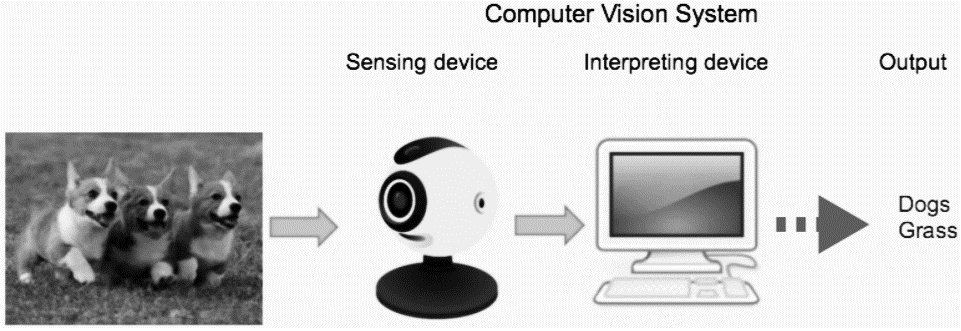
**Chapter 12**

**Computer vision**

**Introduction to Computer Vision**

Computer vision is a scientific field which deals with how computers can be made as high level devices which understand digital images and videos. In terms of engineering, it is an automated task that the human visual system can do. Computer vision has methods for acquiring, processing, analyzing and understanding the digital image. The most important task is to extract high dimensional data from the real world which can produce numerical or symbolic information.

As a scientific discipline, computer vision is related to the theory of artificial systems which can extract information from images. The image data is used in the form of video sequences which can be seen by a human. The ultimate goal of computer vision is to model, replicate, and more importantly exceed human vision using computer software and hardware at different levels. It needs knowledge in computer science, electrical engineering, mathematics, physiology, biology, and cognitive science.

****

***Figure 1: Computer Vision System***

Scientists were inspired by the human vision system and in the past few years we have done an amazing job to extend this visual ability to machines. So, in order to mimic the human vision system, we need the same two main components (Figure 1):

1. **A sensing device:** to mimic the function of the eye.
2. **A powerful algorithm:** to mimic the brain function in interpreting and classifying the image content.

**Some general definitions to computer vision (CV)**

* **CV is an Interdisciplinary scientific** field.
* CV deals with **how computers can be made to gain high-level understanding** from digital images or videos.
* CV Seeks to **automate** **tasks** that the **human visual system can do.**
* **CV** algorithms perform **Automatic extraction**, **analysis** and **understanding** of useful information from a **single image or a sequence of images.**
* CV involved the development of a **theoretical** and **algorithmic** **basis** to **achieve** **automatic** visual understanding.
* CV is concerned with the **theory** behind **artificial systems that extract** information from images.

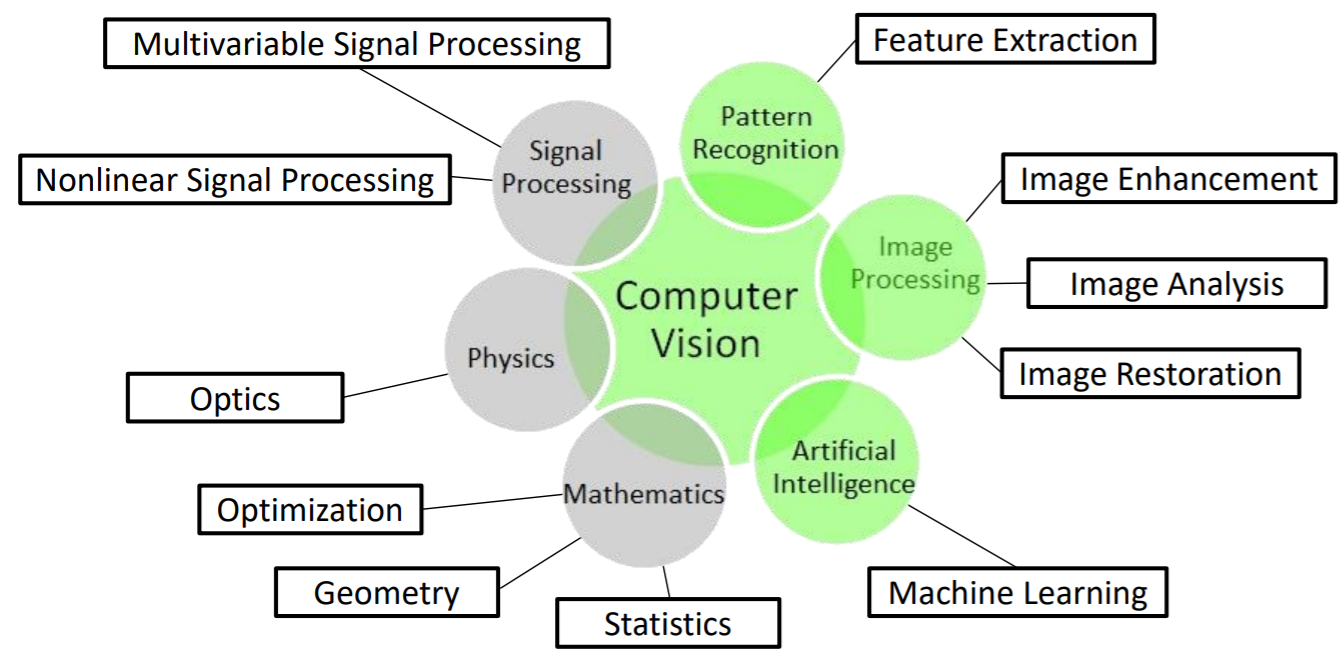
**Computer Vision Hierarchy**

Computer vision is divided into three basic categories that are as following:

