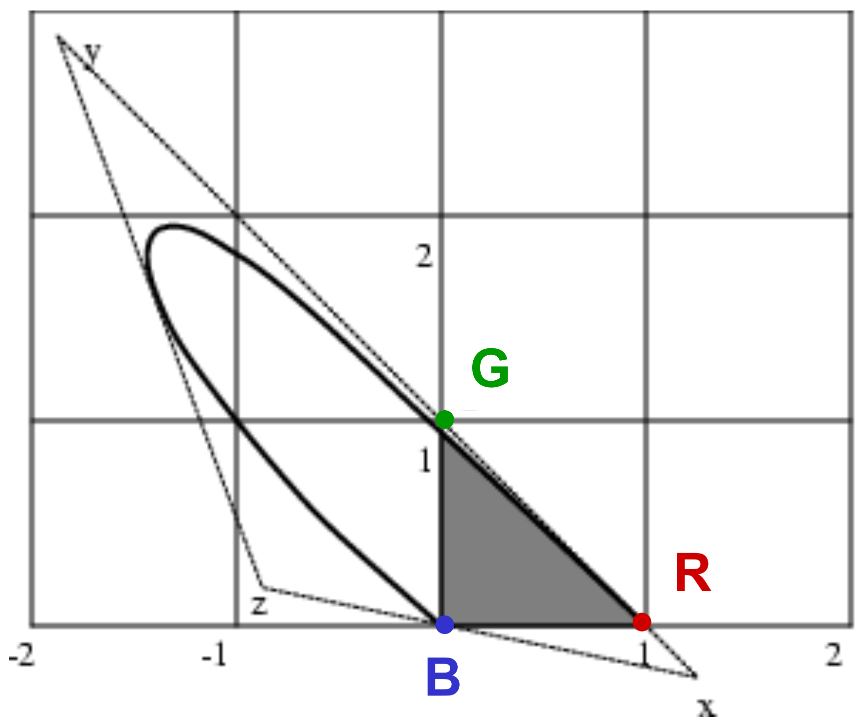
Note that for white $t_1 = t_2 = t_3 = 1/3$

CIE chromaticity diagram

chromaticity coordinates (r, g) for CIE primaries :

$$r = \frac{R}{R+G+B} \qquad g = \frac{G}{R+G+B}$$

The corresponding colour space :



CIE x-y coordinates

In order to get rid of the negative values :
virtual tristimulus colour system X,Y,Z
(no such physically realizable primaries exist !)

linear transf. from R,G,B to X,Y,Z coordinates :

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.490 & 0.310 & 0.200 \\ 0.177 & 0.813 & 0.011 \\ 0.000 & 0.010 & 0.990 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

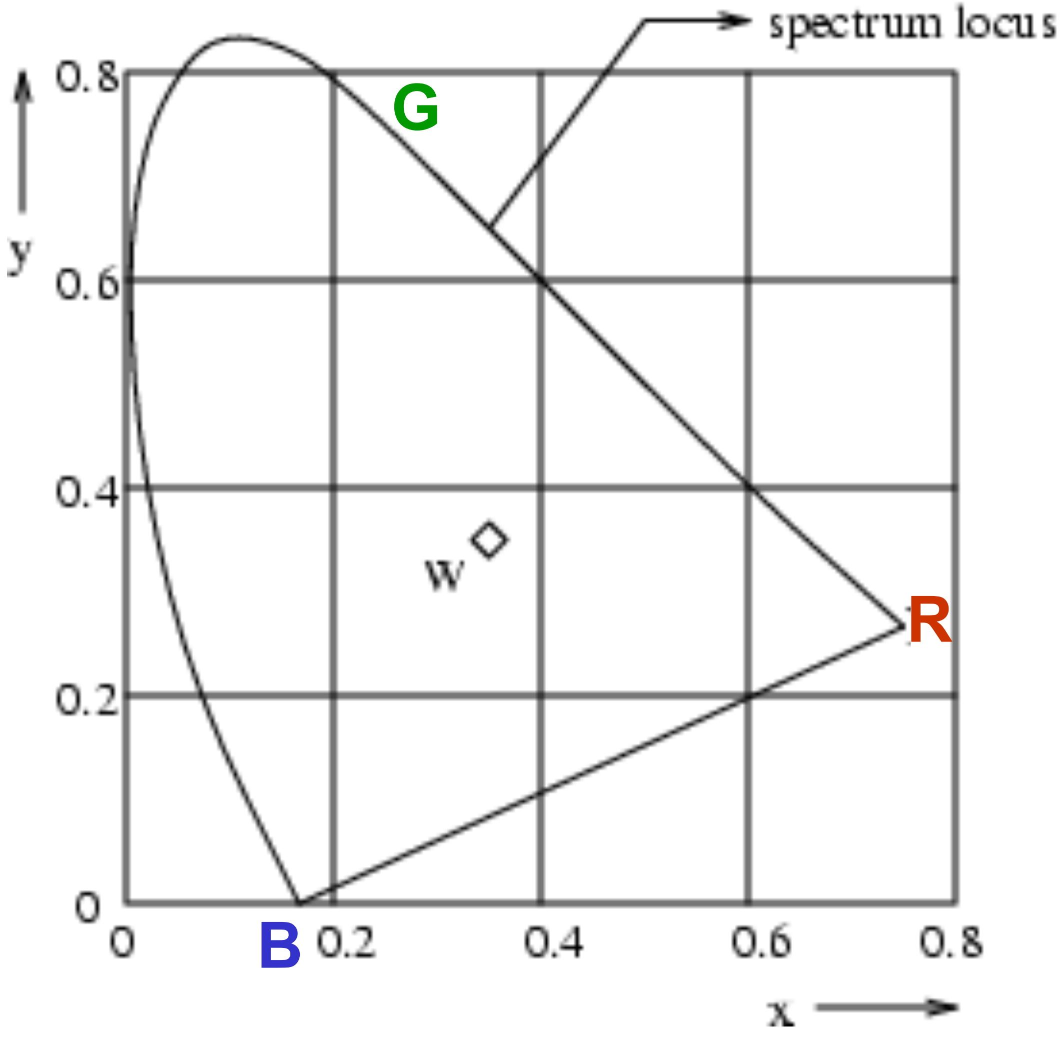
chosen as to make Y represent luminance

white (R=G=B=1) mapped to X=Y=Z=1

$$x = \frac{X}{X+Y+Z} \qquad y = \frac{Y}{X+Y+Z}$$

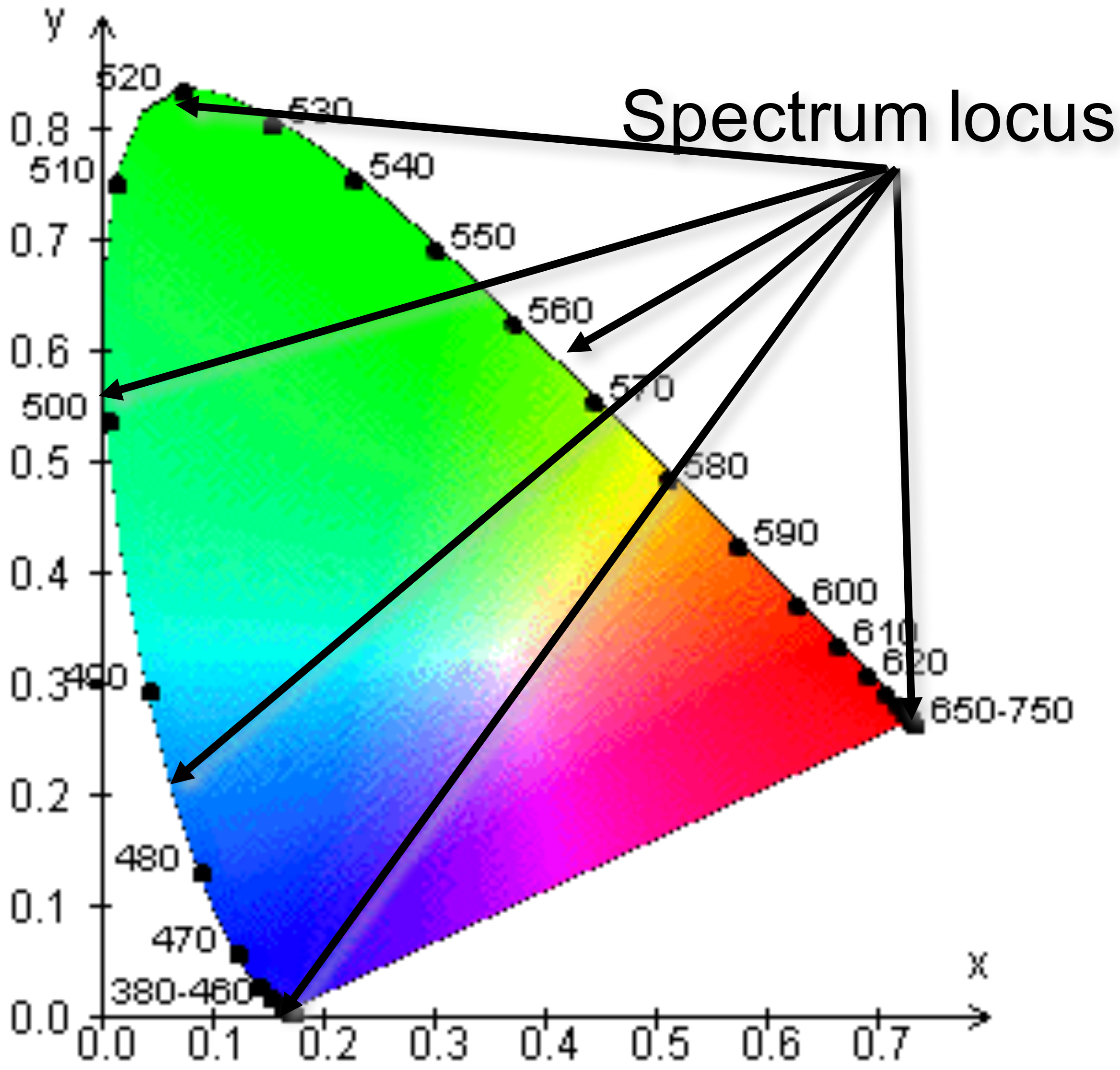
Computer Vision

CIE x,y colour triangle



Computer Vision

CIE x,y colour triangle



TV primaries

the EBU primaries have coordinates

$$R_r: \quad x = 0.64 \quad y = 0.33$$

$$G_r: \quad x = 0.29 \quad y = 0.60$$

$$B_r: \quad x = 0.15 \quad y = 0.06$$

the NTSC primaries have coordinates

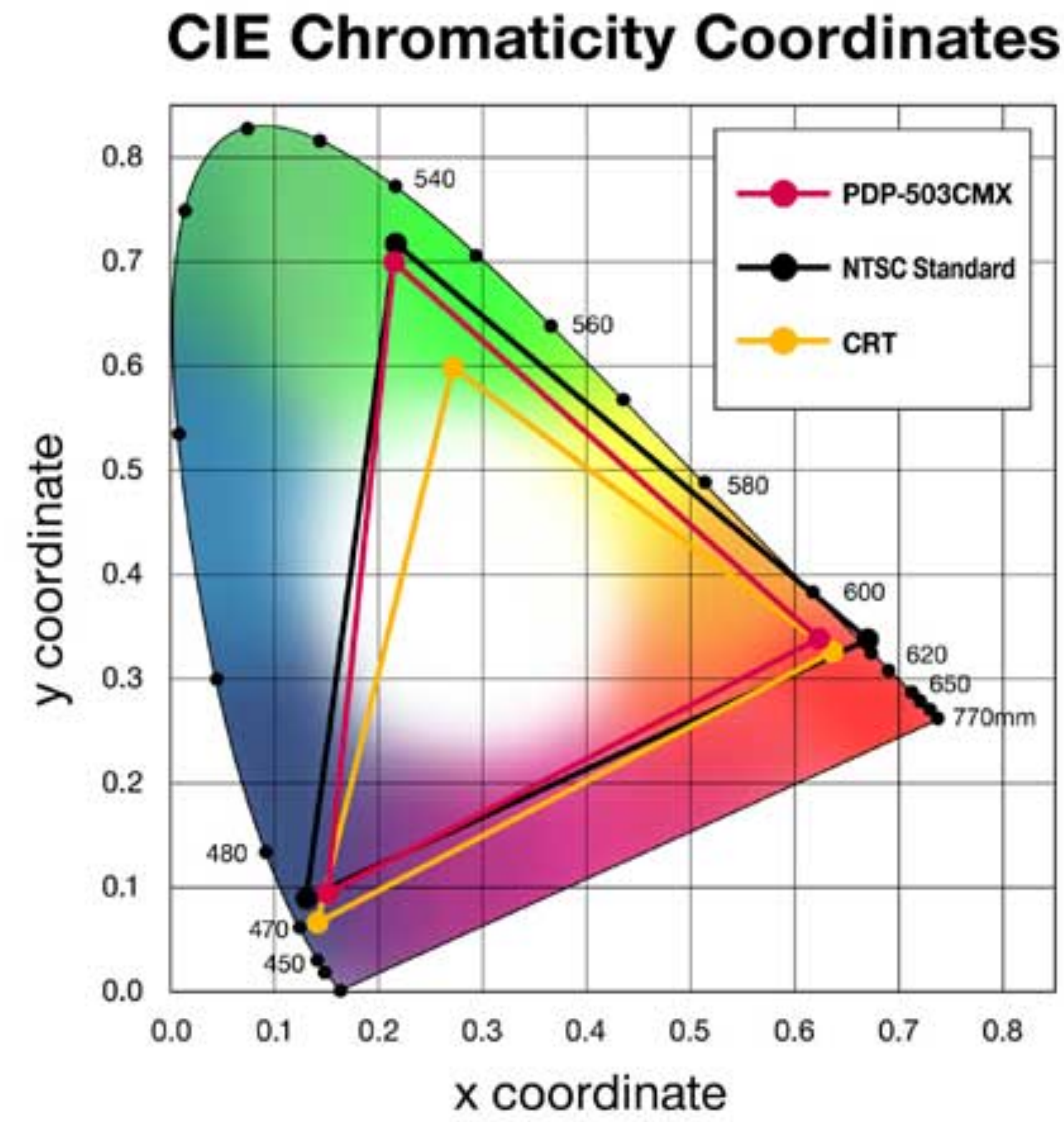
$$R_N: \quad x = 0.67 \quad y = 0.33$$

$$G_N: \quad x = 0.21 \quad y = 0.71$$

$$B_N: \quad x = 0.14 \quad y = 0.08$$

Computer Vision

primaries

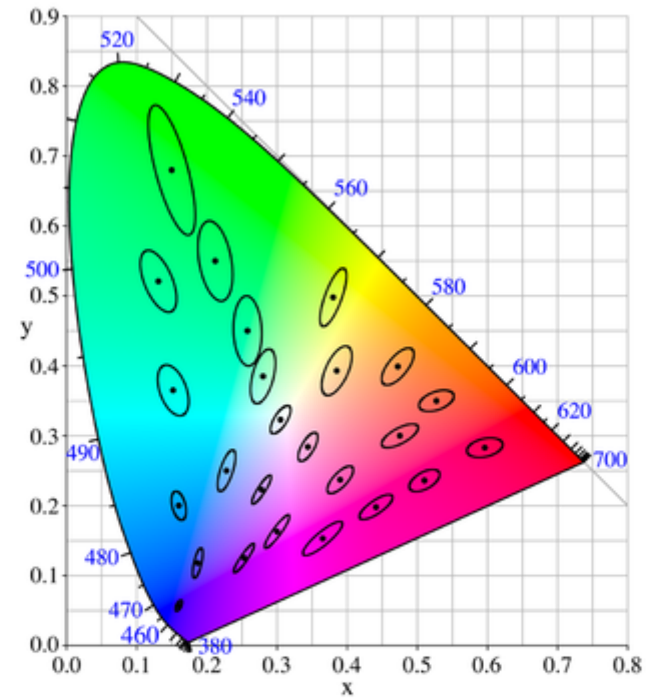


Notes

Minimize colours outside the triangle!

