Images search

Stanislav Protasov

Agenda

- How our eyes work
- Historical approach
- Duplicate search and CBIR
- Image and video understanding

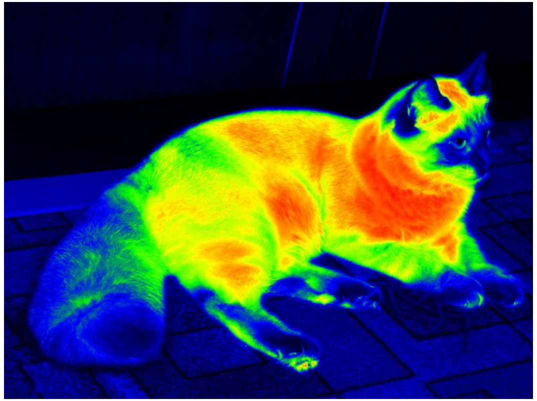
How our vision works

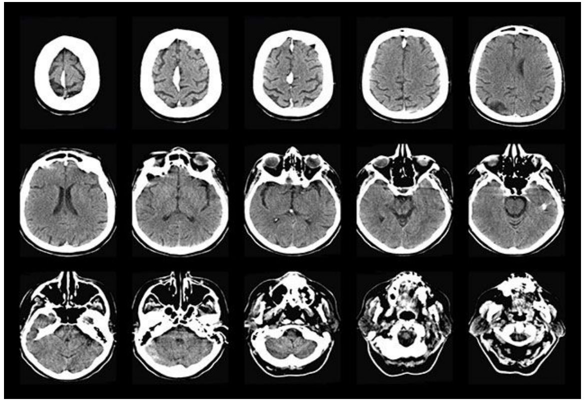
Hint*: very similar to digital camera*

Vision

Vision is a sensor system, that receives information using **electromagnetic waves** [of visible spectrum].

In general, X-ray, infrared and CT can be considered as “vision”.

