# Object Recognition in Images and Video

http://www.micc.unifi.it/bagdanov/obrec

Prof. Andrew D. Bagdanov

Dipartimento di Ingegneria dell'Informazione
Università degli Studi di Firenze
andrew.bagdanov AT unifi.it

20 April 2017

# Outline

# Course introduction

# A little bit about me

UNIVERSITÀ DEGLI STUDI FIRENZE

# There's no time, let me sum up. . .

- 1960s (California): Born, Los Angeles.
- 1970s (Washington): Farm hand, rural Washington.
- 1980s (Las Vegas): High school student; Deadhead; game designer and programmer for Westwood Studios; Emacs user.
- 1990s (Las Vegas): Semi-professional musician; bartender; sports pub bouncer; car counter; math tutor; Math/CS dual Bachelors/Masters student (large cardinal set theory and image processing); Deadhead, Senior Network Analysis, US Department of Energy.
- Early 2000s (Amsterdam): PhD student, University of Amsterdam; Deadhead.
- Late 2000s (New York/Florence/Rome): Postdoc Renselaer Polytechnic Institute; Deadhead; postdoc University of Florence; Senior Development Chief, Food and Agriculture Organization of United Nations.
- Early 2010s (Florence/Barcelona): Project Leader, Computer Vision Center, Barcelona; Adjunct Professor, Universitat Autonoma de Barcelona; Head of Research Unit, Media Integration and Communication Center, University of Florence, Ramon y Cajal Fellow, Computer Vision Center, Barcelona; Deadhead.
- Today: Professor of Information Engineering, University of Florence.

# Yes, but what do you do?

- Document image understanding: style-based interpretation of document layout and content, low-level degradation estimation, inverse halftoning.
- Video surveillance and security: tracking, active camera control, foveal scheduling, face recognition in the wild.
- Object and action recognition: local pyramidal features, color representations for object recognition, semi-supervised and transductive approaches.
- HCI for cultural heritage: visual profiling of museum visitors, knowledge management for cultural heritage resources, personalizing cultural heritage experiences.
- Person re-identification: iterative sparse ranking (this talk), semi-supervised approaches to local manifold estimation.
- Other random interests: functional programming languages, operating systems that dont suck, long-distance bicycle touring, Emacs, the Grateful Dead.

# Some more specific (and relevant) interests

- Color and object recognition. Color is hard to get right, and easy to get wrong.

  *FS Khan, RM Anwer, J van de Weijer, AD Bagdanov, M Vanrell, AM Lopez, "Color attributes for object detection." In: Proceedings of CVPR 2012.*

- Feature fusion for object recognition. How do you bind multiple local modalities in space?

  *FS Khan, J van de Weijer, AD Bagdanov, and M Vanrell, "Portmanteau vocabularies for multi-cue image representation." In: Proceedings of NIPS 2011.*

- Multiresolution description of local image structure. Why use only one local resolution?

  *L Seidenari, G Serra, AD Bagdanov, A Del Bimbo, "Local pyramidal descriptors for image recognition." IEEE TPAMI, 2015.*

- Sparse coding for person re-identification. When you have few samples for each class, exploit all samples for all classes.

  *G Lisanti, I Masi, AD Bagdanov, A Del Bimbo, "Person re-identification by iterative re-weighted sparse ranking." IEEE TPAMI, 2015.*

# Teaching philosophy and style

- Teaching (and learning) is most effective when it is an interactive give-and-take rather than an I-stand-here-and-preach/you-sit-there-and-listen.
- My job as professor is to put my knowledge and know-how at your disposition.
- You job as students is to suck every last bit of knowledge out of my in these 14 weeks.
- If you don't understand something, interrupt me and ask my to clarify.
- I also expect your active participation in the lectures.
- I won't stand here and say "there is no such thing as a stupid question."
- Better: *there should be no question you are too afraid or to shy to ask.*
- [ I know this much parable ]

# What is object recognition?

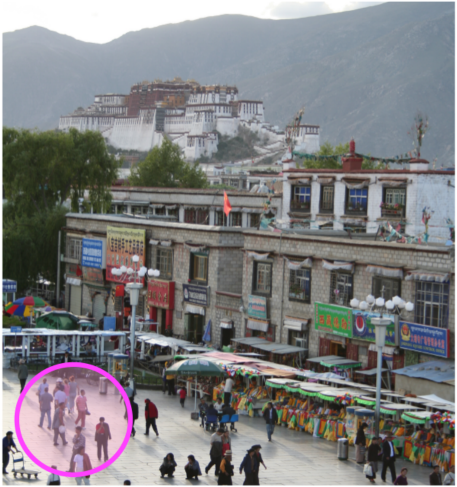
- When we talk about object recognition, we talk about image content.

# What is object recognition?

- This can mean verification: *is this a streetlamp?*

# What is object recognition?

- This can mean detection: *are there people present? If so, where?*



# What is object recognition?

- This can mean identification: *is this Potala Palace?*

# What is object recognition?

- This can mean categorization.

