Hasso Plattner Institut
Digital Engineering · Universität Potsdam


# Medical Image Analysis

Borchert, Dr. Schapranow

Data Management for Digital Health
Winter 2023

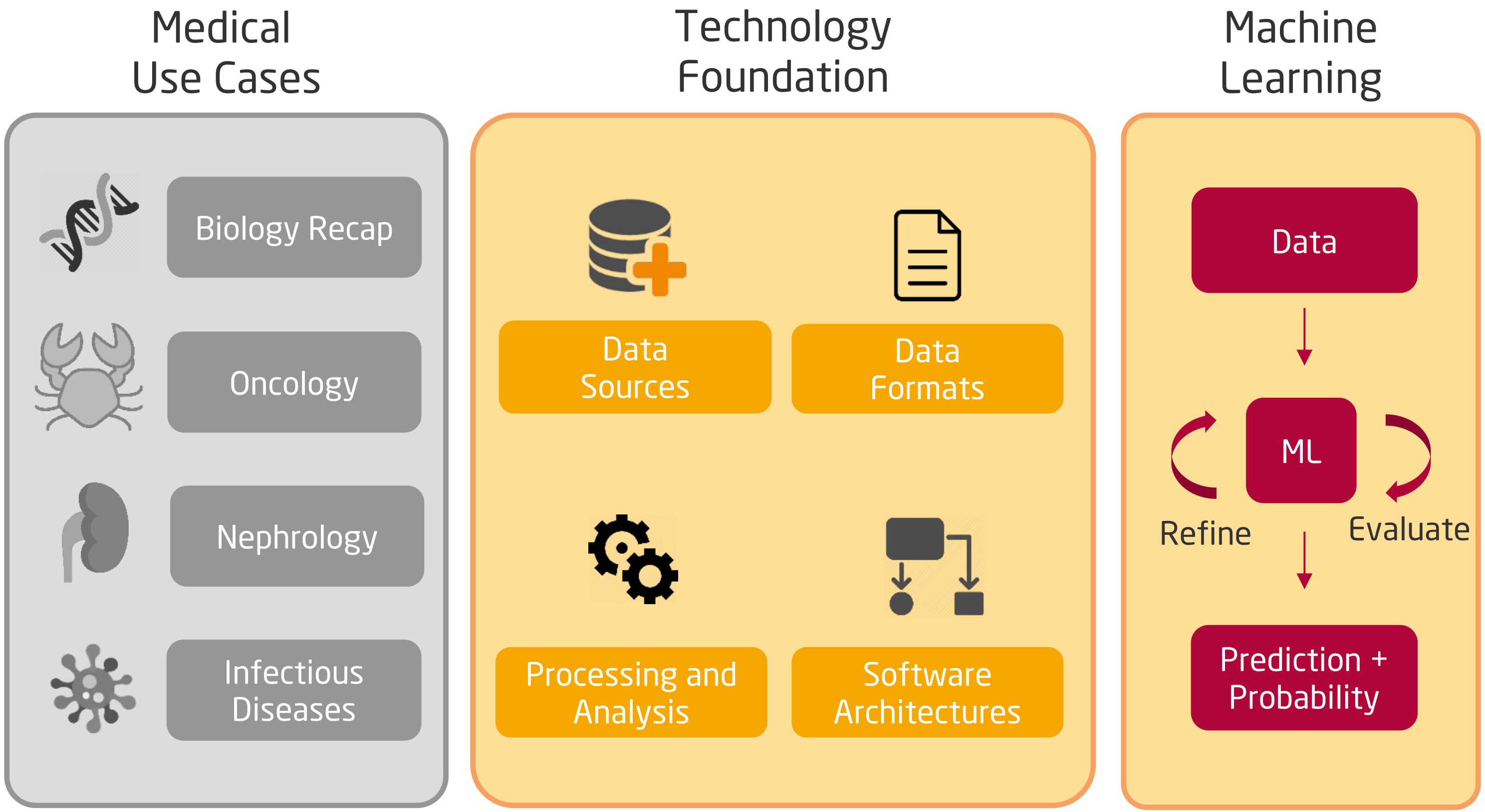
# Agenda
## Pillars of the Lecture

### Medical Use Cases
- Biology Recap
- Oncology
- Nephrology
- Infectious Diseases

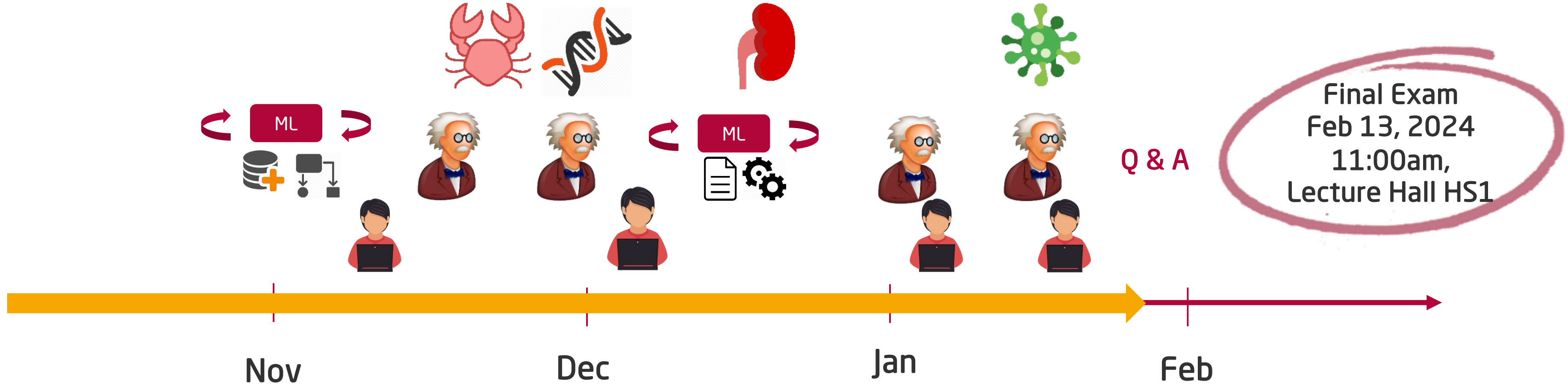
### Technology Foundation
- Data Sources
- Data Formats
- Processing and Analysis
- Software Architectures

### Machine Learning
- Data
- ML (Refine, Evaluate)
- Prediction + Probability

# Agenda
## Pillars of the Lecture



# Lecture Schedule




- Lecture Kickoff
- Actors in Healthcare
- Digital Health Data

- Machine Learning (ML) Foundations
- Use Case Oncology
- Biology Recap

- Natural Language Processing
- Use Case Nephrology & Intensive Care
- Supervised ML & Deep Learning

- Use Case Infectious Diseases
- Unsupervised ML

**Medical Image Analysis**

# Agenda

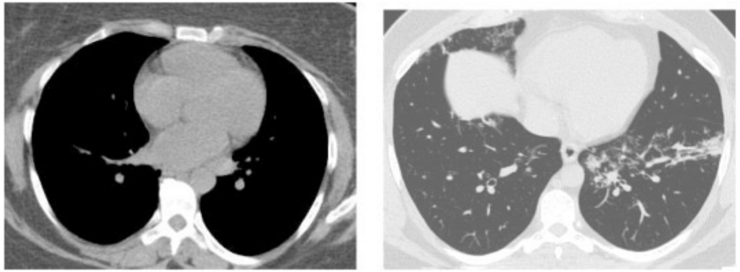
Hasso Plattner Institut

- Imaging Modalities
- Data Formats and Systems
- Computer Vision
- Convolutional Neural Networks

# Medical Imaging

Hasso Plattner Institut

- Applications:
  - Detection and diagnosis of diseases, often in early stages
  - Treatment and surgery planning
  - Monitoring progress over time
  - Research, e.g., aiding in the understanding of diseases and the development of new treatments.



ELSEVIER

Informatics in Medicine Unlocked

Volume 20, 2020, 100427

COVID-19 detection in CT images with deep learning: A voting-based scheme and cross-datasets analysis

Pedro Silva [a], Eduardo Luz [a], Guilherme Silva [b], Gladston Moreira [a], Rodrigo Silva [a], Diego Lucio [c], David Menotti [c]

# Diagnostic Radiology

- Diagnostic radiologists use medical images such as X-rays, ultrasound, CT scans and MRI scans to diagnose diseases anywhere in the body
  - Neuroradiology
  - Pediatric radiology
  - Breast imaging
  - Cardiovascular radiology
  - Gastrointestinal radiology
  - Genitourinary radiology
  - Musculoskeletal radiology
  - Emergency radiology
  - Nuclear radiology

# Interventional Radiology

- A subspecialty of radiology that focuses on the diagnosis and treatment of patients utilizing minimally invasive interventional techniques (non-surgical procedures)
  - Imaging & treatment of blood vessels (angiography)
  - Biopsy procedures
  - Cardiac catheterization
  - Angioplasty (balloon dilation of blood vessels)
  - Stents
  - Laser treatment of varicose veins
  - Fluid abscess drainage

https://www.hiclipart.com/search?clipart=stent

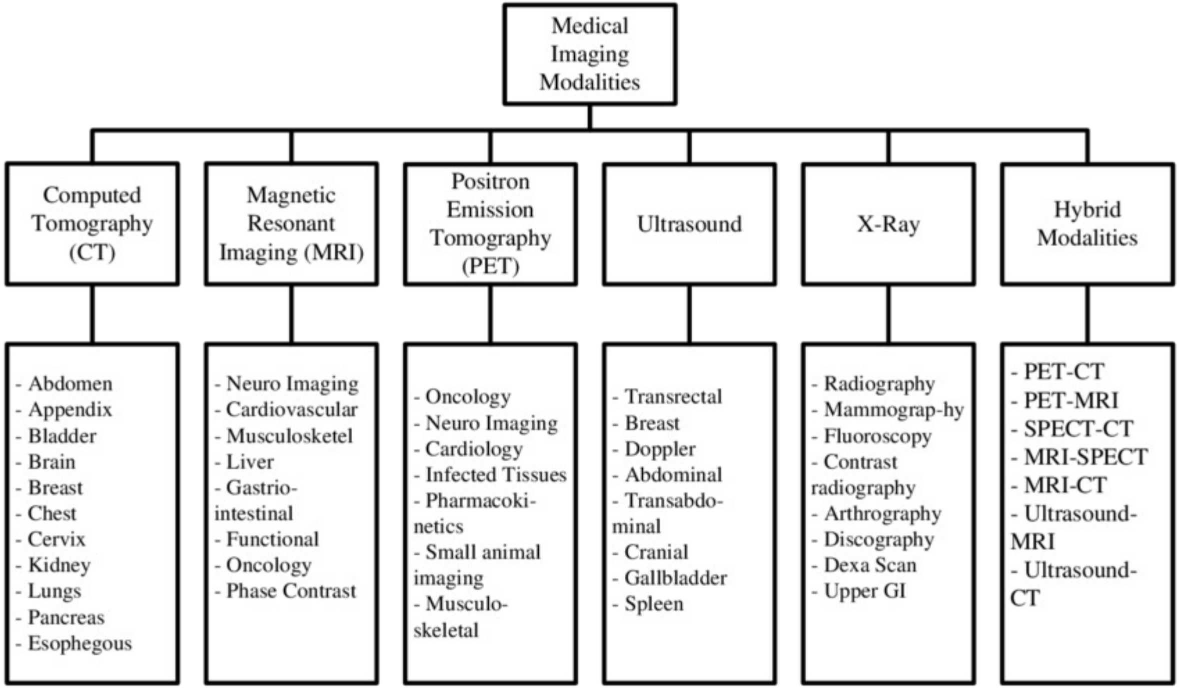
# Medical Imaging Modalities

Zahl der Untersuchungen bei vollstationären Patienten (DRG-Krankenhäuser)

| Diagnostik | 2005 | 2010 | 2011 | 2012 | 2013 | 2016 | 2018[6] |
|---|---|---|---|---|---|---|---|
| Bildgebende Diagnostik | 5.073.309 | 8.417.123 | 9.125.033 | 9.728.437 | 10.255.233 | 12.324.956 | 13.216.070 |
| Computertomografie (CT) | 2.972.307 | 4.183.728 | 4.450.125 | 4.709.286 | 4.957.593 | 5.825.017 | 6.296.363 |
| Magnetresonanztomografie (MRT) | 1.008.944 | 1.518.625 | 1.622.007 | 1.696.235 | 1.767.005 | 2.012.067 | 2.028.008 |

https://www.destatis.de/DE/Themen/Gesellschaft-Umwelt/Gesundheit/Krankenhaeuser/Publikationen/Downloads-Krankenhaeuser/operationen-prozeduren-5231401187014.pdf?__blob=publicationFile

# Medical Imaging Modalities



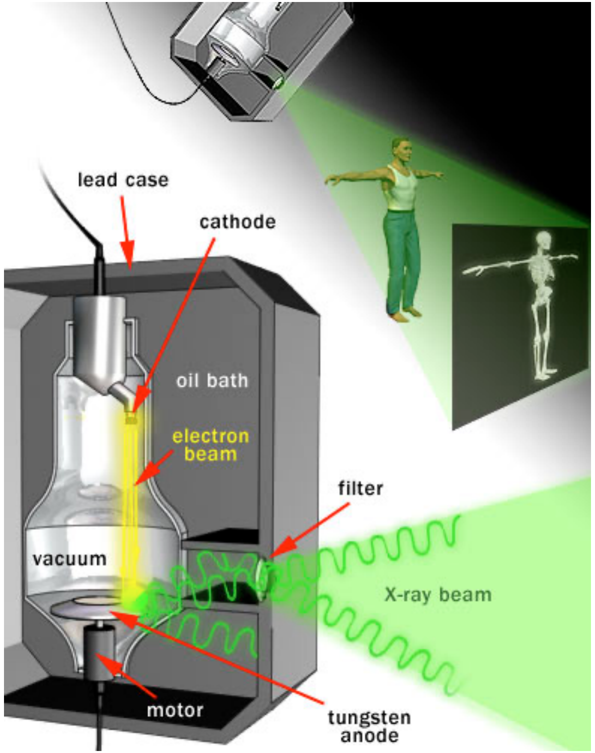

https://www.researchgate.net/figure/Typology-of-Medical-Imaging-Modalities_fig1_319535615