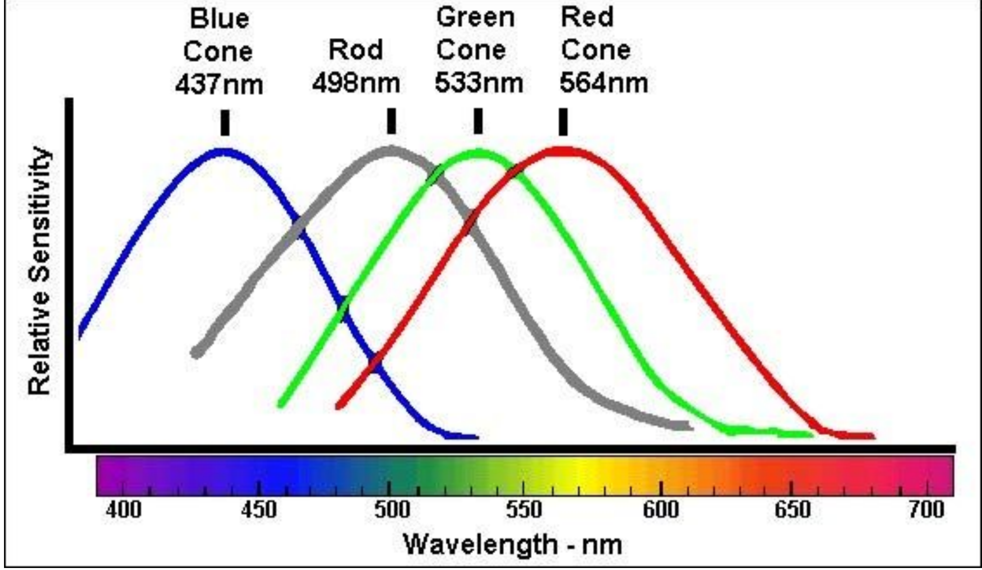


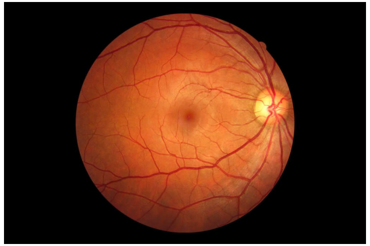


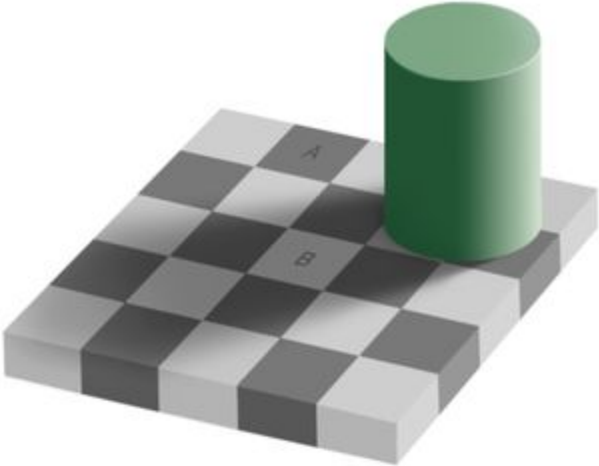
Human vision

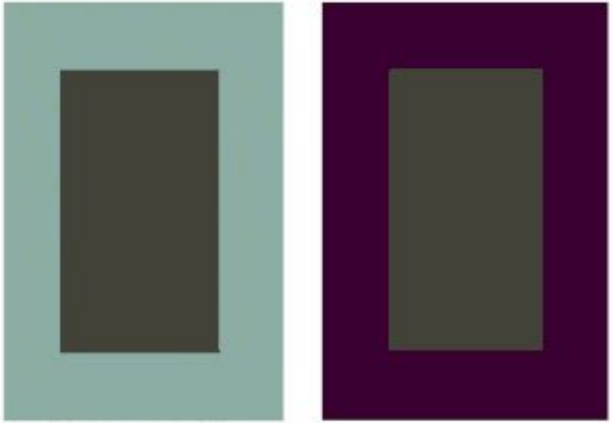
Major facts about vision:

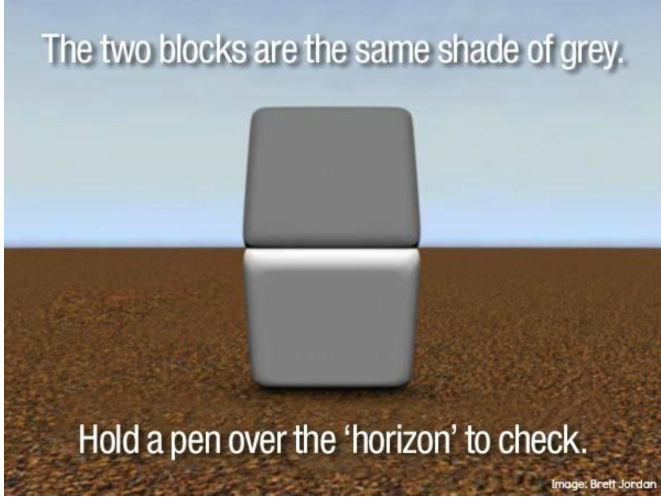
- **Binocular** — allows restore 3D
- **Retina** — discrete
- **Color** — quantized
  - 4 types of sensor cells:
    - S,M,L-*cone* cells
    - *Rod* cells
- Polarization and phase insensitive
- Supports **focus**
- **Opponent**-process theory and
- **Color constancy**
  - Brain process differences of colors







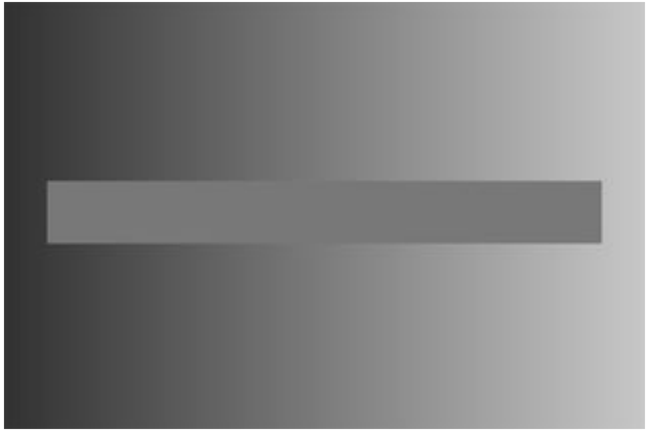




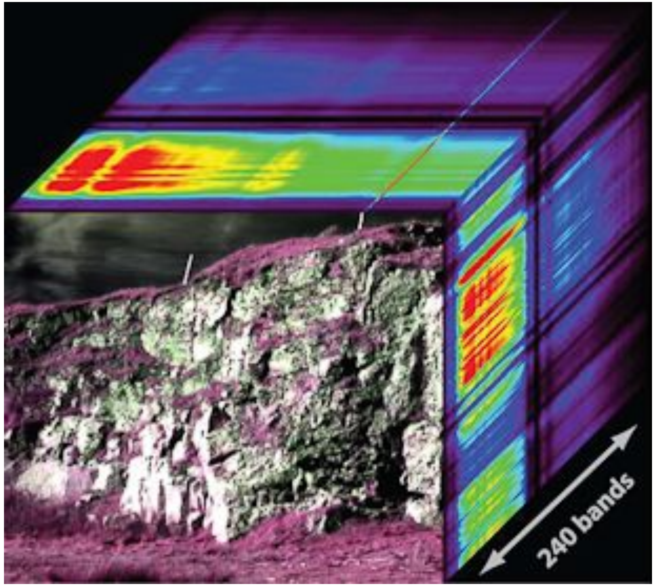
The two blocks are the same shade of grey.