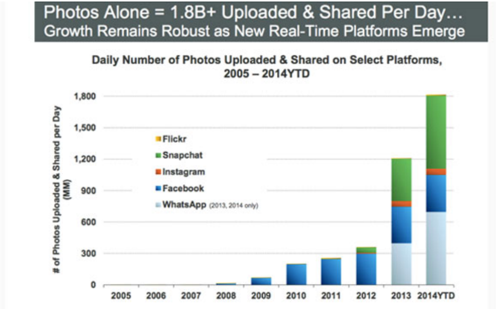
# What is object recognition?

- It can mean segmentation: labeling all pixels with category label.
- It can mean attribution: labeling objects with attributes (flat, hairy, circular, etc).
- It can mean an awful lot of things that have *something* to do with associating meaning to image content.

## A working definition

- For the purposes of this class, we will consider object category recognition and object category detection:
  - Object category recognition: determine if one or more instances of specific object categories are present in an image.
  - Object category detection: if instances of known object categories are present, localize all instances.
- We will relax this distinction for the final lecture when we talk about recent developments at the state-of-the-art.

- Why do we care about object recognition?
- The first motivating factor is the extreme volume of image data generated each month.
- From the graph below[1] we see that 1.8 billion images are uploaded to social media every month.
- Every month.

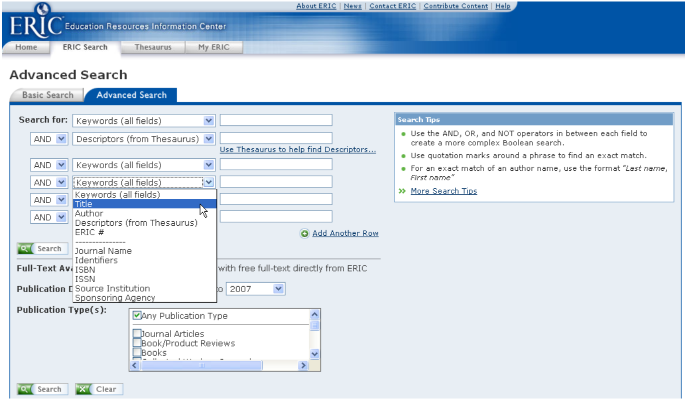



[1] Source: KPCB Meeker Report 2014

# Why is object recognition important?

- You may ask: so what?
- So there are a ton of images on the internet, they are they, I can see them on Facebook/Instagram/Whatever feeds, what does object recognition have to do with this?
- The real problem isn't the volume so much as access to desired content.
- Let's take a minute to consider how we access web content.
- Specifically, how we search for desired web content.

- Actually, let's first look at how we used to access textual content.
- This is an example of a boolean query interface.
- It relies on manually annotated fields for all documents in the collection.



# Why is object recognition important?

- In the 1970s (during the first data explosion, then mainly textual data) we realized that the boolean query model is unsustainable.
- It requires costly and laborious manual annotation of documents.
- And interfaces were clunky and difficult for non-experts.
- Gerald Salton invented an embedding of text documents in a vector space that reflects the word frequency statistics of documents.
- This is the famous TF*IDF model:

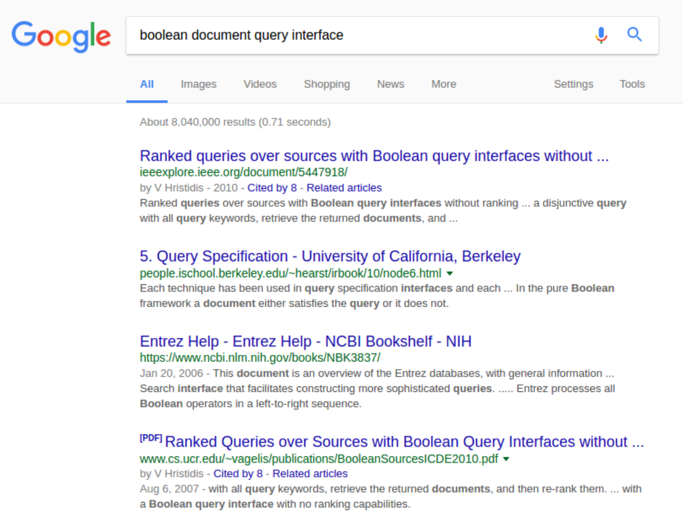
$$\mathrm{tfidf}(t, d, D) = \mathrm{tf}(t, d) \cdot \mathrm{idf}(t, D)$$

- And thus modern information retrieval was born. . .