Area dubious criterion :
projective transf. between chromaticity coordinates

distance in triangle no faithful indication of perceptual difference



pure spectrum colours (on the spectrum locus) are rare in nature

Chromaticity coordinate transitions

So, colour coordinates need for their definition
3 primaries + white : 4 points

4 points define a projective frame :

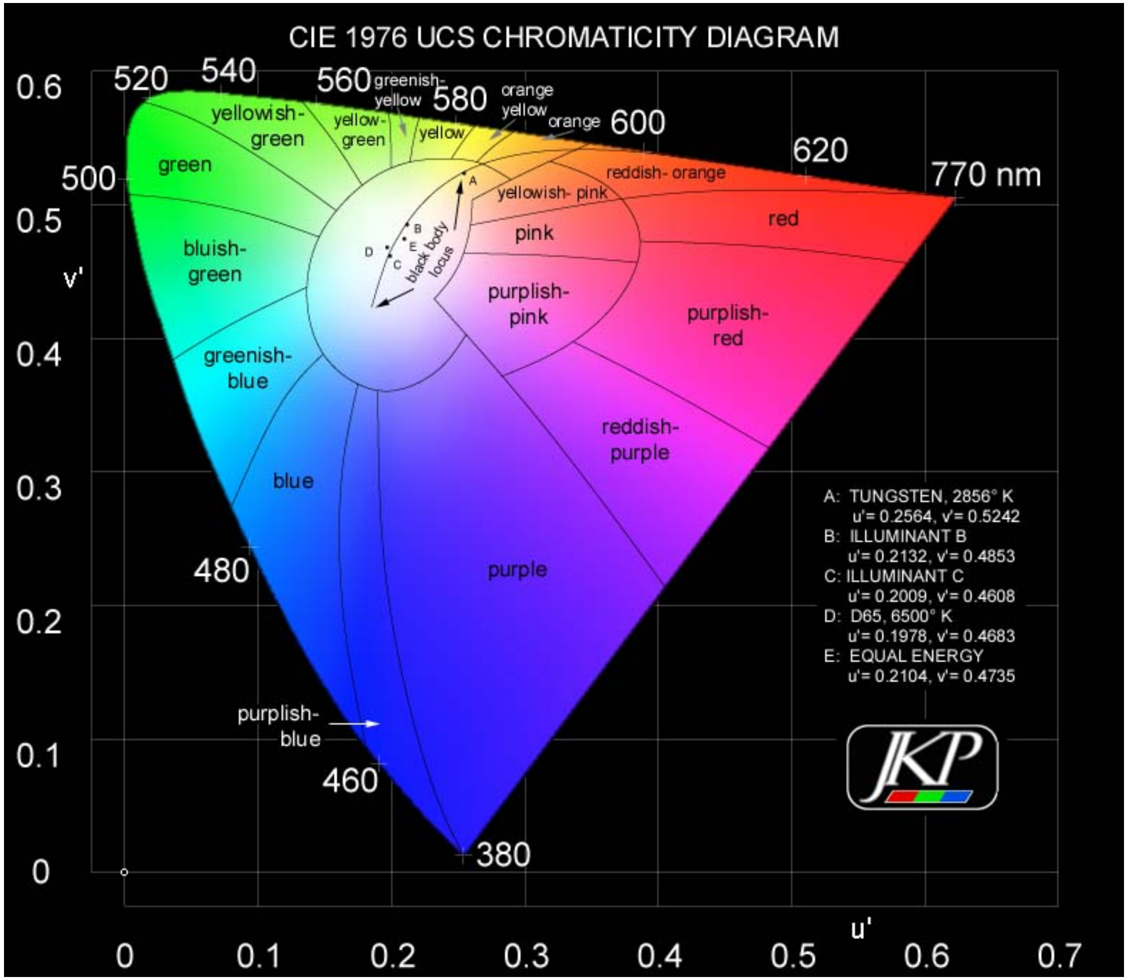
primaries $\Longrightarrow$ (0,0), (1,0), (0,1)
white $\Longrightarrow$ (0.33, 0.33)

a chromaticity coordinate transformation can be shown to be projective, i.e. non-linear

CIE u-v color coordinates

$u-v$ diagram +/- faithfully represents perceptual distance :

$$u = \frac{4x}{-2x+12y+3} \qquad v = \frac{6y}{-2x+12y+3}$$

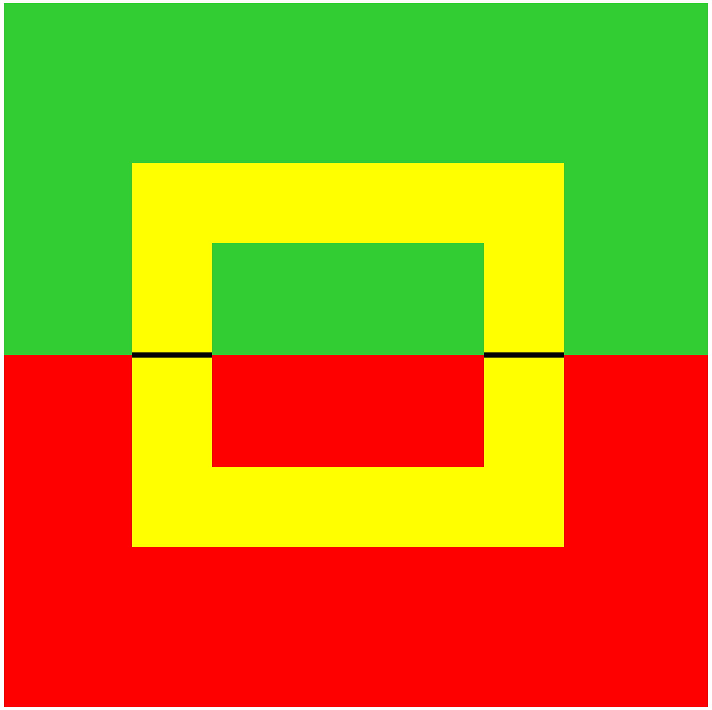


Using colour as a feature:

- colour constancy
- illumination invariant colour features

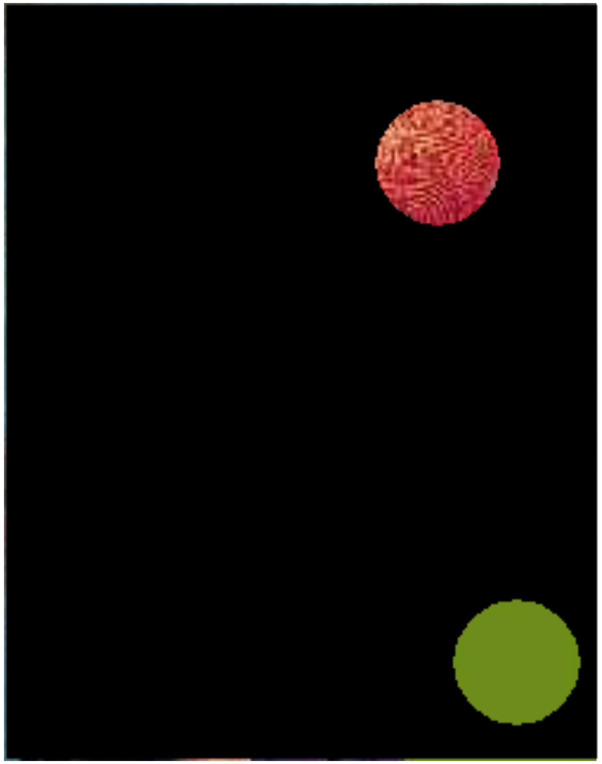
Computer Vision

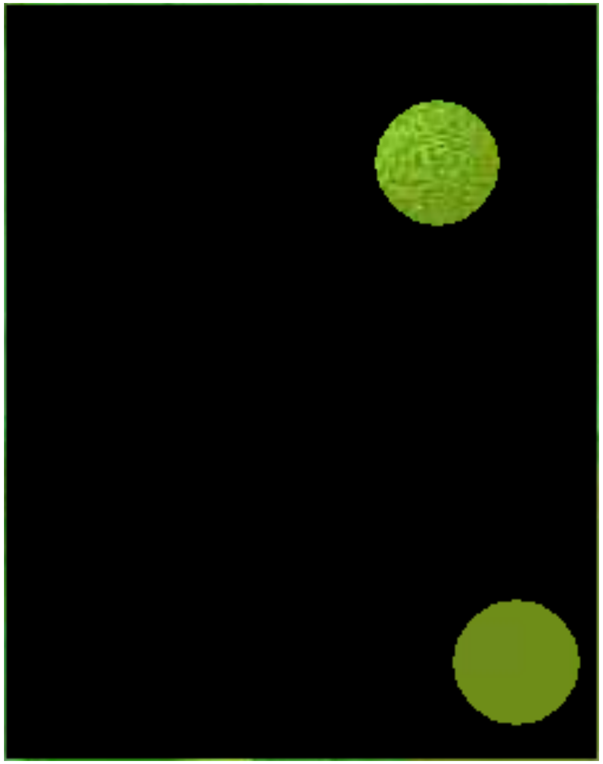
Koffka ring with colours



Computer Vision

Colour constancy





Colour constancy

Patches keep their colour appearance even if they reflect differently (e.g. the hat)

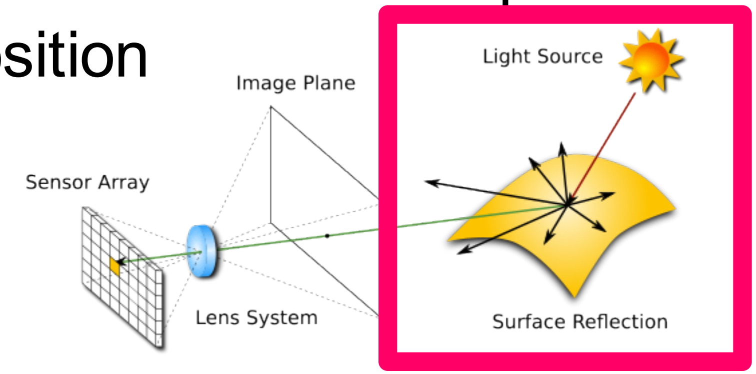
Patches change their colour appearance if they reflect identically but surrounding patches reflect differently (e.g. the background)

There is more to colour perception than 3 cone responses

Edwin Land performed in-depth experiments (psychophysics)

Colour constancy - notes

The colour of a surface is the result of the product of spectral reflectance and spectral light source composition



Our visual system can from a single product determine the two factors, it seems

The colour of the light source can be guessed via that of specular reflections, but the visual system does not critically depend on this trick

Computer Vision

On the menu:

- colour constancy

- illumination invariant colour features

Illumination invariant colour features

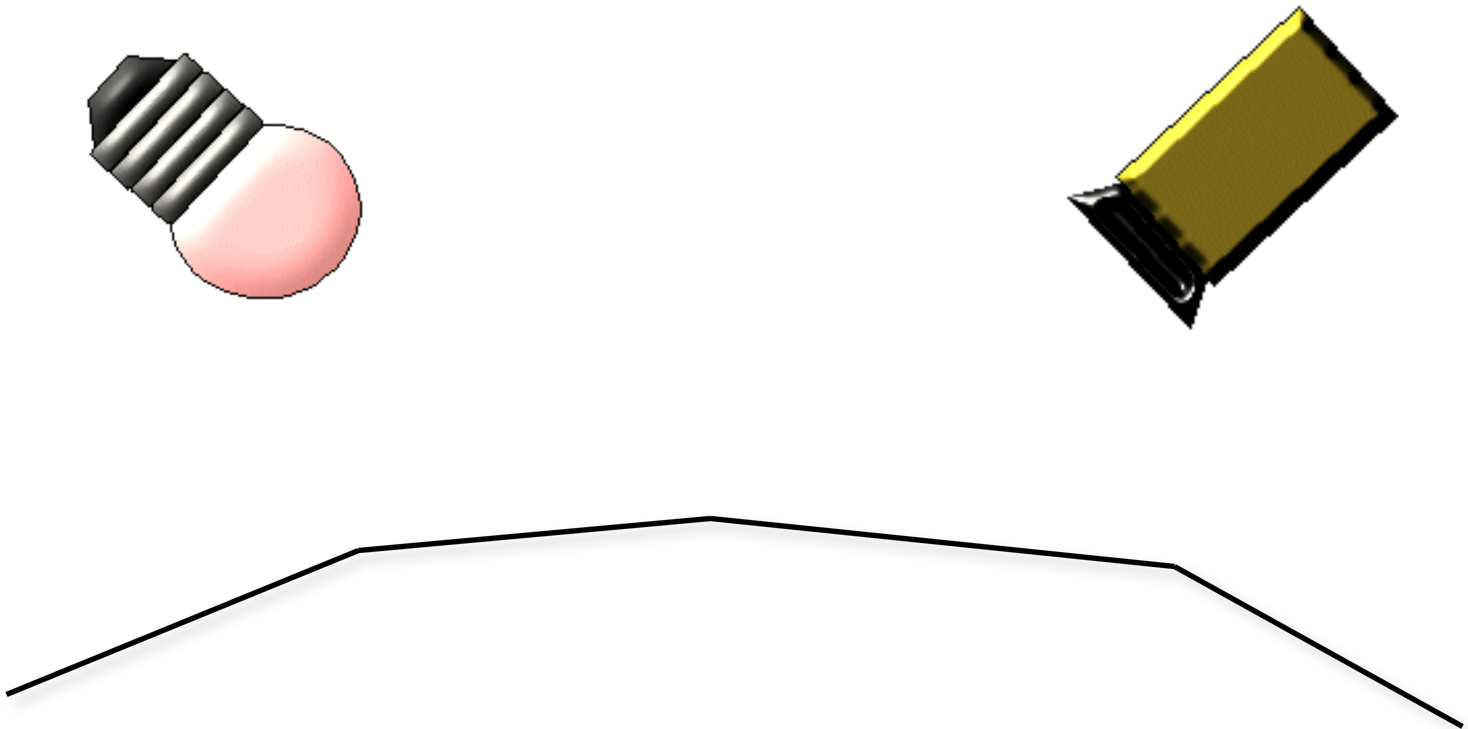
Extracting the true surface colour under varying illumination - as the HVS can - is very difficult

A less ambitious goal is to extract colour features that do not change with illumination

1) Spectral or `internal' changes
2) Geometric or `external' changes
3) Spectral + geometric changes

Illumination invariant colour features

1) Spectral changes



Illumination invariant colour features

1) Spectral changes

Let $I_{R,} I_{G,} I_B$ represent the irradiances at the camera for red, green, blue

A simple model: the irradiances change by

$$\alpha, \beta, \gamma : \left(I'_R, I'_G, I'_B\right) = \left(\alpha\, I_R, \beta\, I_G, \gamma\, I_B\right)$$

Consider irradiances at 2 points:
I_{R1}, I_{G1}, I_{B1} and I_{R2}, I_{G2}, I_{B2}

$$I'_{R1} / I'_{R2} = \left(\alpha I_{R1}\right) / \left(\alpha I_{R2}\right) = I_{R1} / I_{R2}$$

Illumination invariant colour features

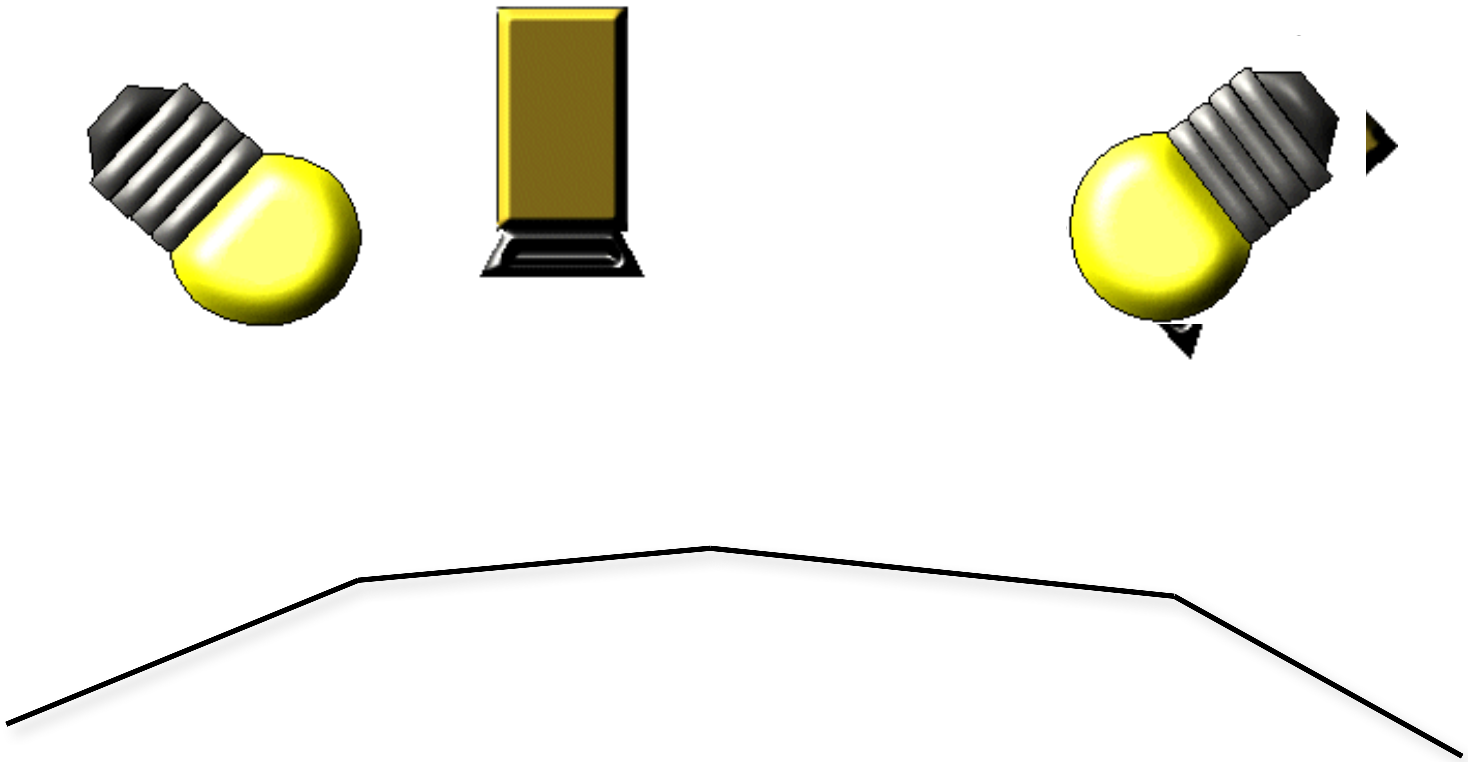
For a camera with a non-linear response:

$$(\alpha I_{R1})^{\gamma}/(\alpha I_{R2})^{\gamma} = (I_{R1})^{\gamma}/(I_{R2})^{\gamma}$$