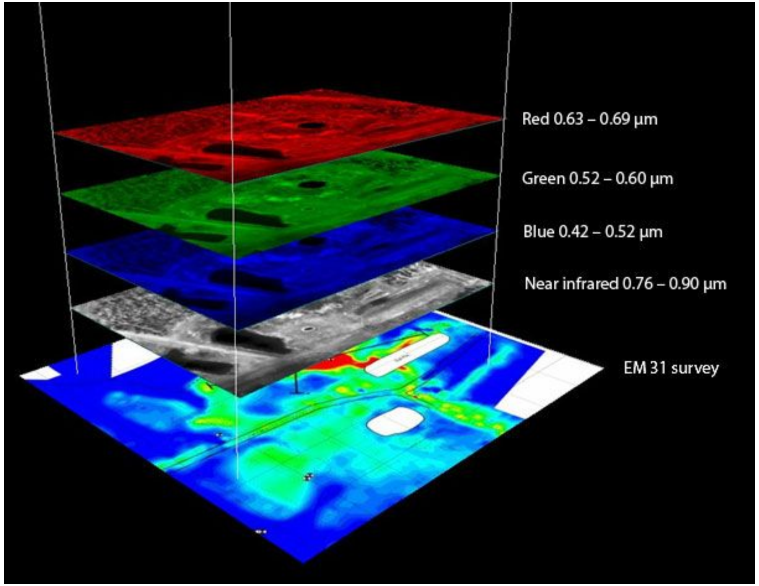
Hold a pen over the 'horizon' to check.

Image: Brett Jordan



Multi- and hyperspectral images





What is digital image

Digital image is a *quantized* and *discrete* vector field (similar to human vision). Each vector component describes:

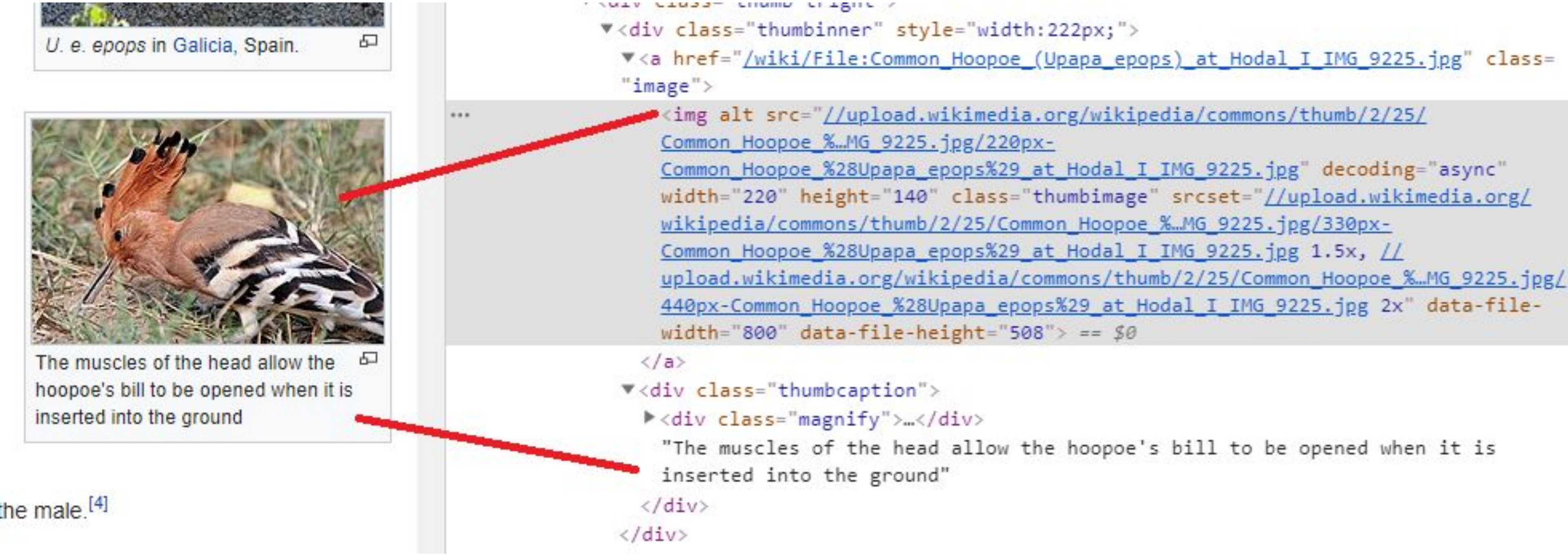
- How much **energy is reflected** in particular spectrum part
  - Images, infrared images, ...