# Why is object recognition important?



- Now, when we search we are performing a content-based comparison between the query and the document corpus:

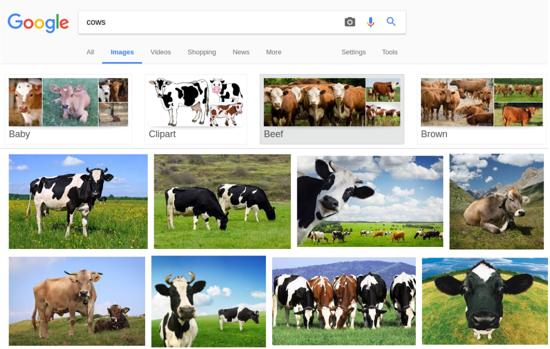


# Why is object recognition important?

- Returning to images: if we knew in the 1970s that manual annotation and boolean queries were unsustainable in light of the "explosion" of text data of the times...
- ... then requiring manual annotation of 1.8 billion images per month is a monumentally unsustainable proposition.
- Without a way to access image by content (similarly to how we access text content), we have few options left to us:
  - Manual categorization: create a taxonomy of images, which reduces the annotation load – but would require a massive number of categories.
  - Naming: give images unique names with which we can recall them – shifts the cognitive burden completely to *user.
  - Tagging: this kinda works – but image tags are noisy and context sensitive.
  - ...

# Why is object recognition important?

- A better solution is to use object recognition to analyze image content.
- This way, we can query images using semantic object categories:



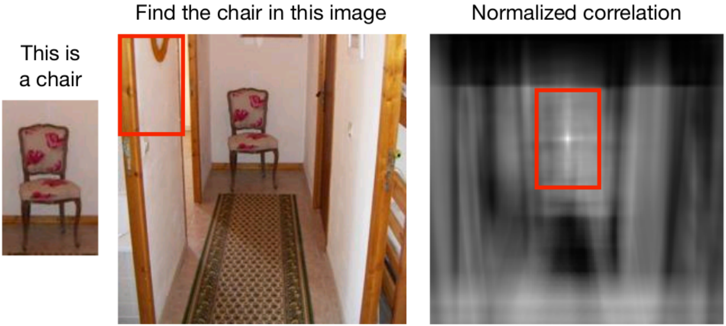
# Why is object recognition hard?

- OK, so what's the real problem?
- The answer: too many to list.
- The TF*IDF model for text retrieval is based on relatively simple analysis of term frequencies in documents and document collections.
- It is not immediately apparent how to do this for images.
- Hence, object recognition can be an intermediate step.
- But, there are innumerable problems with this as well.

# Why is object recognition hard?

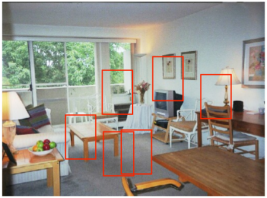


- Let's be naive and do a pixel-by-pixel comparison.
- Take a template and slide it over every position in the image.
- Measure similarity using a normalized correlation:

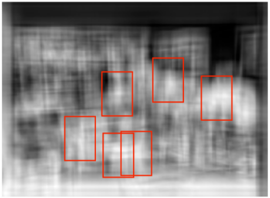


# Why is object recognition hard?

- However, in the presence of:
  - Scale variations: we would need to scan across multiple scales.
  - Rotations (in- and out-of-plane): we should also scan using Rotated templates (and in some way generate out-of-plane rotations).
  - Illumination variations: we should somehow vary illumination of templates or make them invariant in some way.
  - Occlusion: ...
  - Background clutter: ...
  - ...



Find the chair in this image



Will template matching work?

# Why is object recognition hard?

- What we are grasping at is the notion of the semantic and sensory gaps: there is a huge gulf between the raw, pixel-level images captured by sensors and the semantic meaning we associate with image content.
- For text this is easy: we can render our representations invariant to things like document length via simple normalizations.
- For images this is a good deal trickier.
- We will come back shortly to the semantic gap and look at it in more detail.

# Course objectives

- In this course we will trace the development of object recognition (in a general sense) from it's prehistory up through the current state-of-the-art.
- Given the brief nature, our treatment will be necessarily synthetic.
- The goal is not to make you all experts, but rather to give you a broad overview of the field and to leave you in a position to be capable of comprehending the current developments in the field.
- Object recognition (and modern computer vision in general) is an extremely dynamic and vibrant field of study.
- As such, it is difficult to be comprehensive in any meaningful way.

# Course overview

## First steps: how did we get here?

- In this lecture we will see some seminal works from the (pre-) history of object recognition.
- It is important to understand how we arrived at the state-of-the-art we know today.
- We will begin by looking at the work by David Marr on module-based visual recognition.
- Then, we will see a snapshot of the state-of-the-art in contend-based image access at the dawn of the modern era of object recognition.
- This will help us understand the worldview that led to the first big breakthrough in object recognition: the Bag-of-Words (BOW) model.

## The slow march of progress

- In the second lecture we will see how the complementary object detection and object recognition problems were approached through the successes of the early 2000s
- We will look at the HOG feature descriptor and how it led to breakthroughs in object detection.
- We will see how the community revived early theories of Marr and Poggio, integrating them with modern features in the Deformable Part Model (DPM) detector.
- Then we will see how the Bag-of-Words model was incrementally improved through addition of techniques like: spatial pyramids, sparse coding, soft assignment, Fisher vector and VLAD encoding.

## The shot heard 'round the world

- In this lecture we will look at the revolutionary breakthrough that occurred in 2012: the re-introduction of neural networks into the modern discussion on object recognition.
- We will study some of the classic and contemporary models of Convolutional Neural Networks (CNNs) that continue to revolutionize the field.
- We will also look at extensions of these models to the detection problem and to object recognition in video.