* **Low-level vision**: process image for feature extraction (edge, corner, or optical flow).
* **Middle-level vision**: object recognition, motion analysis, and 3D reconstruction using features obtained from the low-level vision.
* **High-level vision**: interpretation of the evolving information provided by the middle level vision as well as directing what middle and low level vision tasks should be performed. Interpretation may include conceptual description of a scene like activity, intention and behavior.

**Computer Vision Related disciplines**

Computer Vision is an overlapping field drawing on concepts from areas such as artificial intelligence, digital image processing, machine learning, deep learning, pattern recognition, probabilistic graphical models, scientific computing and a lot of mathematics (Figure 2).

****

***Figure 2 : Related disciplines to Computer Vision.***

1. **Computer Vision and Artificial Intelligence**

* Deal with **autonomous planning** or **deliberation for robotical systems** to **navigate through an environment.**
* **Detailed understanding of these environments** is required to navigate through them.
* **Information about the environment could be provided by a computer vision** system.
* **Acting as a vision sensor and providing high-level** information about the environment and the robot.
* Artificial intelligence and computer vision share other topics such **as pattern recognition and learning techniques.**

1. **Computer Vision and Information Engineering:**

* Computer vision is often considered ***to be part of information engineering.***
* Information engineering is 🡺 the engineering discipline that deals with the **generation**, **distribution**, **analysis**, and **use of information, data, and knowledge** in systems.

1. **Computer Vision and Solid-state Physics:**

* **Computer vision systems rely on image sensors**, which detect electromagnetic radiation, which is typically in the form of either **visible or infra-red light.**
  + Sensors are designed using quantum physics.

1. **Computer Vision and Neurobiology:**

* **Biological vision system.**
* **How "real" vision systems operate** in order to solve certain vision related tasks
* **Computer Vision studies and describes the processes implemented in software and hardware behind artificial vision systems.**

1. **Computer Vision and Signal Processing**

* **Processing of one-variable signals**, can be extended in a natural way to processing of **two-variable signals or multivariable signals in computer vision.**
* Together with the multi-dimensionality of the signal, this defines a subfield in signal processing as a part of computer vision.

1. **Other Fields**

* Many of the related research topics can also be studied from a purely **mathematical point of view**
* Many methods in computer vision are based on **statistics, optimization or geometry**

**Computer vision and some Closely related fields:**

There is a **significant overlap** in the range of techniques and applications that cover such as (image processing - image analysis - machine vision).

1. **Computer Graphics Vs. Computer Vision**

* Computer Graphics 🡺 **produces image data from 3D models**
* Computer Vision 🡺 **produces 3D models from image data**
* Combination of the two disciplines: **augmented reality**

1. **Image Processing and Image Analysis Vs. Computer Vision**

* Both tend to focus on **2D images** and **how to transform one image to another**
* **By pixel-wise operations such as:**
  + **contrast enhancement**
  + local operations such as **edge extraction or noise removal**
  + geometrical transformations such as **rotating the image**
* Both neither require **assumptions** nor **produce interpretations** about the image content

1. **Computer Vision and 2D Images**

* Computer vision includes **3D analysis from 2D images**
* This analyses **the 3D scene projected onto one or several images.**
* How to **reconstruct structure or other information about the 3D scene from one or several images.**

1. **Machine Vision Vs. Computer Vision**

* **Process of applying a range of technologies** and methods to provide imaging-**based automatic inspection**, **process control** and **robot guidance** in industrial applications
* Machine vision : process of combining **automated image analysis** with other **methods and technologies to provide automated inspection** and robot guidance in industrial applications
* Tends to **focus on applications**.
* **Vision based robots and systems for vision based inspection, measurement**, or picking.
* **Image sensor technologies and control theory** often are **integrated with the processing** of image data to control a robot.