# Major Modalities

- Projection X-ray (Radiography)
- X-ray Computed Tomography (CT)
- Nuclear Medicine (SPECT, PET)
- Magnetic Resonance Imaging (MRI)
- Ultrasound (US)

https://www.hiclipart.com/search?clipart=medical+imaging

# X-ray Imaging

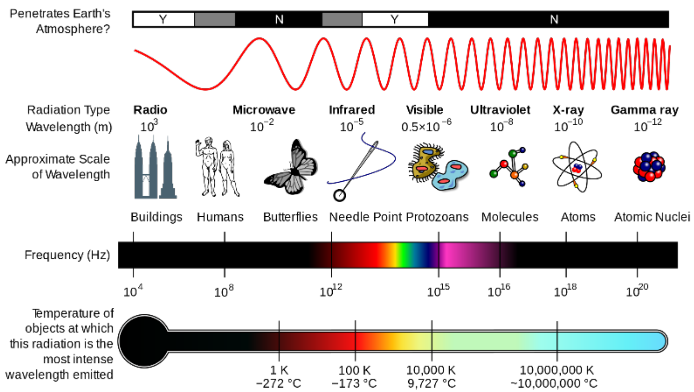


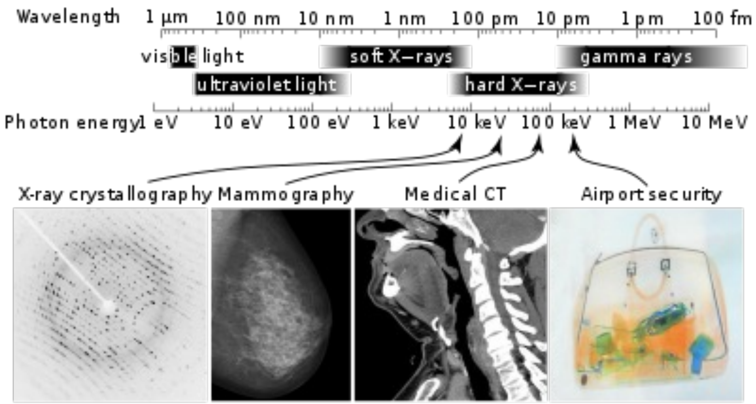

https://science.howstuffworks.com/x-ray.htm

# Projection X-ray Imaging



https://en.wikipedia.org/wiki/Electromagnetic_spectrum

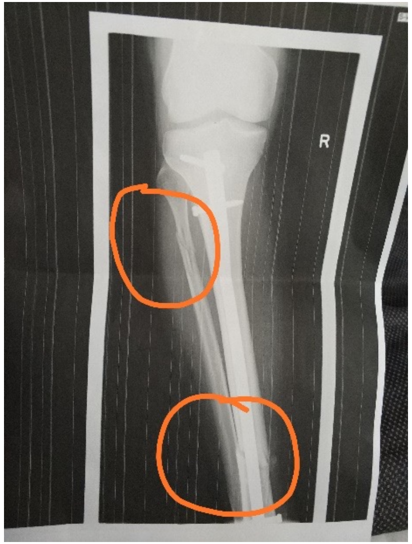



https://en.wikipedia.org/wiki/X-ray

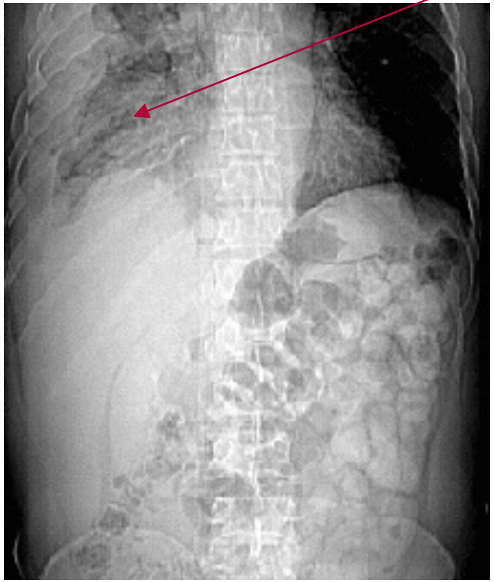
- X-ray imaging requires interactions of x-ray photons with object – work in a specific energy band
- Fast, cheap
- Very effective for solid objects, like bones
- Cons: no depth information, exposure to radiation, less effective for soft tissue

# X-ray Imaging

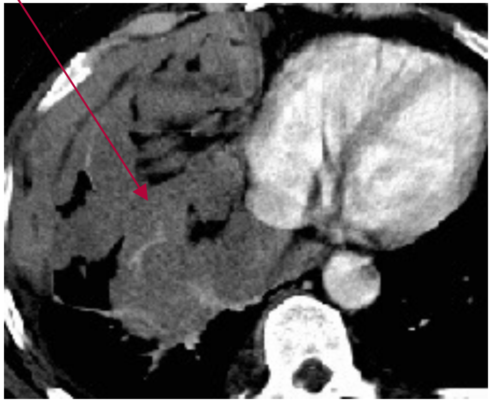


# X-ray Imaging
# Projection vs Tomographic

Chest Mass

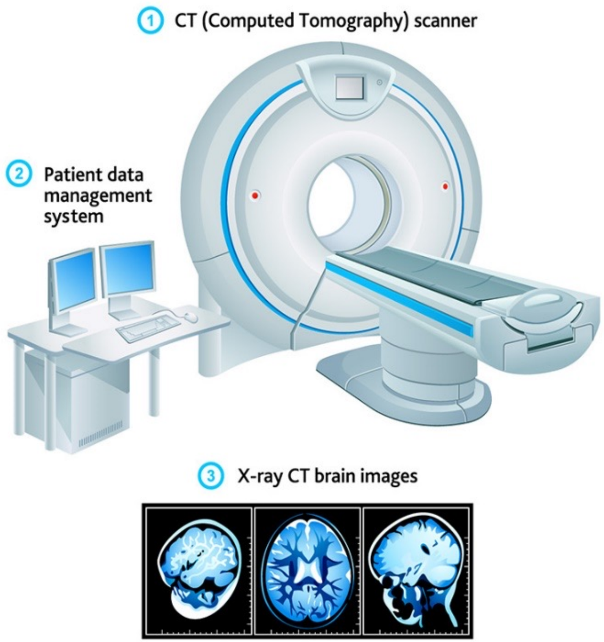


Projection Image



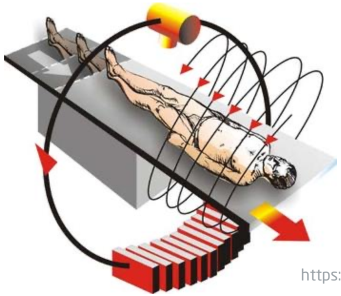
Cross-sectional Image

# Computed Tomography (CT)




https://greenimaging.net/pet-ct/

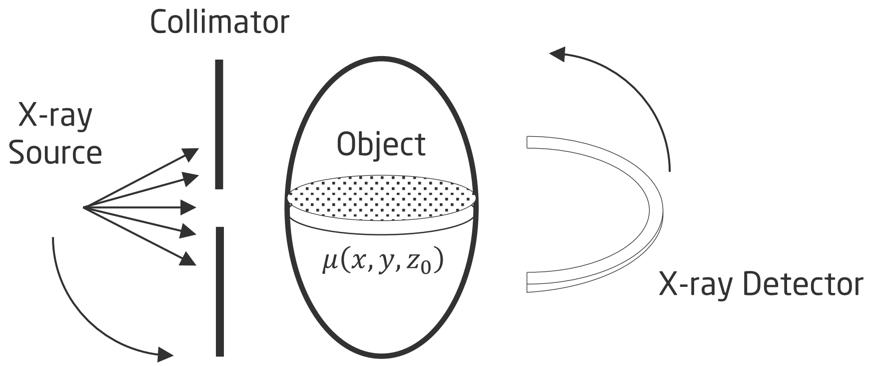
- An X-ray ring rotates within a circular opening as a motorized table moves the patient through the imaging system
- A fan-shaped X-ray beam scans the target body area one section at a time as the patient moves through
- The series of body images is processed and reconstructed by a computer



https://www.stepwards.com/?page_id=21633

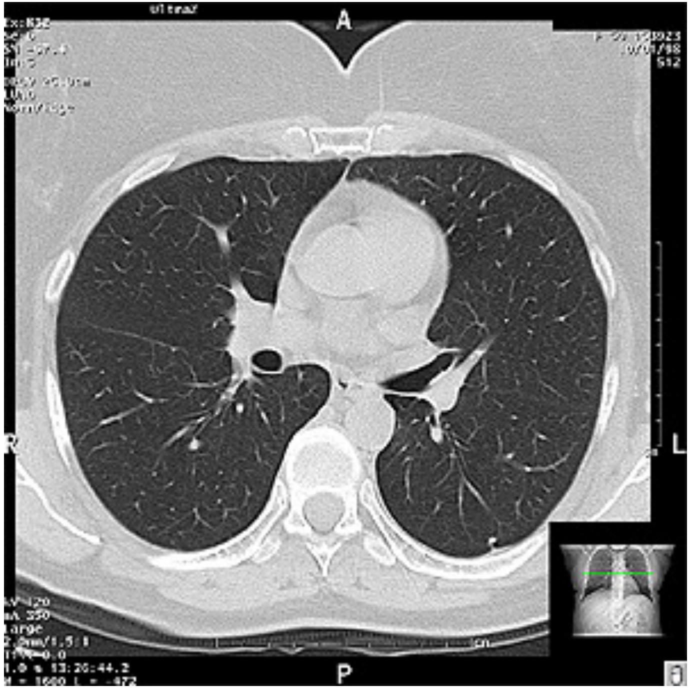
# X-ray Computed Tomography (CT)

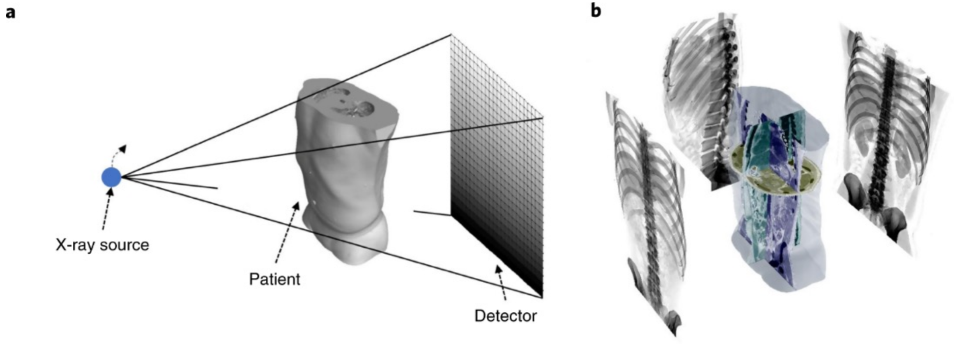
Hasso Plattner Institut




- Uses x-rays, but exposure is limited to a slice (or "a couple of" slices) by a collimator
- Source and detector rotate around object – projections from many angles
- The desired image, $I(x, y) = \mu(x, y, z_0)$, is computed from the projections
- Cons: exposure to radiation

# X-ray Computed Tomography (CT)

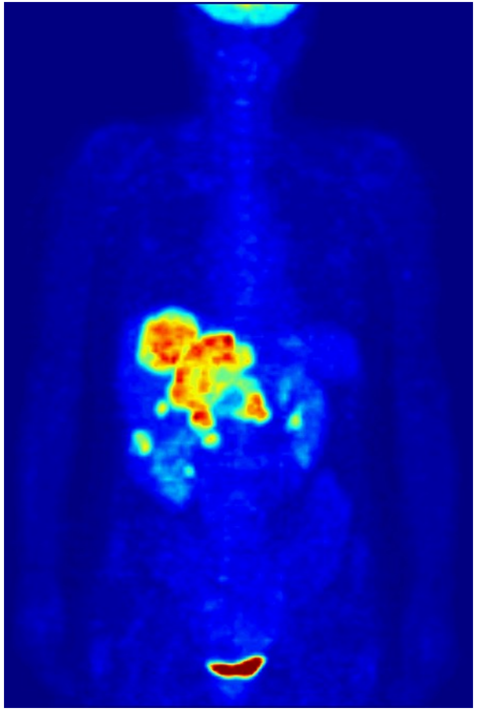



https://www.nature.com/articles/s41551-019-0466-4?proof=t

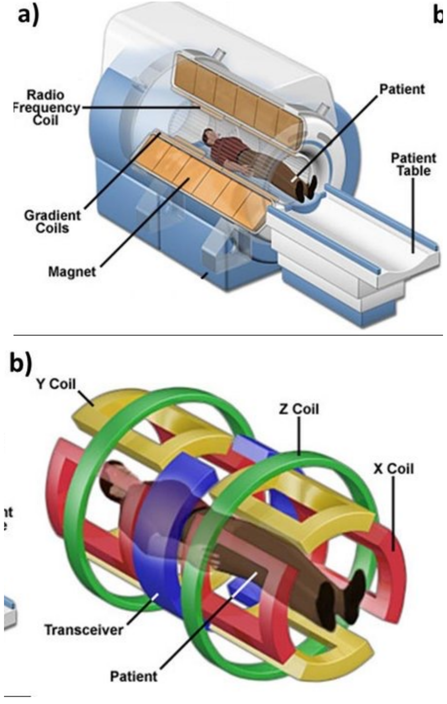
# Positron Emission Tomography (PET)

- Uses radiolabeled biologically active compounds
- Utilizes radioactive tracers to visualize and measure changes at the molecular and cellular level in living organisms
- Produces three-dimensional images that show the distribution of the tracer in the body
- Sensitive to detecting abnormalities in metabolic activity before structural changes become apparent on other imaging tests
- Cons: Exposure to ionizing radiation, limited image quality



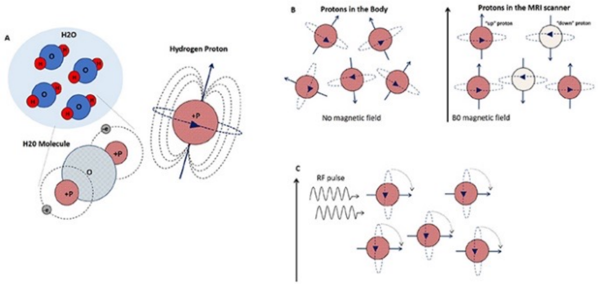
https://en.wikipedia.org/wiki/Positron_emission_tomography#/media/File:PET-MIPS-anim.gif

