

VK Multimedia Information Systems



Mathias Lux, mlux@itec.uni-klu.ac.at

Dienstags, 16.00 Uhr s.t., E.1.42



This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 2.0 License. See <http://creativecommons.org/licenses/by-nc-sa/2.0/at/>

Results



<http://www.uni-klu.ac.at>

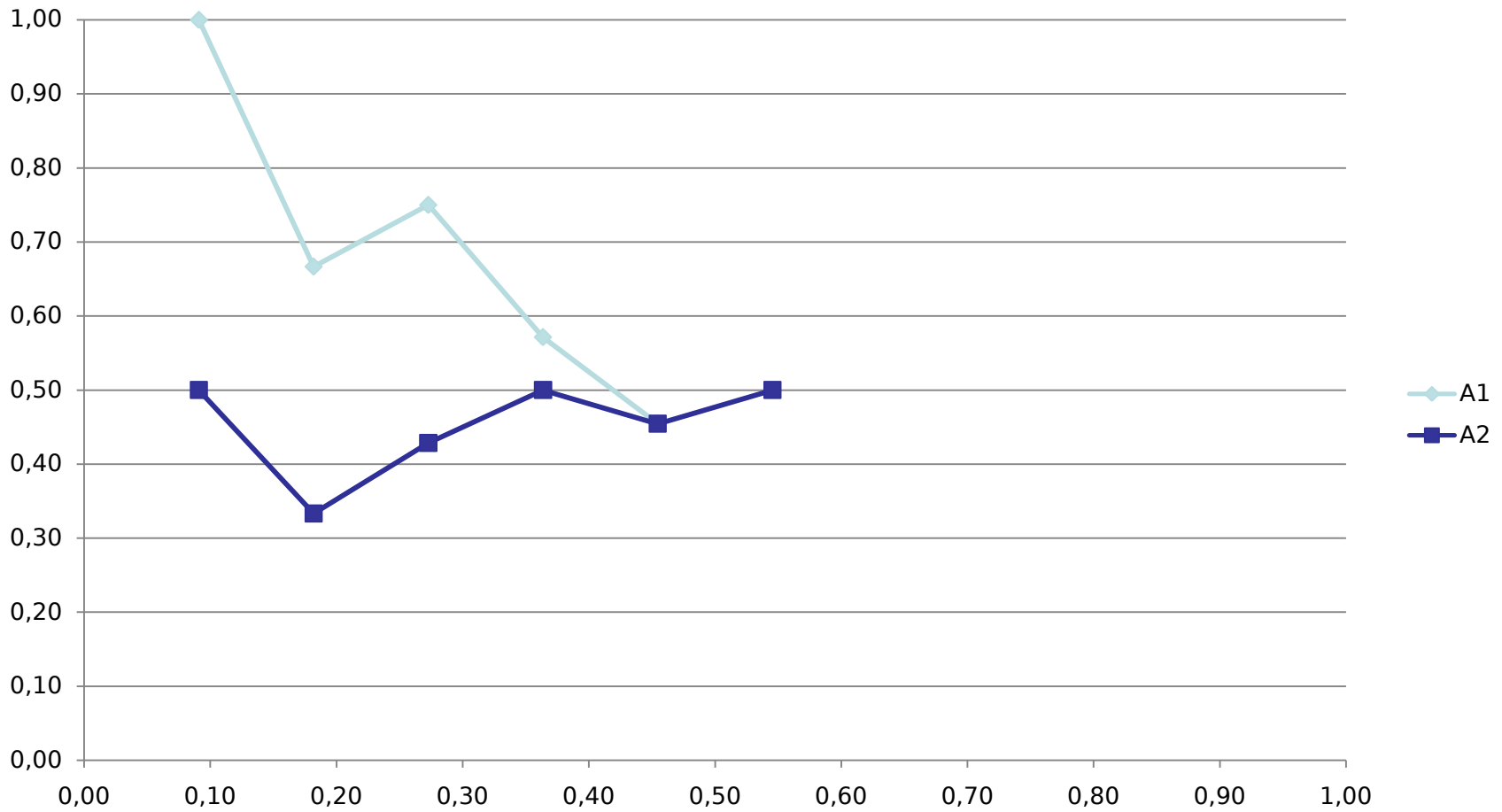
- A1:
 - 0,455 Precision
 - 0,455 Recall
- A2:
 - 0,500 Precision
 - 0,545 Recall

RC-Lev	PR-A1	PR-A2
0,09	1,00	0,50
0,18	0,67	0,33
0,27	0,75	0,43
0,36	0,57	0,50
0,45	0,45	0,45
0,55		0,50
0,64		
0,73		
0,82		
0,91		
1,00		

Results Ex-03



<http://www.uni-klu.ac.at>



Content Based Image Retrieval



<http://www.uni-klu.ac.at>

- Motivation & Semantic Gap
- Perception
- Color Based Features
- Texture Based Features
- Local Features