OR

- How much **energy is absorbed**
  - Medical imaging (X-ray, CT)

How images are (were) retrieved

Neighbouring text and subtitles



U. e. epops in Galicia, Spain.

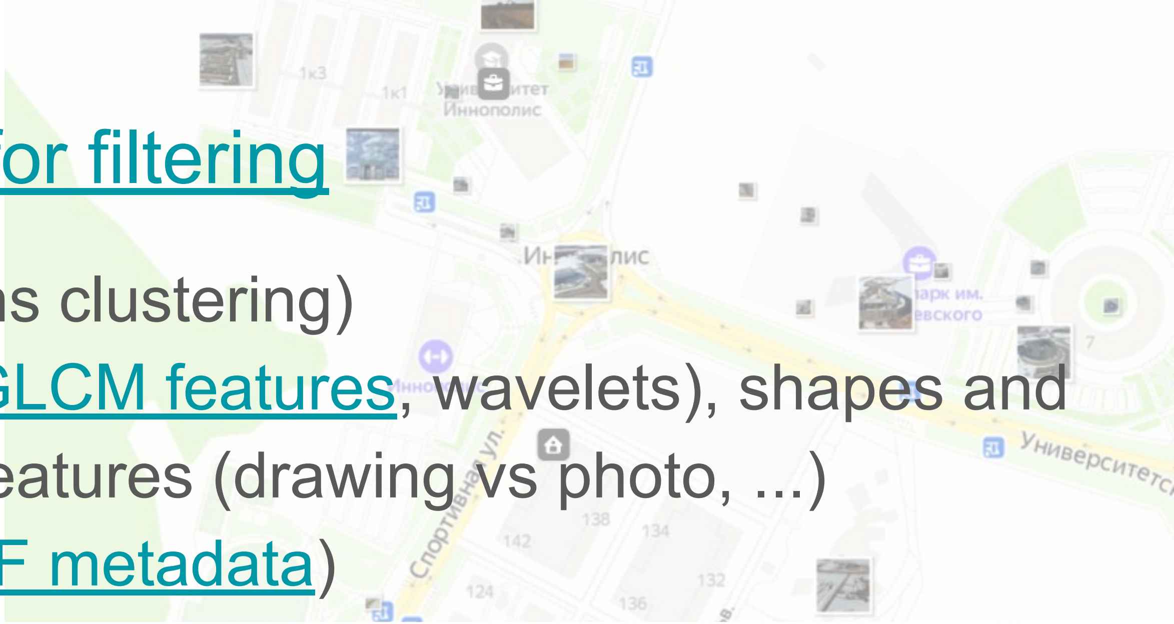
The muscles of the head allow the hoopoe's bill to be opened when it is inserted into the ground

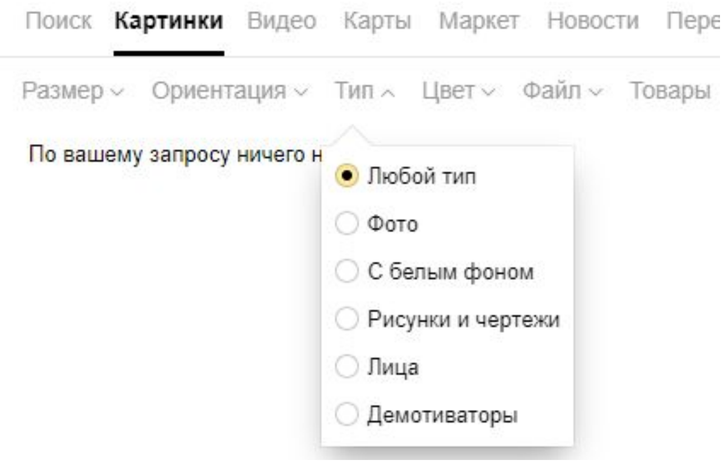
the male.[4]

```
▼<div class="thumb tright">
  ▼<div class="thumbinner" style="width:222px;">
    ▼<a href="/wiki/File:Common_Hoopoe_(Upapa_epops)_at_Hodal_I_IMG_9225.jpg" class=
    "image">
...     <img alt src="//upload.wikimedia.org/wikipedia/commons/thumb/2/25/
        Common_Hoopoe_%…MG_9225.jpg/220px-
        Common_Hoopoe_%28Upapa_epops%29_at_Hodal_I_IMG_9225.jpg" decoding="async"
        width="220" height="140" class="thumbimage" srcset="//upload.wikimedia.org/
        wikipedia/commons/thumb/2/25/Common_Hoopoe_%…MG_9225.jpg/330px-
        Common_Hoopoe_%28Upapa_epops%29_at_Hodal_I_IMG_9225.jpg 1.5x, //
        upload.wikimedia.org/wikipedia/commons/thumb/2/25/Common_Hoopoe_%…MG_9225.jpg/
        440px-Common_Hoopoe_%28Upapa_epops%29_at_Hodal_I_IMG_9225.jpg 2x" data-file-
        width="800" data-file-height="508"> == $0
      </a>
    ▼<div class="thumbcaption">
      ▶<div class="magnify">…</div>
        "The muscles of the head allow the hoopoe's bill to be opened when it is
        inserted into the ground"
      </div>
    </div>
```