# Magnetic Resonance Imaging (MRI)




https://www.researchgate.net/figure/a-Schematic-illustration-of-the-MRI-system-where-the-main-components-are-indicated_fig6_318658798

- The magnetic field is used to align hydrogen protons in the body
- Radio frequency waves are absorbed by the protons and then emitted as a signal
- The computer processes the data, and an image is generated
- Cons: Slow, loud and relatively costly

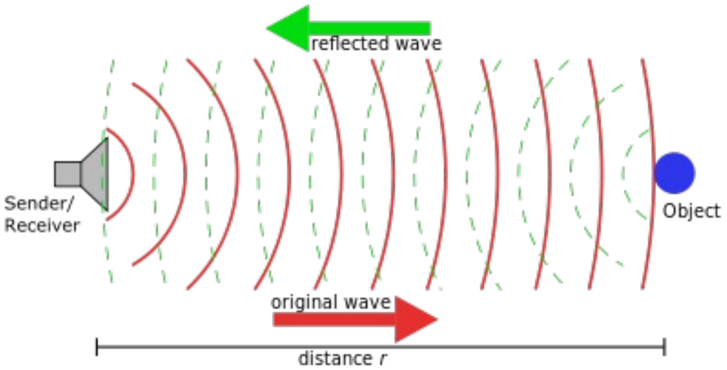


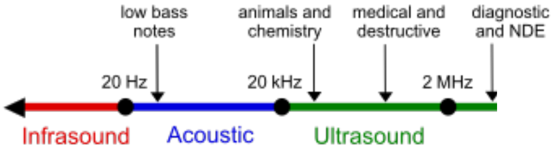

https://kids.frontiersin.org/article/10.3389/frym.2019.00023

# Ultrasound Imaging

- High-frequency sound waves to produce live images
- Direct imaging (e.g., vs. computed) modality – echo data is placed directly into image matrix
- Widely available, portable, and considered safe for all patients, including pregnant women and infants

- Cons: Limited penetration and image quality




https://en.wikipedia.org/wiki/Ultrasound

# Agenda

Hasso Plattner Institut

- Imaging Modalities
- **Data Formats and Systems**
- Computer Vision
- Convolutional Neural Networks

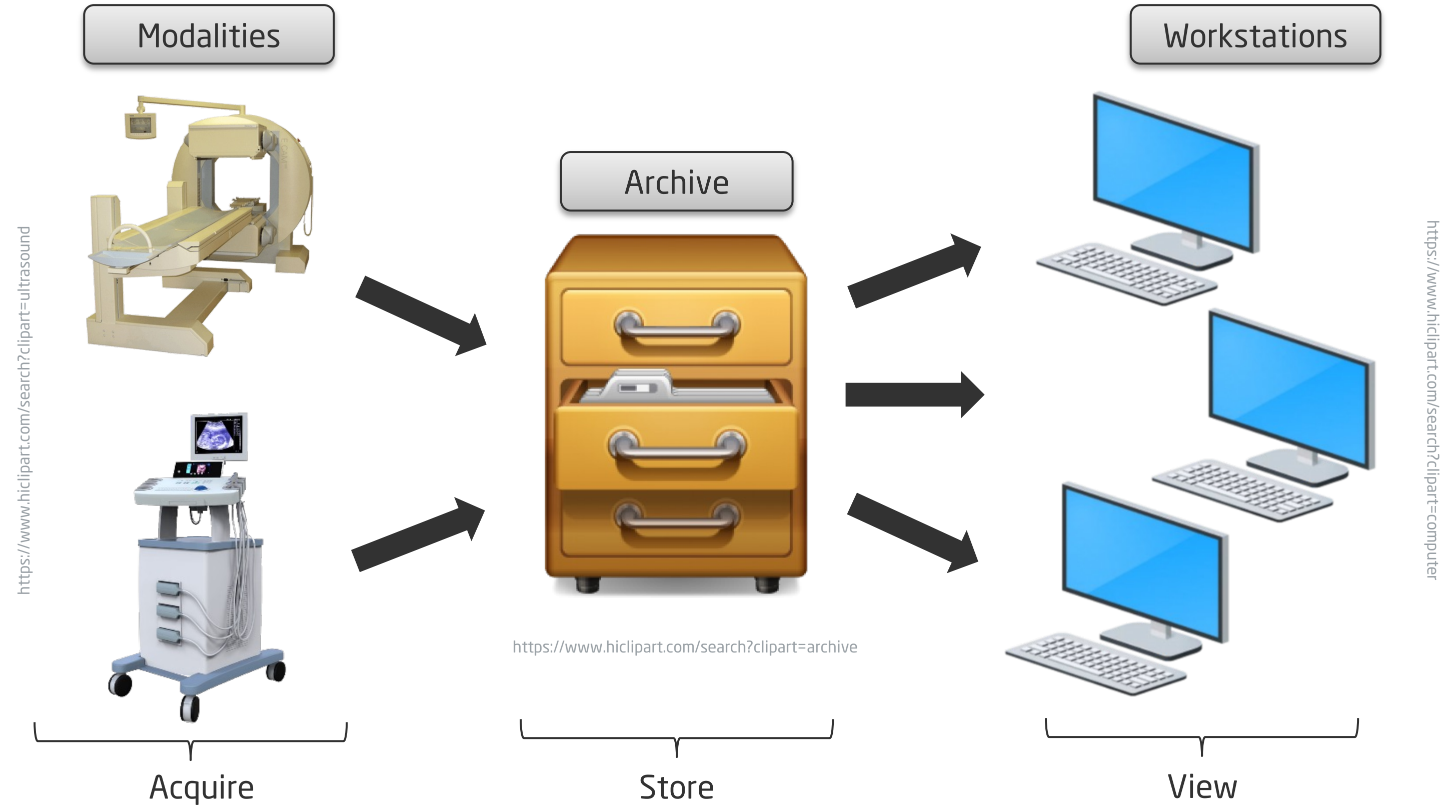
# Digital Imaging and Communications in Medicine (DICOM)

- Widely accepted standard for communication and management of medical imaging information and related data
- Ensures that data can be exchanged and understood across different systems and equipment: scanners, servers, workstations, printers, network hardware, etc. from different manufacturers
- Supports patient / clinical metadata in addition to imaging data

https://en.wikipedia.org/wiki/DICOM#Application_areas

# DICOM Overview

Hasso Plattner Institut



# DICOM Components

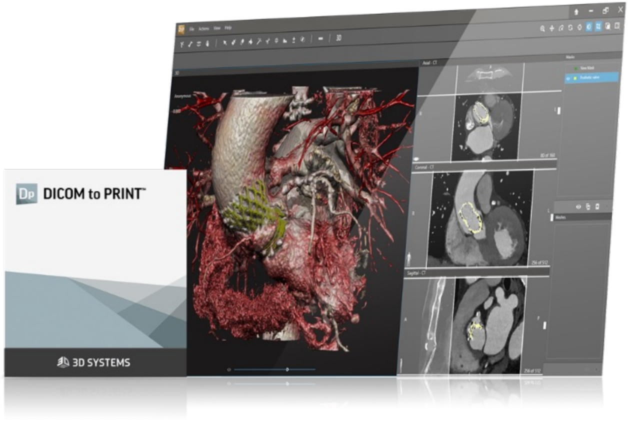
- Two main components
  - DICOM file format
  - DICOM network protocol
- Two elements work together so that images are in a standard format and the exchange of images is also standardized

https://www.hiclipart.com/search?clipart=dicom

# DICOM File Format



- Medical imaging equipment creates DICOM files
- Doctors use DICOM viewers, computer software applications that can display DICOM images, to diagnose the findings in the images
- DICOM files contain more than just images (metadata)
- Every DICOM file holds:
  - Patient information such as name, ID, sex and DOB (date of birth)
  - Important acquisition data such as type of equipment used and settings on the modality




https://www.hiclipart.com/search?clipart=dicom

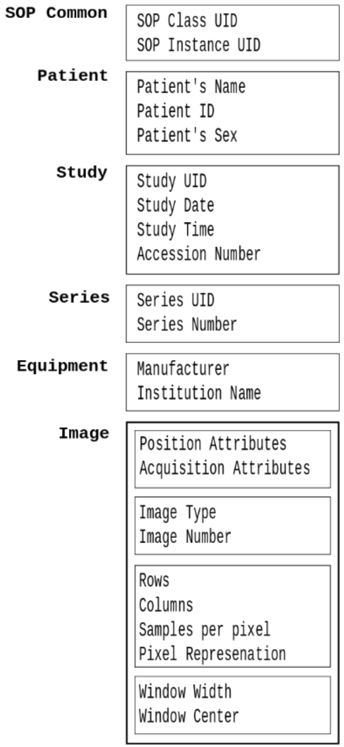
# DICOM Services

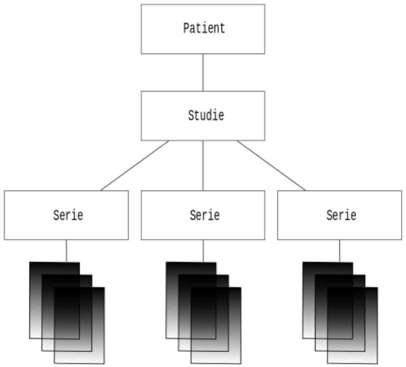
- DICOM objects are acted upon by DICOM services
- DICOM services perform functions like move, find and store DICOM objects
- When a DICOM service is paired with a DICOM object this is called a Service Object Pair or SOP

| DICOM Object | | DICOM Service | | SOP Class |
|---|---|---|---|---|
| CT image | + | Move | = | CT image move |

# DICOM
# Structures and Concepts

- Information model is organized into the levels: patient, study, series and instance
- Each instance of a DICOM object holds all the information needed to assign it to a specific series (for example, image series), study (a specific stay in the hospital or a single examination), and patient
- Information are grouped into data sets
- Data object consists of a number of attributes, including items such as name, ID, etc., and also one special attribute containing the image pixel data




https://en.wikipedia.org/wiki/DICOM#Application_areas

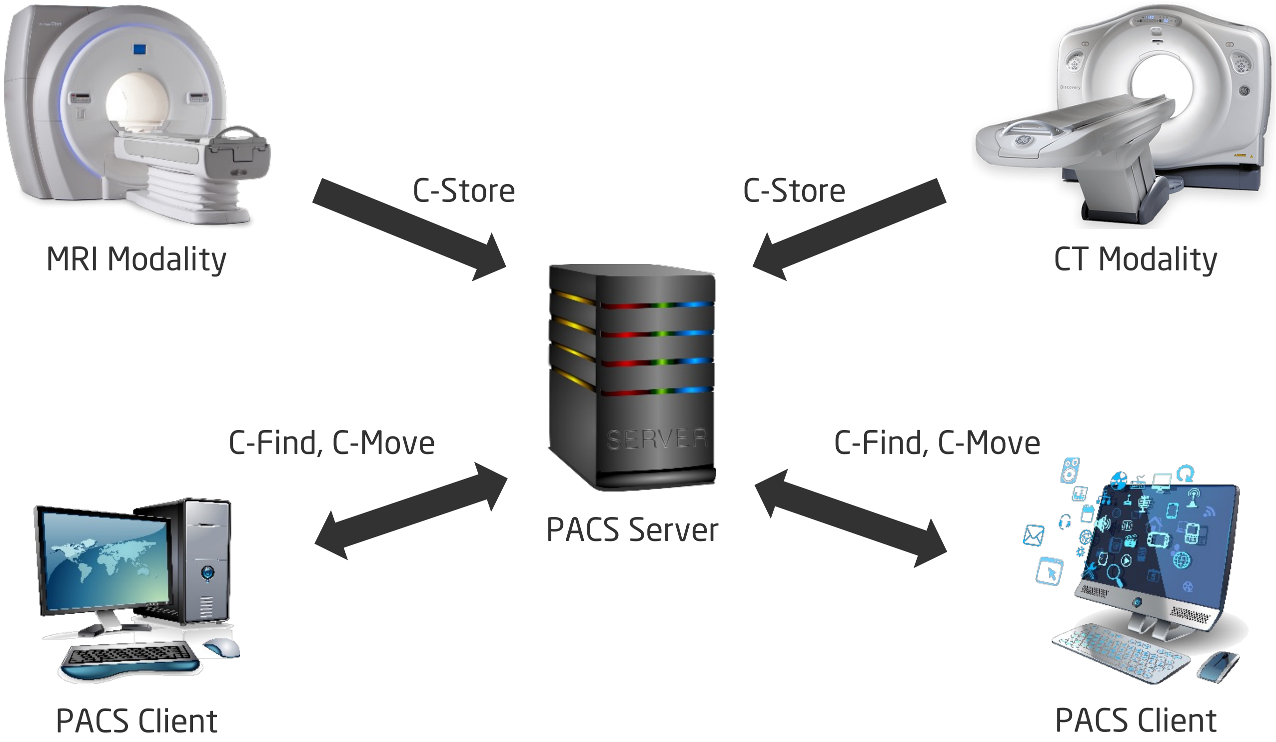
# DICOM
# Network Protocol

Hasso Plattner Institut

- When two DICOM devices want to connect there are standard protocols for how they make the connection
- Protocols to ensure that two communicating DICOM applications are compatible and transfer data in a well-defined format and order
- Uses TCP/IP as the underlying communication protocol