Computer Vision

Illumination invariant colour features

2) Geometric changes



Computer Vision

Illumination invariant colour features

1) Geometric changes

$$\left(I'_R, I'_G, I'_B\right) = \left(s(x,y)\,I_R,\ s(x,y)\,I_G,\ s(x,y)\,I_B\right)$$

$$I'_R / I'_G = I_R / I_G$$

and

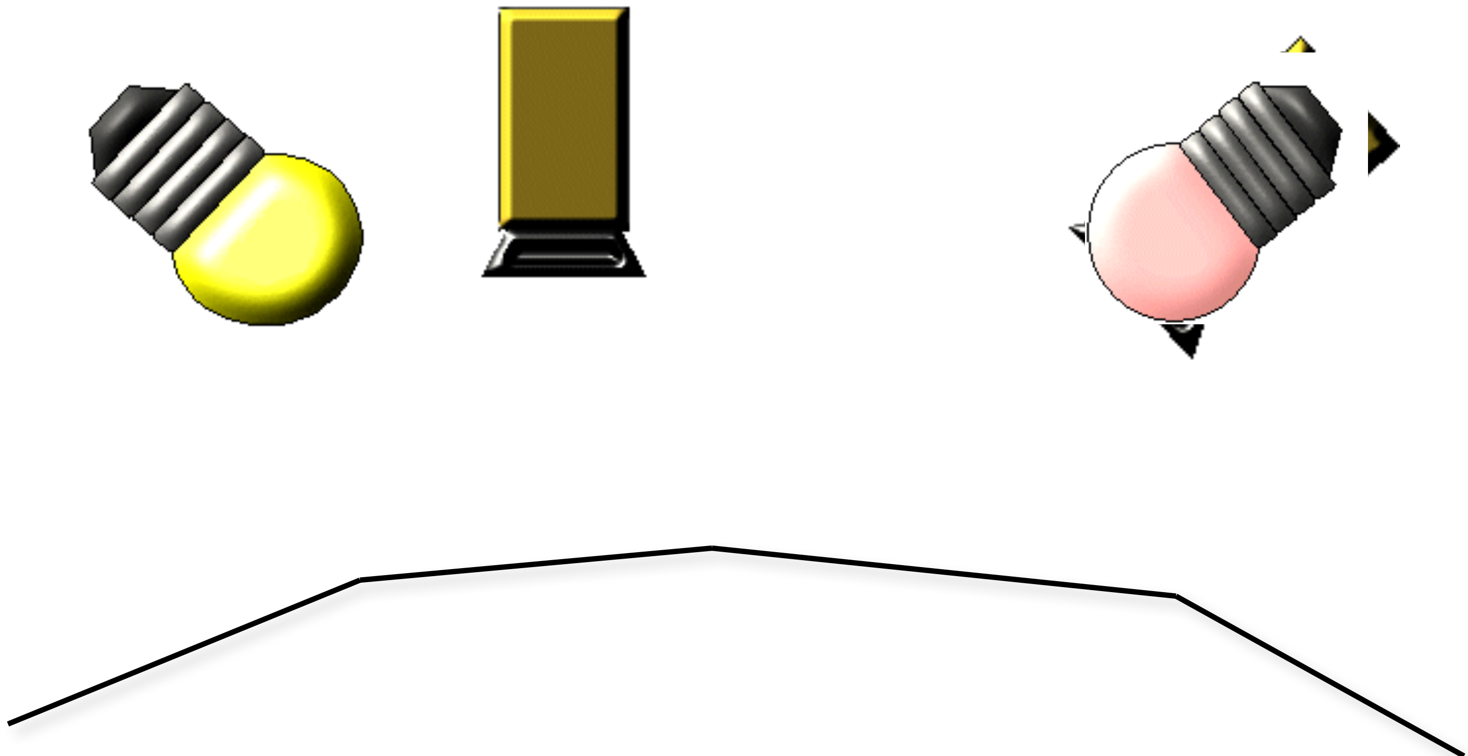
$$I'_R / I'_B = I_R / I_B$$

are invariant

Computer Vision

Illumination invariant colour features

3) Spectral + geometric changes



Illumination invariant colour features

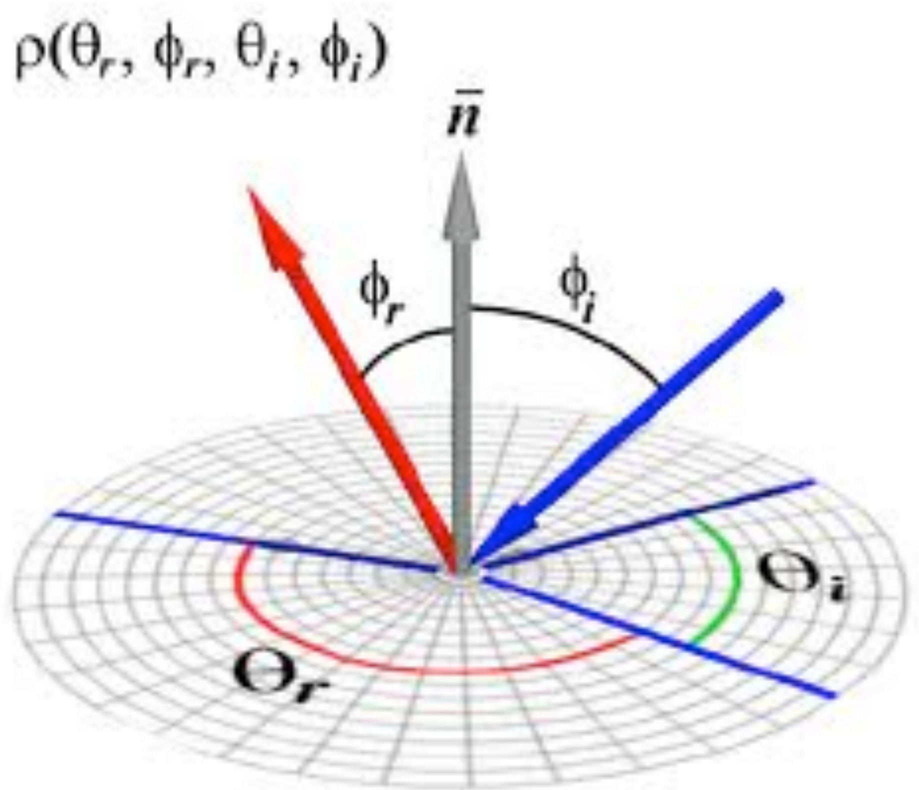
3) Geometric + spectral changes

$$\frac{I'_{R1} I'_{G2}}{I'_{R2} I'_{G1}} = \frac{\alpha\, s(x_1, y_1) I_{R1}\ \beta\, s(x_2, y_2) I_{G2}}{\alpha\, s(x_2, y_2) I_{R2}\ \beta\, s(x_1, y_1) I_{G1}} = \frac{I_{R1} I_{G2}}{I_{R2} I_{G1}}$$

for points on both sides of a colour edge $s(x_1, y_2) \cong s(x_2, y_2)$ and hence I_{R1} / I_{R2} is invariant

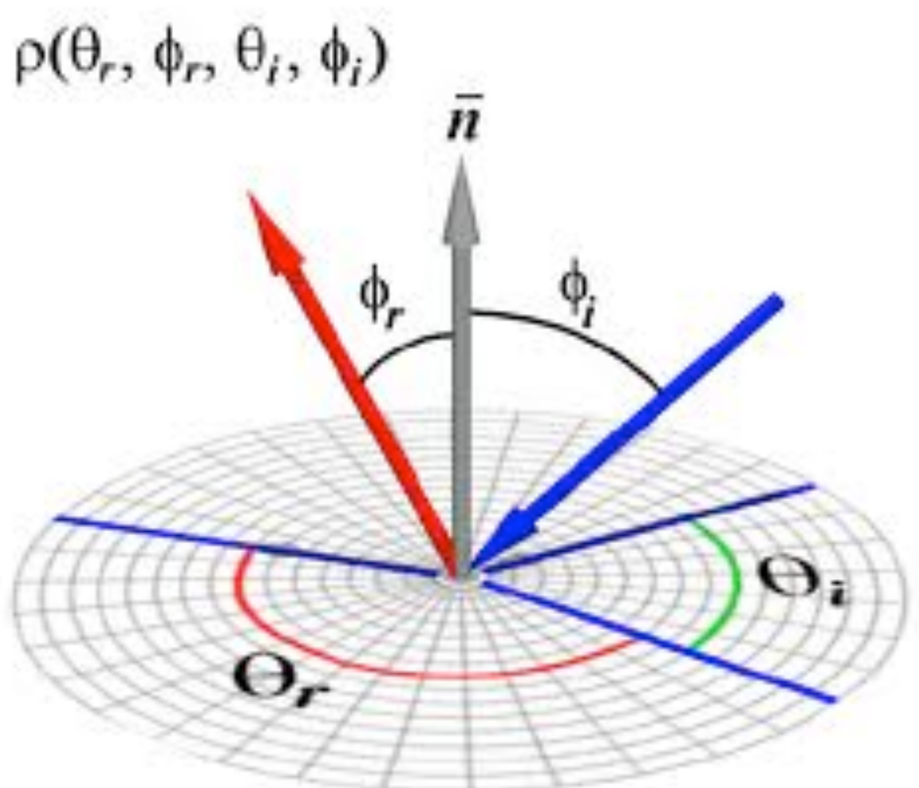
The elusive BRDF

Bidirectional Reflection Distribution Function
…. for different wavelengths



The elusive BRDF

A 4D function, specifying the radiance for an outgoing direction given an irradiance for an incoming direction, relative to the normal and ideally for 1 wavelength at a time



Mini-dome to study reflectance

KATHOLIEKE UNIVERSITEIT
LEUVEN

Computer Vision

Mini-dome to study reflectance

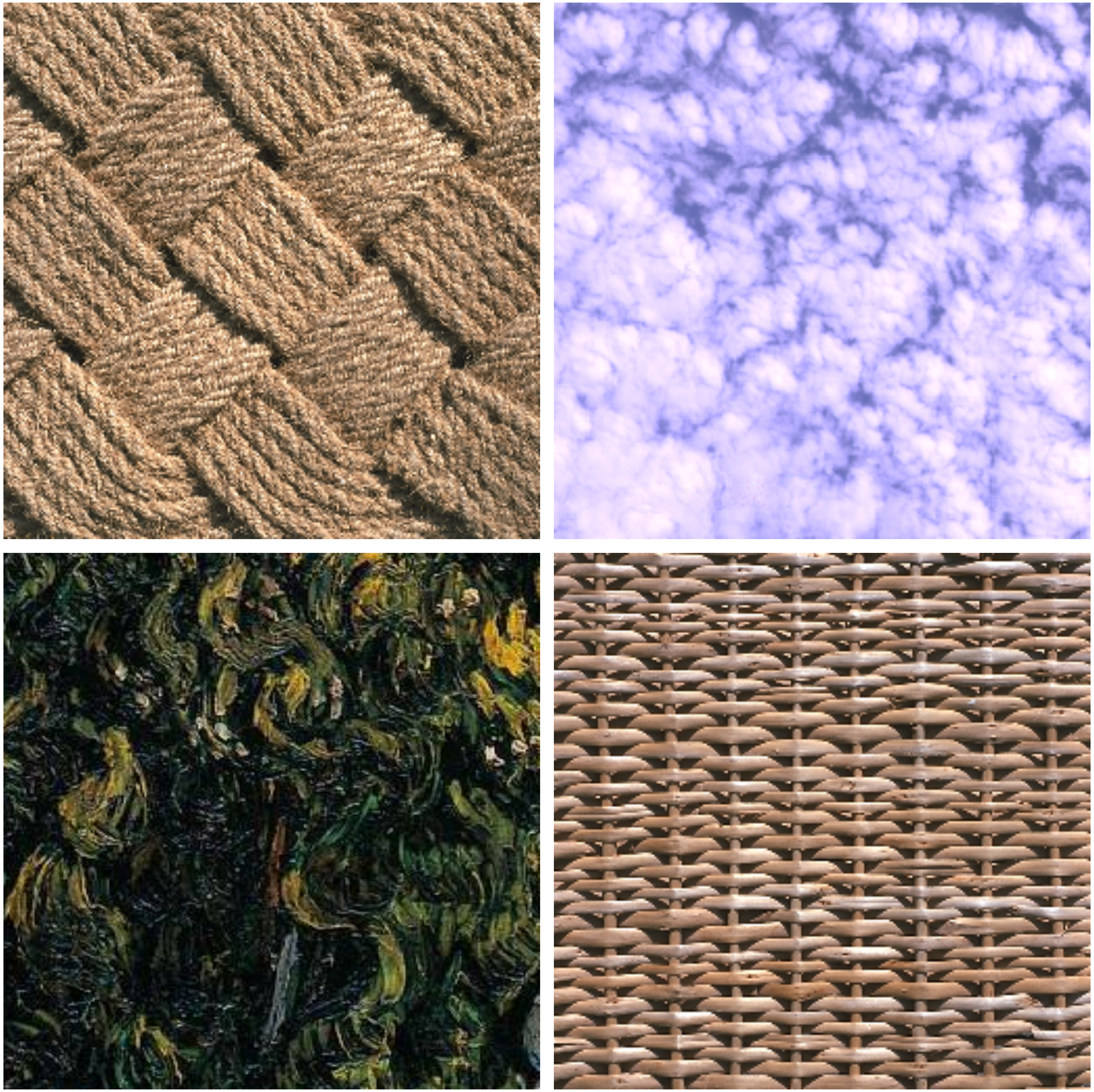
Mini-dome to study reflectance

REAL SIMULATED

Computer Vision

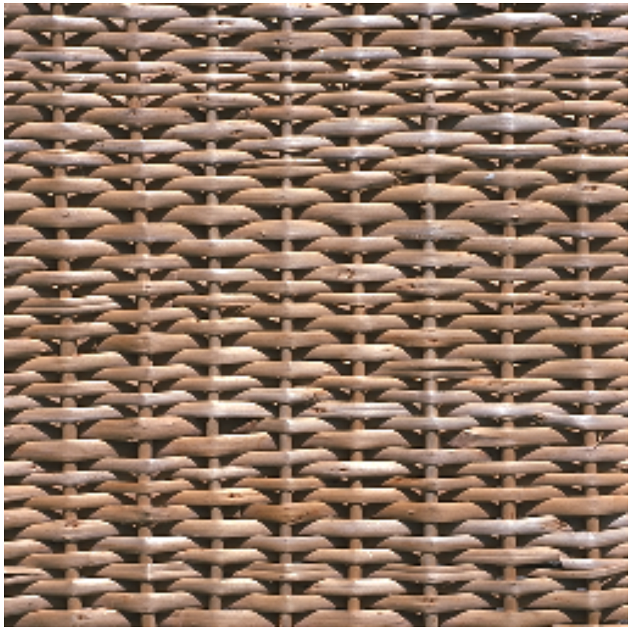
TEXTURE

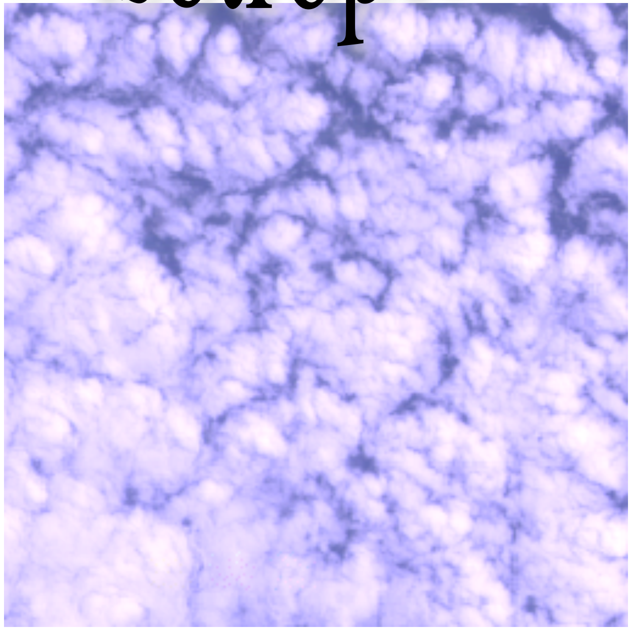
Example textures



Texture characteristics

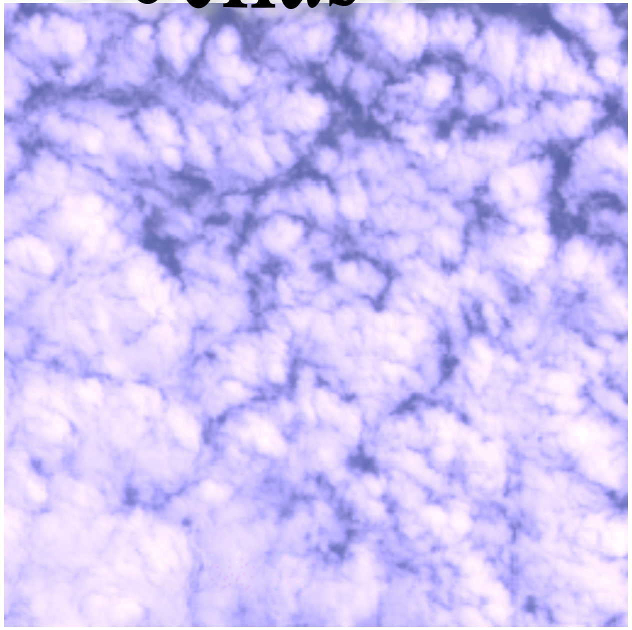
oriented vs. isotropic





Texture characteristics

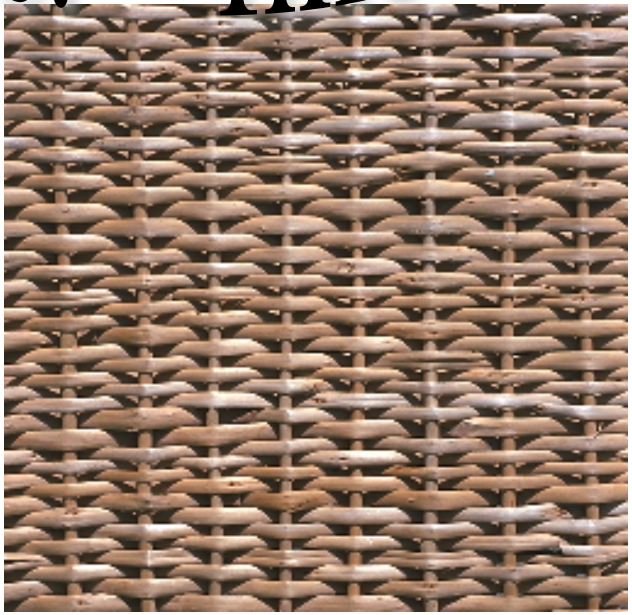
regular vs. stochastic



Computer Vision

Texture characteristics

coarse vs. fine



On the menu:

- Fourier features

- cooccurrence matrices

- filter banks (Laws, Gabor, eigenfilters)

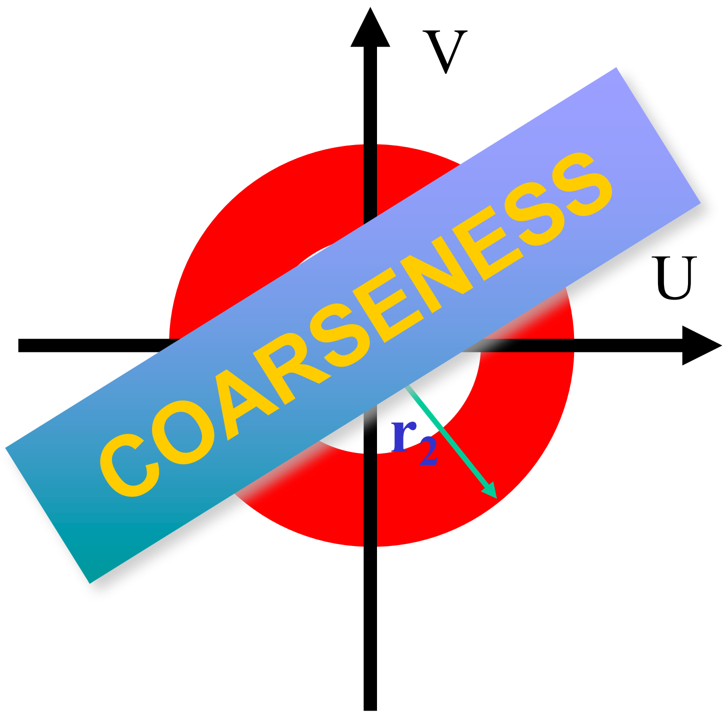
- stochastic models)

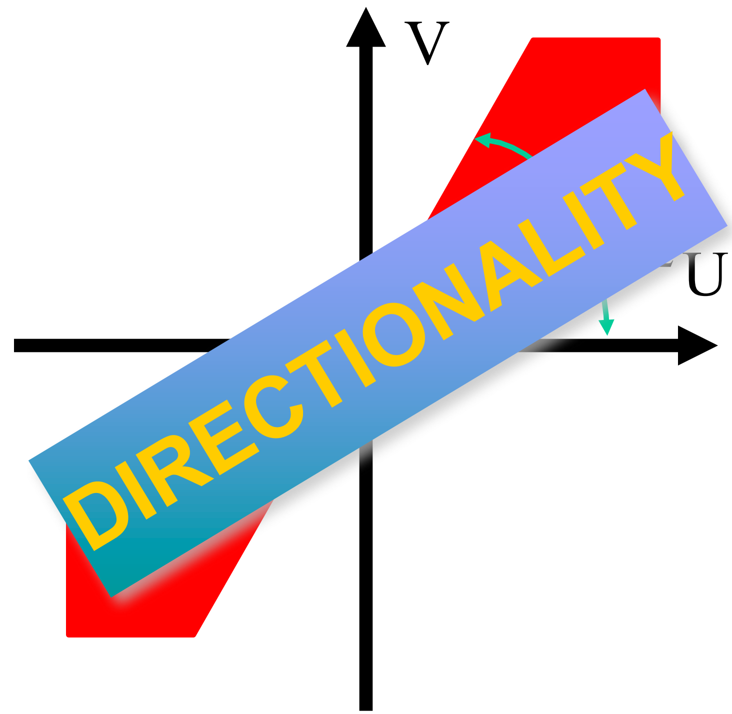
Fourier features

Based on the integration of regions of the Fourier power spectrum $\int_A \int \left| F(u,v) \right|^2 du dv$

Intuitively appealing
- peaks if periodic
- mostly low/high freq. if coarse resp. fine
- the sine patterns each have an orientation

Fourier features





$$r_1^2 \leq u^2 + v^2 < r_2^2$$

$$\theta_1 \leq \arctan\left(\frac{u}{v}\right) < \theta_2$$

Computer Vision

Fourier features

THE FOURIER TRANSFORM COLLECTS INFORMATION GLOBALLY OVER THE ENTIRE IMAGE

NOT GOOD FOR SEGMENTATION OR INSPECTION

On the menu:

- Fourier features
- cooccurrence matrices
- filter banks (Laws, Gabor, eigenfilters)
- stochastic models

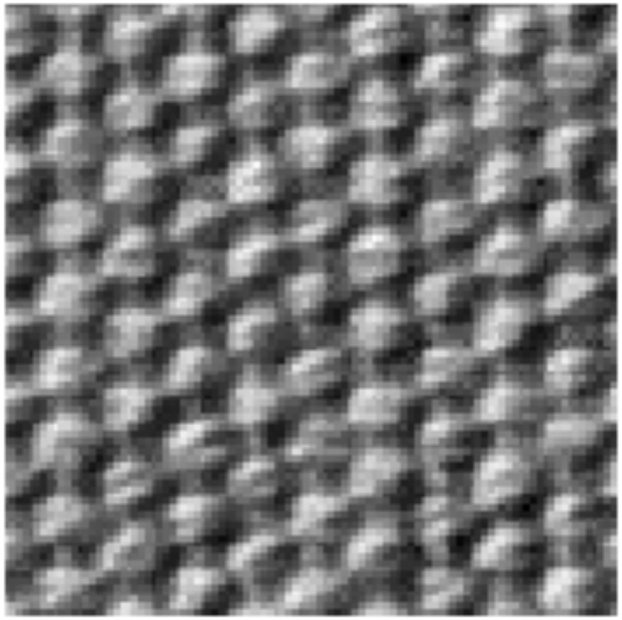
Histograms: principle

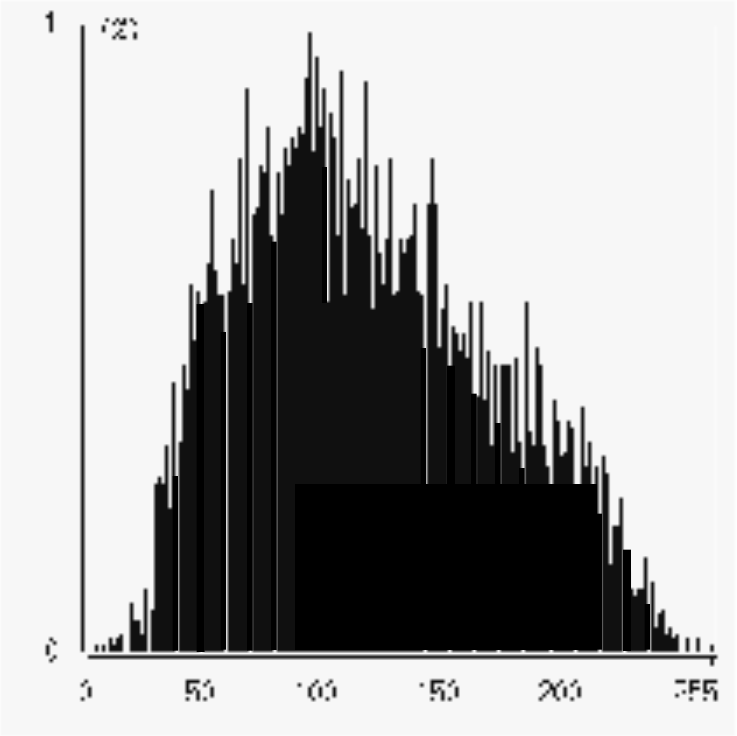
Intensity probability distribution

Captures global brightness information in a compact, but incomplete way

Doesn’t capture spatial relationships

Histograms : example



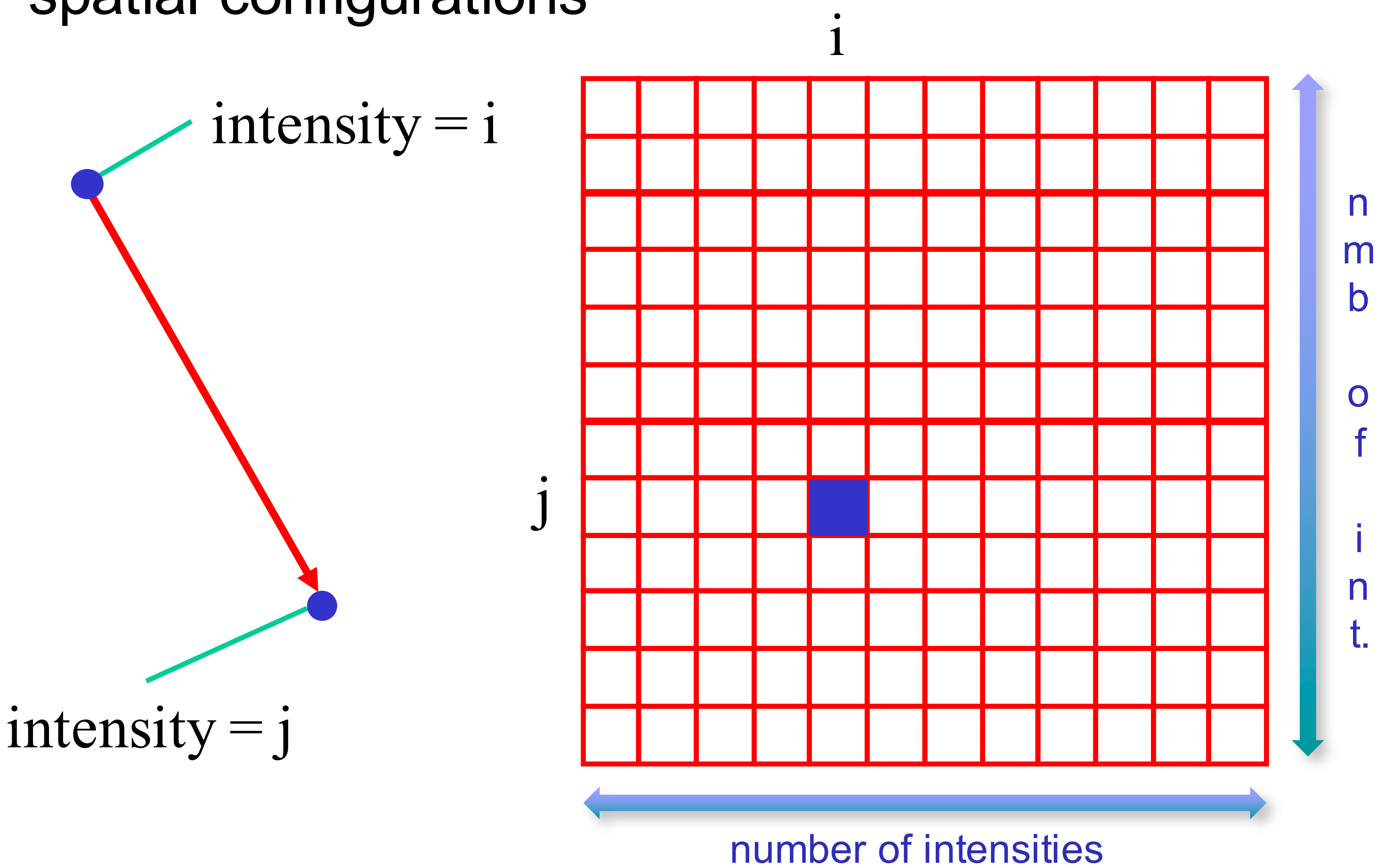


Computer Vision

Cooccurrence matrix

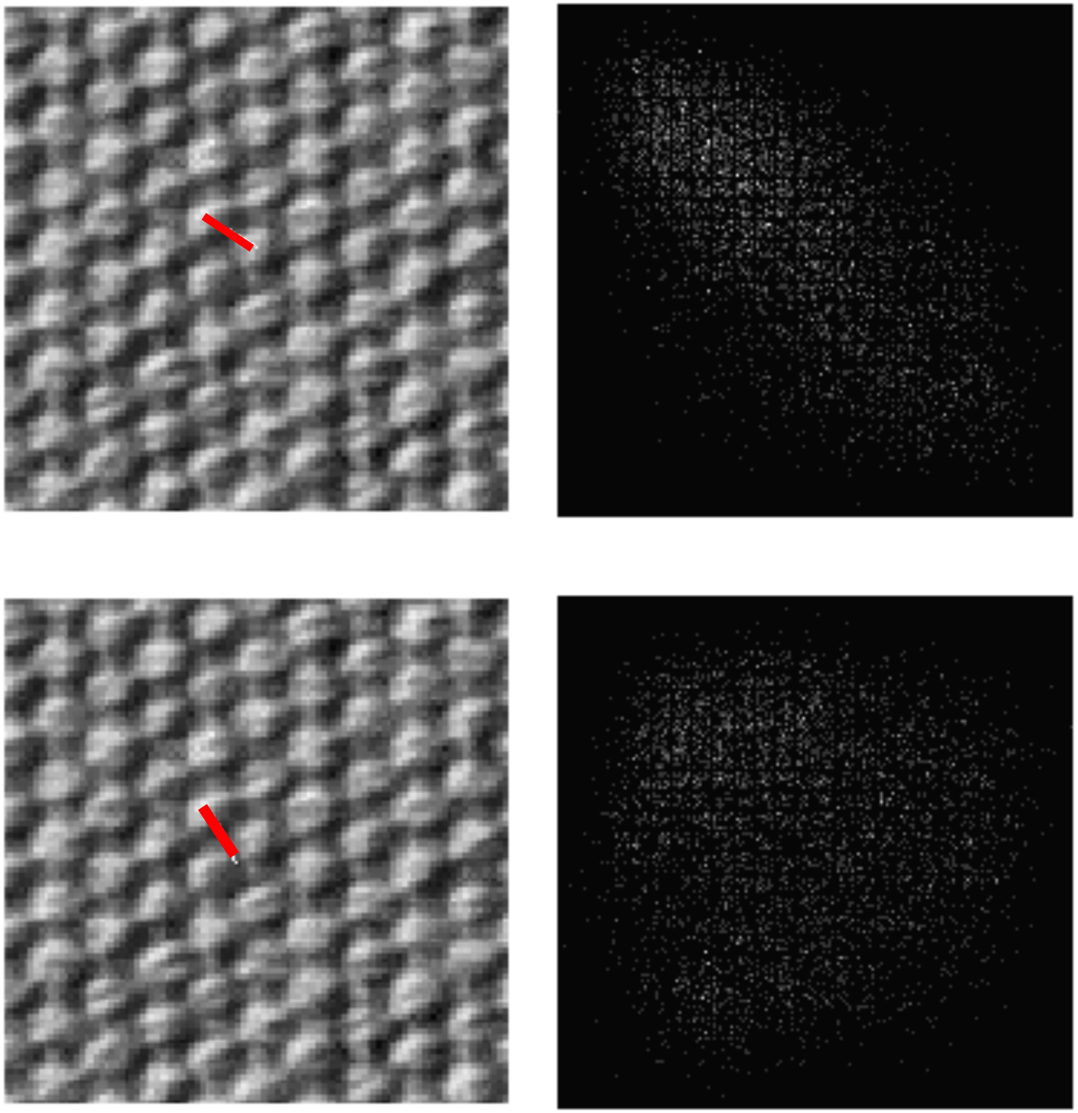
probability distributions for intensity pairs

Contains information on some aspects of the spatial configurations



Computer Vision

Cooccurrence matrix



Cooccurrence matrix

Features calculated from the matrix:

feature	expression
energy	$\sum_i \sum_j C^2(i,j)$
entropy	$-\sum_i \sum_j C(i,j) \log C(i,j)$
contrast	$\sum_i \sum_j (i-j)^2 C(i,j)$
homogeneity	$\sum_i \sum_j \mathrm{C}(\mathrm{i},\mathrm{j}) / (1+\lvert \mathrm{i}-\mathrm{j} \rvert)$
max. probability	$\max_{i,j} C(i,j)$

On the menu:

- Fourier features
- cooccurrence matrices
- filter banks (Laws, Gabor, eigenfilters)
- stochastic models)