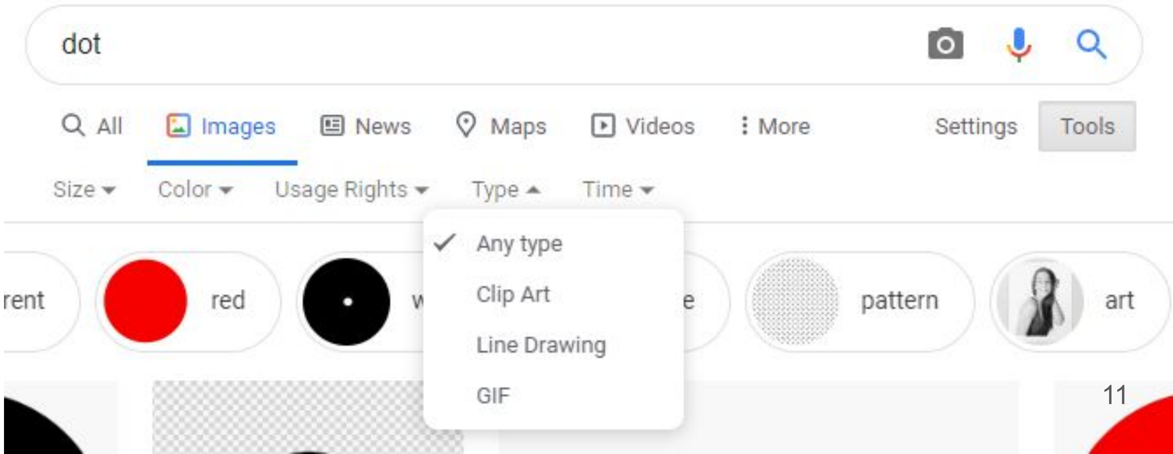
High-level features for filtering

- Color (using k-Means clustering)
- Textures (Haralick/GLCM features, wavelets), shapes and easily computable features (drawing vs photo, ...)
- Metadata (size, EXIF metadata)




Поиск Картинки Видео Карты Маркет Новости Пере
Размер Ориентация Тип Цвет Файл Товары
● Любой тип
○ Фото
○ С белым фоном
○ Рисунки и чертежи
○ Лица
○ Демотиваторы

По вашему запросу ничего н



dot

All Images News Maps Videos More Settings Tools
Size Color Usage Rights Type Time
✓ Any type
Clip Art
Line Drawing
GIF
rent
red
w
e
pattern
art

CBIR = Content Based Image Retrieval

CBIR

Problems (sensitivity increases)

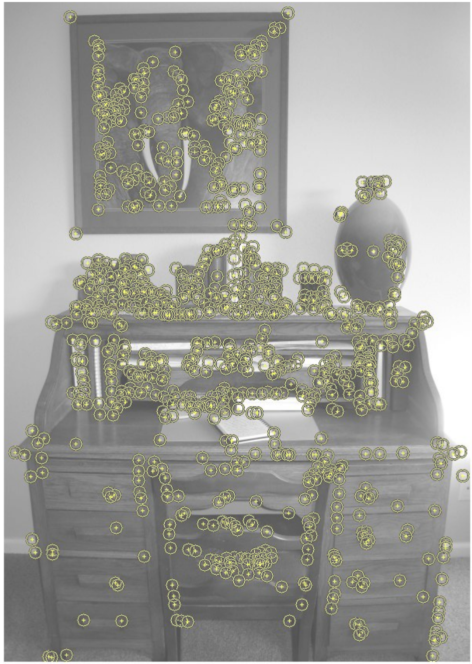
- Similarity search
- Duplicate search
- Identification (exactly the same, but with respect to e.g. compression)

Similarity and duplicate search: image as a bag of words

In CV ... a **feature [point]** is defined as an "interesting" part of an image.