1. **Imaging Vs. Computer Vision**

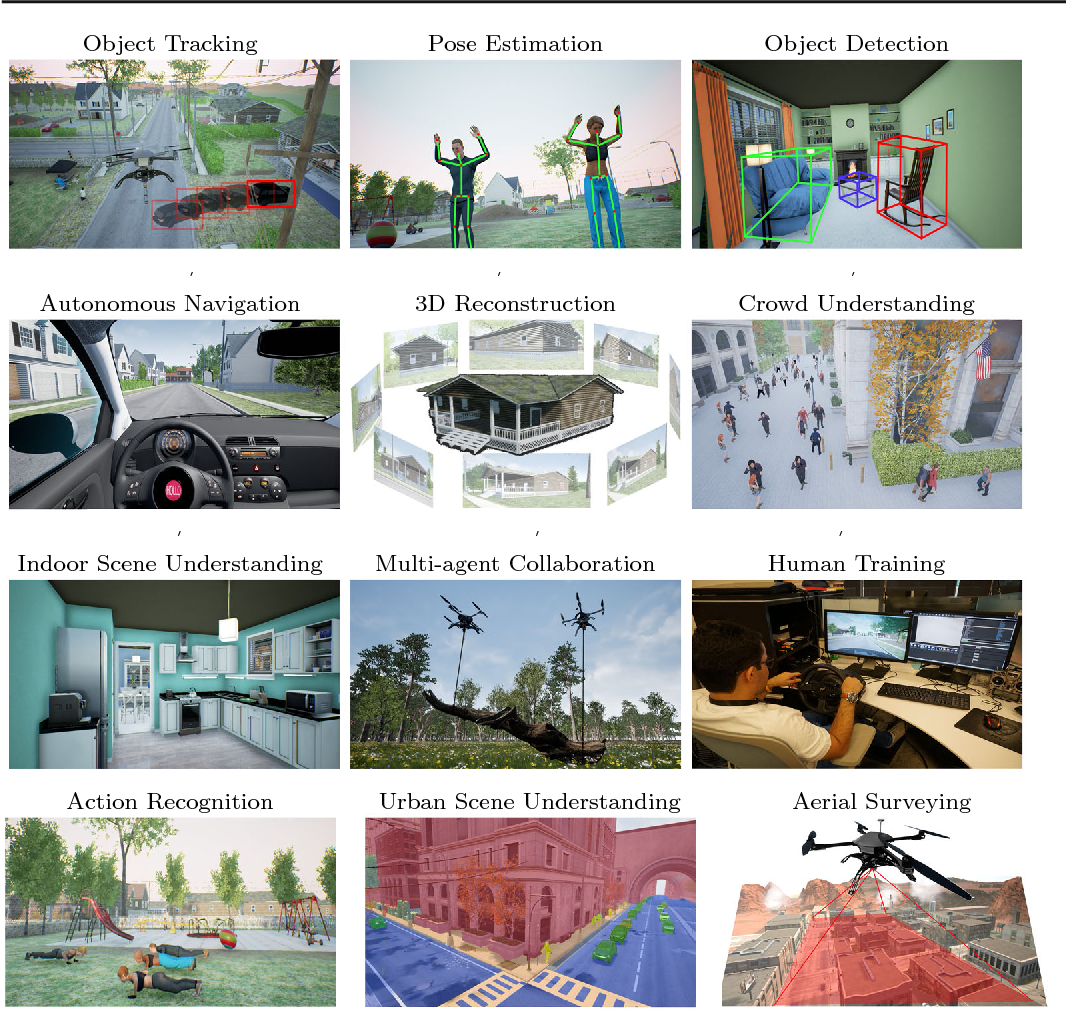
* **Focus on the process of producing images**
* Sometimes deals **with processing and analysis of images**
* **Medical imaging** includes **work on the analysis of image data in medical applications**

1. **Pattern Recognition Vs. Computer Vision**

* Field which uses various methods to **extract information from signals in general**
* Mainly **based on statistical approaches** and **artificial neural networks**
* Significant part of this field is **devoted to applying these methods to image data**

**Computer Vision Applications**

Computer vision is a sector of artificial intelligence which uses machine and deep learning to allow computers to “see” and analyze their surroundings. Computer vision has a massive impact on companies of all industries, from retail to agriculture. It is especially useful for problems where we would need a human’s eye to view the situation. Because of the broad amount of problems that exist in that criteria, thousands of applications (figure 3) of computer vision have not been discovered or exhausted yet.



***Figure 3: some applications of computer vision***

1. **Learning 3D Shapes**

* **Challenging task**
* Recent advances in **deep learning** has **enabled researchers to build models that** are **able to generate and reconstruct 3D shapes** from **single or multi-view depth** maps seamlessly and efficiently.
* **Automatic inspection**, e.g., in manufacturing applications;
* **Assisting humans in identification tasks**, e.g., a species identification system;
* **Controlling processes**, e.g., an industrial robot;
* **Detecting events**, e.g., for visual surveillance or people counting;
* **Interaction**, e.g., as the input to a device for computer-human interaction;
* **Modelling objects or environments**, e.g., medical image analysis or topographical modelling;
* **Navigation**, e.g., by an autonomous vehicle or mobile robot; and
* **Organizing information**, e.g., for indexing databases of images and image sequences.