Computer Vision

Laws filters

Feature	1D filter
L3	[1 2 1]
E3	[-1 0 1]
S3	[-1 2 -1]
L5	[1 4 6 4 1]
E5	[-1 -2 0 2 1]
S5	[-1 0 2 0 -1]
W5	[-1 2 0 -2 1]
R5	[1 -4 6 -4 1]
L7	[1 6 15 20 15 6 1]
E7	[-1 -4 -5 0 5 4 1]
S7	[-1 -2 1 4 1 -2 -1]
W7	[-1 0 3 0 -3 0 1]
R7	[1 -2 -1 4 -1 -2 1]
O7	[-1 6 -15 20 -15 6 -1]

Laws filters

This fixed filter set yields simple convolutions but has proven very effective in some cases

Gabor filters

Gaussian envelope multiplied by cosine

$$g(x,y) = e^{-\frac{x^2+y^2}{4\Delta_{x,y}^2}} \cos\left(2\pi u^* x + \varphi\right)$$

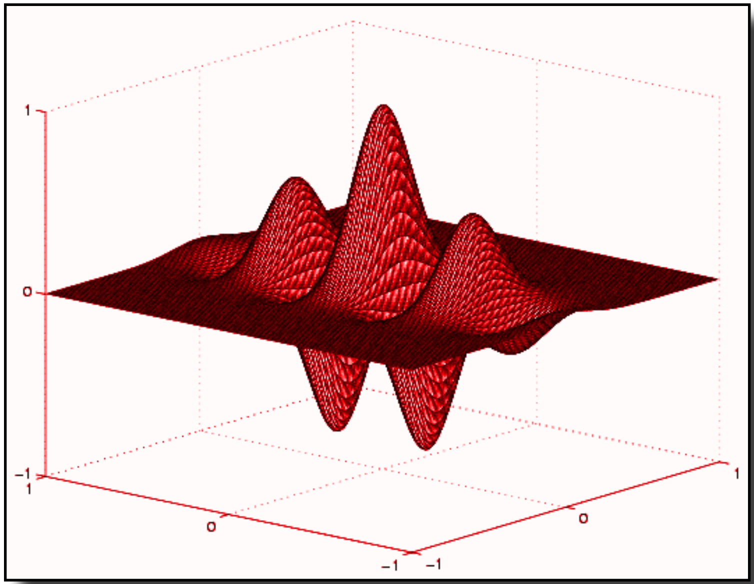
The filter's Fourier power spectrum

$$G(u,v) =$$

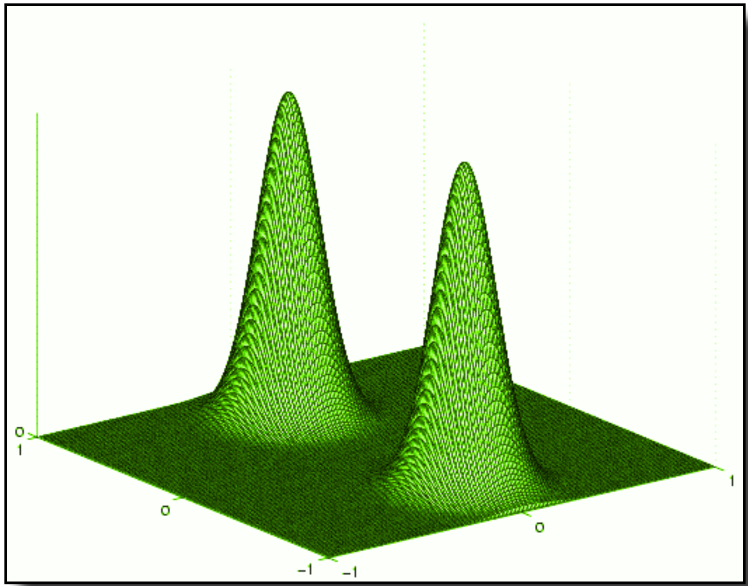
$$\frac{1}{4\pi\Delta_{u,v}^2}\left(e^{-\left(\left(u-u^*\right)^2+v^2\right)/\left(4\Delta_{u,v}^2\right)} + e^{-\left(\left(u+u^*\right)^2+v^2\right)/\left(4\Delta_{u,v}^2\right)}\right)$$

Gabor filters

Spatial domain



Frequency domain



Good localisation in both domains

Computer Vision

Gabor filters

$$f = f(x, y) \qquad F = F(u, v)$$

$$x_{av} = \frac{\int_{-\infty}^{\infty} x\, f\, \bar{f}\, dx}{\int_{-\infty}^{\infty} f\, \bar{f}\, dx} \qquad u_{av} = \frac{\int_{-\infty}^{\infty} uF\bar{F}\, du}{\int_{-\infty}^{\infty} F\bar{F}\, du}$$

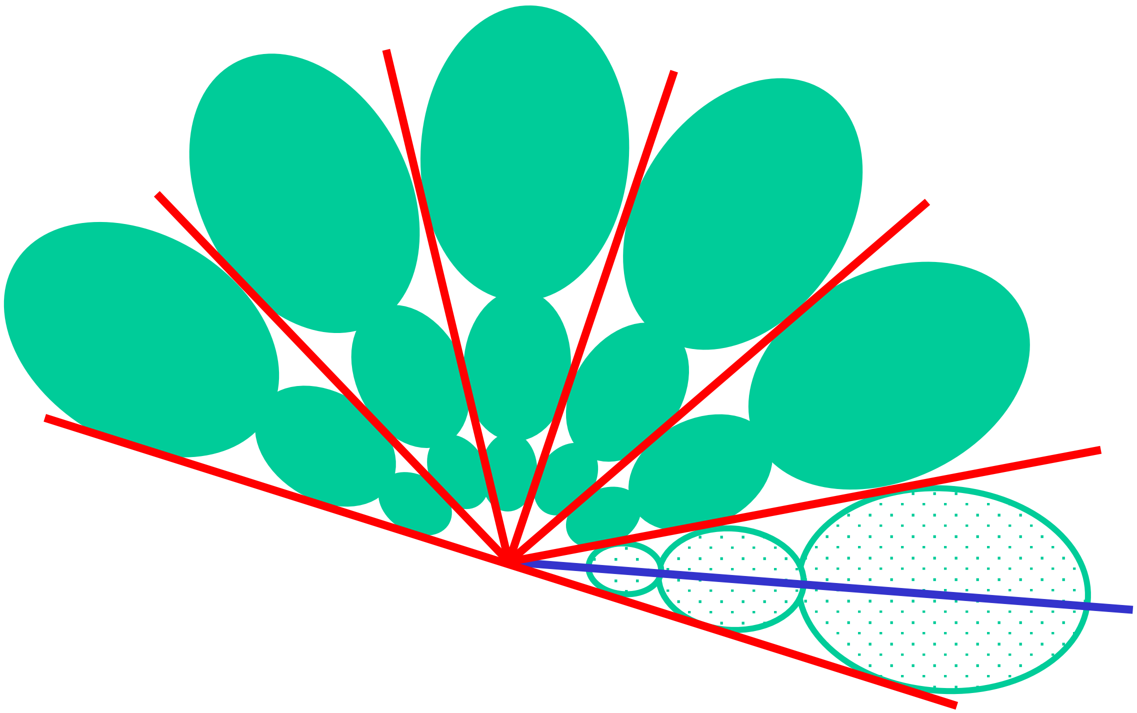
$$(\Delta x)^2 = \frac{\int_{-\infty}^{\infty} (x - x_{av})^2 f\, \bar{f} dx}{\int_{-\infty}^{\infty} f\, \bar{f} dx} \qquad (\Delta u)^2 = \frac{\int_{-\infty}^{\infty} (u - u_{av})^2 F\, \bar{F} du}{\int_{-\infty}^{\infty} F\, \bar{F} du}$$

the Heisenberg uncertainty principle

$$\Delta x\, \Delta u = 1/4\pi$$

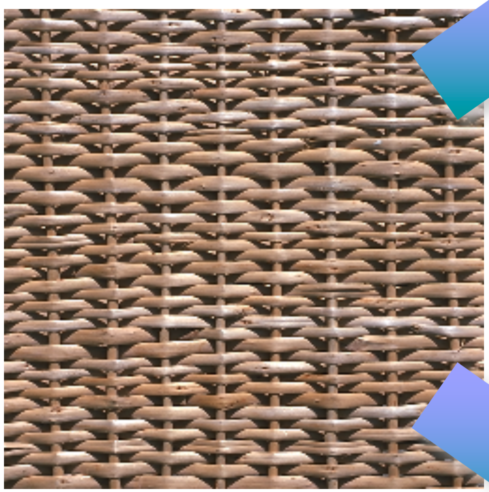
Gabor filters

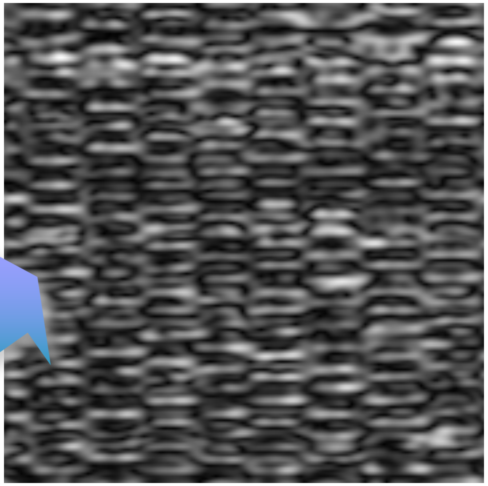
Covering the Fourier domain with responses
- to probe for directionality
- to look at different scales



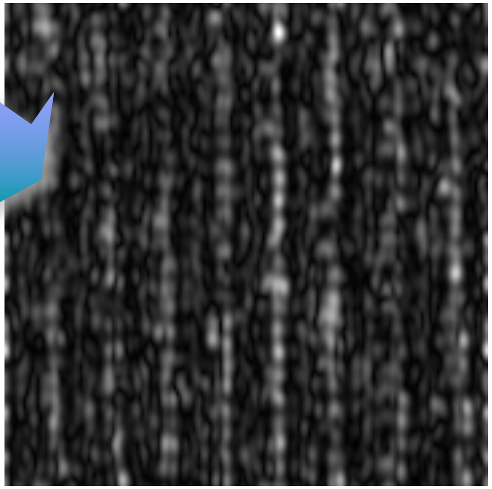
Gabor filters

Input texture





Output for filter responsive to horizontal structures



Output for filter responsive to vertical structures

Eigenfilters (Ade, ETH)

Filters adapted to the texture

1) shift mask over training image

2) collect intensity statistics

3) PCA -> eigenvectors -> `eigenfilters’

4) energies of eigenfilter outputs

Eigenfilters

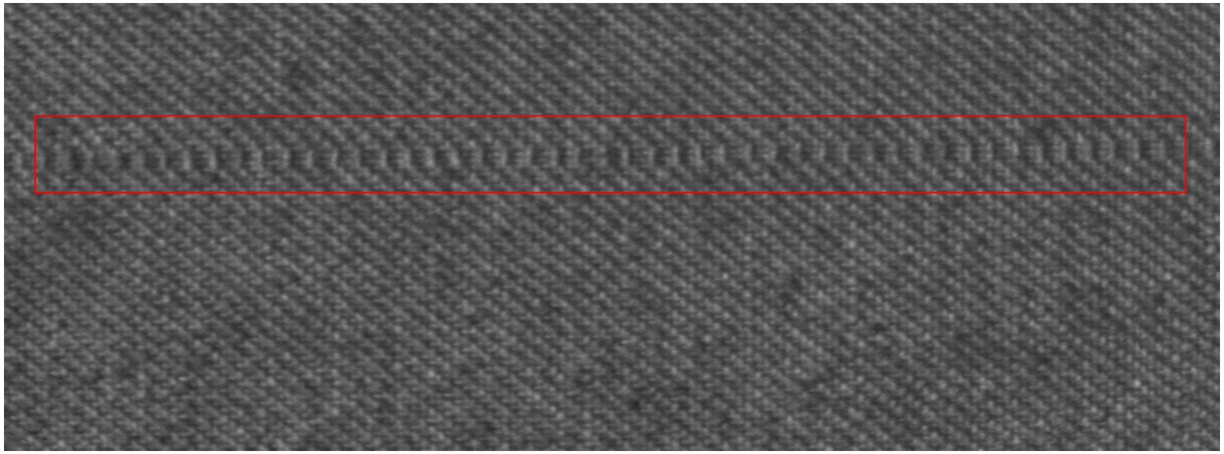
Filters adapted to the texture

but small filters may reduce efficacy

hence large, but sparse filters

Eigenfilters

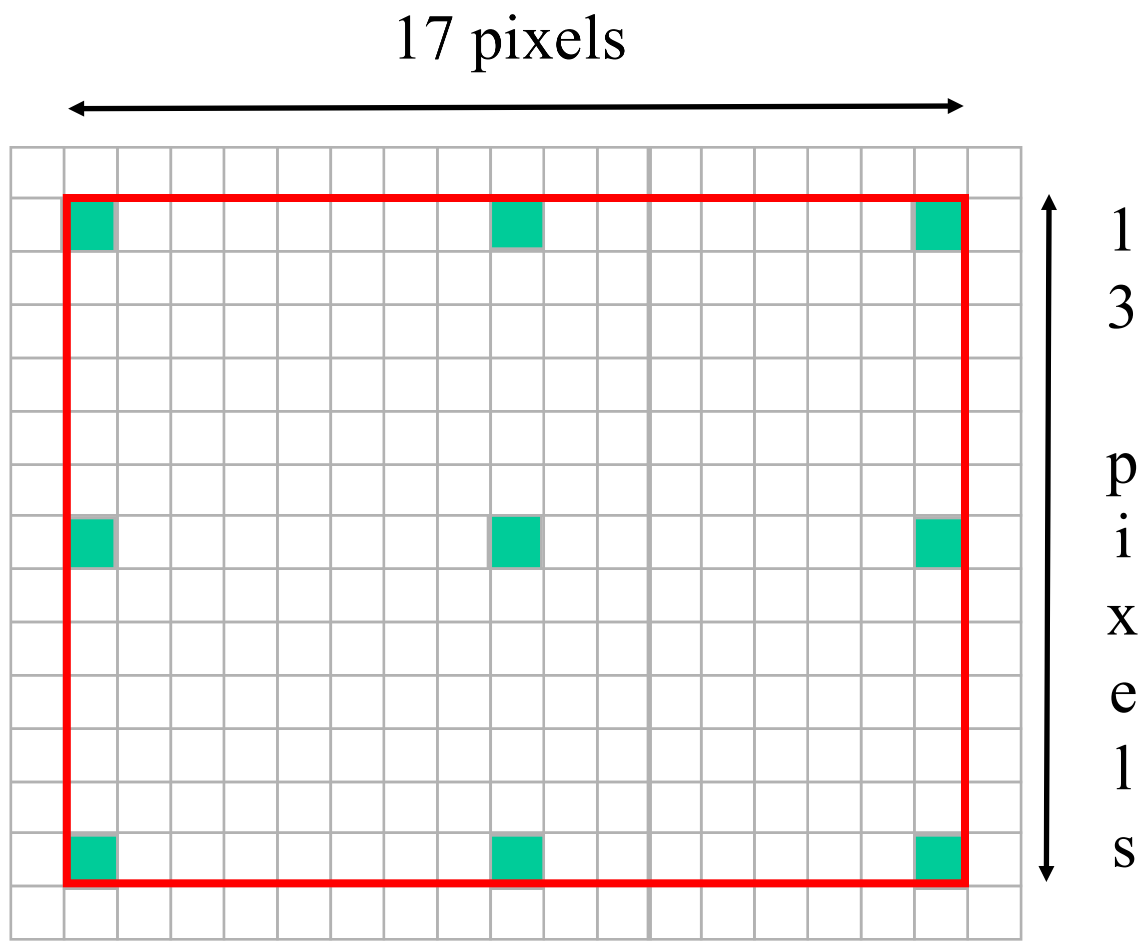
Example applications: textile inspection



Filters with size of one period

(period found as peak in autocorrelation)

Eigenfilters



Computer Vision

Eigenfilters

Covariance matrix needed for PCA

$$\begin{pmatrix}
174.1 & -77.6 & 101.6 & -60.8 & 72.7 & -71.5 & 116.5 & -77.9 & 91.4 \\
-77.6 & 173.9 & -78.2 & 71.5 & -61.7 & 73.1 & -76.4 & 116.4 & -78.4 \\
101.6 & -78.2 & 173.5 & -70.4 & 71.7 & -62.1 & 95.3 & -77.0 & 116.3 \\
-60.8 & 71.5 & -70.4 & 173.9 & -76.5 & 101.0 & -59.4 & 71.8 & -70.1 \\
72.7 & -61.7 & 71.7 & -76.5 & 173.7 & -77.1 & 70.6 & -60.3 & 72.1 \\
-71.5 & 73.1 & -62.1 & 101.0 & -77.1 & 173.4 & -69.3 & 70.9 & -60.7 \\
116.5 & -76.4 & 95.3 & -59.4 & 70.6 & -69.3 & 173.4 & -75.3 & 99.8 \\
-77.9 & 116.4 & -77.0 & 71.8 & -60.3 & 70.9 & -75.3 & 173.2 & -75.9 \\
91.4 & -78.4 & 116.3 & -70.1 & 72.1 & -60.7 & 99.8 & -75.9 & 172.8
\end{pmatrix}$$