## Where we are today

- In this final lecture we will leverage what we have learned about the historical development of modern object detection to study some state-of-the-art topics in object recognition.
- We will see how captioning, for example, can be thought of as a natural generalization of the classical recognition problem.
- We will also study several advanced CNN architectures for recognition and detection.

# Administrivia

# Schedule

## Course schedule

- 20/04/2017 10:00 – 13:00: Introduction
- 27/04/2017 10:00 – 13:00: Detection and Advanced Bag-of-Words
- 04/05/2017 10:00 – 13:00: Deep Convolutional Neural Networks
- 11/05/2017 10:00 – 13:00: The state-of-the-art

## Course policies

- This course is organized as a reading group course.
- Each lecture (except today) will have 4-5 required papers to read.
- You must read the required papers and you must be prepared to participate in the discussions.
- In each lecture, I will give an overview presentation of each article and open the discussion.

# Final exam

UNIVERSITÀ DEGLI STUDI FIRENZE

## Exam

- For the final exam, you will will be required to prepare a 20-minute presentation on a paper of your choice.
- This paper should:
  - have something to do with object recognition; and
  - have been published at a top conference in the last year.
- A date will be fixed for final presentations approximately three weeks after the end of the course.

# A complementary lab

- There is a laboratory course on object recognition in practice being offered in the PhD in Smart Computing.
- This lab is highly complementary to the material we will cover here.
- In fact, we have designed the schedules to overlap for the last two lectures of this class and the first two laboratory sessions.
- That way, you can attend my lectures in the morning (om May 4th and May 11th), and then attend the laboratory sessions in the afternoon.
- See the PhD in Smart Computing page for more details.

Questions?

# Questions?

- Questions?
- (plus an informal survey)

# Introduction to Object Recognition

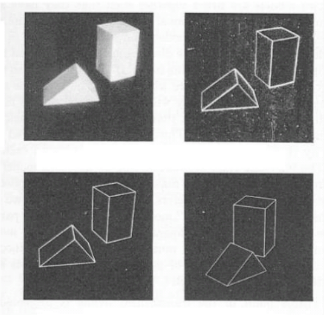
# First steps

- From the very beginning of the digital computing revolution we have been interesting in analyzing visual media.
- It is one of the uniquely human things that we all do (interpret visual content).
- And thus, visual recognition is one of the oldest categories of problems in artificial intelligence.
- Most early work built upon biologically-inspired features (e.g. wavelets and other frequency-based filters).

# First steps

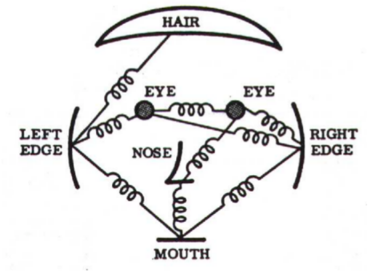
- A first classic approach was from Roberts[2] (known as Blocks World).
- It is typical of early works: use constrained 3D models to recognize objects from simple image features.
- These works seem naive from a modern perspective because they assume the need to explicitly model 3D reality.


L. Roberts



[2]L Roberts: Machine perception of 3-d solids. In: PhD. Thesis (1965)

- This type of explicit model of recognition gave way to part-based representations.
- An object was represented by a set of parts arranged in an elastic configuration.[3]
- The trend: move away from models and move towards the image.

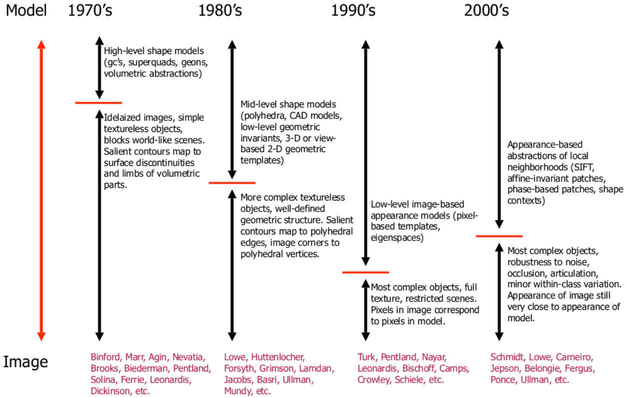



[Fischler & Elschlager 73]

[3] MA Fischler, and RA Elschlager: The representation and matching of pictorial structures. IEEE Transactions on computers, 1973

# First steps (jumping to the chase)

- Here is a high-entropy summary of historical developments.[4]




[4] S Dickinson: The evolution of object categorization and the challenge of image abstraction, 2009

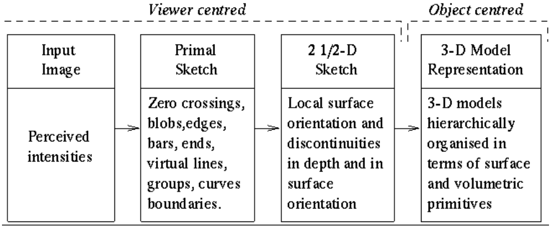
- And now we come to the first paper in our list of readings.
- In reality, it isn't a paper, but rather a book chapter.
- It is the introduction to the book *Vision* by David Marr.
- This book (published posthumously in 1980) is considered one of the seminal works in computer vision.
- It is important not because it proposes a workable – or even tractable – theory of vision and object recognition.
- Rather, it is important because it is one of the first works to propose a complete theory of vision systems.
- Before this, most works concentrated on highly specialized sub-problems and not on end-to-end vision as a whole.