Usually for **interesting points** consider:

- Edges
- Corners
- Regions



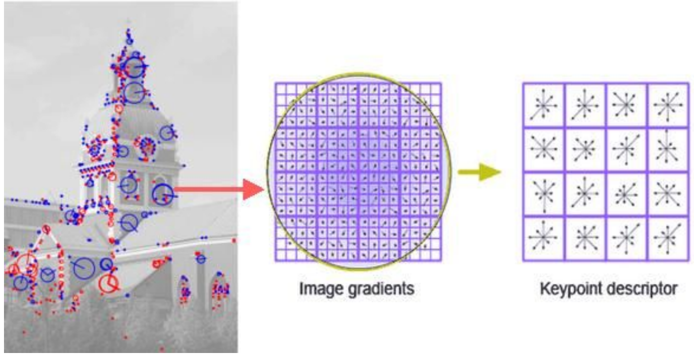
After detector *feature vector* (**descriptor**) is computed.

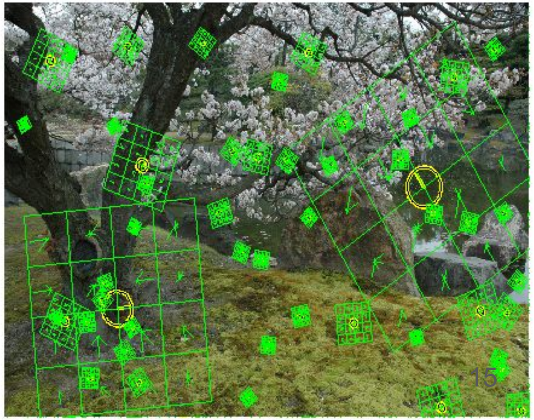
Use feature vector sets to describe **objects**

$$\Delta[G_\sigma(x,y) * f(x,y)] = [\Delta G_\sigma(x,y)] * f(x,y) = LoG * f(x,y)$$