https://www.hiclipart.com/search?clipart=handshake

# DICOM
# Network Protocol



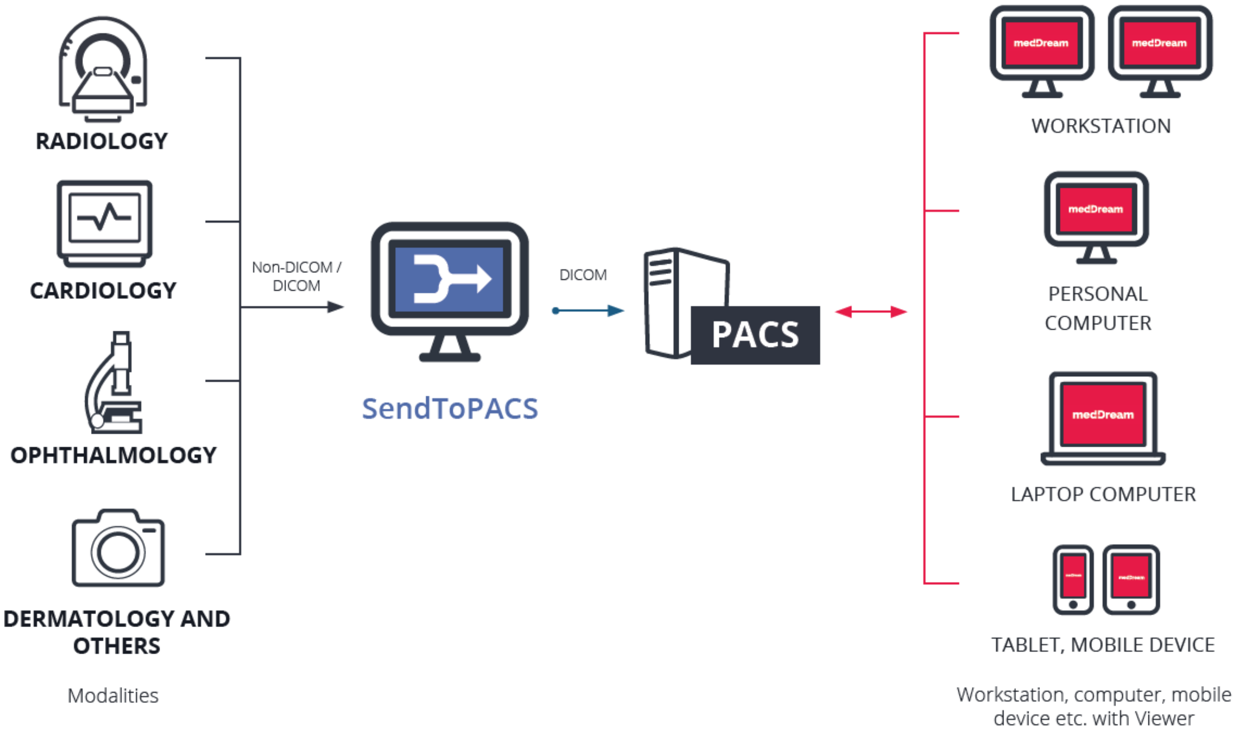

https://www.hiclipart.com/search?clipart=server

# Medical Image Archiving
# Picture Archiving and Communication System (PACS)

- A picture archiving and communication system (PACS) provides economical storage and convenient access to images from multiple modalities
- Eliminates the need to manually file and store, retrieve, or transport physical film images
- Includes both hardware (servers, workstations, network) and software for managing image data

https://en.wikipedia.org/wiki/Picture_archiving_and_communication_system

# PACS
# Overview

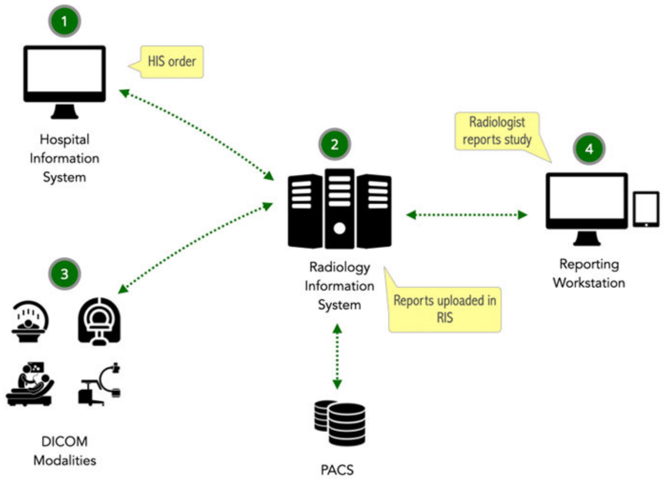



https://www.softneta.com/solutions/dicom-anonymization-for-medical-education-and-research/

# Radiology Information System (RIS)

Hasso Plattner Institut

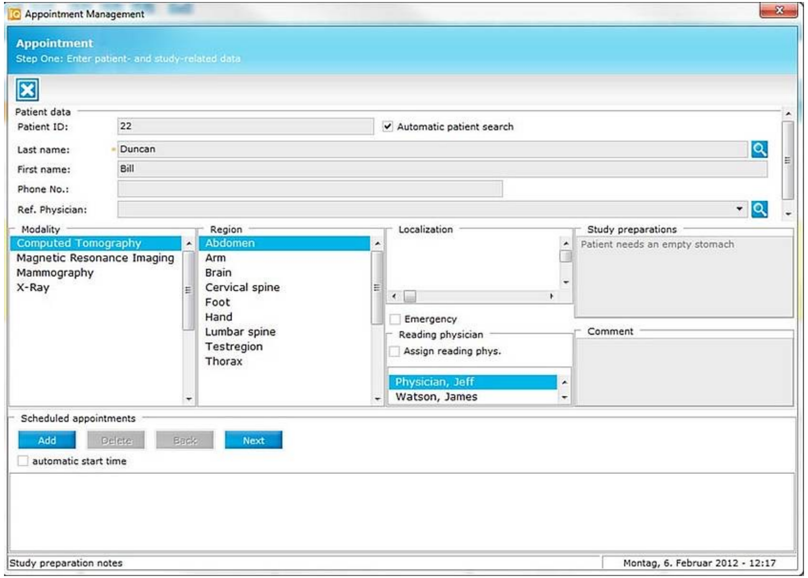
- A networked software system for managing medical imagery and associated data
- Especially useful for tracking radiology imaging orders and billing information
- Often used in conjunction with PACS to manage image archives, record-keeping and billing




https://www.raster.in/ris.php

# RIS Basic Functions

- Patient Management
- Scheduling
- Patient tracking
- Results reporting
- Image tracking
- Billing




https://image-systems.biz/products/ris/

# Agenda

- Imaging Modalities
- Data Formats and Systems
- **Computer Vision**
- Convolutional Neural Networks

# Computer Vision

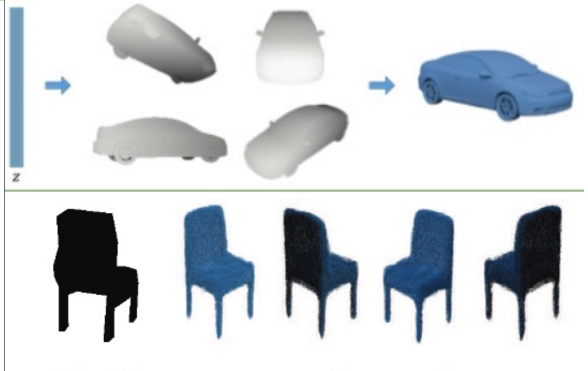

Computer vision for people counter purposes in public places, malls, shopping centers

Interdisciplinary scientific field that deals with how computers can gain high-level understanding from digital images or videos

New iPad includes lidar sensor

Tracking of vehicles movement using Python (cvlib library)

Learning 3D shapes has been a challenging task in computer vision. Recent advances in deep learning have enabled researchers to build models that are able to generate and reconstruct 3D shapes from single or multi-view depth maps or silhouettes seamlessly and efficiently




Images: https://en.wikipedia.org/wiki/Computer_vision

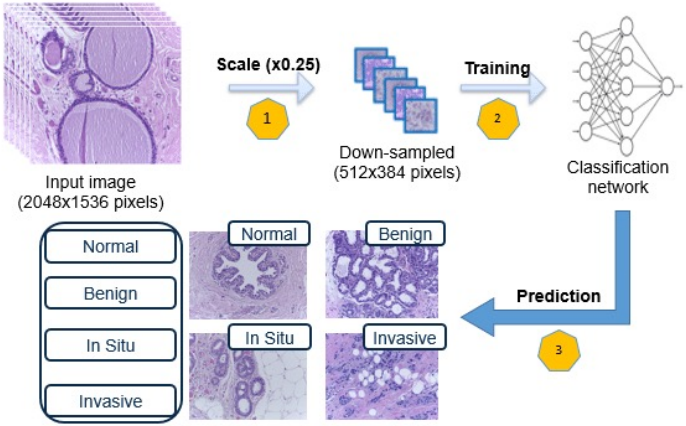
# Computer Vision Tasks (non-exhaustive)

Hasso Plattner Institut

- Classification (optionally Localization)
- Object Detection
- Object Segmentation
- Colorization
- Reconstruction
- Super-Resolution
- Synthesis

# Image Processing Examples
# Image Classification

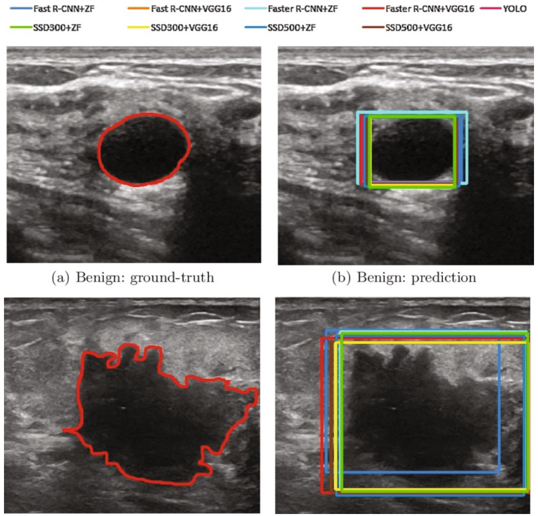
- Also referred to as “*object classification*” and perhaps more generally as “*image recognition*”
- Some examples of image classification include:
  - Labeling an x-ray as cancer or not (binary classification)
  - Classifying a handwritten digit (multiclass classification)
  - Assigning a name to a photograph of a face (multiclass classification)




https://www.researchgate.net/figure/The-overview-of-breast-cancer-image-classification-using-CNN_fig1_337527353

# Image Processing Examples
# Image Classification With Localization

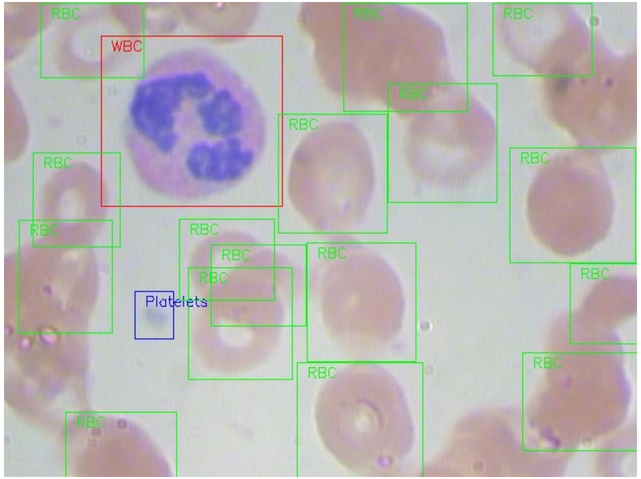
- Assigning a class label to an image and showing the location of the object in the image by a bounding box
- Some examples of image classification with localization include:
  - Labeling an x-ray as cancer or not and drawing a box around the cancerous region
  - Classifying photographs of animals and drawing a box around the animal in each scene




https://www.researchgate.net/figure/Ground-truth-annotations-and-predicted-bounding-boxes-of-different-methods-for-four_fig1_319389518

# Image Processing Examples
# Object Detection

- Task of image classification with localization, although an image may contain multiple objects that require localization and classification
- Some examples of object detection include:
  - Drawing a bounding box and labeling each object in a street scene
  - Drawing a bounding box and labeling each object in an indoor photograph
  - Drawing a bounding box and labeling each object in a landscape



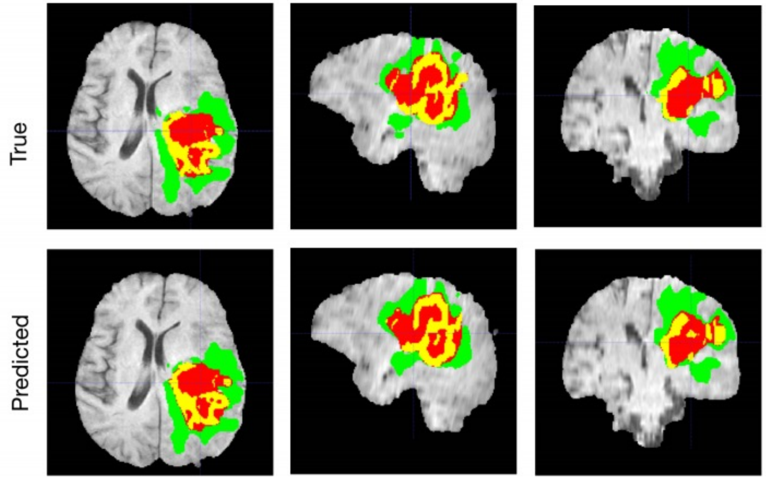

https://www.analyticsvidhya.com/blog/2018/11/implementation-faster-r-cnn-python-object-detection/