Motivation

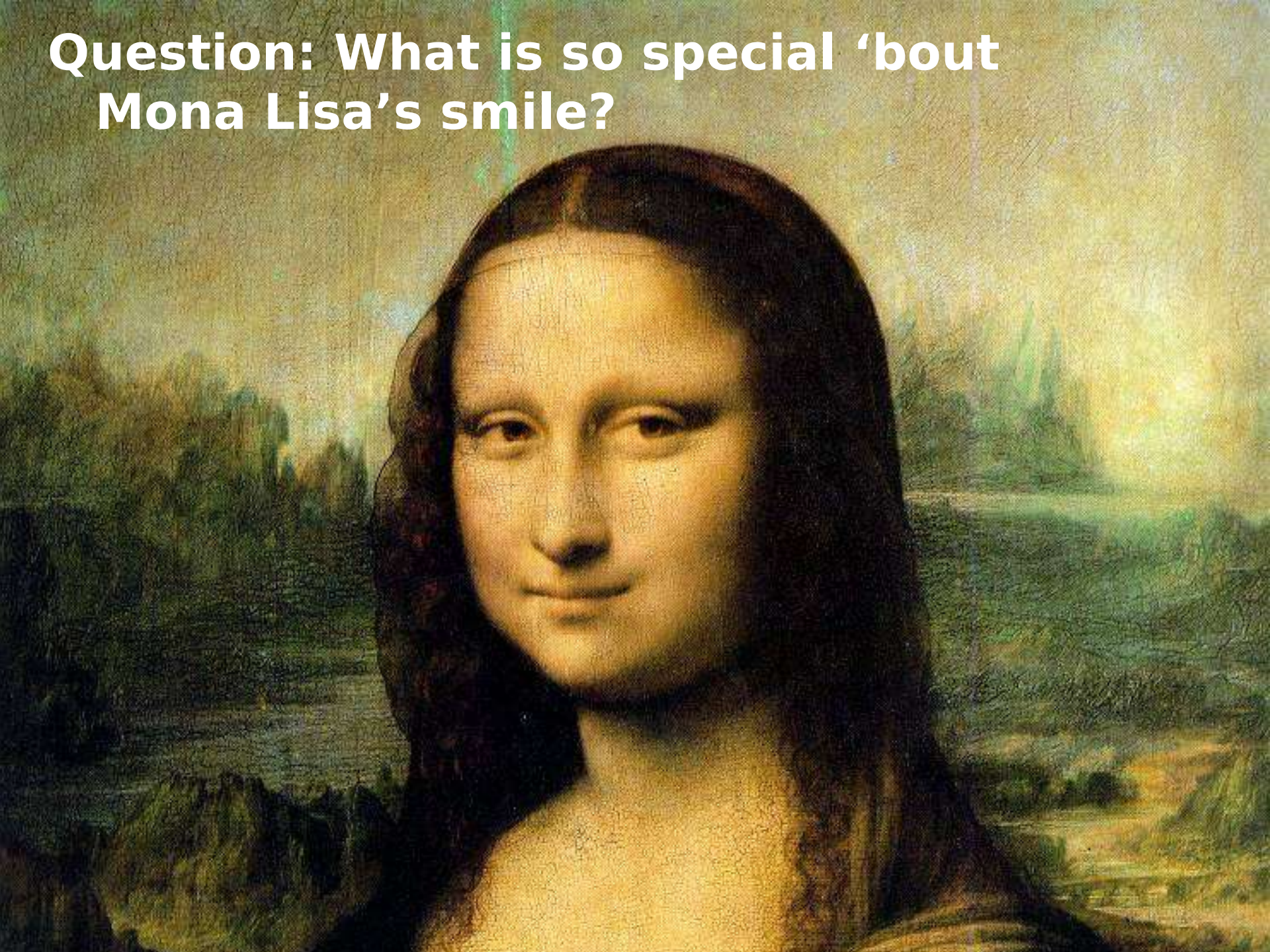


<http://www.uni-klu.ac.at>

Lots of good reasons ...

- Visual information overload
 - Devices (cameras, mobile phones, etc.)
 - Communication (email, mo-blogs, etc.)
- Metadata not available
 - Time consuming
 - No automation

**Question: What is so special 'bout
Mona Lisa's smile?**



Semantic Gap



<http://www.uni-klu.ac.at>

- Defined as
 - Inability of automatic understanding
 - Gap between high- and low-level features / metadata
- Actually hard task for humans also

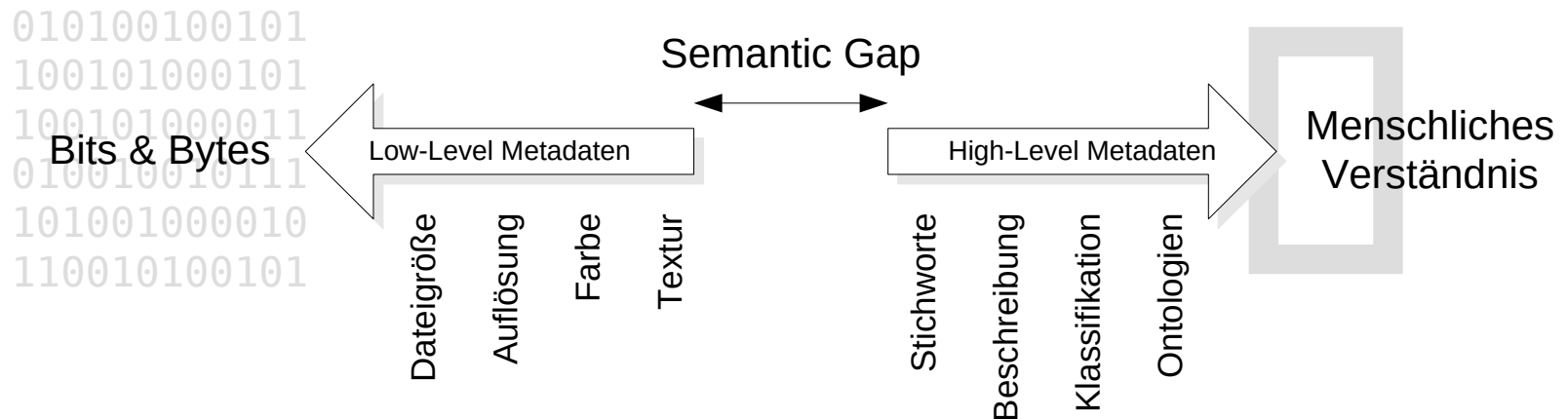


Semantic Gap (1)