1. **Medical Computer Vision**

* Characterized by the **extraction of information from image data to diagnose a patient**
* **Detection of tumours**
* **Arteriosclerosis or other malign changes**
* **Measurements of organ dimensions**
* **Blood flow, etc.**

1. **Industry**

* Information is extracted for the purpose of **supporting a manufacturing process**
* **Quality Control** where details or final products are being **automatically inspected in order to find defects**
* **Measurement of position and orientation** of details to be picked up by **a robot** arm
* **Heavily used in the agricultural process**.

1. **Other Application Areas**

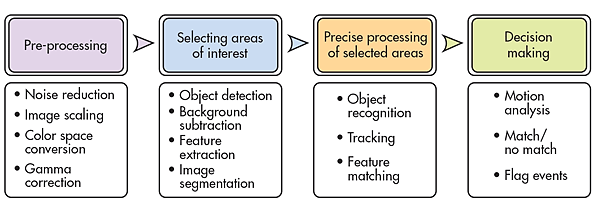
* Military
* Autonomous Vehicles
* Visual Effects creation for cinema and broadcast
* Surveillance
* Tracking and counting organisms in the biological sciences

**Some other general applications of computer vision:**

* Automatic face recognition, and interpretation of expression.
* visual guidance of autonomous vehicles.
* automated medical image analysis, interpretation, and diagnosis.
* robotic manufacturing: manipulation, grading, and assembly of parts.
* OCR: recognition of printed or handwritten characters and words.
* agricultural robots: visual grading and harvesting of produce.
* smart offices: tracking of persons and objects; understanding gestures.
* biometric-based visual identification of persons.
* visually endowed robotic helpers.
* security monitoring and alerting; detection of anomaly.
* Machine inspection: inspection for quality assurance (looking for defects).
* intelligent interpretive prostheses for the blind.
* tracking of moving objects; collision avoidance; stereoscopic depth.
* object-based (model-based) compression of video streams.
* General scene understanding.
* Improved image and video searching

**General Computer Vision Processing Pipeline**

Computer vision pipeline – the series of steps that a computer vision application goes through for acquiring, processing, and performing an action on images. A typical computer vision pipeline includes **image acquisition** using image sensors; **pre-processing** to enhance the image such as reducing noise; **feature extraction** that would reveal lines, edges, shapes, textures or motion; **image segmentation** to identify areas or objects of interest; **high-level processing** (also called post-processing) as relevant to the application; and finally, **decision making** such as classifying a medical scan as true or false for tumor (figure 4).



***Figure 4: Typical computer vision data Pipeline***

**Typical functions which are found in many Computer Vision systems:**

Image Acquisition, Pre-Processing, Feature Extraction, Detection- Segmentation, High-Level Processing Decision Making.

1. **Image Acquisition**

* Digital image is produced by **one or several image sensors**, besides various **types of light-sensitive cameras**, include **range sensors, tomography devices,** radar, ultrasonic cameras, etc
* **Depending on the type of sensor**, the resulting image data is an ordinary 2D image, a 3D volume, or an image sequence
* **The pixel values** correspond to light intensity in one or **several spectral bands** (gray images or color images)

1. **Pre-Processing**

**Before a computer vision method can be applied to image data in order to extract some specific piece of information**, it is necessary to process the data in order to assure that it satisfies certain assumptions implied by the method.

1. **Pre-Processing Examples**

* **Re-sampling** to assure that the image coordinate system is correct
* **Noise reduction** in order to assure that sensor noise does not introduce false information
* **Contrast enhancement** to assure that relevant information can be detected
* **Scale space representation** to enhance image structures at locally appropriate scales.

1. **Feature Extraction:**

* Image features at various levels of **complexity are extracted from the image data**
* **Typical examples of such features are**
  + Lines, edges and ridges
* Localized interest points such as **corner or points**
* More complex features **may be related to texture, shape or motion**

1. **Detection / Segmentation**

A decision can be made about **which image points or regions of the image are relevant for further processing**

**Detection / Segmentation Examples**

* **Segmentation of one or multiple image regions** which contain a specific object of interest
* **Segmentation of image into nested scene architecture comprised foreground, object groups**, **single objects or salient object parts**.
* **Segmentation or co-segmentation of one or multiple videos into a series of per-frame foreground masks**, while maintaining its temporal semantic continuity.

1. **High-Level Processing**

* **At this step the input is a small set of data**, for example a set of points or an image region which is assumed to contain a specific object.
* **The remaining processing deals with, for example:**
  + **Verification that the data satisfy model-based** and application specific assumptions
  + **Estimation of application specific parameters**, such as object pose or object size
  + **Image recognition** (classifying a detected object into different categories)
  + **Image registration** ( comparing and combining two different views of the same object).