# Object Detection and Tracking


https://github.com/hkchengrex/Tracking-Anything-with-DEVA

# Image Processing Examples
# Object Segmentation

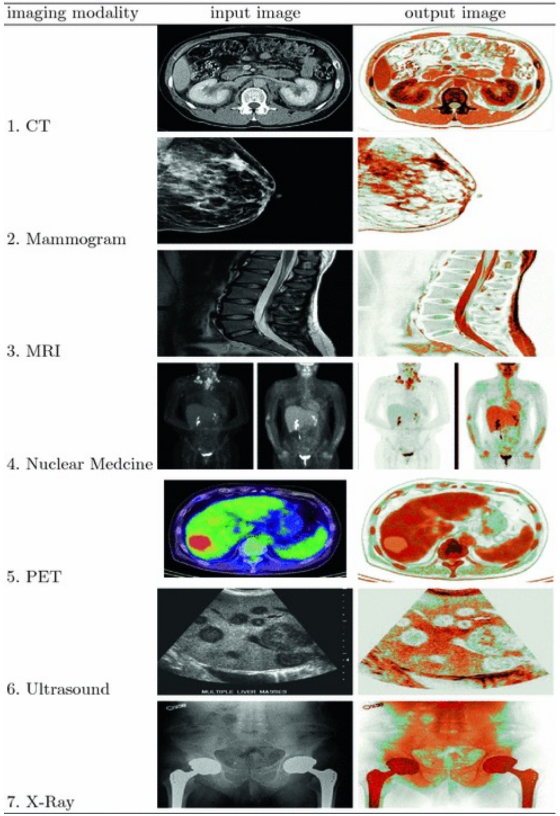
- Task of object detection where a line is drawn around each object detected in the image
- Unlike object detection that involves using a bounding box to identify objects, object segmentation identifies the specific pixels in the image that belong to the object
- Some examples of object segmentation include:
  - Locate tumors and other pathologies
  - Intra-surgery navigation
  - Pedestrian detection




https://news.developer.nvidia.com/automatically-segmenting-brain-tumors-with-ai/

# Image Processing Examples
# Image Colorization

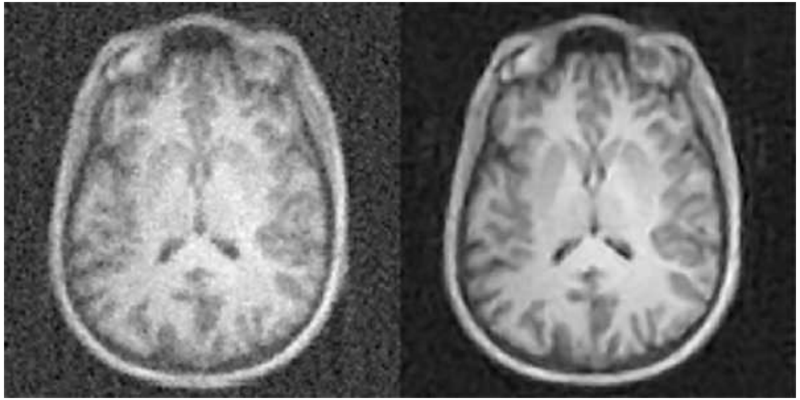
- Converting a grayscale image to a full color image
- Examples include:
  - Colorizing old black and white photographs and movies
  - Improving the contrast of anatomical structures to facilitate precise segmentation




https://link.springer.com/chapter/10.1007/978-3-319-60964-5_50

# Image Processing Examples
# Image Reconstruction

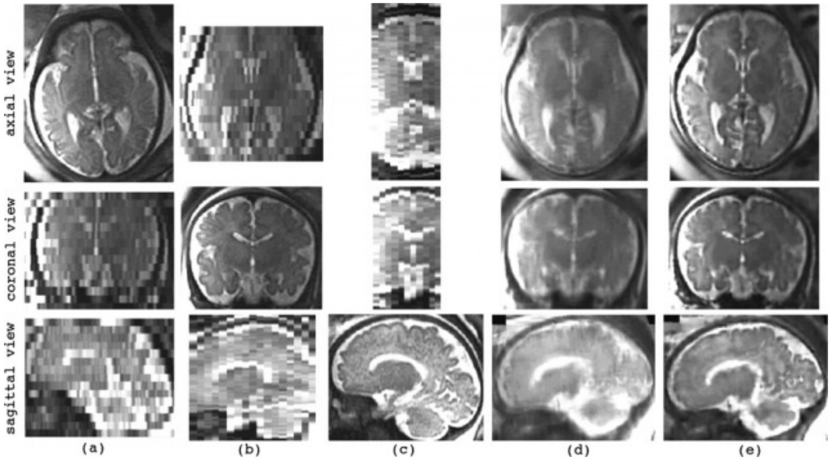
- Task of filling in missing or corrupt parts of an image
- Examples include:
  - Reconstructing old, damaged black and white photographs and movies (e.g., photo restoration)
  - Retain quality with less radiation dose for CT and PET or avoid uncomfortably long scan times for magnetic resonance imaging (MRI)



https://phys.org/news/2018-03-artificial-intelligence-technique-quality-medical.html

# Image Processing Examples
# Image Super-Resolution

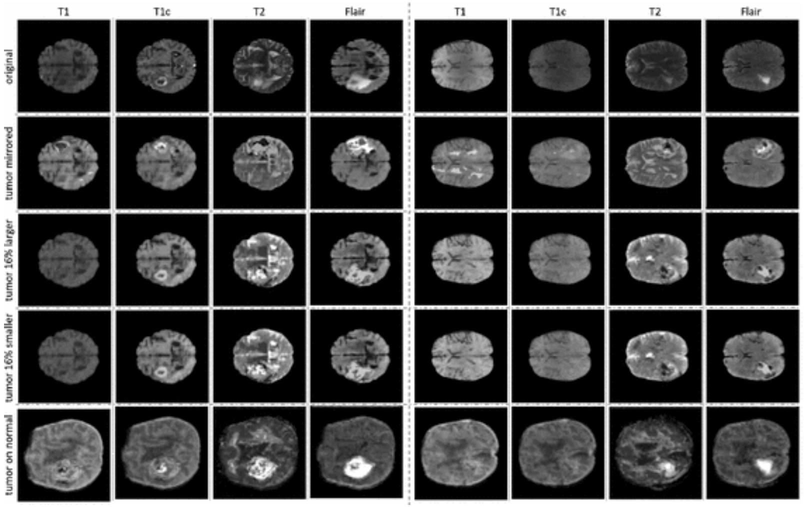
- Task of generating a new version of an image with a higher resolution and detail than the original image
- Examples include:
  - Image restoration
  - Enhance medical image analysis




https://mediantechnologies.com/super-resolution-imaging-for-better-medical-image-analysis/

# Image Processing Examples
# Image Synthesis

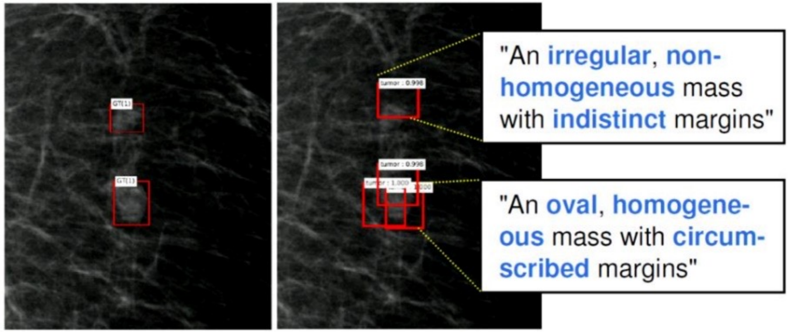
- Task of generating targeted modifications of existing images or entirely new images
- Examples include:
  - Changing the style of an object in a scene
  - Adding an object to a scene
  - Adding a face to a scene
  - Medical image synthesis for data augmentation and anonymization




https://link.springer.com/chapter/10.1007/978-3-030-00536-8_1

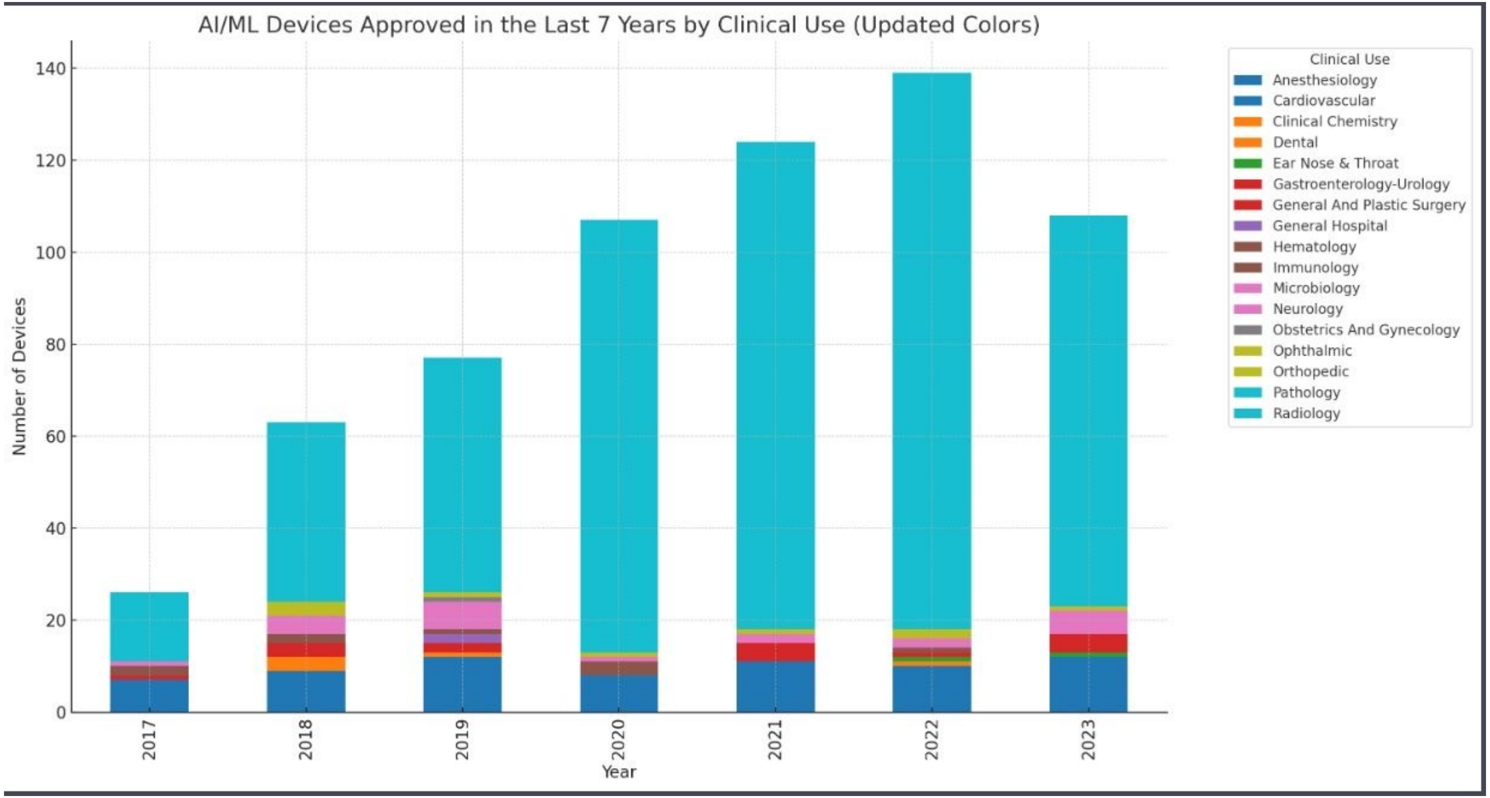
# Multi-Modality

- Not purely computer vision tasks
- Examples include:
  - Image Captioning: Generating a textual description of an image
  - Image Describing: Generating a textual description of each object in an image
  - Text to Image: Synthesizing an image based on a textual description




https://www.semanticscholar.org/paper/Medical-image-captioning-%3A-learning-to-describe-CNN-Kisilev-Sason/592baf974b02598d43eda669a652853f8d44a8da

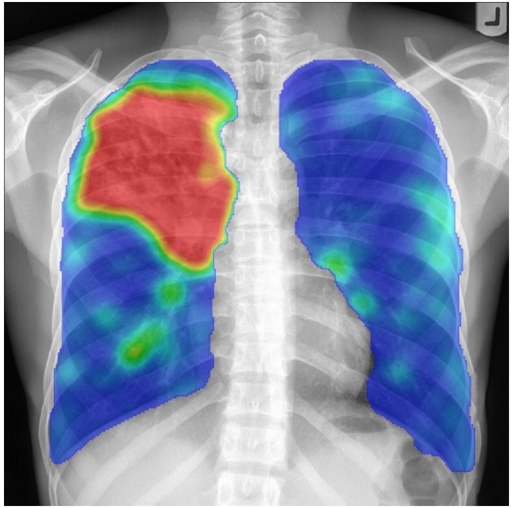
# AI/ML-Enabled Medical Devices (FDA Approved)




https://www.fda.gov/medical-devices/software-medical-device-samd/artificial-intelligence-and-machine-learning-aiml-enabled-medical-devices

# Computer Vision and Medical Imaging

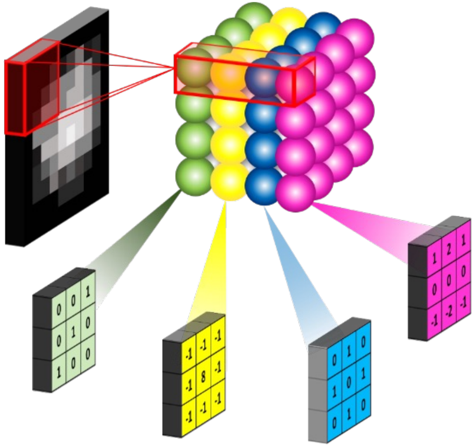
- Various applications for CV in medical imaging, e.g., computer-aided diagnosis (via classification or object detection)
- CV today is dominated by (supervised) deep learning approaches
- Primary deep learning architecture for images: convolutional neural network (CNN)



https://medium.com/analytics-vidhya/detecting-anomalies-in-x-ray-using-cnn-1e4c2e49f23a

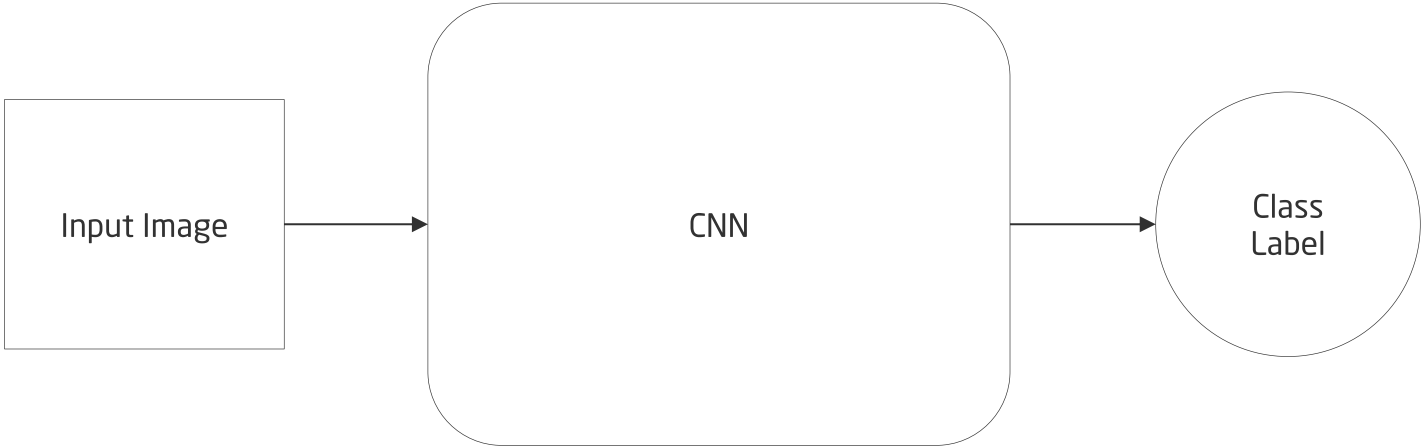
# Convolutional Neural Network (CNN)