<http://www.uni-klu.ac.at>

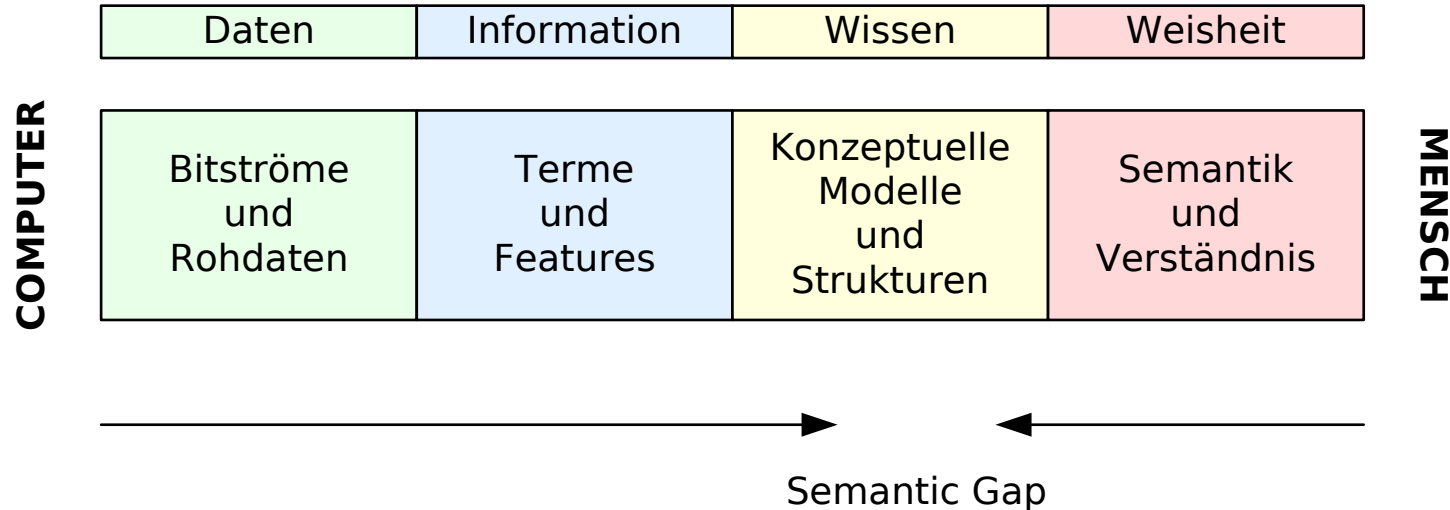
- General Definition: Santini & Jain (1998)



Semantic Gap (2)



<http://www.uni-klu.ac.at>



- Where actually is the Semantic Gap?
 - Classification based on Concepts
 - Segmentation & Object Recognition
 - ...

Applications



<http://www.uni-klu.ac.at>

- Home User & Entertainment
 - Find picture of / from / at
 - Search & browse personal digital library
- Graphics & Design
 - Find picture representing something (Color in CD/CI, feeling, etc.)
- Medical Applications
 - Diagnosis, segmentation & classification
 - X-Ray images, patient monitoring

Applications



<http://www.uni-klu.ac.at>

- Accessibility
 - 'Explain' image to visually disabled people
- Industrial application
 - Select / Sort out products (chips, buns)
 - Monitor processes (e.g. sensors unavailable)
- Security
 - Match fingerprints
 - Search face database

Applications



<http://www.uni-klu.ac.at>

- Biology
 - Analysing cell samples
 - Recognizing animals, insects & plants
- Astronomy
 - Classifying stars & events
- Weather forecasting
 - Satellite images, clouds
- Cartography
 - Mapping (e.g. aerial photo - earth model)

Content Based Image Retrieval



<http://www.uni-klu.ac.at>

- Motivation & Semantic Gap
- Perception
- Color Based Features
- Texture Based Features
- Local Features

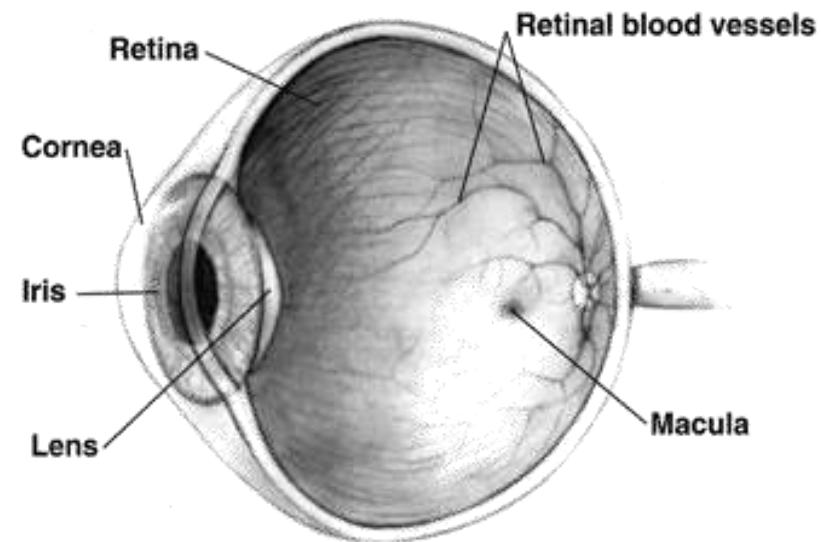


Perception



<http://www.uni-klu.ac.at>

- The **eye** as instrument of perception
- Sensory capabilities
 - Cones (bright light): 6-7 Mio.
 - Rods (dim-light): 75-150 Mio.
 - Brain 'corrects' vision
 - e.g. blind spot