Computer Vision

Eigenfilters

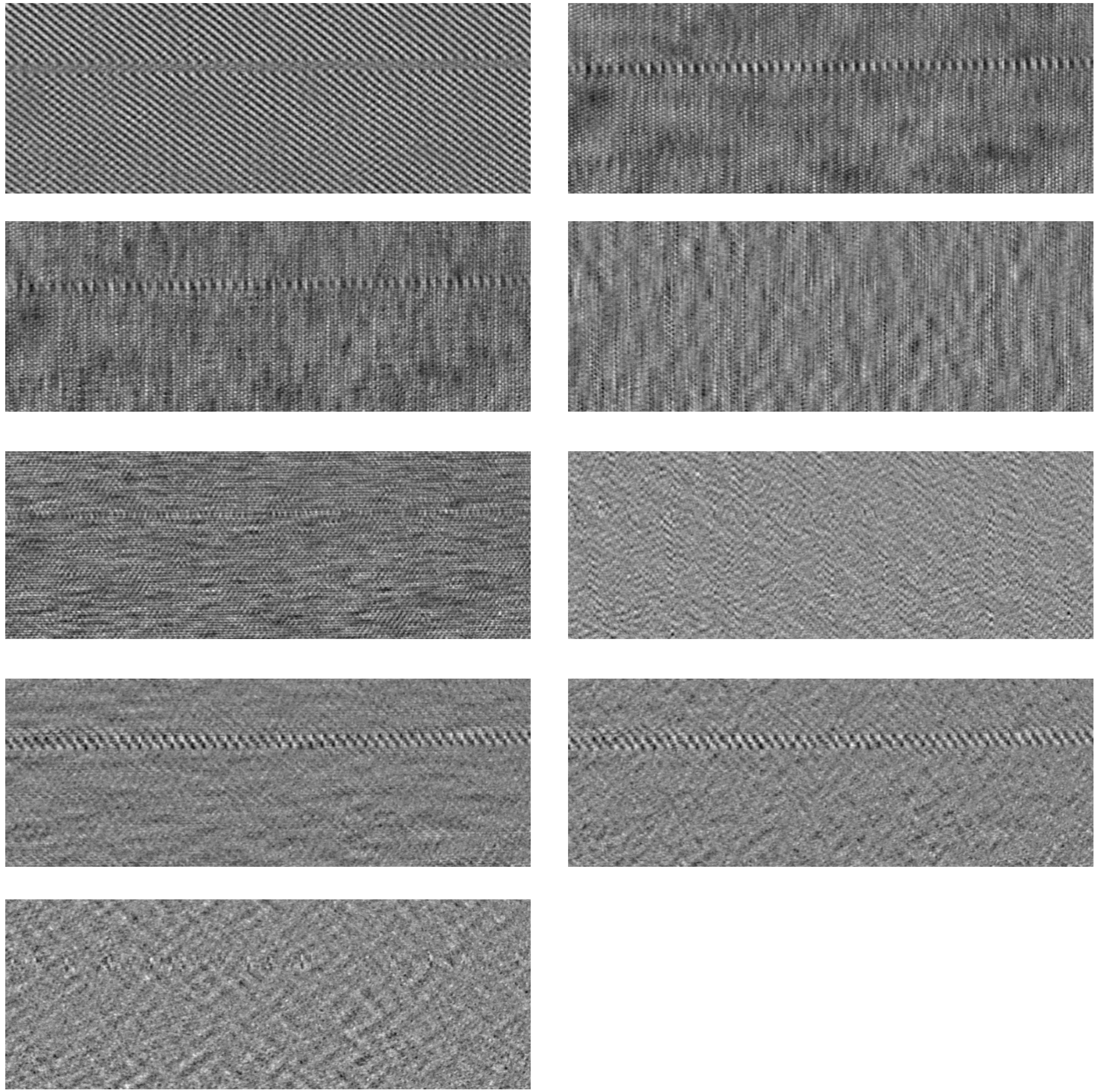
$$
\begin{pmatrix} 0.35 \\ -0.33 \\ 0.35 \\ -0.30 \\ 0.30 \\ -0.30 \\ 0.35 \\ -0.33 \\ 0.35 \end{pmatrix}
\quad
\begin{pmatrix} 0.31 \\ 0.12 \\ 0.31 \\ 0.51 \\ -0.20 \\ 0.49 \\ 0.34 \\ 0.12 \\ 0.31 \end{pmatrix}
\quad
\begin{pmatrix} 0.10 \\ 0.58 \\ 0.10 \\ -0.19 \\ 0.41 \\ -0.19 \\ 0.11 \\ 0.59 \\ 0.09 \end{pmatrix}
\quad
\begin{pmatrix} 0.46 \\ -0.00 \\ -0.43 \\ 0.30 \\ 0.03 \\ -0.28 \\ 0.41 \\ -0.02 \\ -0.49 \end{pmatrix}
\quad
\begin{pmatrix} -0.10 \\ -0.15 \\ -0.09 \\ 0.33 \\ 0.83 \\ 0.33 \\ -0.13 \\ -0.14 \\ -0.06 \end{pmatrix}
$$

$$
\begin{pmatrix} 0.27 \\ 0.11 \\ -0.11 \\ -0.62 \\ 0.00 \\ 0.63 \\ 0.11 \\ -0.12 \\ -0.27 \end{pmatrix}
\quad
\begin{pmatrix} -0.43 \\ -0.06 \\ -0.54 \\ -0.10 \\ 0.00 \\ 0.09 \\ 0.55 \\ 0.05 \\ 0.40 \end{pmatrix}
\quad
\begin{pmatrix} 0.08 \\ -0.69 \\ 0.02 \\ -0.09 \\ -0.00 \\ 0.10 \\ -0.02 \\ 0.69 \\ -0.09 \end{pmatrix}
\quad
\begin{pmatrix} -0.50 \\ -0.00 \\ 0.50 \\ 0.00 \\ 0.02 \\ 0.00 \\ 0.47 \\ -0.01 \\ -0.50 \end{pmatrix}
$$

Computer Vision

Eigenfilters

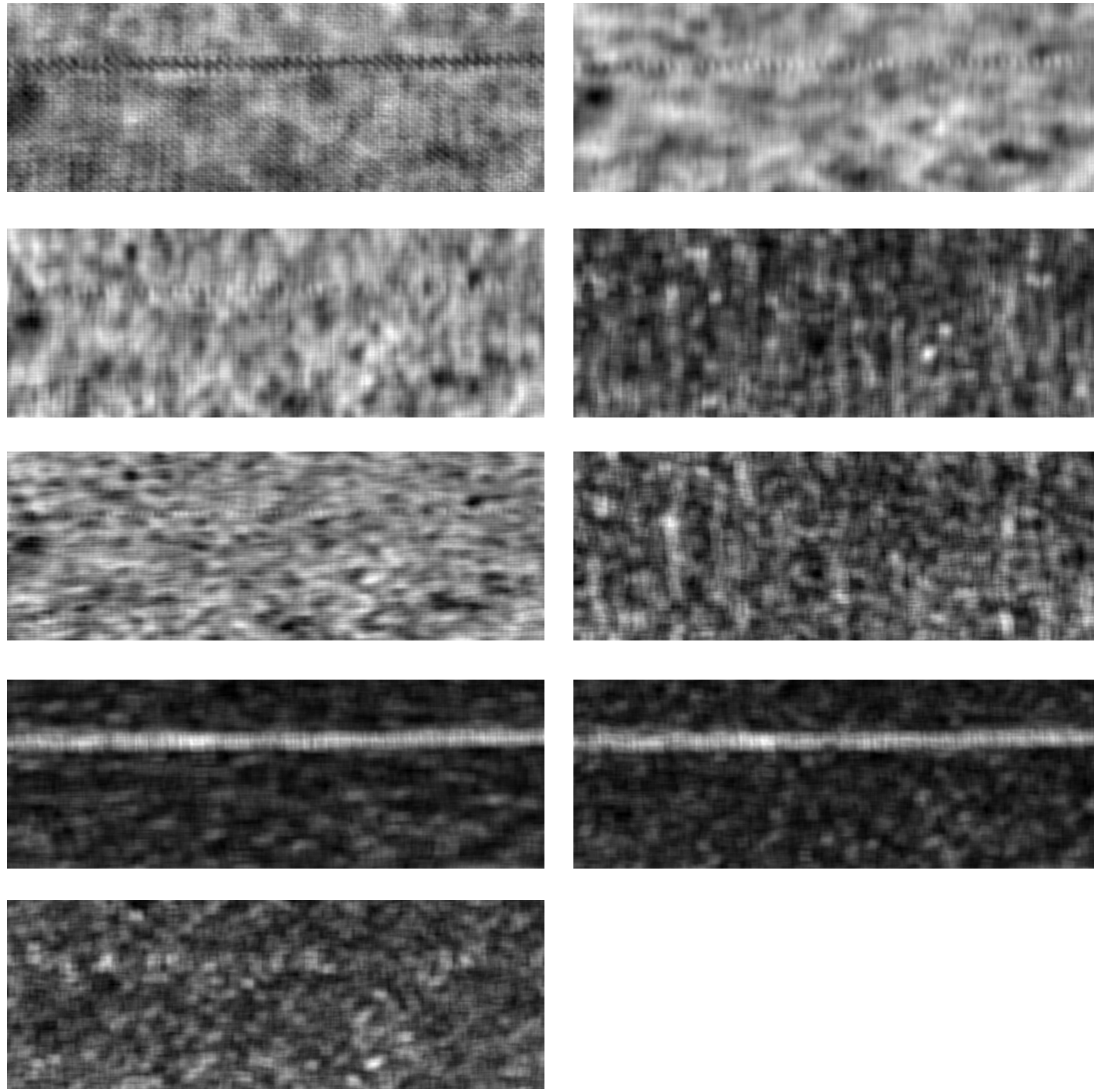
eigenfilter outputs



Computer Vision

Eigenfilters

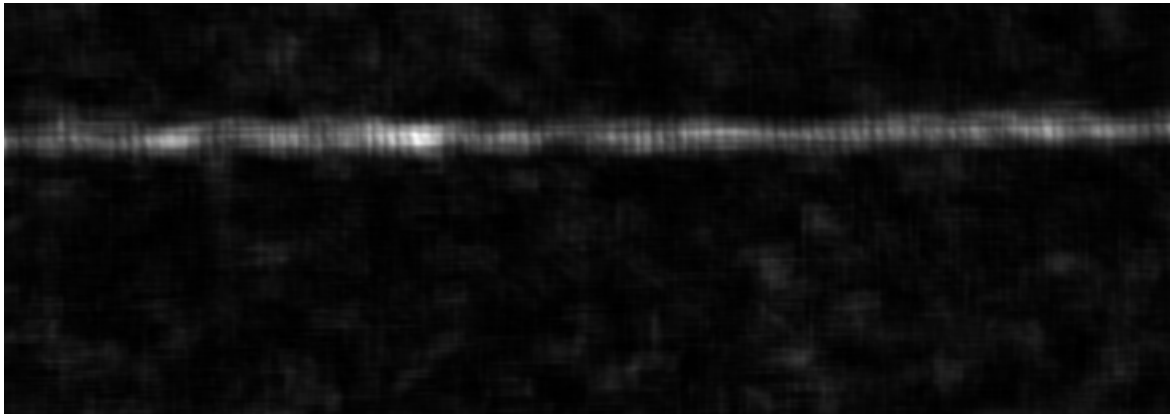
eigenfilter energies



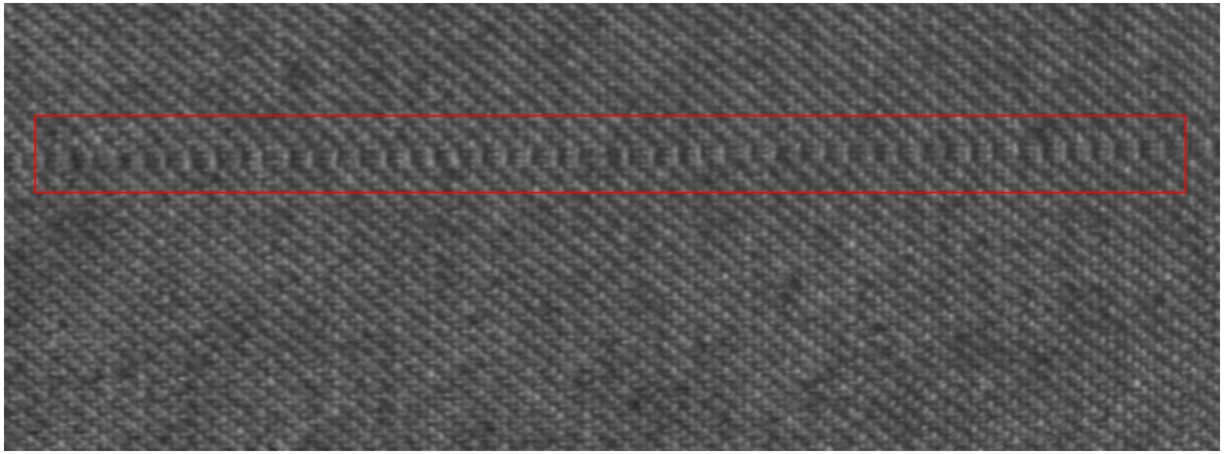
Computer Vision

Eigenfilters

Mahalanobis distance of the energies:

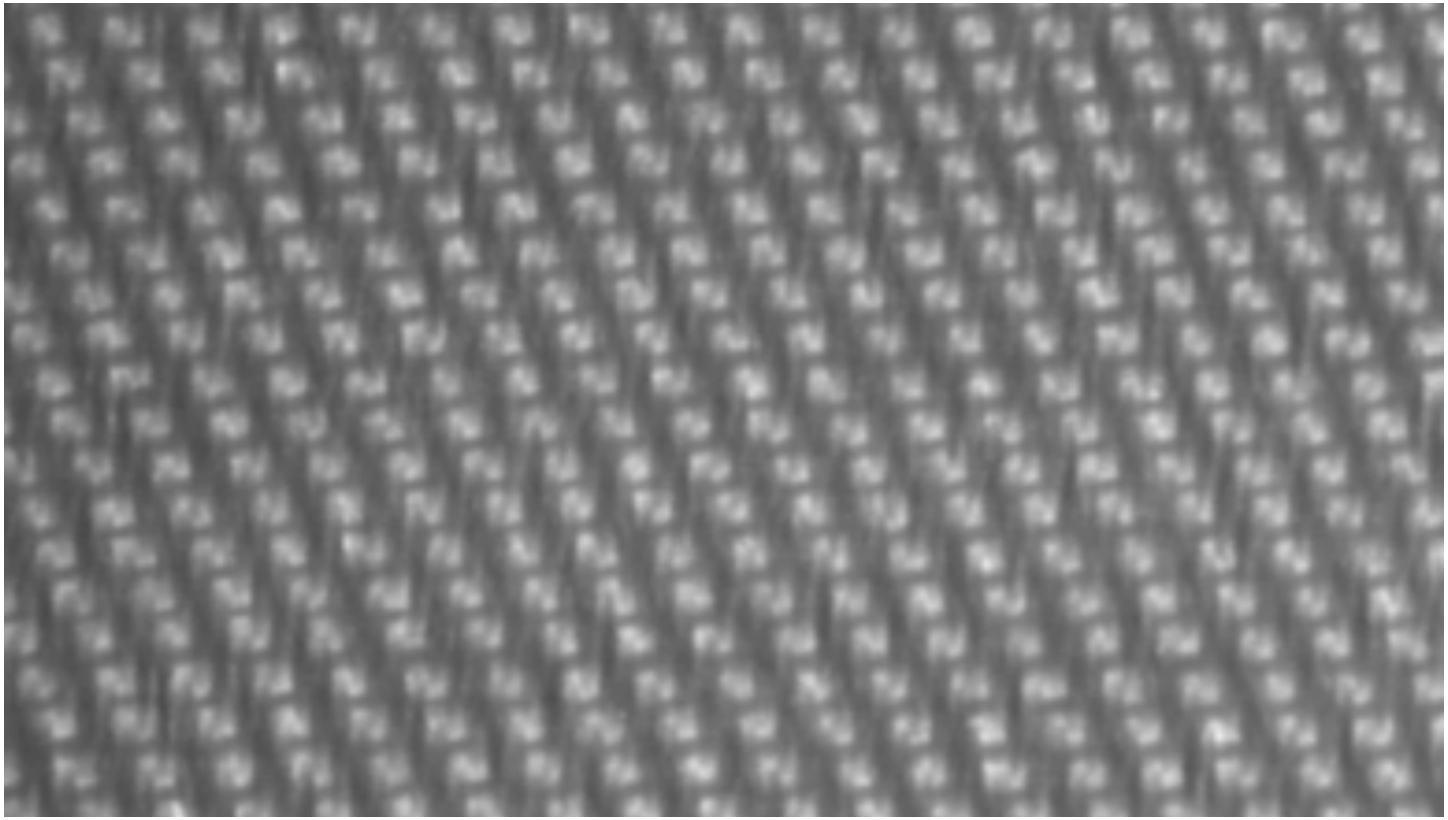


Flaw region found by thresholding:



Eigenfilters

Textile inspection: a second example



The texture is coarser, the filters are larger…

Eigenfilters

Textile inspection: eigenfilter blueprint

1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1

47 rows

21 columns

Computer Vision

Eigenfilters

The covariance matrix

Covariance Matrix :