# Marr's Vision

- A central tenet of Marr's theory is that *vision is a complex information processing task*.
- The goal of which is to *capture and represent various aspects of the world that are of use to us* (e.g. objects).
- Marr approached object recognition in a systematic way, dividing the process into:
  - Computational Theory: what is the goal, why is it appropriate, and how can it be carried out?
  - Representation and Algorithmics: how can the theory be implemented, and what representations are necessary?
  - Implementation: how can the representations and algorithms be realized?
- These are not hard and fast divisions, but Marr argued that no explanation is complete unless it covers all three.

# Marr's Vision

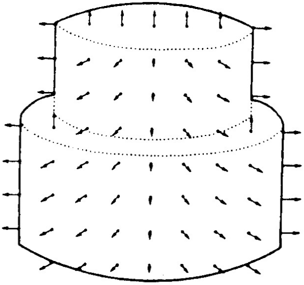
- In his book, Marr developed a modular framework for computer vision.
- This framework consists of three representations that are created, maintained, and interpreted by the process of vision:



# Marr's Vision

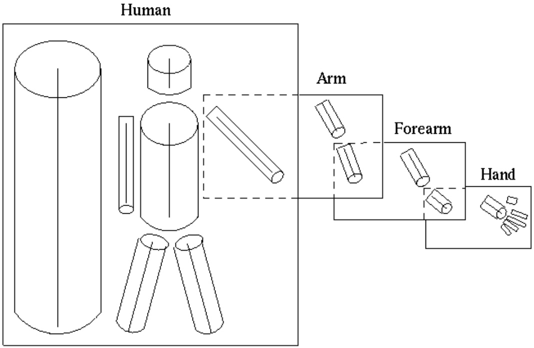
- The Primal Sketch is a description of the intensity changes in the image and their local geometry.
- It is based on the assumption that intensity variations are likely to correspond to physical realities like object boundaries.


- The 2.5D Sketch is a viewer-centric representation of orientation and depth of visible surfaces drawing from the primal sketch.
- Note that no grouping is done yet: we are only associating weak geometry to image elements.
- Hence the metaphor 2.5D sketch.



# Marr's Vision

- The 3D Model is an object-centric representation of 3D objects in the image.
- The goal of this model is to enable object manipulation and recognition.



# Marr's Vision

- Marr's contribution is primarily of historical interest at this point.
- The most characteristic feature of his theory is a tireless attempt at rigor in the study of human visual information processing.
- Marr's theory is intended as a computational model of human vision.
- Many of the approaches proposed in his body of work (especially at the primal sketch level and how image features are identified) are still in use today.
- So how did the field advance after Marr? We will flash forward in time 20 years from 1980 to 2000...

# CBIR: Consolidating the Field

- Between the years 1980 and 2000, great advances were made in the field of Content-based Image Retrieval (CBIR).
- These advances were instrumental in putting the tools in place that led to modern approaches to object recognition.
- A milestone was the publications of a survey of CBIR: 0.1in*

{

*Content-based image retrieval at the end of the early years*,
Smeulders, A. W., Worring, M., Santini, S., Gupta, A., and Jain, R.
In: IEEE Transactions on pattern analysis and machine intelligence, 2000.

- This paper is long and reviews more than 200 papers in CBIR – nonetheless it is well worth a skim to understand the historical context.

# CBIR: Consolidating the Field

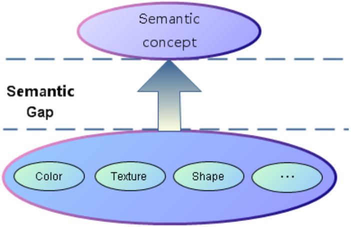
- The paper introduced two concepts into the discussion on object recognition (and computer vision in general).
- The first is the sensory gap:

  *The sensory gap is the gap between the object in the world and the information in a (computational) description derived from a recording of that scene.*

- Think about this for a moment: we are always working with an imperfect reconstruction of the real world.
- Images have limitations: they have finite resolution, they are subject to noise processes, they are acquired with a sensor which is another free object in the world.
- This sensory gap must be surpassed in order to render object recognition invariant to scene-incidental artifacts.

- The other key concept is the semantic gap:

  *The semantic gap is the lack of coincidence between the information that one can extract from the visual data and the interpretation that the same data have for a user in a given situation.*



# CBIR: Consolidating the Field

- This paper changed the dialog on object-centered image analysis by subtly shifting the focus.
- Like Marr, the semantic and sensory gaps give a natural division between analysis modules.
- From 2000 on, the discussion shifted away from holistic recognition, towards a semi-conscious recognition of whether a proposed technique was bridging the sensory or semantic gap.
- Another key idea promoted in this paper is the importance of invariance to addressing both problems.