1. **Decision Making**

* Making the final decision required for the application, **for example:**
  + Pass/fail on automatic inspection applications
  + Match / no-match in recognition applications

**In order to produce numerical or symbolic information**, e.g., in the forms of decisions

1. **Recognition**

Determine whether or not the image data contains **some specific object, feature, or activity**

* + **Object Recognition**
  + **Identification**
  + **Detection**

**Object Recognition (Object Classification)**

* One or several pre-specified or **learned objects can be recognized**
* Usually together **with their 2D positions in the image or 3D poses in the scene.**

**Identification**

* **Individual instance of an object is recognized**
* Identification of a specific person’s face or fingerprint
* Identification of **handwritten digits**
* Identification of a **specific vehicle**

**Detection**

* **Image data are scanned for a specific condition**
* **Detection of possible abnormal cells or tissues** in medical images or detection of **a vehicle in an automatic road toll system**
* Detection based on relatively simple and fast computations is **used for finding smaller regions of interesting image data** which can be **further analyzed by more computationally** demanding techniques to produce a correct interpretation

**Specialized Tasks based on Recognition:** Several specialized tasks based on recognition such as:

1. **Content-based Image Retrieval**

* Finding **all images in a larger set of images which have a specific content**
* Content **can be specified in different ways**
  + **In terms of similarity relative a target image**
* give me all images similar to image X
  + **In terms of high-level search criteria given as text input**
* give me all images which contains many houses
* are taken during winter
* and have no cars in them

1. **Pose Estimation:**

* Estimating **the position or orientation of a specific object relative to the** camera
* Assisting **a robot arm in retrieving objects.**

1. **Optical Character Recognition (OCR)**

* **Identifying characters** in images of **printed or handwritten text**
* **Usually with a view to encoding the text in a format more amenable to editing or indexing**.

1. **2D Code Reading**

* **Reading of 2D codes** such as data matrix and **QR codes**

1. **Facial Recognition**

* **Technology capable of identifying or verifying a person** from a digital image or a video frame from a video source

1. **Shape Recognition**

* **Shape Recognition Technology (SRT)** in people counter systems **differentiating human beings (head and shoulder patterns) from objects**

1. **Motion Analysis**

Image sequence is **processed to produce an estimate of the velocity either at each point** in **the image or in the 3D scene**, **or even of the camera that** produces the images.

**Examples of such tasks are:**

* **Egomotion** 🡺 determining the **3D rigid motion (rotation and translation) of the camera** from an image sequence produced by the camera.
* **Tracking** 🡺 following the **movements of a smaller set of interest points or objects (e.g., vehicles, humans or other organisms) in the image sequence.**
* **Optical flow** 🡺 to **determine, for each point in the image, how that point is moving relative to the image plane**, i.e., its apparent motion.

1. **Scene Reconstruction**

* Given one or more images of a scene, or a video, scene reconstruction aims at **computing a 3D model of the scene**
* In the simplest case the **model can be a set of 3D points**
* More sophisticated methods **produce a complete 3D surface model**
* **Grid-based 3D sensing** can be used to acquire 3D images from **multiple angles**

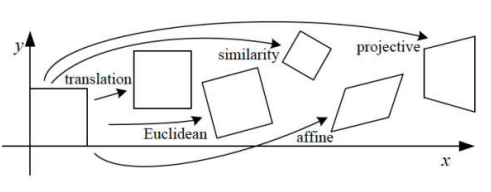
1. **Image Restoration:**

* The aim is the **removal of noise (sensor noise, motion blur, etc.) from images**
* The simplest possible approach for noise removal is 🡺 **types of filters such as low-pass filters or median filters**
* More sophisticated methods **assume a model of how the local image structures look like,** a model which distinguishes them from the noise
* By first analysing the image data in terms of the local image structures, such as lines or edges, then controlling the filtering based on local information from the analysis step

**Some General Computer Vision Tasks**

* **Image Point matching problem**

Suppose I have two images related by some transformation. Or have two images of the same object in different positions.

****

How to find the **transformation** of image 1 that would **align** it with image 2 ?

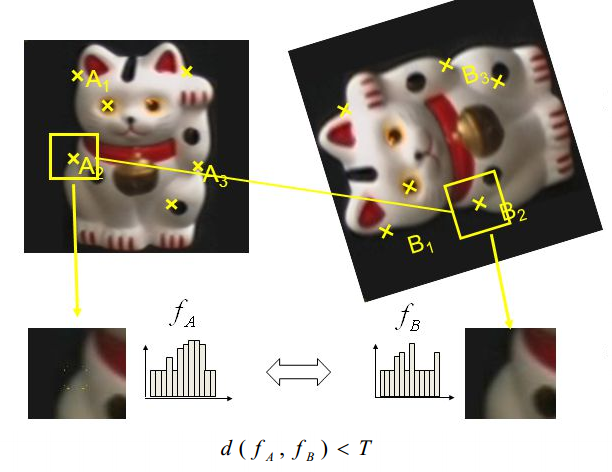
* We want Local Features
* **Features**: Also known as **interesting points**, **salient points** or **keypoints**.