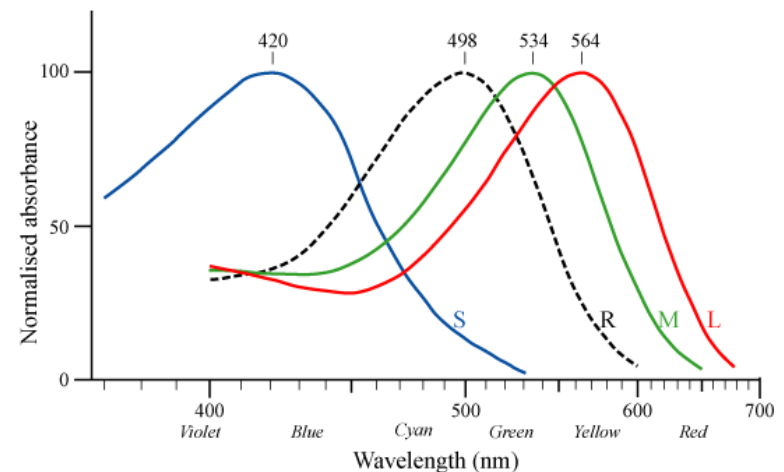
Color & Color Spaces



<http://www.uni-klu.ac.at>

S-, M- and L-cones: Blue, green and red

- RGB based on these three colors
- CIE models perception better
 - Responsiveness of cone types
 - Number of cones / types
 - etc.

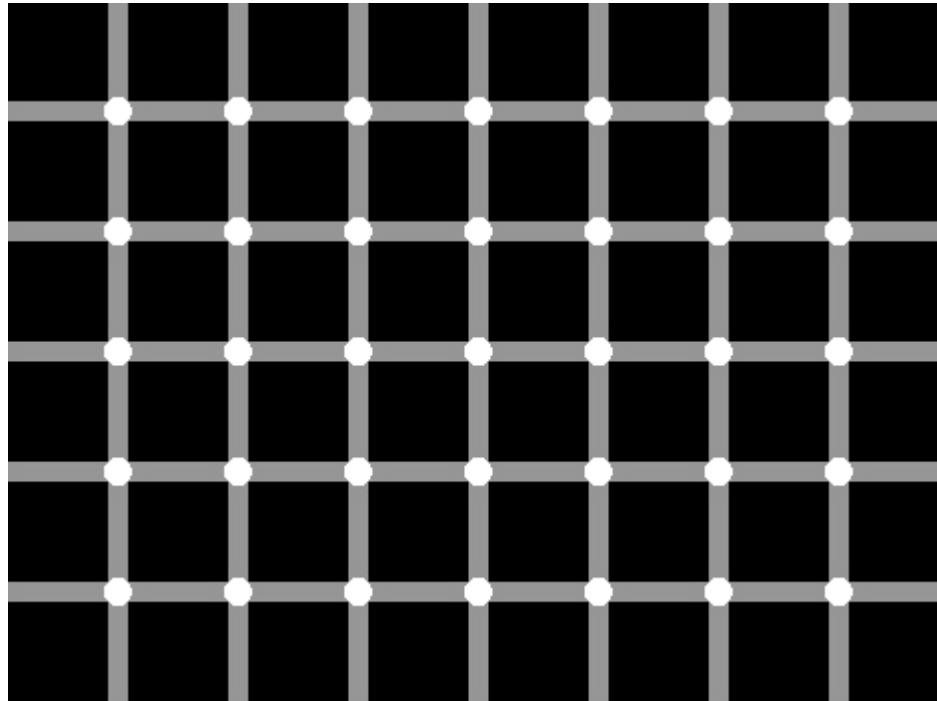


The human eye ...



<http://www.uni-klu.ac.at>

- Count the black dots on the image:



The human eye ...



<http://www.uni-klu.ac.at>

- Rabbit or duck?



The human eye ...



<http://www.uni-klu.ac.at>

- Anamorphic illusions



See e.g. <http://users.skynet.be/J.Beever/pave.htm>

Anamorphic Illusions (Julian Beever)



<http://www.uni-klu.ac.at>



What are (digital) images?



<http://www.uni-klu.ac.at>

- An Image is
 - Created by a set of photons
 - With different frequency
 - Moving from different sources
 - Along different vectors
 - A representation of sensor unit activation
 - Activated by the set of photons
- Storing an image
 - Based on the set of photons ???