- Name “convolutional neural network” indicates that the network employs a mathematical operation called **convolution**
- Convolution is a specialized kind of linear operation
- CNNs are very similar to ordinary Neural Networks – they are made up of neurons that have learnable weights and biases
- Each neuron receives some inputs, performs a dot product and optionally follows it with a non-linearity

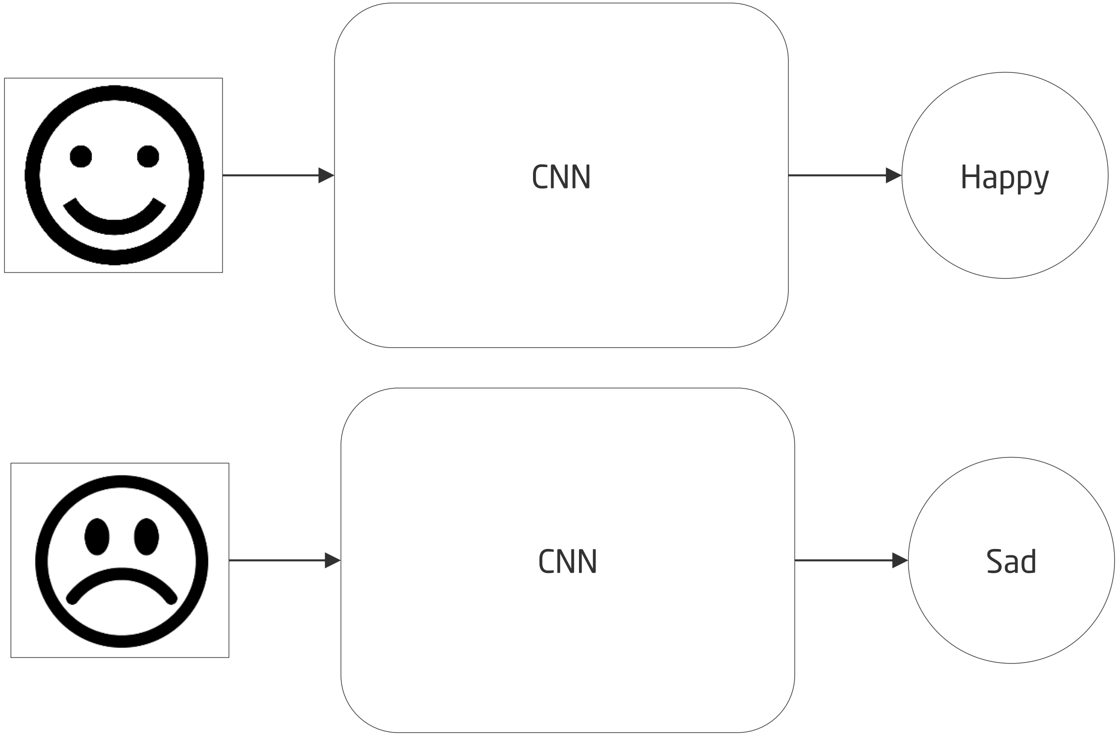


https://www.hiclipart.com/search?clipart=convolutional+neural+network

# CNNs for Image Classification



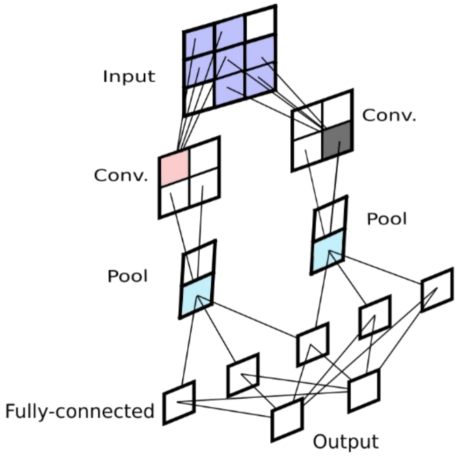
# CNNs for Image Classification Example




**Medical Image Analysis**

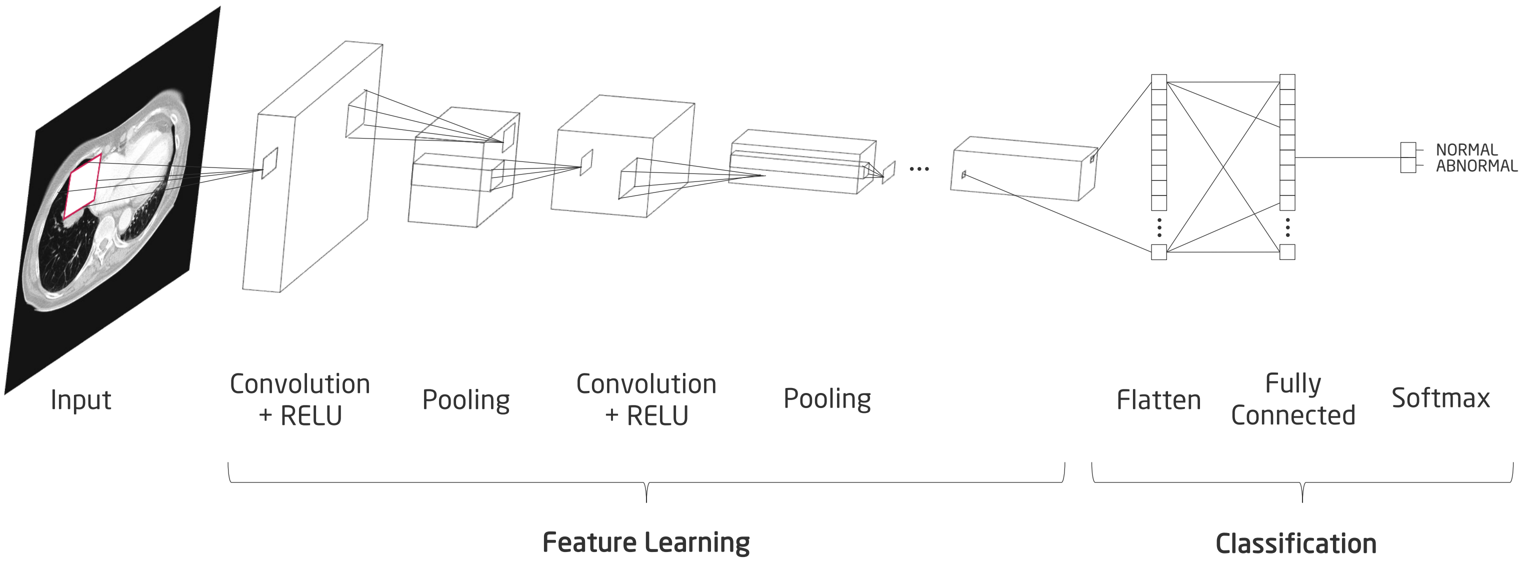
# CNNs for Image Classification

- A CNN for image classification is a combination of two basic building blocks:
  1. The Convolution Block consists of the Convolution Layer and the Pooling Layer. This layer forms the essential component for feature extraction
  2. The Fully Connected Block consists of a fully connected (feed forward) neural network. This layer performs the task of *Classification* based on the input from the convolutional block.

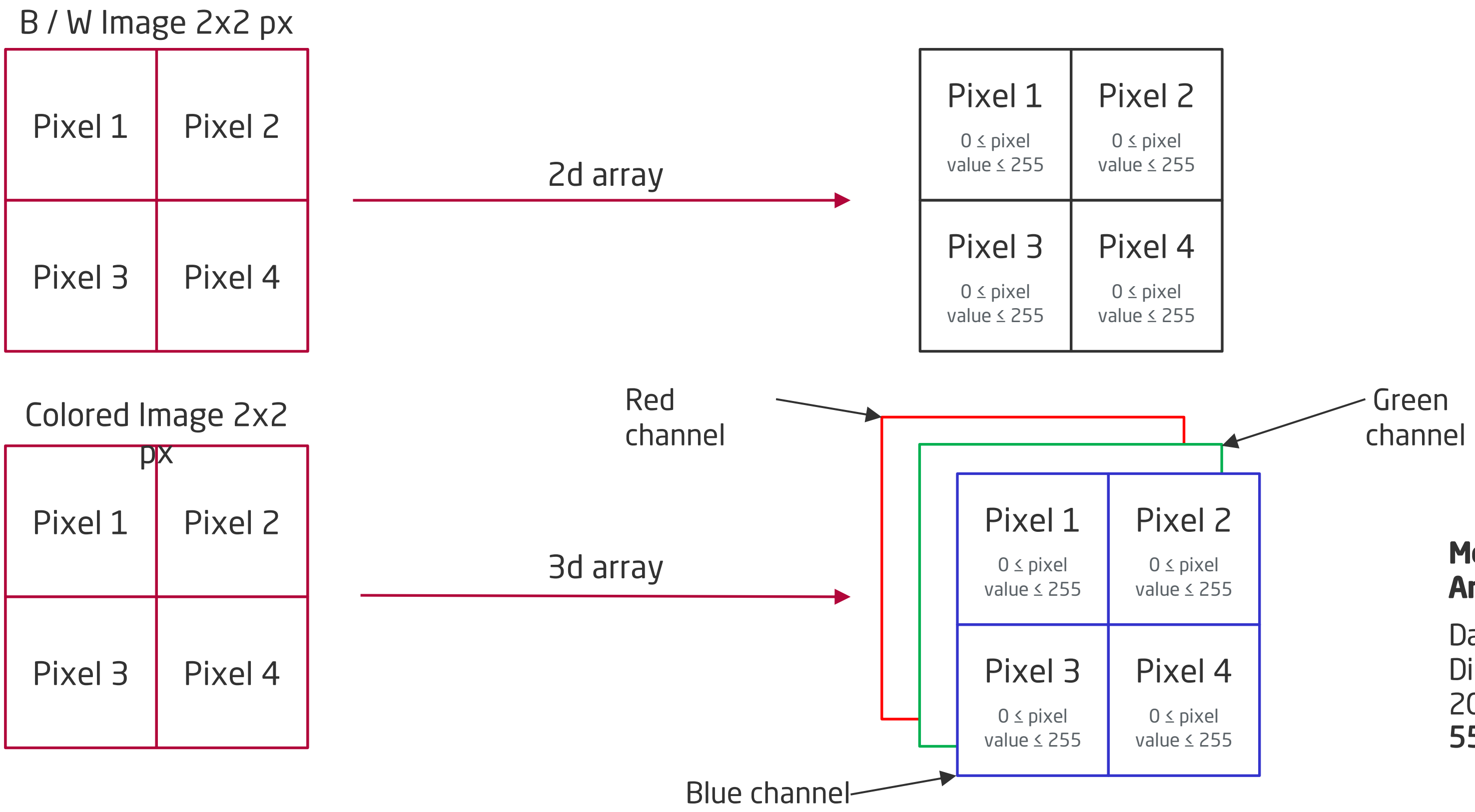



https://www.hiclipart.com/search?clipart=convolutional+neural+network

# CNNs for Image Classification Details



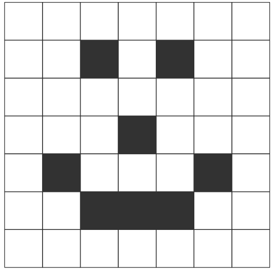
# CNN



B / W Image 2x2 px
Pixel 1
Pixel 2
Pixel 3
Pixel 4
2d array
Pixel 1
0 ≤ pixel value ≤ 255
Pixel 2
0 ≤ pixel value ≤ 255
Pixel 3
0 ≤ pixel value ≤ 255
Pixel 4
0 ≤ pixel value ≤ 255
Colored Image 2x2 px
Pixel 1
Pixel 2
Pixel 3
Pixel 4
3d array
Red channel
Green channel
Blue channel
Pixel 1
0 ≤ pixel value ≤ 255
Pixel 2
0 ≤ pixel value ≤ 255
Pixel 3
0 ≤ pixel value ≤ 255
Pixel 4
0 ≤ pixel value ≤ 255

# CNN
# What We See Vs. What Computers See

What we see



| | | | | | | |
|---|---|---|---|---|---|---|
| 255 | 255 | 255 | 255 | 255 | 255 | 255 |
| 255 | 255 | 0 | 255 | 0 | 255 | 255 |
| 255 | 255 | 255 | 255 | 255 | 255 | 255 |
| 255 | 255 | 255 | 0 | 255 | 255 | 255 |
| 255 | 0 | 255 | 255 | 255 | 0 | 255 |
| 255 | 255 | 0 | 0 | 0 | 255 | 255 |
| 255 | 255 | 255 | 255 | 255 | 255 | 255 |

What the computer sees

# Convolution Layer

Also called 'filter' or 'kernel'

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

# Convolution Layer



$$(I * K)_{xy} = \sum_{i=1}^{h} \sum_{j=1}^{w} K_{ij} \cdot I_{x+i-1, y-j-1}$$

Input Image (I)

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$\otimes$

Feature Detector (K)

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

$=$

Feature Map

| | | | | |
|---|---|---|---|---|
| 0 | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

[0*0 + 0*0 + 0*1 + 0*1 + 1*0 + 0*0 + 0*0 + 0*1 + 0*1] = [0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0] = [0]

# Convolution Layer



| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

=

| | | | | |
|---|---|---|---|---|
| 0 | 1 | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Feature Map

[0*0 + 0*0 + 0*1 + 1*1 + 0*0 + 0*0 + 0*0 + 0*1 + 0*1] = [0 + 0 + 0 + 1 + 0 + 0 + 0 + 0 + 0] = [1]