**Goal: Find points in an image that can be:**

**► Found in other images  
► Found precisely – well localized  
► Found reliably – well matched**

* A local feature is an image **pattern which differs from its immediate  
  neighborhood**. e.g. Point where the direction of the boundary of the object changes abruptly.
* **Local features** can be points, but also edges or small image patches.
* Typically, some measurements are taken from a region centered on a  
  local feature and converted into **descriptor**

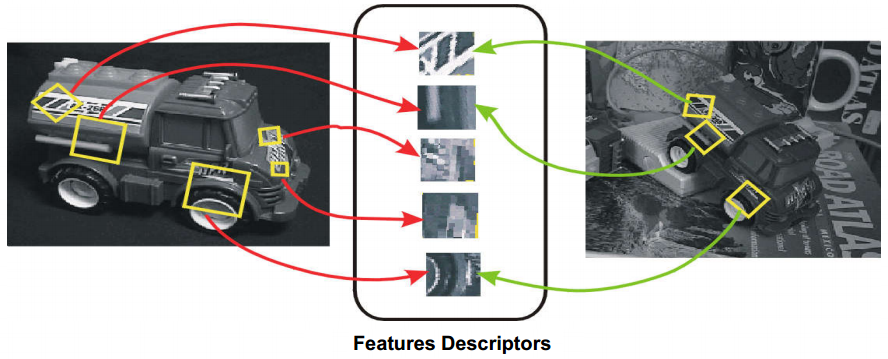
**Overview of Keypoint Matching**

****

1. **Find a set of distinctive keypoints**
2. **Define a region around each keypoint**
3. **Extract and normalize the region content**
4. **Compute a local descriptor from the normalized region**
5. **Match local descriptors**

**Invariant Local Features**

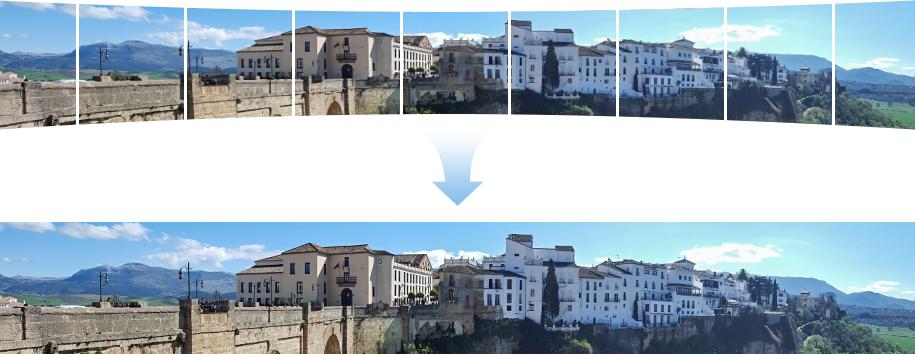
Image content is transformed into local feature coordinates that are invariant to translation, rotation, scale, and other imaging parameters

****

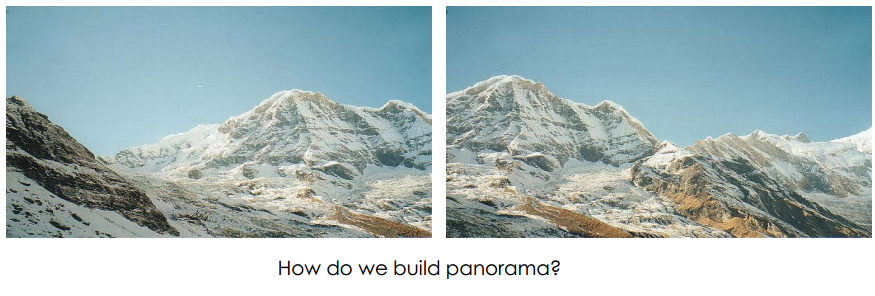
**Main challenges**

**► Change in position and scale ► Change in viewpoint  
► Occlusion ► Articulation**

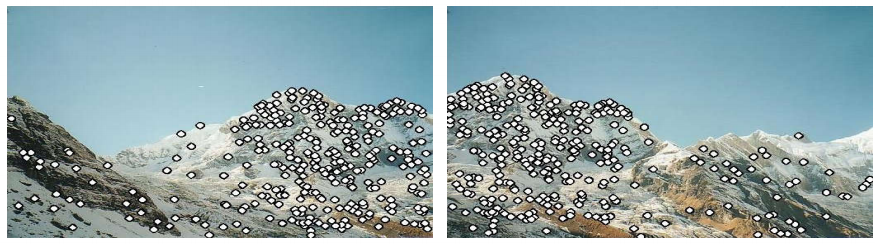
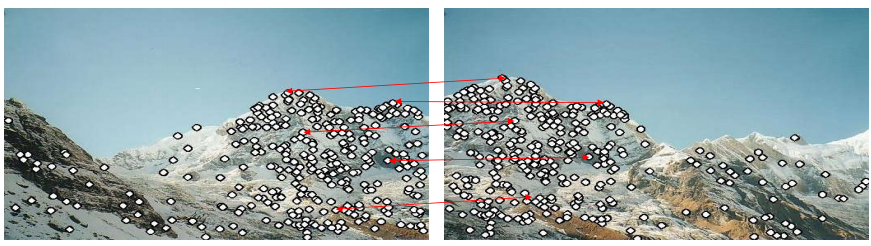
**Application: Image Matching**