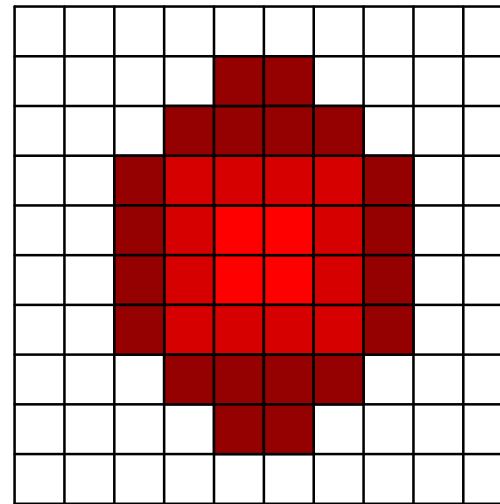
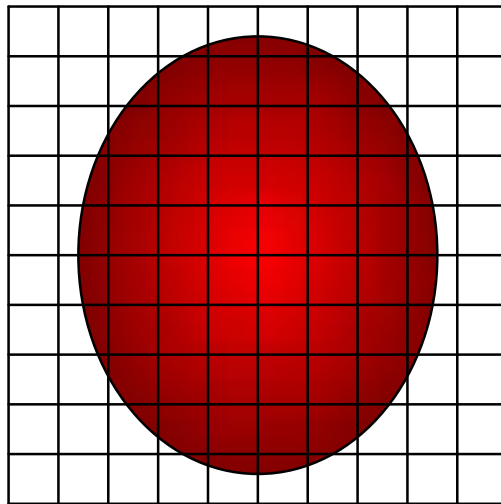


Sampling & Quantization



<http://www.uni-klu.ac.at>

- Capturing continuous images on sensors
 - Sampling: Continuous to matrix
 - Quantization: Continuous color to value



Sampling & Quantization



<http://www.uni-klu.ac.at>

- Size of a captured image:
 - # of samples (width*height) * # of colors



Content Based Image Retrieval



<http://www.uni-klu.ac.at>

- Motivation & Semantic Gap
- Perception
- Color Based Features
- Texture Based Features
- Local Features

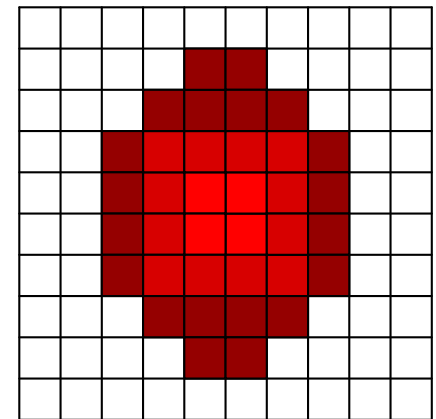


Color Histogram



<http://www.uni-klu.ac.at>

- Count how often which color is used
- Algorithm:
 - Allocate int array h with dim = # of colors
 - Visit next pixel -> it has color with index i
 - Increment h[i]
 - IF pixels left THEN goto line 2
- Example: 4 colors, 10*10 pixels
 - histogram: [4, 12, 20, 64]



Color Histogram



<http://www.uni-klu.ac.at>

- Strategies:
 - Quantize if too many colors
 - Normalize histogram (different image sizes)
 - Weight colors according to use case
 - Use (part of) color space according to domain
- Distance / Similarity
 - Assumption: All images have the same colors
 - L_1 or L_2 is quite common

Color Histogram



<http://www.uni-klu.ac.at>

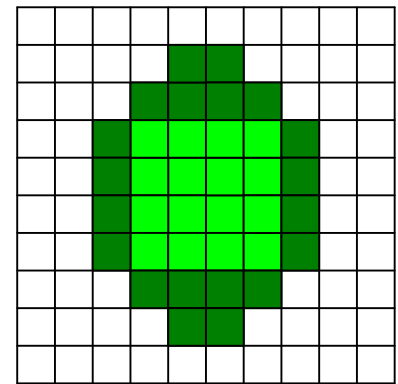
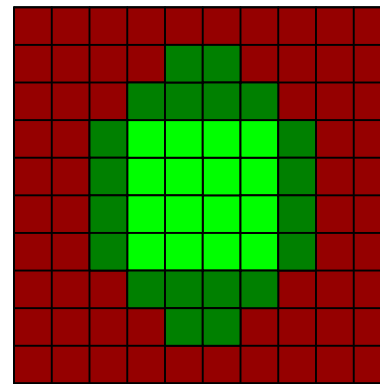
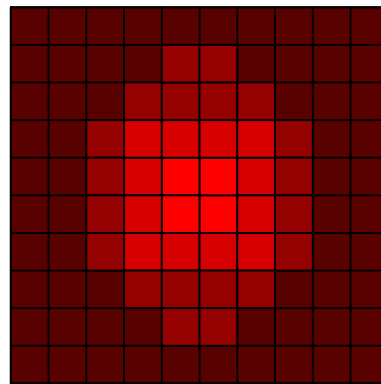
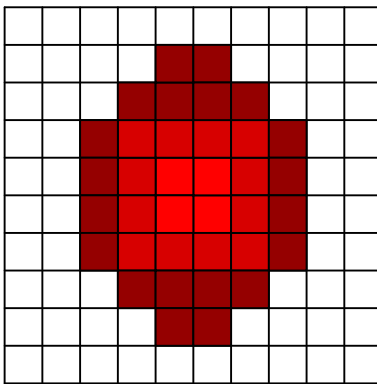
- Benefits
 - Easy to compute, not depending on pixel order
 - Matches human perception quite good
 - Quantization allows to scale size of histogram
 - Invariant rotation & reflection
- Disadvantages
 - Distribution of colors not taken into account
 - Colors might not represent semantics
 - Find quantization fitting to domain / perception
 - Image scaling might be a problem