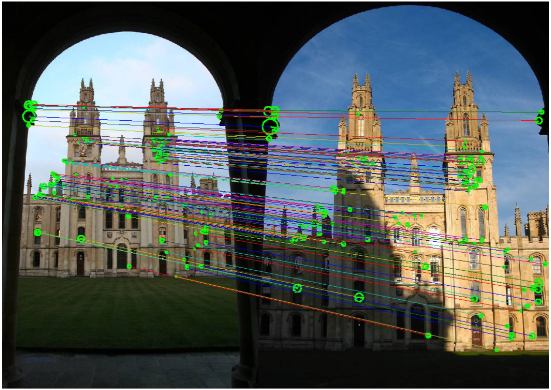
# Local Descriptors of Image Structure

# Two missing links

- Returning to our analogy with text retrieval, if we want to apply a similar approach we have a three fundamental problems.
- The first is how to decompose image content into a set of primitive components.
- The second is how to describe these components in a discriminative, yet sufficiently invariant way.
- A solution to these two problems was proposed in one of the first landmark papers in modern object recognition: *Distinctive Image Features from Scale-Invariant Keypoints*, David G. Lowe. In: International Journal of Computer Vision, 2004.
- The Scale Invariant Feature Transform (SIFT) descriptor is proposed in this paper.
- We will now take a quick look at how it is computed.

# The SIFT Descriptor

- The descriptor was originally proposed as a descriptor for local feature matching.
- For such problems, a stable feature invariant to translation, scale and affine/perspective transformation, and rotation is needed.



# The SIFT Descriptor

## The basic SIFT pipeline

1. Scale-space extrema detection: The first stage searches over all scales and image locations using a difference-of-Gaussian filter to identify potential interest points that are invariant to scale and orientation.
2. Keypoint localization: At each candidate location, a detailed model is fit to determine location and scale. Keypoints are selected based on their stability.
3. Orientation assignment: An orientation is assigned to each keypoint based on local image gradient directions. All future operations are performed on image data that has been transformed relative to the assigned orientation, scale, and location for each feature, thereby providing invariance to these transformations.
4. Keypoint descriptor: The local image gradients are measured at the selected scale in the region around each keypoint. These are transformed into a representation that allows for significant levels of local shape distortion and change in illumination.

## Build a scale-space pyramid

- Use iterated Gaussian smoothing to analyze blob structure in image $I$:

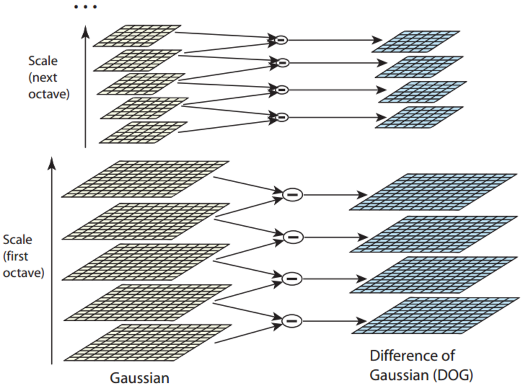
$$\begin{aligned} L(x, y, \sigma) &= G(x, y, \sigma) * I(x, y) \\ G(x, y, \sigma) &= \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \end{aligned}$$

- Then, compute differences at each level of the pyramid:

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned}$$

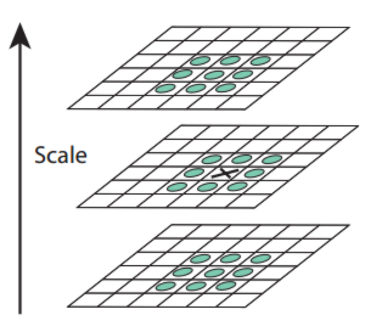
- This is basically a discrete approximation of the (more expensive) Laplacian of Gaussian filter $\sigma^2 \nabla^2 G$:

- What's really going on:



## Finding extrema

- Then, we search for extrema in this scale space.
- By extrema we mean extrema in space and in scale.
- At each point in every image in the DoG pyramid, we check to see it it is larger or smaller than its 26 neighbors.
- This localizes "interesting" points in space and scale.



- What is happening is we are locating where features are distinct and disappearing:

UNIVERSITÀ DEGLI STUDI FIRENZE

### Extrema: localization refinement

- Problem: scale-space extrema localization can be unstable in the presence of even small amounts of noise.
- In some applications it is thus necessary to refine locations of detected extrema.
- Lowe uses a 2nd-order Taylor expansion of the scale-space function:

$$D(\mathbf{x}) = D + \frac{\partial D^T}{\partial \mathbf{x}} + 0.5\mathbf{x}^T \frac{\partial^2 D}{\partial \mathbf{x}^2}\mathbf{x}$$

- The extrema of this function (found by evaluating the derivatives at the discrete, DoG-localized extrema and setting the derivative of the above function to 0) are used as the new keypoint location $\widehat{\mathbf{x}}$.