**Image Stitching:** We need to match (align) images

****

**Matching with Features: Procedure**

1. **Detect feature points in both images**
2. **Find corresponding pairs**
3. **Use these pairs to align images**

****

**Characteristics of good features**

* Region extraction needs to be **repeatable** and **accurate**
  + The same feature can be found in several images despite geometric and photometric transformations
* **Locality**: Features are local, therefore robust to occlusion and clutter.
* **Saliency**: Each feature has a distinctive description.
* **Compactness** and **efficiency**: Many fewer features than image pixels.
* **Fitting and Alignment problem**

**Fitting:** find the parameters of a model that best fit the data.

**Alignment:** find the parameters of the transformation that best align

matched points.

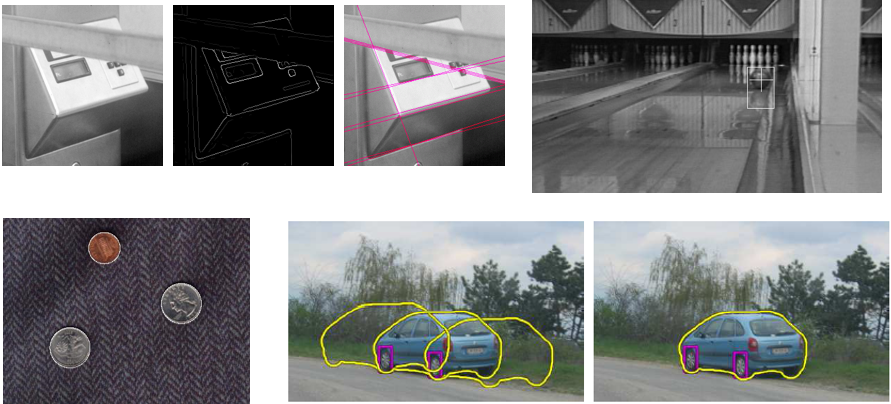
**Parametric Model: Hough Transform**

A parametric model can represent a class of instances where each is  
defined by **a value of the parameters**.

**Examples** of parametric model include:

* lines, or circles, or even a parameterized template.

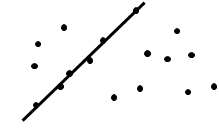
**Want to associate a model with observed features**

****

**Example: Line Fitting**

**Why fit lines?** Many objects characterized by presence of straight lines

****

**Difficulty of line fitting**

* **Extra edge points (clutter), multiple models:**
  + - Which points go with which line?
* **Only some parts of each line detected, and some parts are missing:**
  + - How to find a line that bridges missing evidence?
* **Noise in measured edge points, orientations:**
  + - How to detect true underlying parameters?

**Fitting Lines**

* Given points that belong to a line, what is the line?
* How many lines are there?
* Which points belong to which lines?

**Hough Transform**

Is a voting technique that can be used to answer all of these

**Main idea:**

1. Each edge point votes for compatible lines.  
2. Look for lines that get many votes

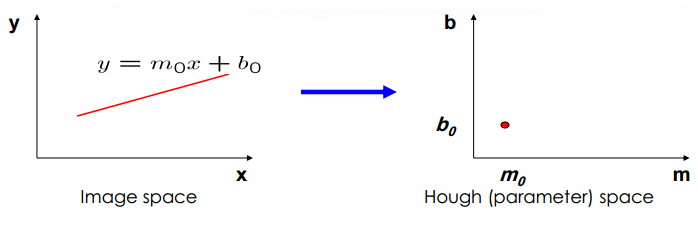
**Voting**

* **Voting is a general technique where we let the features vote for all models that are compatible with it.**

1. Cycle through features, cast votes for model parameters.
2. Look for model parameters that receive a lot of votes**.**

* **Noise & clutter features will cast votes too, *but* typically their votes should be inconsistent with the majority of “good” features.**
* The Hough transform (HT) can be used to detect lines circles or , circles or other parametric curves (known parametric equation)

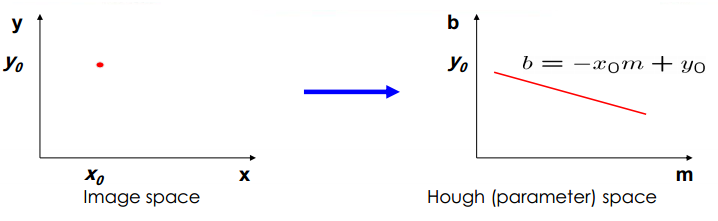
**Hough Space**

****

**Connection between image (x,y) and Hough (m,b) spaces**

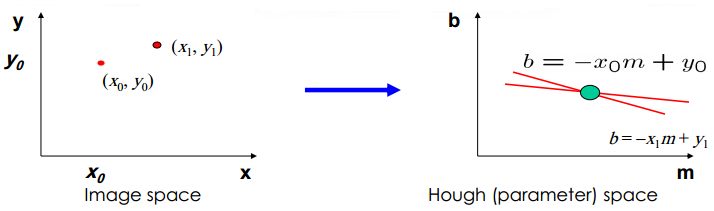
* A line in the image corresponds to a point in Hough space
* To go from image space to Hough space:
  + given a set of points (x,y), find all (m,b) such that y = mx + b