# Convolution Layer



| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

=

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Feature Map

[0*0 + 0*0 + 0*1 + 0*1 + 0*0 + 0*0 + 0*0 + 0*1 + 0*1] = [0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0] = [0]

# Convolution Layer

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

$=$

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Feature Map

# Convolution Layer

Hasso Plattner Institut

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

$=$

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Feature Map

# Convolution Layer

Hasso Plattner Institut

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

$=$

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | | | | |
| | | | | |
| | | | | |
| | | | | |

Feature Map

# Convolution Layer



Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$\otimes$

Feature Detector

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

$=$

Feature Map

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | | | |
| | | | | |
| | | | | |
| | | | | |

# Convolution Layer

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

$=$

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | | |
| | | | | |
| | | | | |
| | | | | |

Feature Map

# Convolution Layer

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

$=$

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | |
| | | | | |
| | | | | |
| | | | | |

Feature Map

# Convolution Layer

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| | | | | |
| | | | | |
| | | | | |

Feature Map

# Convolution Layer

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

$=$

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | | | | |
| | | | | |
| | | | | |

Feature Map

# Convolution Layer

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

=

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | | | |
| | | | | |
| | | | | |

Feature Map

# Convolution Layer



| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

=

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | | |
| | | | | |
| | | | | |

Feature Map

# Convolution Layer

Hasso Plattner Institut

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

⊗

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

=

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | |
| | | | | |
| | | | | |

Feature Map

# Convolution Layer

Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$\otimes$

Feature Detector

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

=

Feature Map

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| | | | | |
| | | | | |

# Convolution Layer



| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

=

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | | | | |
| | | | | |

Feature Map

# Convolution Layer

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

$=$

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | | | |
| | | | | |

Feature Map

# Convolution Layer

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

$=$

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | | |
| | | | | |

Feature Map

# Convolution Layer

Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$\otimes$

Feature Detector

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

$=$

Feature Map

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | |
| | | | | |

# Convolution Layer

Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$\otimes$

Feature Detector

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

$=$

Feature Map

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| | | | | |

# Convolution Layer



| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

$=$

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| 0 | | | | |

Feature Map

# Convolution Layer



| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

$=$

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| 0 | 0 | | | |

Feature Map

# Convolution Layer

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

$=$

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| 0 | 0 | 1 | | |

Feature Map

# Convolution Layer



| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

Feature Detector

$=$

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| 0 | 0 | 1 | 2 | |

Feature Map

# Convolution Layer



| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

$\otimes$

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |

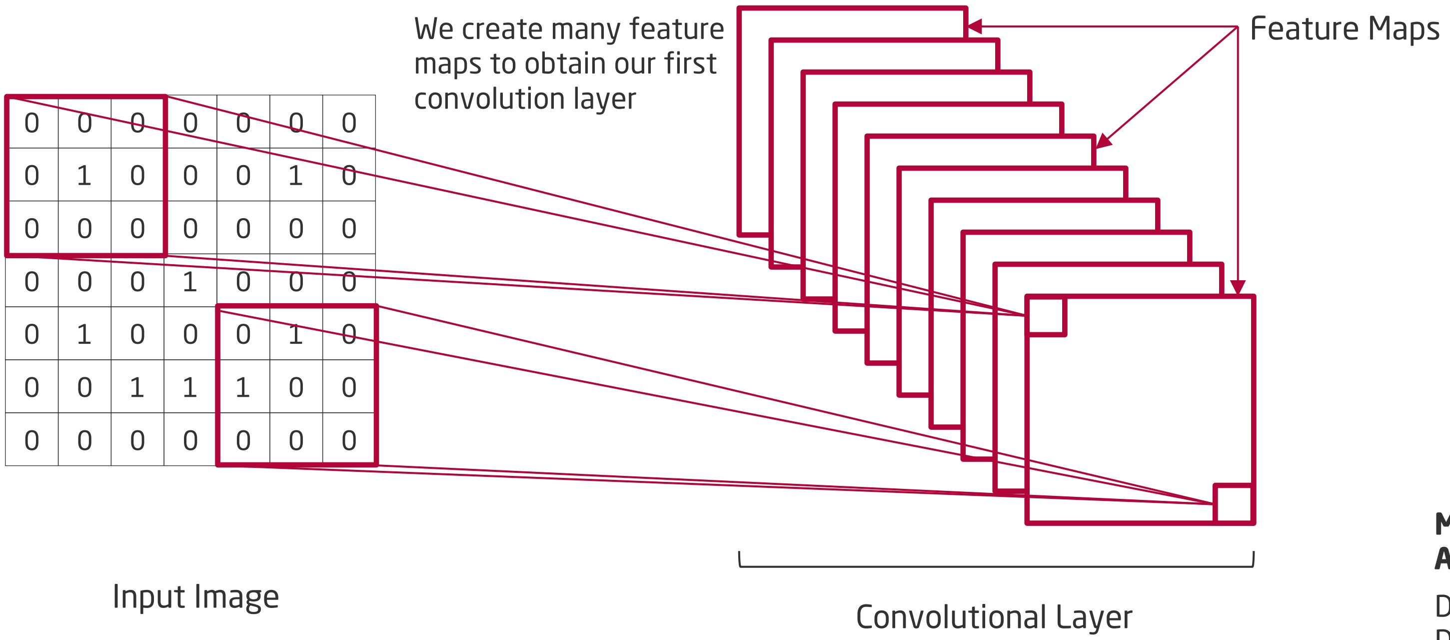
Feature Detector

=

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| 0 | 0 | 1 | 2 | 1 |

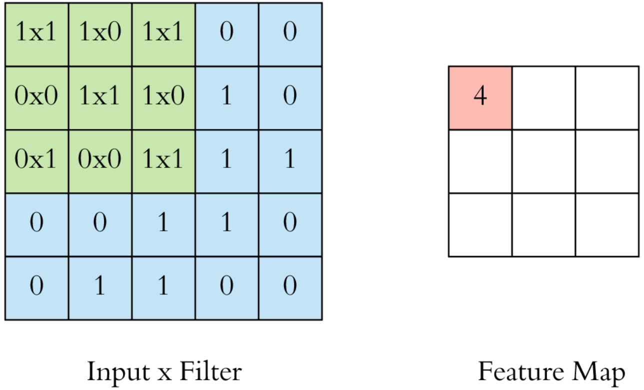
Feature Map

# Convolution Layer



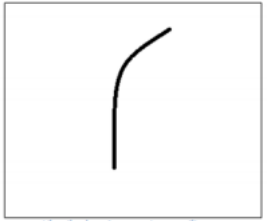
# Convolution Layer
# Why Feature Maps

- Recognize features like edges, … automatically
- Input for ML algorithms
- Example for manually engineered features:
  - Texture Features: Histogram based, Entropy, Haralick features (Co-occurrence matrix), Gray-level run length metrics, Local Binary Pattern, Fractal, etc.
  - Morphological Features: Hu’s moments, Shape features, Granulometry, Bending Energy, Roundness ratio, etc.




https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2

# Convolution Layer
# Why Feature Maps

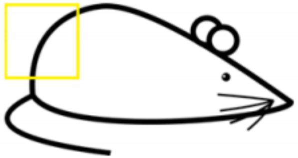


Visualization of a curve detector filter

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 30 | 0 |
| 0 | 0 | 0 | 0 | 30 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Pixel representation of filter

Original image



Visualization of the filter on the image

Visualization of the receptive field

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 30 |
| 0 | 0 | 0 | 0 | 50 | 50 | 50 |
| 0 | 0 | 0 | 20 | 50 | 0 | 0 |
| 0 | 0 | 0 | 50 | 50 | 0 | 0 |
| 0 | 0 | 0 | 50 | 50 | 0 | 0 |
| 0 | 0 | 0 | 50 | 50 | 0 | 0 |
| 0 | 0 | 0 | 50 | 50 | 0 | 0 |

Pixel representation of the receptive field

*

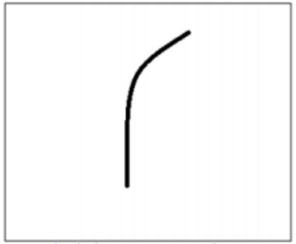
| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 30 | 0 |
| 0 | 0 | 0 | 0 | 30 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Pixel representation of filter

Multiplication and Summation = (50*30)+(50*30)+(50*30)+(20*30)+(50*30) = 6600 (A large number!)

https://lipyeow.github.io/ics491f17/morea/deepnn/ImageClassification-CNN.pdf

# Convolution Layer
# Why Feature Maps



Visualization of a curve detector filter

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 30 | 0 |
| 0 | 0 | 0 | 0 | 30 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Pixel representation of filter

Visualization of the filter on the image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 40 | 0 | 0 | 0 | 0 | 0 |
| 40 | 0 | 40 | 0 | 0 | 0 | 0 |
| 40 | 20 | 0 | 0 | 0 | 0 | 0 |
| 0 | 50 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 50 | 0 | 0 | 0 | 0 |
| 25 | 25 | 0 | 50 | 0 | 0 | 0 |

Pixel representation of receptive field

*

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 30 | 0 |
| 0 | 0 | 0 | 0 | 30 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 30 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

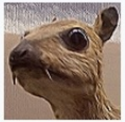
Pixel representation of filter

Multiplication and Summation = 0

https://lipyeow.github.io/ics491f17/morea/deepnn/ImageClassification-CNN.pdf

# Convolution Layer
# Convolutional Operator

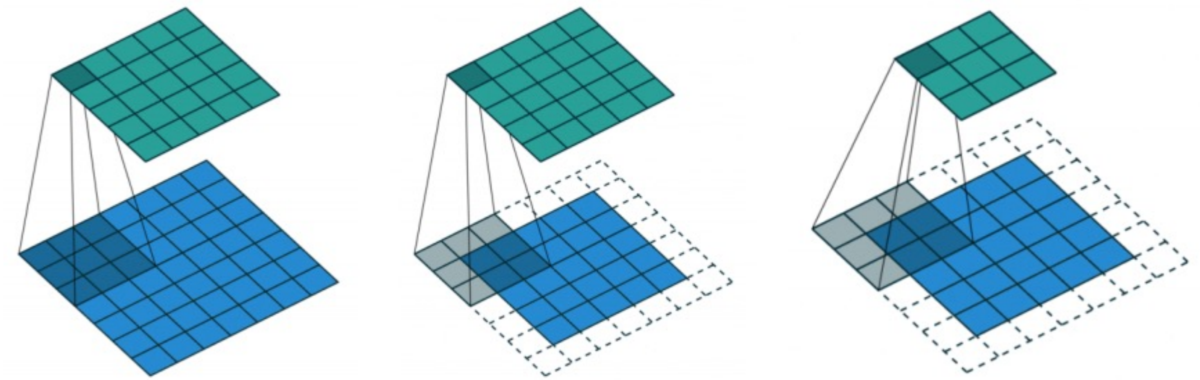
Different filters will produce different Feature Maps for the same input image. Examples:

| Operation | Kernel ω | Image result g(x,y) |
|---|---|---|
| Identity | $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ |  |
| Edge detection | $\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$ |  |
| | $\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$ |  |
| | $\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$ |  |
| Sharpen | $\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$ |  |
| Box blur (normalized) | $\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ |  |
| Gaussian blur 3 × 3 (approximation) | $\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$ |  |
| Gaussian blur 5 × 5 (approximation) | $\frac{1}{256}\begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$ |  |
| Unsharp masking 5 × 5 Based on Gaussian blur with amount as 1 and threshold as 0 (with no image mask) | $\frac{-1}{256}\begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & -476 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$ |  |

https://en.wikipedia.org/wiki/Kernel_(image_processing)

# CNN
# Convolution Layer

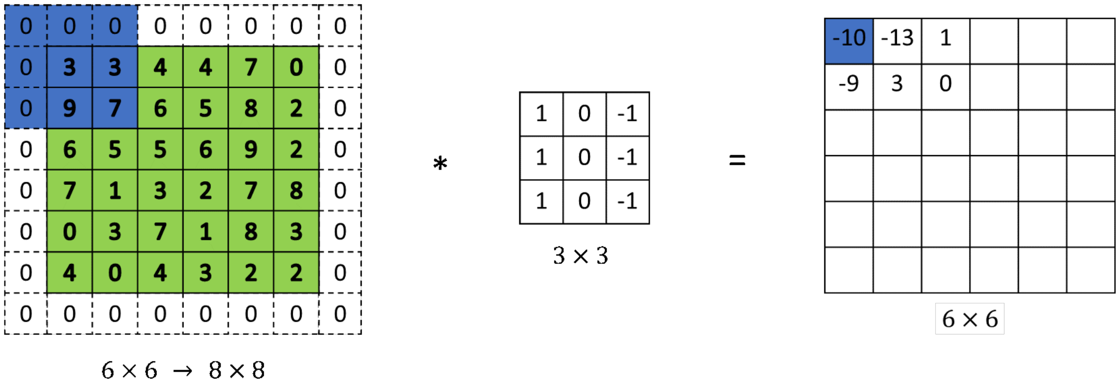
- In practice, a CNN learns the values of these filters on its own during the training process
- Although we still need to specify parameters such as number of filters, filter size, padding, and stride before the training process



https://lipyeow.github.io/ics491f17/morea/deepnn/ImageClassification-CNN.pdf

# Convolution Layer
## Padding

- Adding extra pixels of filler around the boundary of input image → Increasing the effective size of the image
- Typically, set values of the extra pixels to zero
- **Valid convolutions** ↔️ no padding
- **Same convolution** ↔️ pad so that output size is the same as the input size




http://datahacker.rs/what-is-padding-cnn/

# Convolution Layer
# Striding

- Stride controls how the filter convolves around the input
- In the example we had, the filter convolves around the input by shifting one unit at a time
- The amount by which the filter shifts is the stride
- Stride is normally set in a way so that the output volume is an integer and not a fraction