Color Histogram



<http://www.uni-klu.ac.at>

- Example: 4 images, 7 colors
 - 1: [0, 4, 12, 20, 64, 0, 0]
 - 2: [66, 4, 12, 20, 0, 0, 0]
 - 3: [0, 0, 0, 64, 0, 20, 16]
 - 4: [0, 0, 0, 0, 64, 20, 16]



Color Histogram



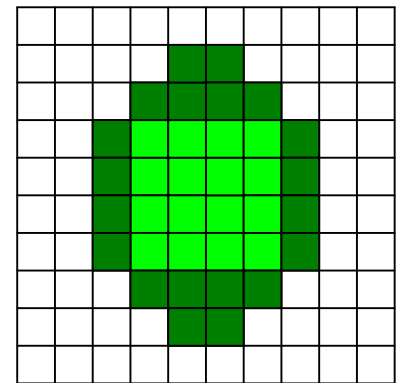
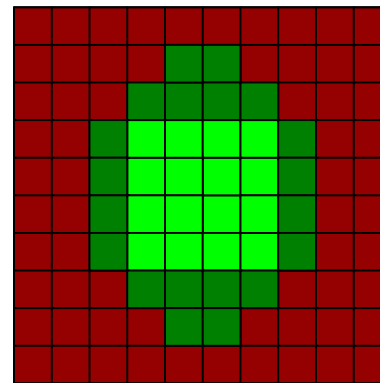
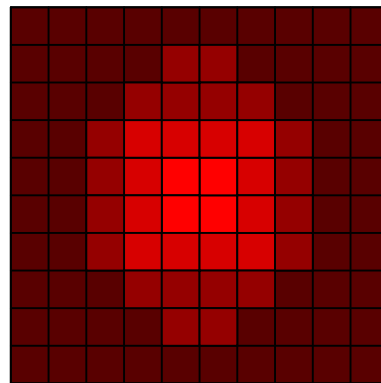
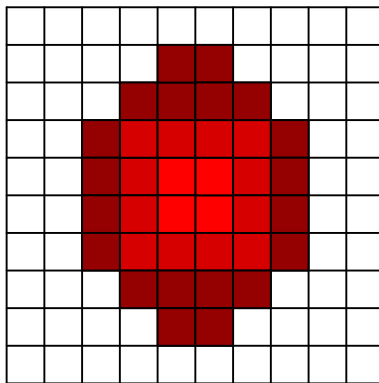
<http://www.uni-klu.ac.at>

Bilder:

- 1: [0, 4, 12, 20, 64, 0, 0]
- 2: [66, 4, 12, 20, 0, 0, 0]
- 3: [0, 0, 0, 64, 0, 20, 16]
- 4: [0, 0, 0, 0, 64, 20, 16]

Distanzfunktion d: L1

- $d(1,2) = 130$
- $d(1,3) = 160$
- $d(1,4) = 52$



Dominant Color



<http://www.uni-klu.ac.at>

- Reduce histogram to dominant colors
 - e.g. for 64 colors c0-c63:
 - image 1: c12 -> 23%, c33 -> 6%, c2 -> 2%
 - image 2: c11 -> 43%, c2 -> 12%, c54 -> 10%
- Distance function in 2 aspects:
 - Difference in amount (percentage)
 - Difference between colors (c11 vs. c12)
- Further aspects:
 - Diversity and distribution

Dominant Color



<http://www.uni-klu.ac.at>

- Benefits:
 - Small feature vectors
 - Easily understandable & intuitive
 - Similarity of color pairs (light vs. dark red, etc.)
 - Invariant to rotation & reflection
- Disadvantages
 - Similarity of color pairs no trivial problem
 - Dominant colors might not represent semantics

Color Distribution



<http://www.uni-klu.ac.at>

- Index dominant color in image segment
 - e.g. $8 \times 8 = 64$ image segments
 - feature vector has 64 dimensions
 - One for each segment
 - color index is the entry on segment dimension
 - e.g. 16 colors [2, 0, 3, 3, 8, 4, ...]



Color Distribution



<http://www.uni-klu.ac.at>

- Similarity
 - L_1 or L_2 are commonly used
- Benefits
 - Works fine for many scenarios
 - clouds in the sky, portrait photos, etc.
 - Mostly invariant to scaling
- Disadvantages
 - Colors might not represent semantics
 - Find quantization fitting to domain / perception
 - Rotation & reflection are a problem

Color Correlogram



<http://www.uni-klu.ac.at>

- Histogram on
 - how often **specific colors** occur
 - in the **neighbourhood** of each other
- Histogram size is $(\# \text{ of colors})^2$
 - For each color an array of neighboring colors

Color Correlogram



<http://www.uni-klu.ac.at>

- Extraction algorithm
 - Allocate array $h[\text{\#colors}][\text{\#colors}]$ all zero
 - Visit next pixel p
 - For each pixel q in neighborhood of p :
 - increment $h[\text{color}(p)][\text{color}(q)]$
 - IF pixels left THEN goto line 2
- Algorithm is rather slow
 - Depends on size of neighborhood
 - Typically determined by city block (L1) distance

Color Correlogram



<http://www.uni-klu.ac.at>

- Similarity
 - L_1 or L_2 are commonly used
- Benefits
 - Integrates color as well as distribution
 - Works fine for many scenarios
 - Mostly invariant to rotation & reflection
- Disadvantages
 - Find appropriate neighborhood size
 - Find quantization fitting to domain / perception
 - Rather slow indexing / extraction