**Finding Lines in an Image: Hough Space**

****

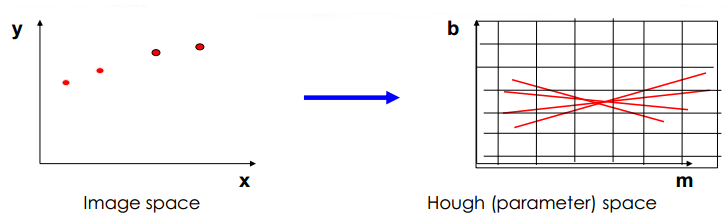
**What does a point (x0, y0) in the image space map to?**

* + - **Answer: the solutions of b = -x0 m + y0**
    - **this is a line in Hough space**

****

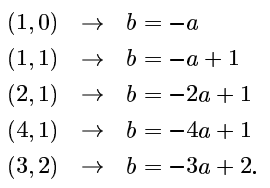
**What are the line parameters for the line that contains both (x0, y0) and (x1, y1)?**

It is the intersection of the lines b = –x0m + y0 and b = –x1m + y1

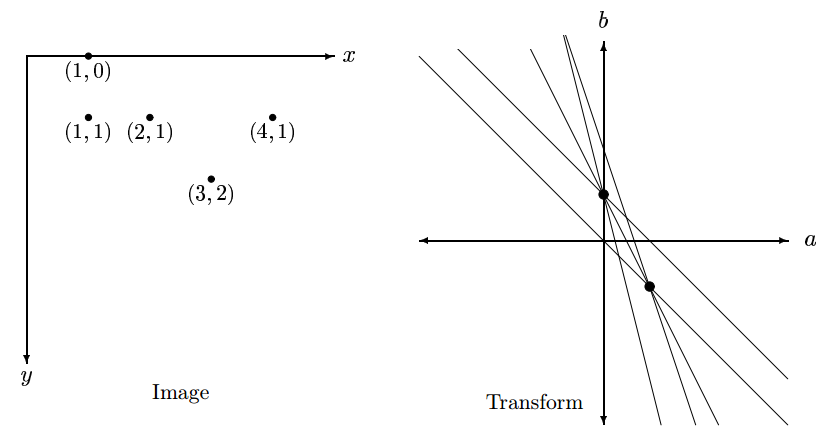
****

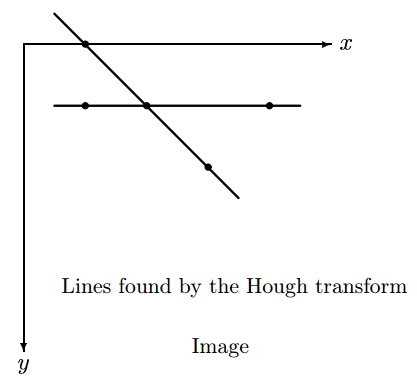
**How can we use this to find the most likely parameters (m,b) for the most prominent line in the image space?**

* Let each edge point in image space *vote* for a set of possible parameters in Hough space
* Accumulate votes in a discrete set of bins; parameters with the most votes indicate line in image space.

**Example:**

suppose we consider an image with five points (1,0) , (1,1) (,2,1) ,(4,1) and (3,2) Each of these points corresponds to a line as follow



**The coordinates of these dots are:** 

**(a,b) = (0,1) and**

**(a,b) = (1,-1)**

**These values correspond to the lines:**

**Y = 1 and**

**y = 1.X + (-1)**

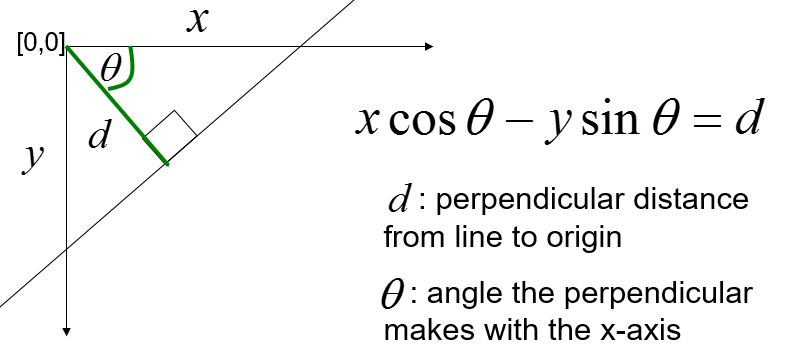
**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Polar Representation for Lines**

**Issues** with usual (*m,b*) parameter space:

**Can take on infinite values, undefined for vertical lines.**

**In polar representation,** any line can be described in terms of **d , θ**

****

**Point in image space 🡪 sinusoid segment in Hough space.**