743,89	-331,82	632,59	-316,03	618,47	-298,31	632,41	-302,50	548,84
-331,82	741,71	-345,42	641,76	-330,40	614,36	-334,25	629,22	-316,68
632,59	-345,42	738,28	-338,50	638,50	-343,03	618,37	-347,55	624,75
-316,03	641,76	-338,50	746,54	-328,93	634,29	-314,59	619,83	-297,11
618,47	-330,40	638,50	-328,93	743,82	-343,16	643,10	-329,00	615,32
-298,31	614,36	-343,03	634,29	-343,16	739,56	-337,44	639,40	-342,03
632,41	-334,25	618,37	-314,59	643,10	-337,44	748,53	-328,18	636,26
-302,50	629,22	-347,55	619,83	-329,00	639,40	-328,18	745,57	-342,19
548,84	-316,68	624,75	-297,11	615,32	-342,03	636,26	-342,19	741,04

Computer Vision

Eigenfilters

Eigenvectors / eigenvalues

Eigenvalues :

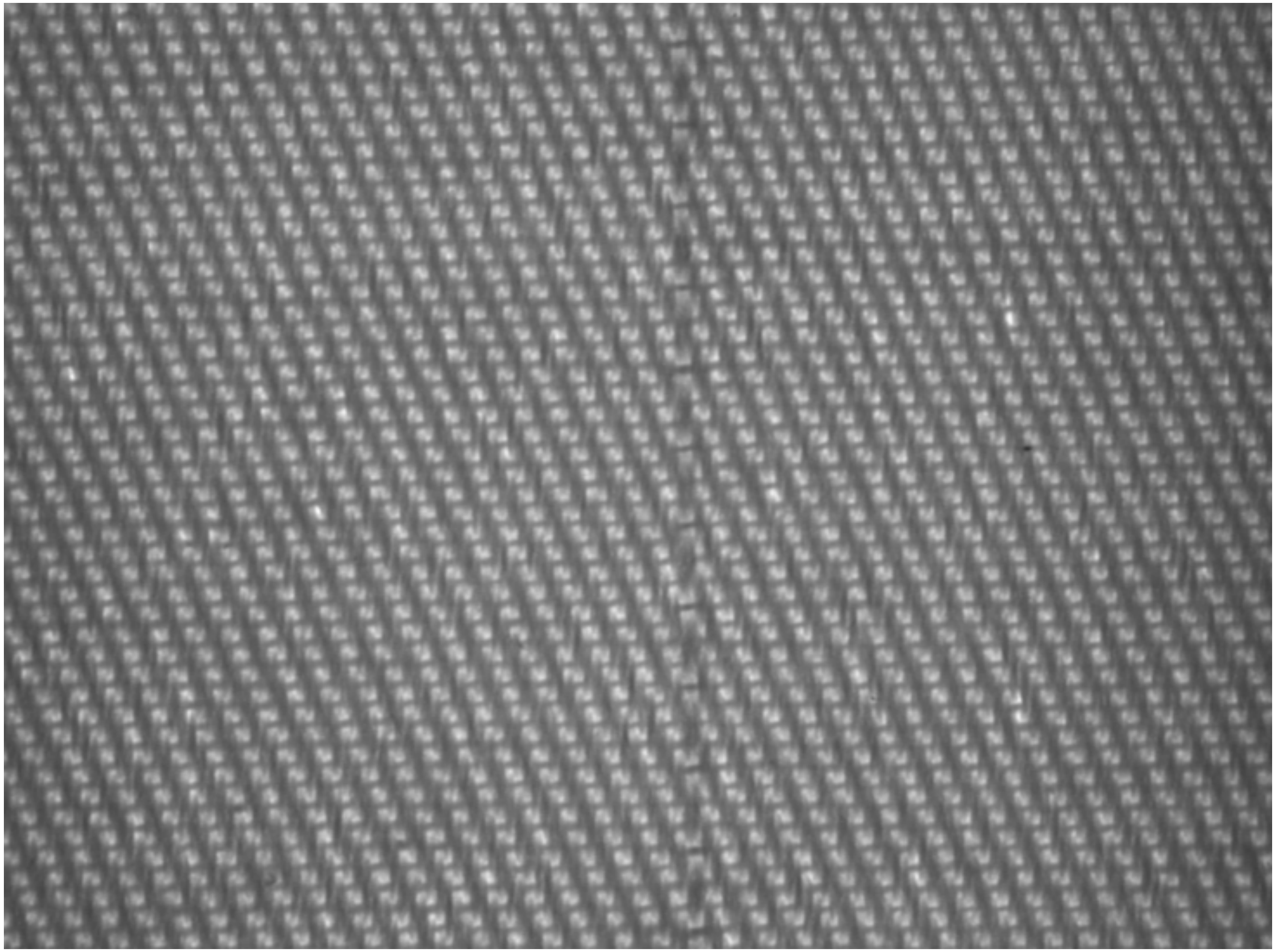
4428,70	1432,36	215,64	126,57	126,19	113,86	98,41	85,21	62,01

Eigenvectors :

0,337	0,292	-0,621	-0,121	0,119	-0,319	-0,229	0,040	-0,480
-0,317	0,382	0,225	0,273	-0,369	-0,465	-0,302	-0,429	-0,025
0,353	0,271	-0,055	-0,434	-0,536	0,069	-0,179	0,155	0,513
-0,314	0,400	0,240	-0,294	0,259	-0,399	0,266	0,548	0,034
0,350	0,285	-0,005	0,404	-0,292	-0,005	0,734	-0,004	-0,096
-0,318	0,381	-0,234	-0,410	0,161	0,353	0,295	-0,541	0,033
0,350	0,294	0,056	0,312	0,610	-0,056	-0,144	-0,163	0,518
-0,317	0,386	-0,219	0,421	-0,081	0,530	-0,261	0,412	-0,025
0,340	0,274	0,630	-0,173	0,079	0,330	-0,198	-0,046	-0,475

Eigenfilters

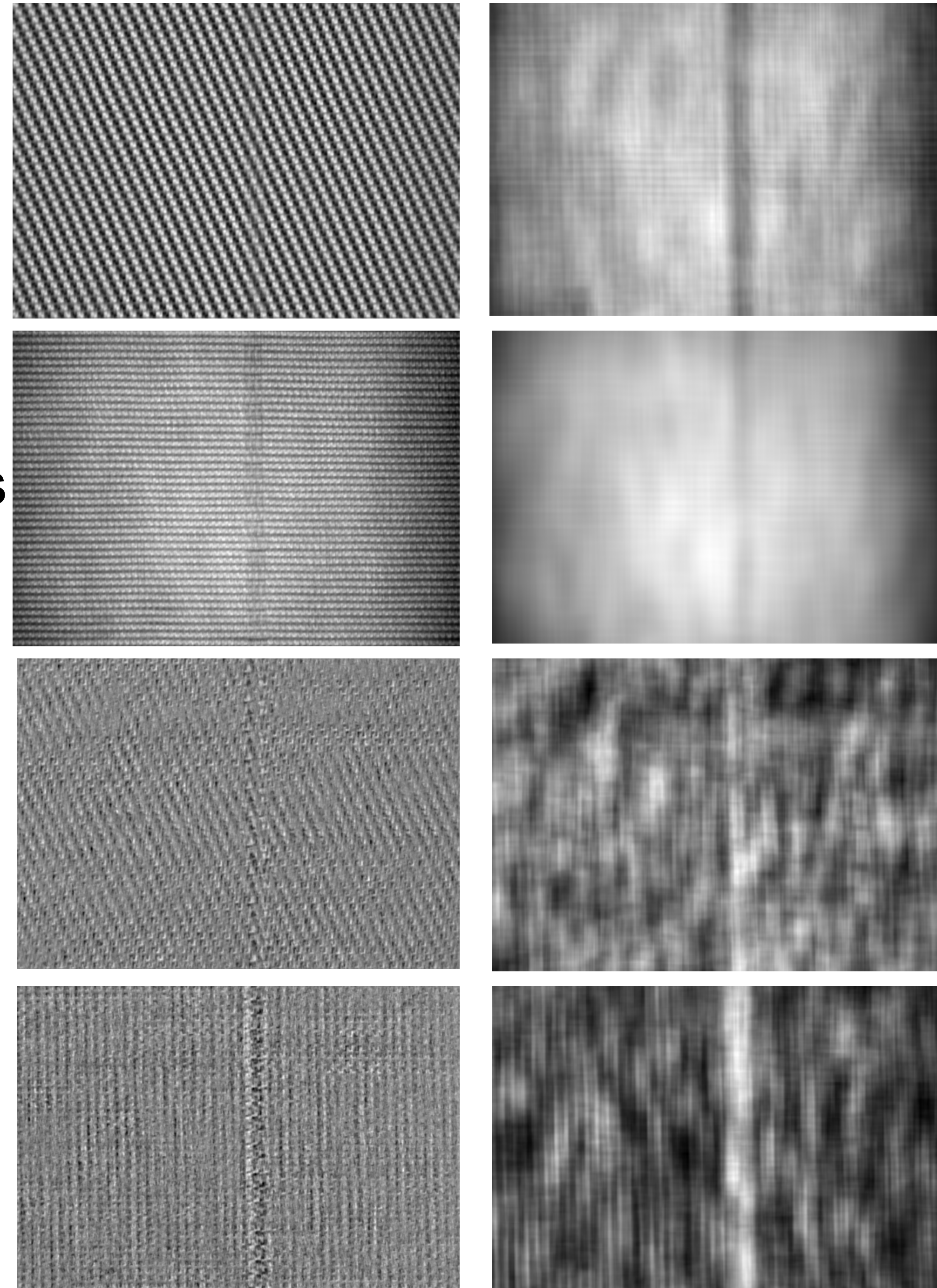
example of defect



Eigenfilters

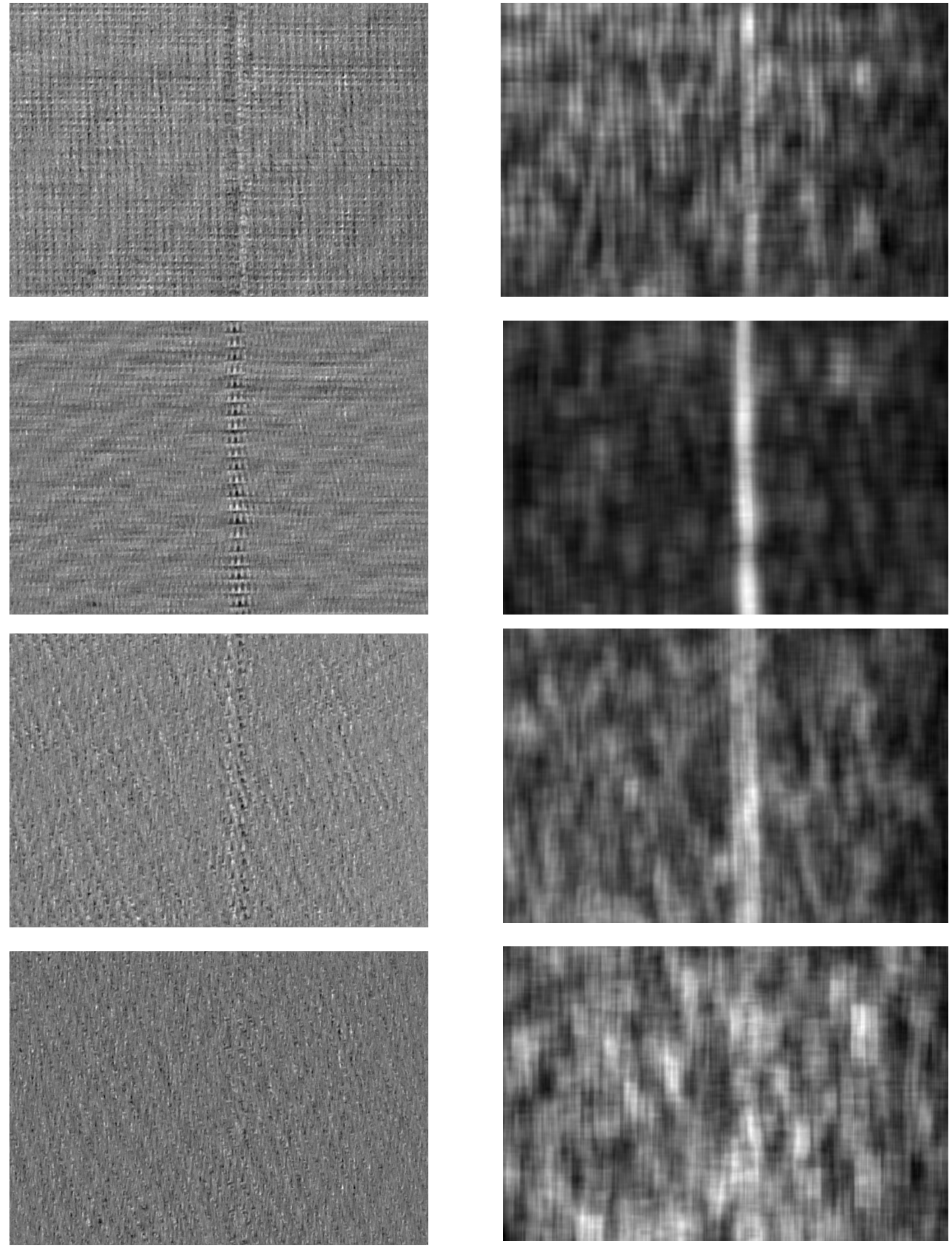
Outputs/energies for the

4 largest eigenvalues



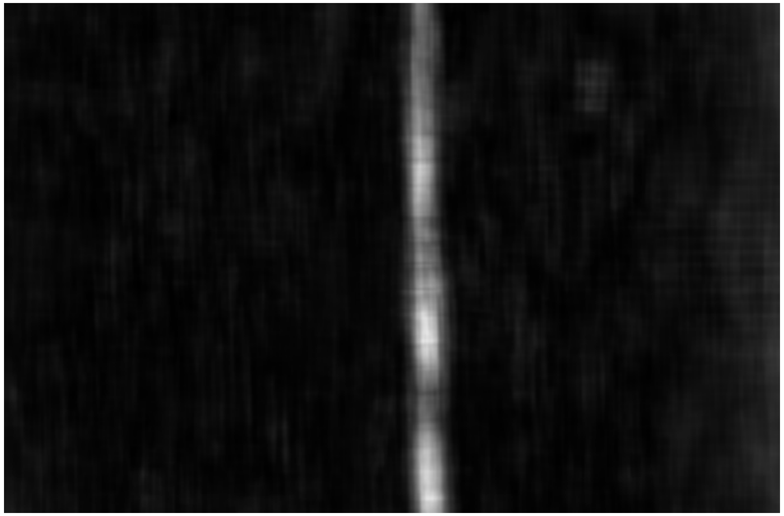
Eigenfilters

4 smallest eigenvalues



Eigenfilters

Mahalanobis distance



Threshold

On the menu:

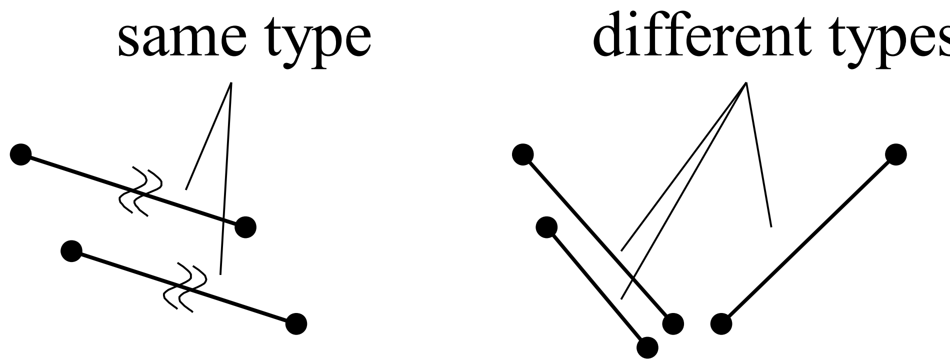
- Fourier features
- cooccurrence matrices
- filter banks (Laws, Gabor, eigenfilters)
- stochastic models

A 2-component texture model

1. Neighbourhood system

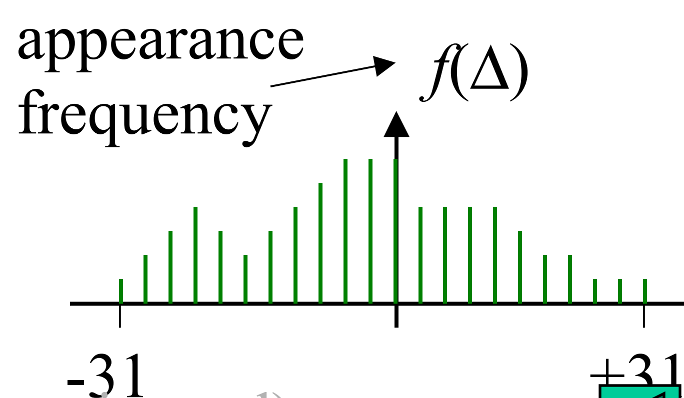
Pairs of pixels (Cliques)

Clique types:



2. Statistical parameter set

Intensity difference histograms:



(additionally histogram of singletons is used)

Histogram distance:

Δ - signal difference

$f(\Delta)$ - appearance frequency

$$\forall type\ \ t: \quad d_t = \sqrt{\sum_{\Delta}\left(f_t(\Delta) - f_t^0(\Delta)\right)^2}$$