Color Correlogram



<http://www.uni-klu.ac.at>

- Auto Color Correlogram
 - Just indexing how often $color(p)$ occurs in neighborhood of pixel p
 - Simplifies the histogram to size # of colors

Color Correlogram



<http://www.uni-klu.ac.at>

- Integrating different pixel features to correlate
 - **Gradient Magnitude** (intensity of change in the direction of maximum change)
 - **Rank** (intensity variation within a neighborhood of a pixel)
 - **Texturedness** (number of pixels exceeding a certain level in a neighborhood)

Content Based Image Retrieval



<http://www.uni-klu.ac.at>

- Motivation & Semantic Gap
- Perception
- Color Based Features
- Texture & Shape Features
- Local Features



Texture & Shape Features



<http://www.uni-klu.ac.at>

- Indexing non color features in image
 - Outlines, edges of regions
 - Overall characteristics like coarseness and regularity



Spatial Filtering



<http://www.uni-klu.ac.at>

- Methods for *enhancing* the image
- Normally a kernel or filter is used:
 - A matrix which is applied to the image
 - In a linear transformation

Spatial Filtering



<http://www.uni-klu.ac.at>

194	128	102	197	69
162	68	103	144	115
121	85	57	27	14
24	183	192	239	150
92	93	154	138	170

194	128	102	197	69
162	68	103	144	115
121	85	122	27	14
24	183	192	239	150
92	93	154	138	170

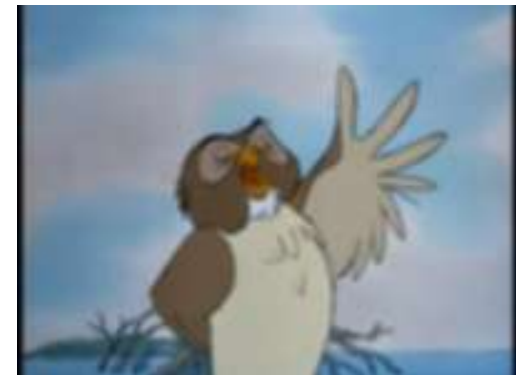
1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Spatial Filtering



<http://www.uni-klu.ac.at>

- This is a simple smoothing kernel
- Other operations
 - Sharpen
 - Unsharp Mask
 - Gradient



Edge Detection



<http://www.uni-klu.ac.at>

- Based on the gradient
 - Denotes the amount of change at specific point
 - Can be estimated with Gradient $\nabla f \approx |G_x| + |G_y|$
- Different kernels based on estimation
 - Test e.g. with Gimp



Edge & Texture Features



<http://www.uni-klu.ac.at>

- Edge based on gradient map
 - Aims to represent edges in number vector
 - e.g. Edge Histogram (MPEG-7)
 - Problems with rotation & reflection
- Texture features
 - Statistics on the image representing
 - Heavily depends on domain
 - Mostly invariant to rotation & reflection
 - Problems with scaling

Tamura Features

Tamura & Mori (1978)



<http://www.uni-klu.ac.at>

- Widely used in CBIR
 - E.g. IBM QBIC
- 6 texture features
 - Coarseness, contrast, directionality
 - Line-likeness, regularity, and roughness
- Good overview is provided in:
 - Thomas Deselaers, “Features for Image Retrieval”, Thesis, RWTH Aachen, Dec. 2003

Tamura Features

Tamura & Mori (1978)



<http://www.uni-klu.ac.at>

- Coarseness
 - Pixel diversity in neighborhoods
- Contrast
 - Using mean and variance of an image
- Directionality
 - Horizontal and vertical derivatives (like Sobel)

Shapes



<http://www.uni-klu.ac.at>

- Indexing of
 - **Boundaries** and **Regions**
 - Invariant to scaling, rotation and translation
- Features depend on domain, e.g.
 - Length of outline w.r.t. the image
 - Convexity & Concavity
 - Holes & Connectivity

Joint Histograms



<http://www.uni-klu.ac.at>

- Determine color & texture properties per pixel
 - E.g. color (64 bins) + gradient (4 bins)
- Use a color histogram per texture bin
 - E.g. 4 times a color histogram
- Classify pixel based on texture
- Add value to the corresponding color histogram.

Joint Histograms



<http://www.uni-klu.ac.at>

- Strategies
 - Find good texture properties
 - Minimize overall number of bins
- Benefits
 - Works better than pure color histograms
- Disadvantages
 - Typically slower extraction
 - Higher number of bins

Content Based Image Retrieval



<http://www.uni-klu.ac.at>

- Motivation & Semantic Gap
- Perception
- Color Based Features
- Texture Based Features
- Local Features



Global vs. Local



<http://www.uni-klu.ac.at>

- All features try to pertain ...
 - Similarity of similar pictures
 - Dissimilarity of dissimilar pictures
- Often a “global view” is not what is intuitively “similarity”