# The SIFT Descriptor

## Extrema: contrast thresholding

- Another problem in DoG-localized keypoints is that they might have low contrast.
- This can be determined by inspecting the scale-space function directly: high magnitudes implies high contrast.
- Lowe uses a threshold of $|D(\hat{\mathbf{x}}) < 0.03|$ to filter low-contrast points.

## Extrema: curvature thresholding

- A final problem in keypoint selection is when keypoints are localized on scale-space extrema corresponding to edges.
- In this case, the location of the keypoint is sharp and robust to noise in one direction, but unstable in the other.
- Lowe uses a trick similar to the Harris keypoint detector to using the Hessian $\mathbf{H}$:

$$\begin{aligned}
\mathrm{Tr}(\mathbf{H}) &= D_{xx} + D_{yy} = \alpha + \beta \\
\mathrm{Det}(\mathbf{H}) &= D_{xx} + D_{yy} = \alpha + \beta \\
\frac{\mathrm{Tr}(\mathbf{H})^2}{\mathrm{Det}(\mathbf{H})} &< \frac{(r+1)^2}{r}
\end{aligned}$$

# The SIFT Descriptor



## Orientation assignment

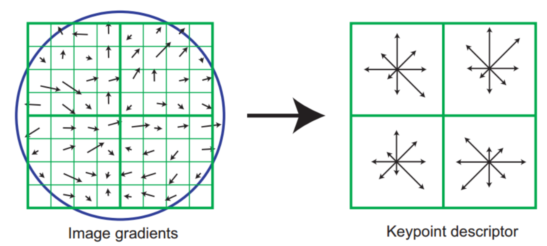
- OK, we have now detected keypoints, localized them to sub-pixel accuracy, and filtered unstable candidates.
- Now we must describe the local image structure around keypoints in a suitable invariant way.
- First step: assign an orientation to keypoints using local orientations and the gradient magnitude:

$$
\begin{aligned}
m(x,y) &= \sqrt{(L(x+1,y)-L(x-1,y))^2+(L(x,y+1)-L(x,y-1))^2} \\
\theta(x,y) &= \tan^{-1}((L(x,y+1)L(x,y-1))/(L(x+1,y)-L(x-1,y)))
\end{aligned}
$$

- A histogram of 36 quantized orientations is computed around the keypoint.
- The contribution of each is weighted by the gradient magnitude and a Gaussian centered at the keypoint location ($\sigma = 1.5 * \text{scale of keypoint}$).
- The maximum in this histogram is the dominant orientation of the keypoint.

## Local Structure Description

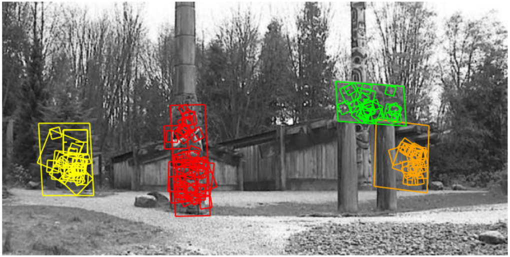
- First we compute gradient magnitude and orientation at each image sample point in a region around the keypoint location
- These samples are then accumulated into orientation histograms summarizing the contents over $4 \times 4$ subregions.
- Standard configuration: $4 \times 4$ subwindows, 8-bin orientation histograms, yielding a SIFT descriptor of 128 dimensions.



## Local Structure Description: some important details

- In order to enhance invariance to rotation, all orientation values are relativized with respect to the keypoint orientation before binning.
- To enhance invariance to illumination changes, the final descriptor (a concatenation of local orientation histograms) is normalized to unit length.
- Lowe performs extensive experiments in the paper to demonstrate the robustness to noise, orientation, scale, and illumination changes.

# The SIFT Descriptor: Applications



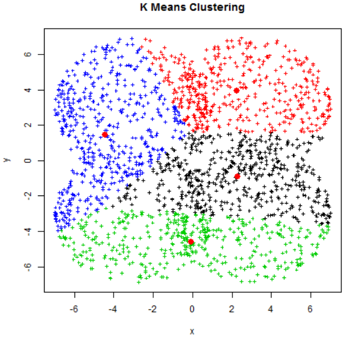
- It is hard to overstate the importance of the SIFT descriptor in the history of the development of object recognition.
- It was literally the feature descriptor of choice for more than a decade.
- It is an example of a nearly perfect balance of theory and engineering.
- Funny story. . .

# The Bag-of-Words Model

- Now we will shift our discussion to one of the first Big Breakthroughs in modern object recognition. *Visual Categorization with Bags of Keypoints*, Gabriella Csurka, Christopher R. Dance, Lixin Fan, Jutta Willamowski, Cédric Bray. In: European Conference on Computer Vision (ECCV), 2004.
- These ideas were developed independently, in many places, at the same time.
- This paper is one of the first, and in my opinion the simplest explanation of the basic Bag-of-Words pipeline.
- Again returning to our analogy with *text retrievalř, we now have a reasonably invariant way to describe local image structure.
- However, we still don't have a concept corresponding to words.
- SIFT features are 128-dimensional vectors, which are not discrete enough to use in a TF*IDF model.