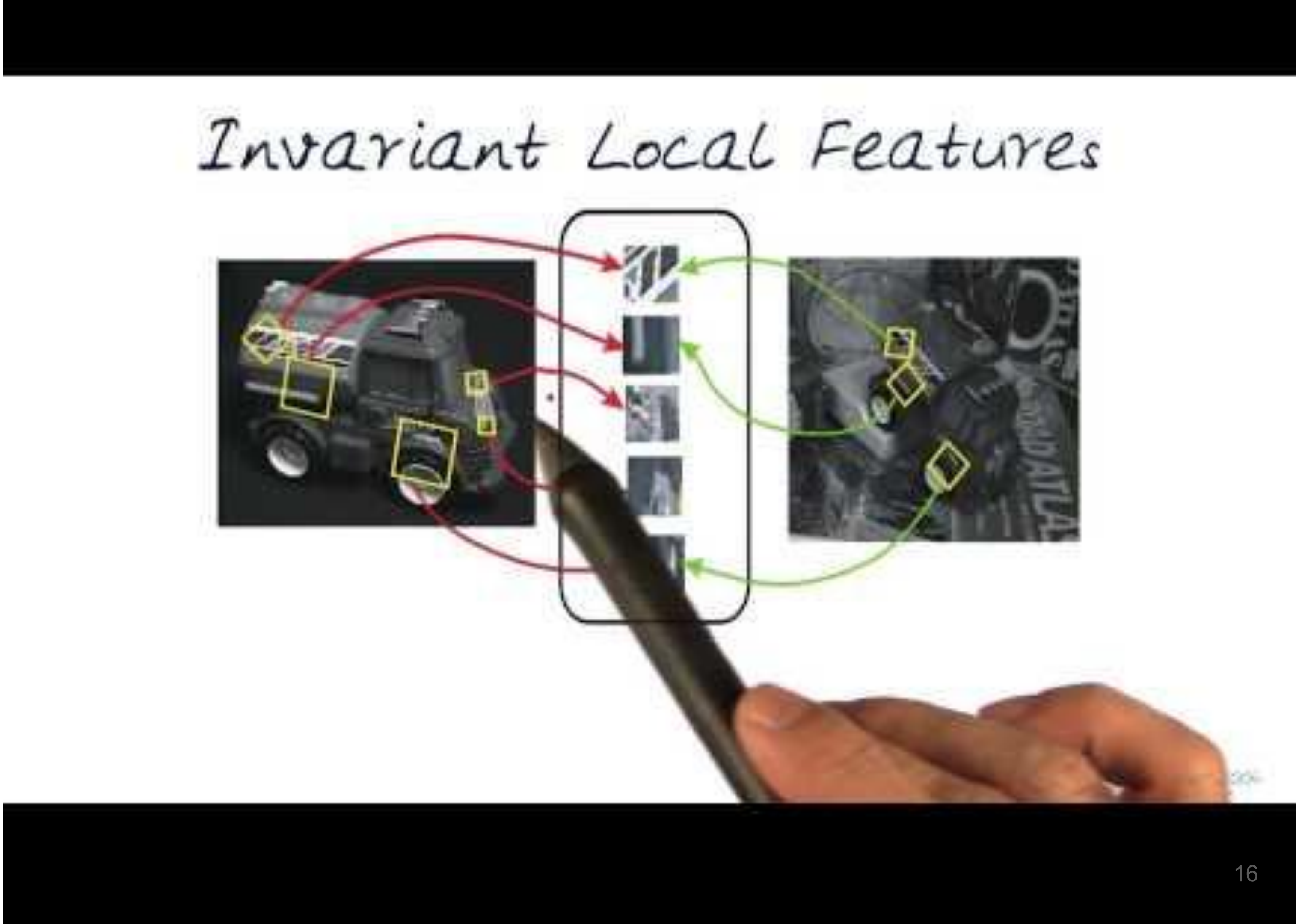
SIFT: Scale-invariant feature transform

1) Compute gradients for images in *image pyramid* using difference of Gaussians (DoG). (Image pyramid ~ Scale invariant)
2) Search for local extrema in scale and space (*keypoints*)
3) Compute *direction* (*rotation invariant*)
4) Create descriptor: in 16x16 neighbourhood make 16 blocks, compute gradients (8 bins for angles) and make a vector.
5) Normalize (*intensity invariant*)





SIFT overview



Invariant Local Features

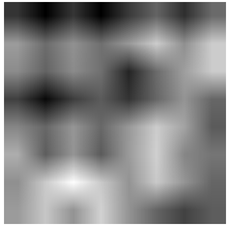
Image fingerprinting for **duplicate search**

1. Use `PoI`. Allows cropping, need ~100 points, fails for texts
2. Use hash functions:
   a. Image.Match based on Xerox features
   - Grayscale color image
   - Place 9x9 uniform grid of pixels
   - Each point is described with 8-neighbourhood {darker = -2, mild darker , … , lighter = +2 }
   - Concatenate

Image fingerprinting for **duplicate search (2)**

- Hash functions (pip install ImageHash):
- [average] aHash
  - Resize to 8x8
  - Grayscale
  - Binarize by average
  - Use Hamming dist

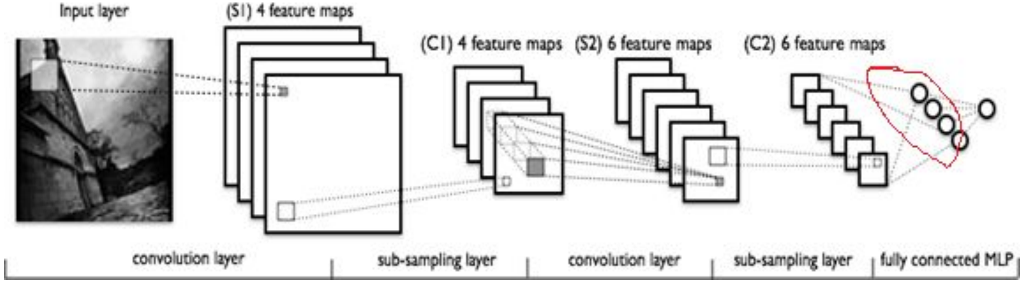
- [perception] pHash and [wavelet] wHash
  - pHash uses DCT
  - wHash - DWT, both coarse grained
  - Use Hamming dist



- [difference] dHash
  - Resize to 9x8
  - Grayscale
  - Compute `I[x+1, y] <> I[x, y]` and use this as a bit

Deep networks for specific and general **similarity** search

1. Images are of **different types** (classes, e.g. ImageNet). Train classification network (AlexNet, VGG16, …) and use embeddings (from inner layer) as index.