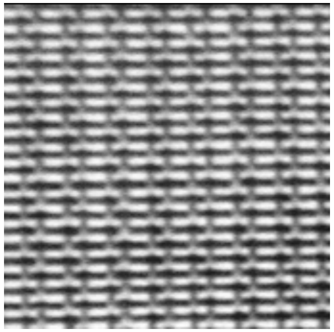
reference texture

Analysis (clique type selection algorithm)

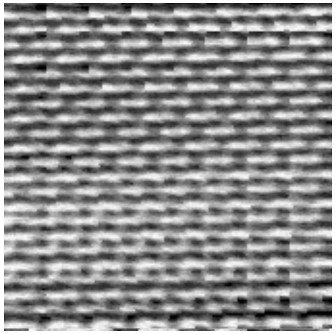
0. Collect reference histograms for all clique types (restriction on maximal clique length)
1. Collect histograms for the current synthesized texture (initially random noise)
2. Select the clique type with a maximal distance
3. If maximal distance < threshold => **STOP**
4. Add clique type to the texture model
5. Synthesize texture based on new model
6. Go to 1

Clique type selection

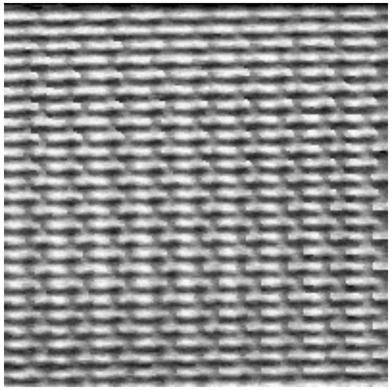
reference texture

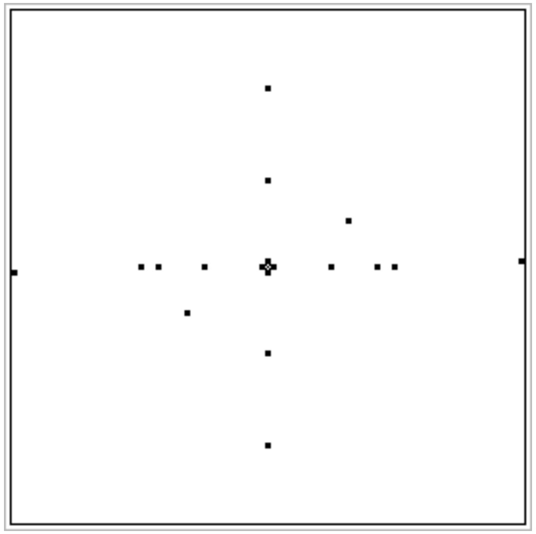


analysis steps



final synthesis

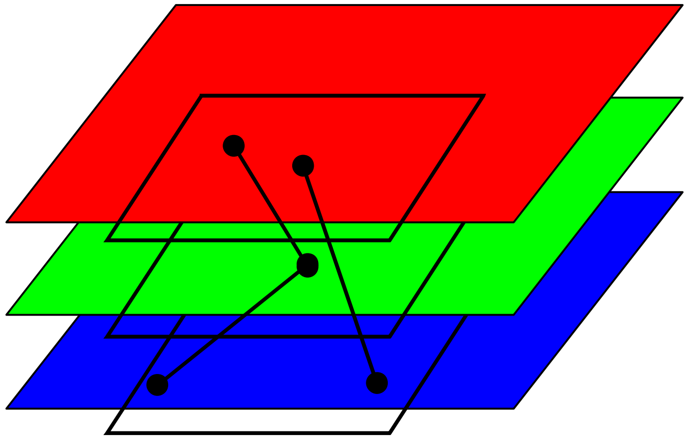




Successive neighborhood system update

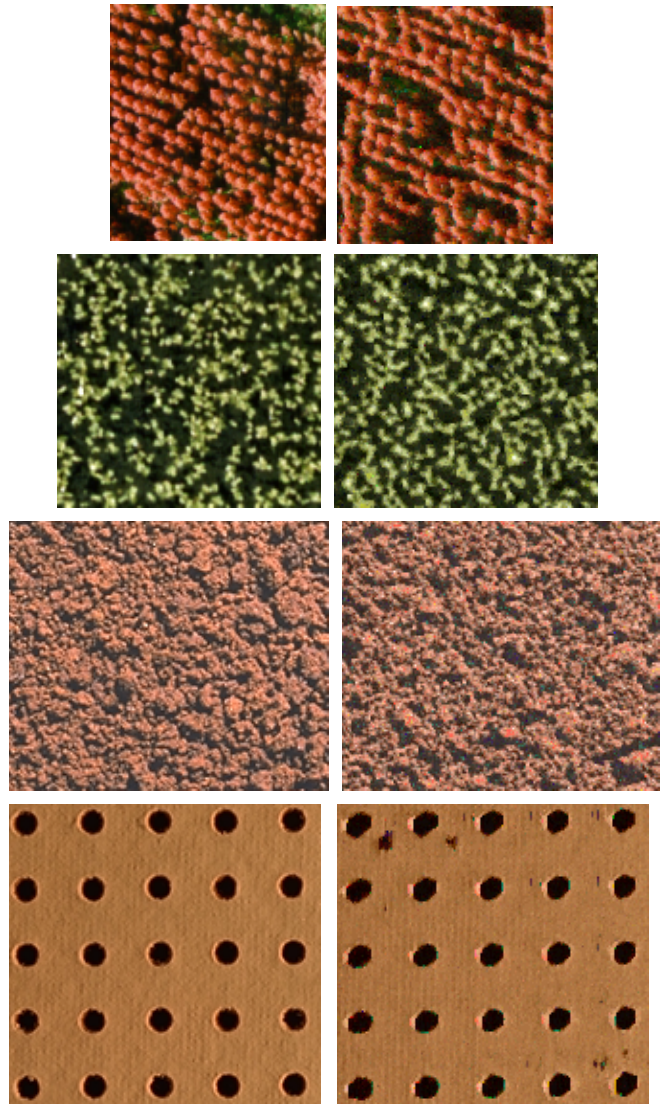
9 clique types

Interaction structure for color textures



Synthesized textures

Original Synthesis



Ex2: cedars on Sagalassos’ mountain

Computer Vision

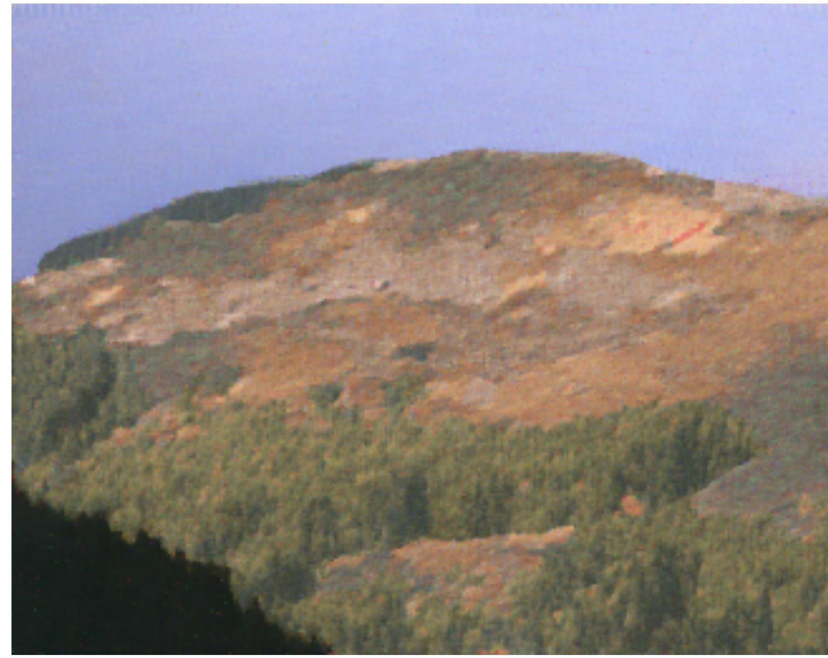
Ex.: cedars on Sagalassos’ mountain

ready

Ex.: cedars on Sagalassos' mountain

Note that texture synthesis was also used to remove the crane and van

Filling in of vegetation maps



ready

Sagalassos landscape, synthetic

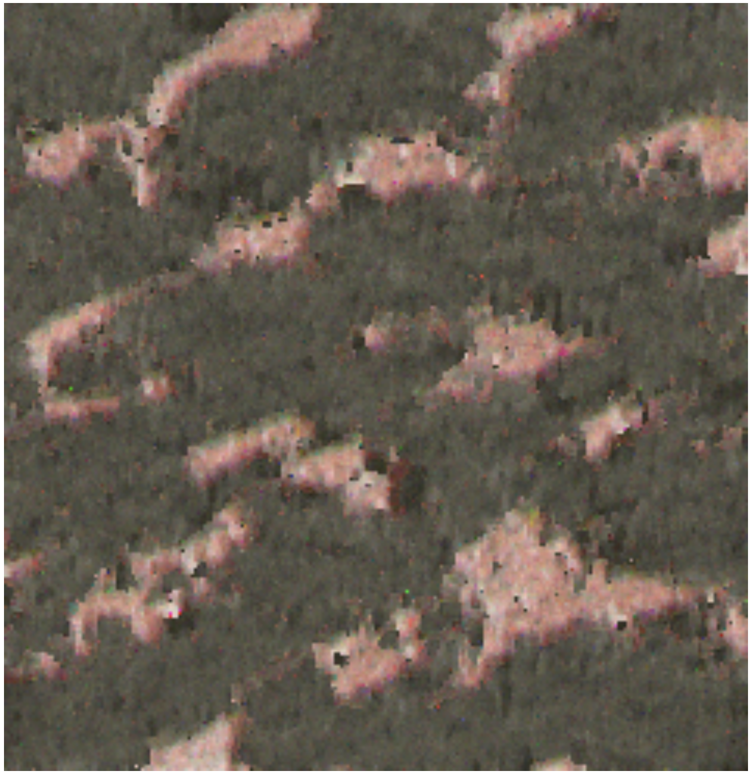
AUTOMATIC

Sagalassos landscape,with interactions

AUTOMATIC

Sagalassos landscape with interactions

Example image



Synthetic image

AUTOMATIC

ready

Computer Vision

Viewpoint / lighting dependent textures

Real orange

Synthetic orange

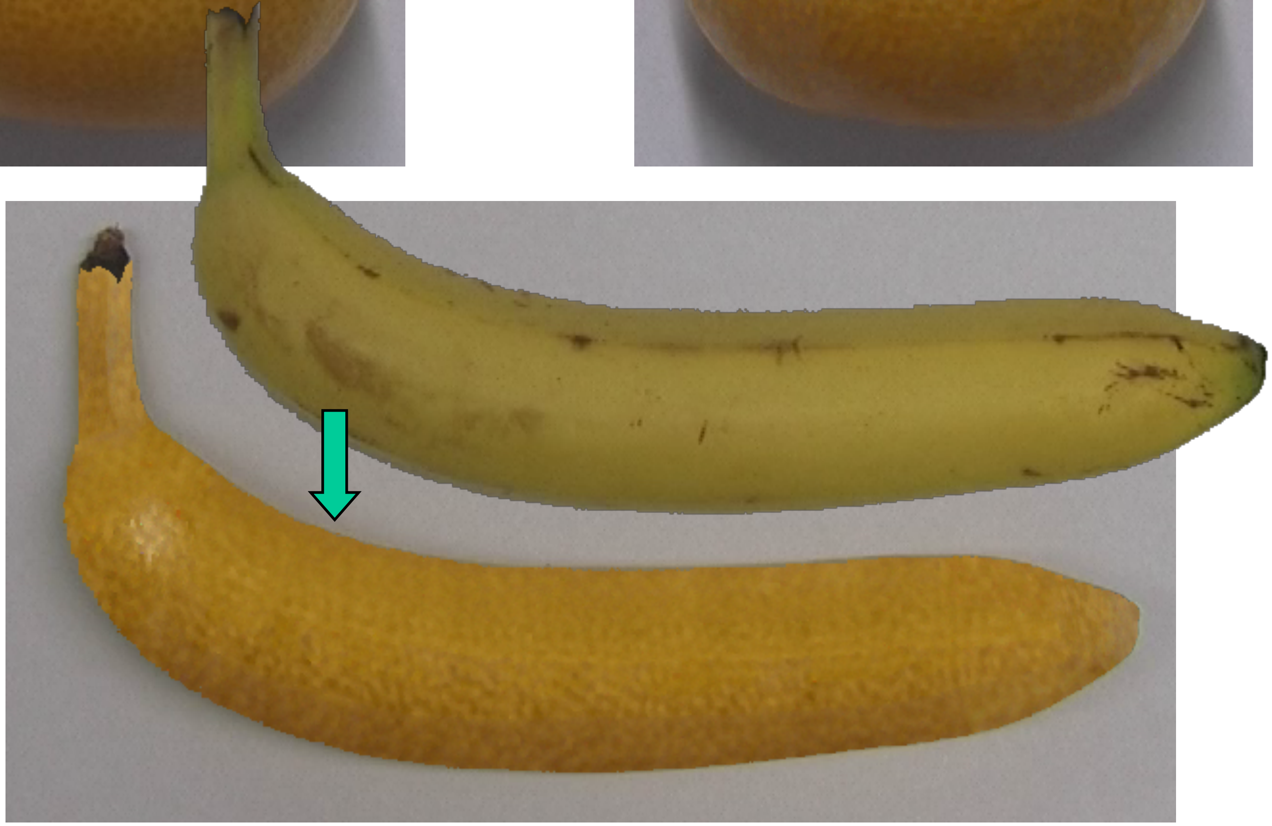
ready

Computer Vision

"Banarine"

Real tangerine

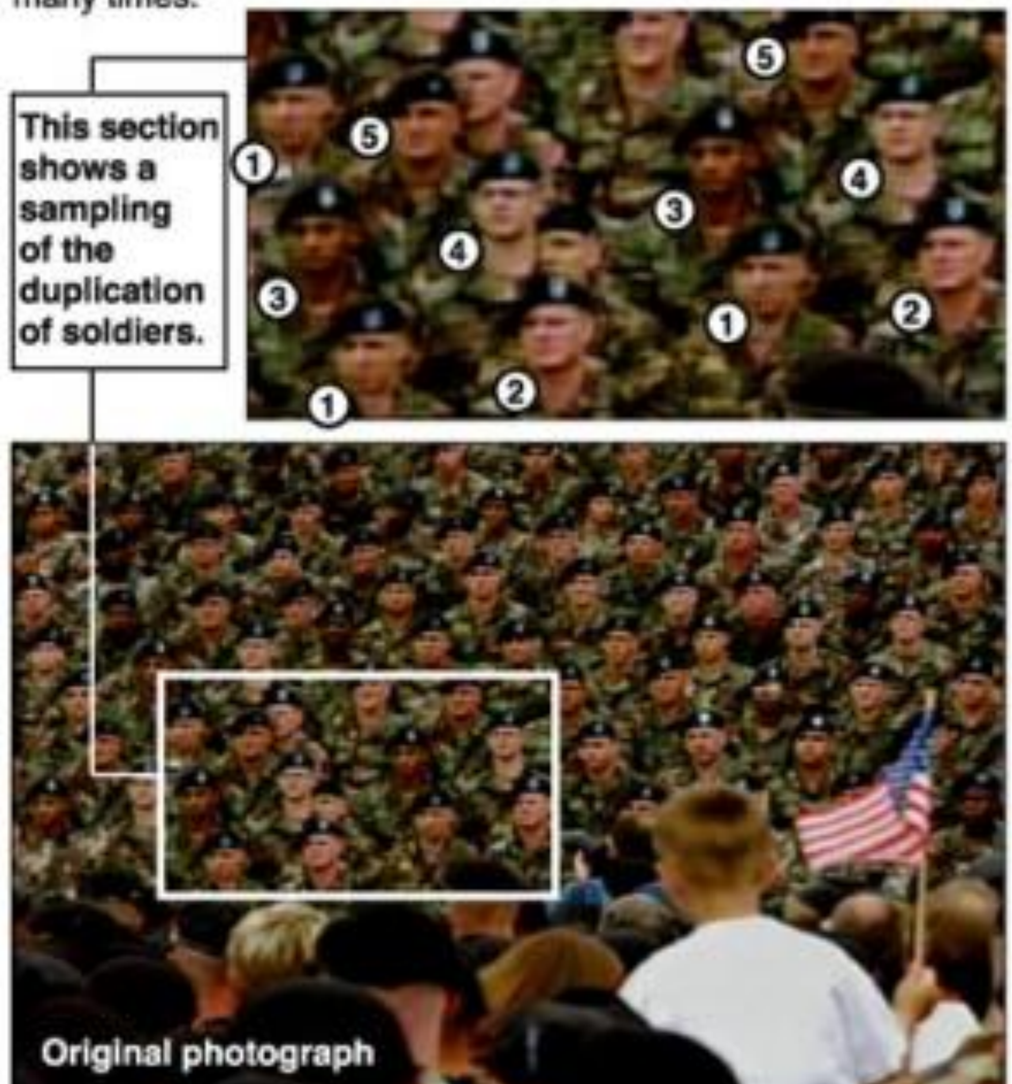
Synthetic tangerine



Computer Vision

Example of `smart copying'

Modified Bush 2004 election campaign ad



Bush campaign digitally altered TV ad

President Bush's campaign acknowledged Thursday that it had digitally altered a photo that appeared in a national cable television commercial. In the photo, a handful of soldiers were multiplied many times.