# Feature Quantization

- Key idea: use clustering to identify groups of SIFT points using a training set.
- The centers are used as a visual vocabulary – words in our model.
- All SIFT descriptors extracted from training or test images are quantized to the closest visual word in our vocabulary.
- We have gone from an infinite class of SIFT descriptors, to a finite class of visual words.

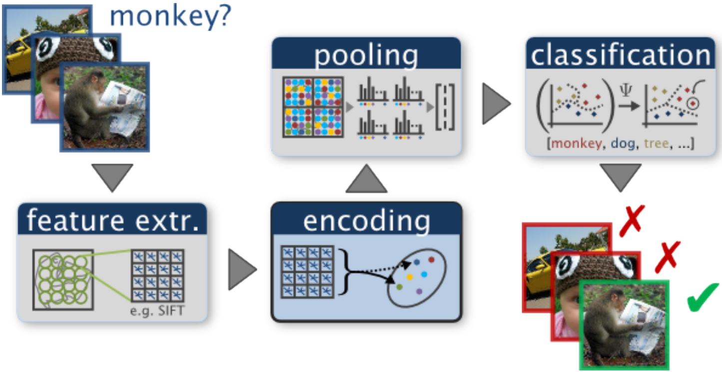


# Feature Pooling

- One last problem: the number of SIFT descriptor is variable: each image will yield a different number of points.
- Also, the order of points (for comparison, for example) is crucial.
- This problem makes it hard to apply standard, machine learning techniques to our representation (e.g. SVM, naive-Bayes, nearest neighbor, etc).
- The solution: like in text retrieval, use pooling to build a fixed-length descriptor of images that is invariant to descriptor order.
- Our descriptor is a histogram of frequencies of visual word occurrences in the image.
- To compare images we can now use: inner products (like TF*IDF), SVMs, and a vast array of tried and true classifiers.
- This last point is most important: given a training set of images labeled with object categories, we can train classifiers to recognize objects in unseen test images.

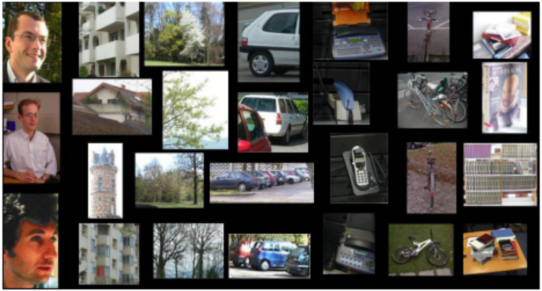
# The Bag-of-Words Model

- This full pipeline is best explained graphically



# The Bag-of-Words Model

- Csurka et al. demonstrated the BOW approach on a dataset with 7 object categories.
- They extract BOW descriptors from training images and train a multiclass, one-versus-all, linear SVM for each.



# The Bag-of-Words

- The punchline: the results on this challenging dataset are impressive.
- The approach uses a small vocabulary of 1000 visual words (in text retrieval, 100K+ word dictionaries are common).
- It also uses an extremely simple linear SVM for classification.

| True classes → | *faces* | *buildings* | *trees* | *cars* | *phones* | *bikes* | *books* |
|---|---|---|---|---|---|---|---|
| ***faces*** | **98** | 14 | 10 | 10 | 34 | 0 | 13 |
| ***buildings*** | 1 | **63** | 3 | 0 | 3 | 1 | 6 |
| ***trees*** | 1 | 10 | **81** | 1 | 0 | 6 | 0 |
| ***cars*** | 0 | 1 | 1 | **85** | 5 | 0 | 5 |
| ***phones*** | 0 | 5 | 4 | 3 | **55** | 2 | 3 |
| ***bikes*** | 0 | 4 | 1 | 0 | 1 | **91** | 0 |
| ***books*** | 0 | 3 | 0 | 1 | 2 | 0 | **73** |
| ***Mean ranks*** | 1.04 | 1.77 | 1.28 | 1.30 | 1.83 | 1.09 | 1.39 |

- Added bonus: visual words are semantically meaningful (note, this example from Csurka et al. is highly cherry-picked):

- Another bonus: the one-versus-all SVM architecture can recognize multiple object categories in images.

| | | |
|---|---|---|
| phones, books, cars | bikes, buildings, cars | buildings, cars, faces |

# Discussion

# Discussion

- Like the SIFT descriptor, it is hard to overstate the impact and influence the Bag-of-Words model has had on the development of modern object recognition.
- It is a hallmark result, despite its extreme simplicity (in hindsight).
- The paper of Csurka et al. was the first to demonstrate the plausibility of efficient, accurate, and robust object category recognition over a large number of categories with extreme visual variance.
- Clearly, this simple BOW model was only the beginning.
- The next ten years of computer vision was dominated by incremental improvements and refinements of this model.
- In the next lecture we will head of in that direction with a survey of advanced Bag-of-Words models that came after.
- Note: see the course website for the required reading for next week.