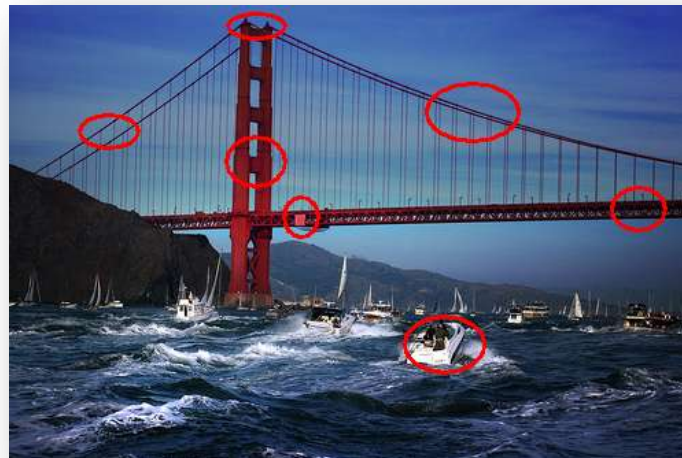


Local Features



<http://www.uni-klu.ac.at>

- Instead of global view
 - Points of “local interests” are selected
 - These points are “described” by vectors
 - Images are compared based on these points



Local Features



<http://www.uni-klu.ac.at>

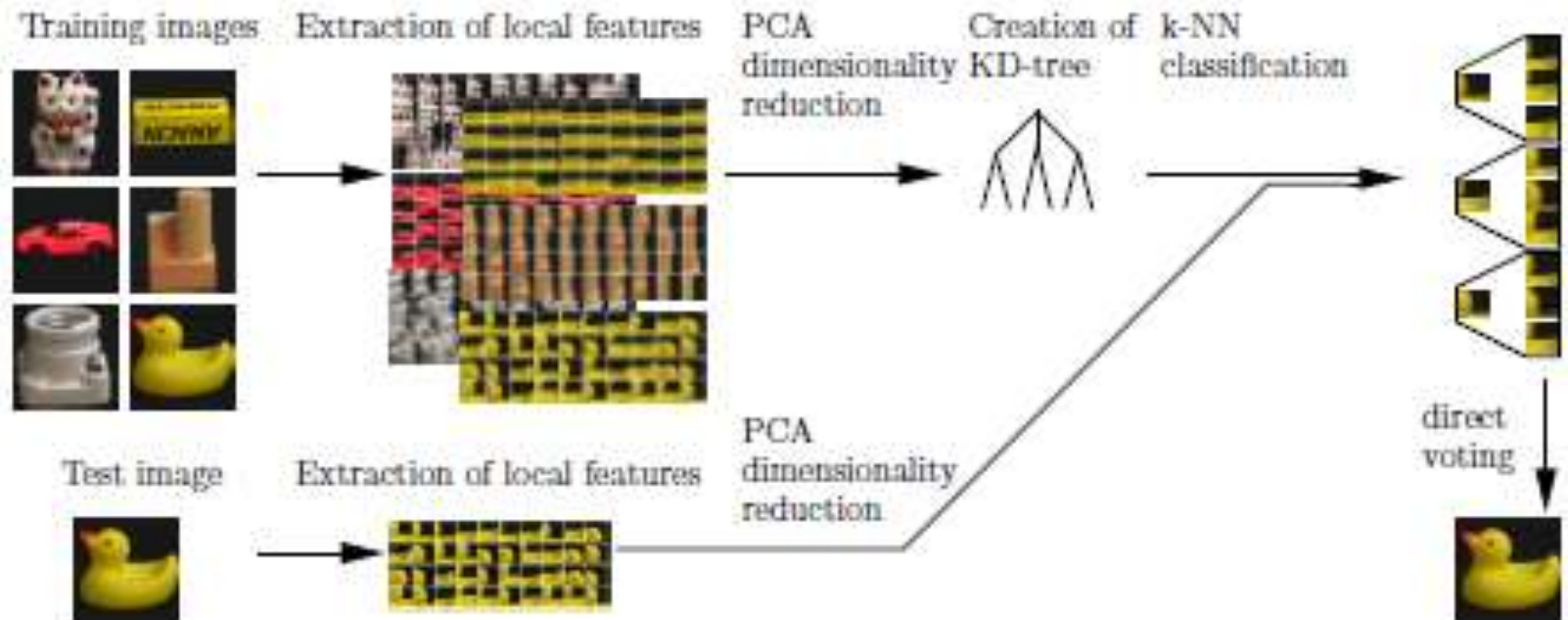


Figure 4.7: Classification using local features.

src: T. Deselaers, Features for Image Retrieval, Aachen 2003

Salient Points



<http://www.uni-klu.ac.at>

- Objective: Find areas of local interest
 - Reduces local features to a minimum
- Salient points, objects & regions
 - Sub-image of importance for assessment
 - Motivated by perception
 - Still recognizable after transformation
 - Scaling, rotation, different viewpoint, angle, etc.

Salient Points



<http://www.uni-klu.ac.at>

- High Gradients
 - Patches with high “energy” are chosen
- SIFT
 - Points invariant to scaling are chosen
 - Used in image stitching
- SURF
 - Faster alternative to SIFT

Local Features



<http://www.uni-klu.ac.at>

- Approach
 - Find salient patches
 - Define histogram after certain amount of analyzed images
- Benefits
 - Very often works better than global feature
- Disadvantages
 - Significant bigger load of indexing work & data

Implementation



<http://www.uni-klu.ac.at>

- Find the source on the course home page
- Extend the source and implement
 - 256 bin color histogram
 - L2 distance
 - Evaluation:
 - Error Classification Rate
 - Precision @ 10
 - MAP

Implementation



<http://www.uni-klu.ac.at>

- Start with any of the task
 - Average color and L1 are provided
 - Simple search is implemented

Thank you ...



<http://www.uni-klu.ac.at>

... for your attention