2. Images are of the same type (faces). Train deep convolutional autoencoder which creates small-dimensional embeddings.

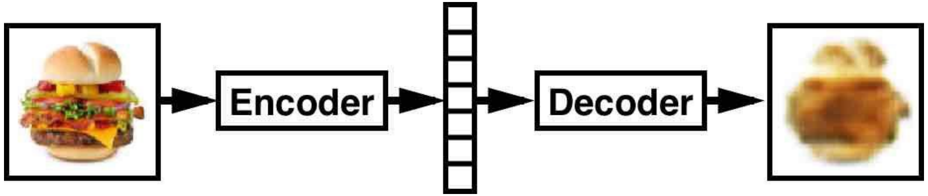


Image understanding, video structure

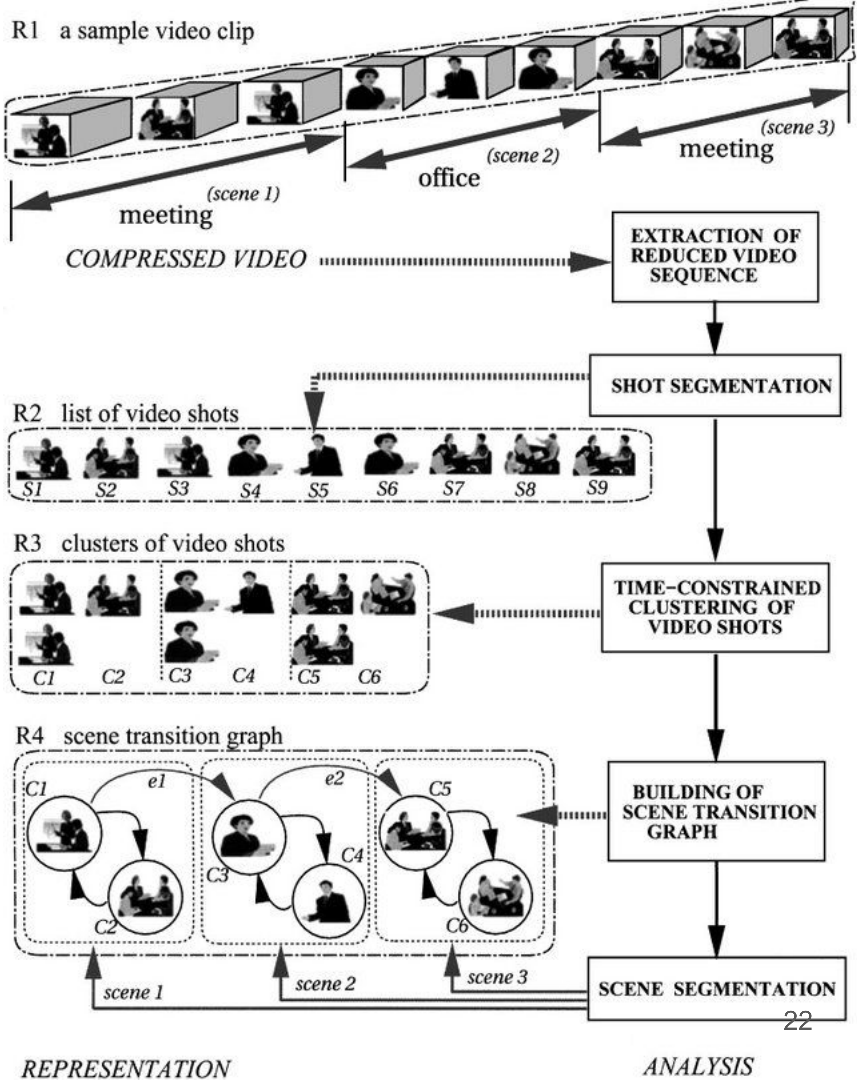
Semantic retrieval

Deep classification and region-based networks allow adding semantic indices.

NB: how many $$ will single inference cost for 20B of images?

Video structure mining

As text can be searched for a **paragraph**, Long videos should be also indexed with **scenes**. [demo]


R1 a sample video clip
(scene 1)
meeting
(scene 2)
office
(scene 3)
meeting
COMPRESSED VIDEO
EXTRACTION OF REDUCED VIDEO SEQUENCE
SHOT SEGMENTATION
R2 list of video shots
S1
S2
S3
S4
S5
S6
S7
S8
S9
R3 clusters of video shots
C1
C2
C3
C4
C5
C6
TIME-CONSTRAINED CLUSTERING OF VIDEO SHOTS
R4 scene transition graph
C1
e1
e2
C5
C2
C3
C4
C6
BUILDING OF SCENE TRANSITION GRAPH
scene 1
scene 2
scene 3
SCENE SEGMENTATION
REPRESENTATION
ANALYSIS