O - Output height/length

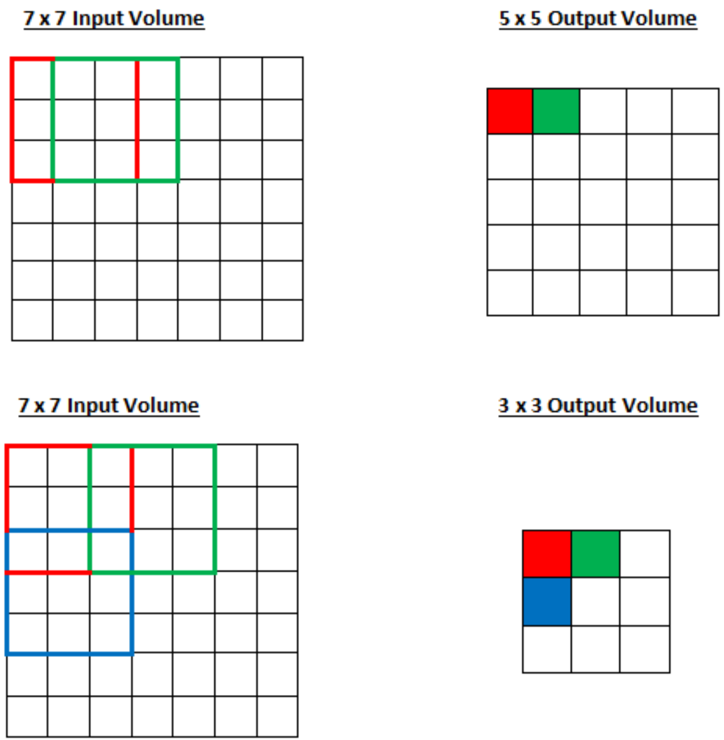
I - Input height/length

K - Filter size

P - Padding

S - Stride

$$O = \frac{(I - K + 2P)}{S} + 1$$

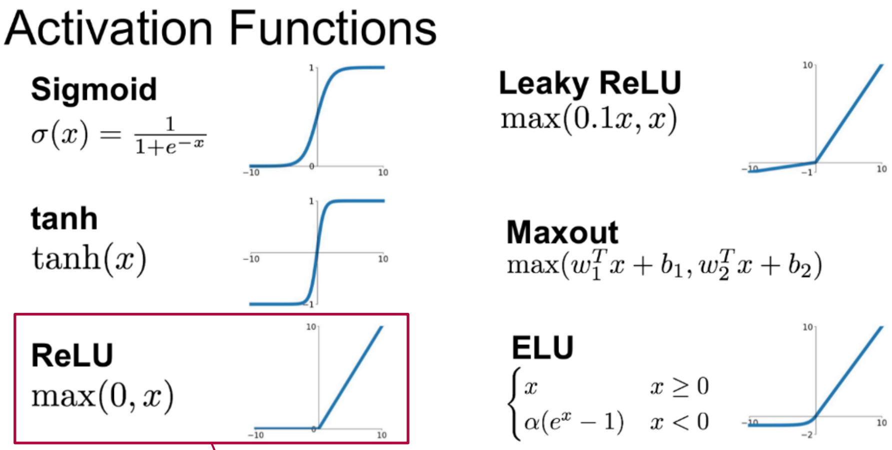



https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks-Part-2/

# Convolution Layer
# Activation (ReLU)

Hasso Plattner Institut

- An additional operation called Rectified Linear Unit (ReLU) has been used after every Convolution operation
- Basically, ReLU is an element wise operation (applied per pixel) and replaces all negative pixel values in the feature map by zero
- The purpose of ReLU is to introduce non-linearity to the network
- Other nonlinear functions such as tanh or sigmoid can also be used instead of ReLU, but ReLU has been found to perform better in most situations




$$(I * K)_{xy} = \sigma\left(b + \sum_{i=1}^{h}\sum_{j=1}^{w} K_{ij} \cdot I_{x+i-1,y-j-1}\right)$$

https://medium.com/dataseries/basic-overview-of-convolutional-neural-network-cnn-4fcc7dbb4f17

# Output layer

- Activation functions for output layer :
  - Softmax
  - Sigmoid
- Sigmoid activation functions are used when the output of the neural network is continuous.
- Softmax activation functions are used when the output of the neural network is categorical

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

$\sigma$ = softmax

$\vec{z}$ = input vector

$e^{z_i}$ = standard exponential function for input vector

$K$ = number of classes in the multi-class classifier

$$S(x) = \frac{1}{1 + e^{-x}}$$

$S(x)$ = sigmoid function

$e$ = Euler's number

# Convolution Layer
# << QUIZ >>

- Which matrix represents vertical edge detection?

**A**

| | | |
|---|---|---|
| 1 | 0 | -1 |
| 1 | 0 | -1 |
| 1 | 0 | -1 |

**B**

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 0 | 0 | 0 |
| -1 | -1 | -1 |

# Convolution Layer
# << QUIZ >>




- Which matrix represents vertical edge detection?

**A**

| | | |
|---|---|---|
| 1 | 0 | -1 |
| 1 | 0 | -1 |
| 1 | 0 | -1 |

✓

**B**

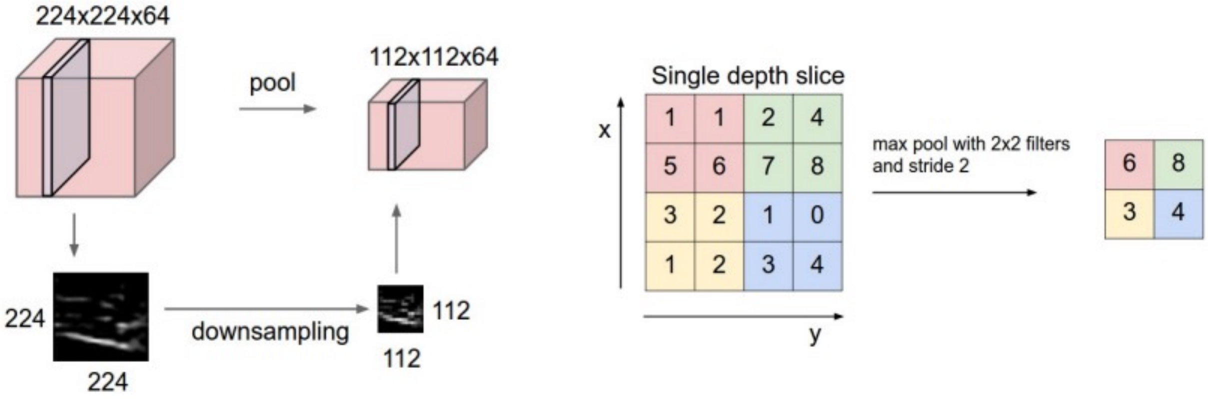
| | | |
|---|---|---|
| 1 | 1 | 1 |
| 0 | 0 | 0 |
| -1 | -1 | -1 |

✗

# CNN
# Pooling Layer

- Pooling layer downsamples the volume spatially, independently in each depth slice of the input

https://lipyeow.github.io/ics491f17/morea/deepnn/ImageClassification-CNN.pdf




- The most common downsampling operation is max, giving rise to max pooling, here shown with a stride of 2

# CNN
## Pooling Layer

Hasso Plattner Institut

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| 0 | 0 | 1 | 2 | 1 |

Feature Map

Max Pooling →

| | | |
|---|---|---|
| | | |
| | | |
| | | |

Pooled Feature Map

# CNN
## Pooling Layer

Hasso Plattner Institut

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| 0 | 0 | 1 | 2 | 1 |

Feature Map

Max Pooling →

| | | |
|---|---|---|
| 1 | | |
| | | |
| | | |

Pooled Feature Map

# CNN
## Pooling Layer

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| 0 | 0 | 1 | 2 | 1 |

Feature Map

Max Pooling

| | | |
|---|---|---|
| 1 | 1 | |
| | | |
| | | |

Pooled Feature Map

# CNN
# Pooling Layer

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| 0 | 0 | 1 | 2 | 1 |

Feature Map

Max Pooling

| | | |
|---|---|---|
| 1 | 1 | 0 |
| | | |
| | | |

Pooled Feature Map

# CNN
## Pooling Layer

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| 0 | 0 | 1 | 2 | 1 |

Feature Map

Max Pooling →

| | | |
|---|---|---|
| 1 | 1 | 0 |
| 4 | | |
| | | |

Pooled Feature Map

# CNN
## Pooling Layer



| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| 0 | 0 | 1 | 2 | 1 |

Feature Map

Max Pooling →

| | | |
|---|---|---|
| 1 | 1 | 0 |
| 4 | 2 | |
| | | |

Pooled Feature Map

# CNN
## Pooling Layer



| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| 0 | 0 | 1 | 2 | 1 |

Feature Map

Max Pooling →

| | | |
|---|---|---|
| 1 | 1 | 0 |
| 4 | 2 | 1 |
| | | |

Pooled Feature Map

# CNN
# Pooling Layer



| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| 0 | 0 | 1 | 2 | 1 |

Feature Map

Max Pooling →

| | | |
|---|---|---|
| 1 | 1 | 0 |
| 4 | 2 | 1 |
| 0 | | |

Pooled Feature Map

# CNN
## Pooling Layer



| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| 0 | 0 | 1 | 2 | 1 |

Feature Map

Max Pooling →

| | | |
|---|---|---|
| 1 | 1 | 0 |
| 4 | 2 | 1 |
| 0 | 2 | |

Pooled Feature Map

# CNN
## Pooling Layer

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 2 | 1 |
| 1 | 4 | 2 | 1 | 0 |
| 0 | 0 | 1 | 2 | 1 |

Feature Map

Max Pooling →

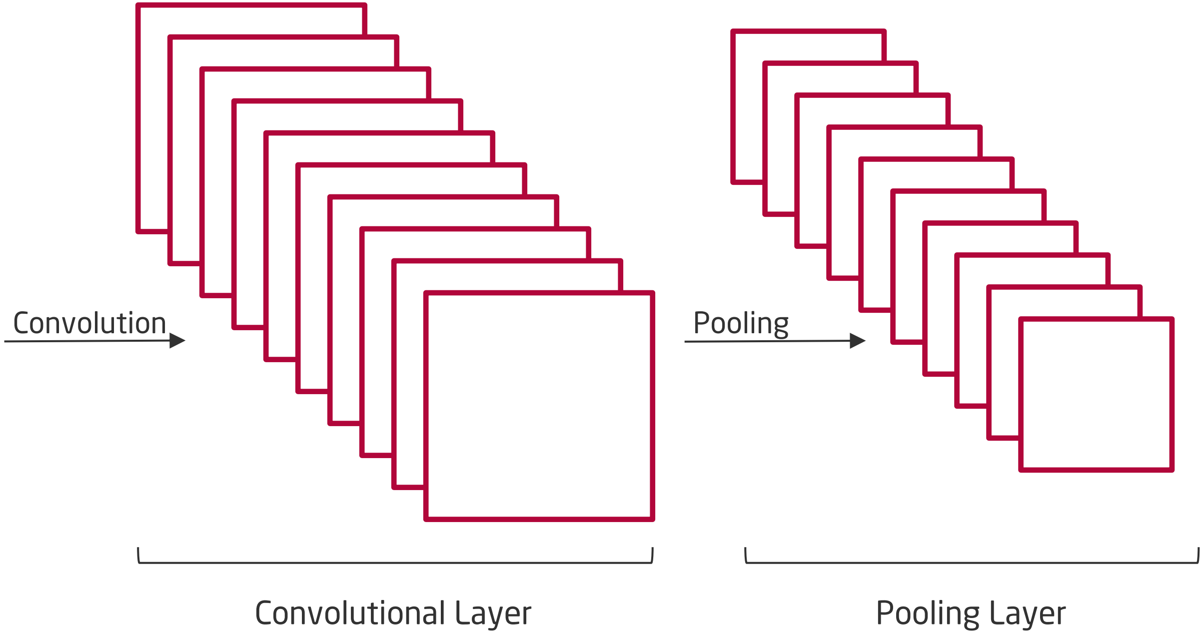
| | | |
|---|---|---|
| 1 | 1 | 0 |
| 4 | 2 | 1 |
| 0 | 2 | 1 |

Pooled Feature Map

# CNN
# Pooling Layer

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Input Image

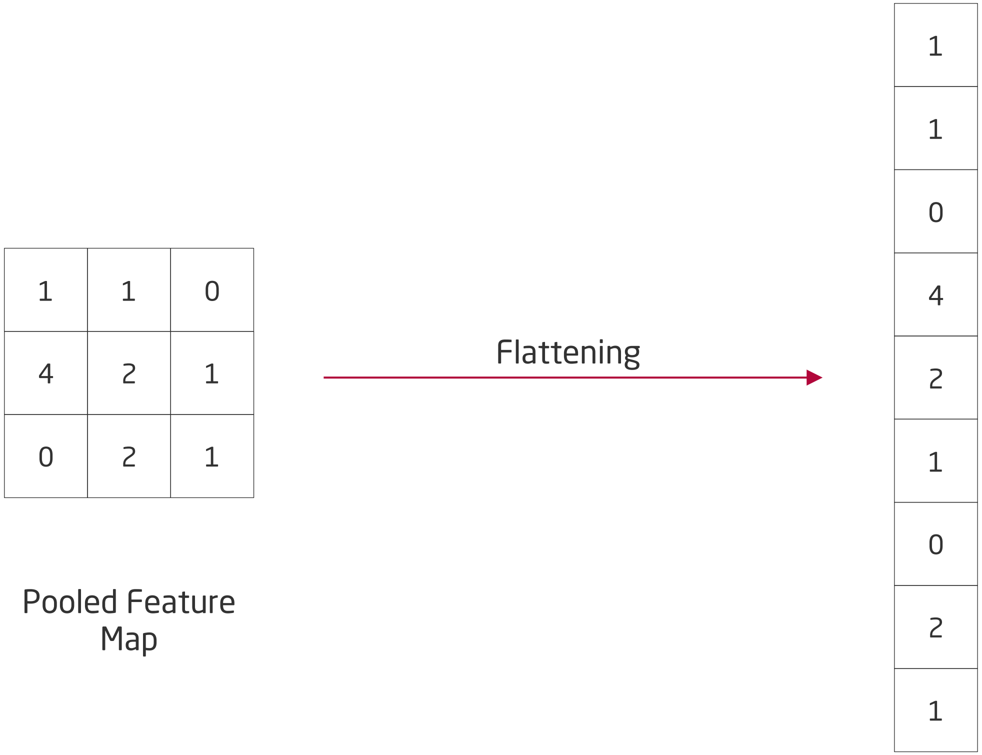


# CNN
# Flattening

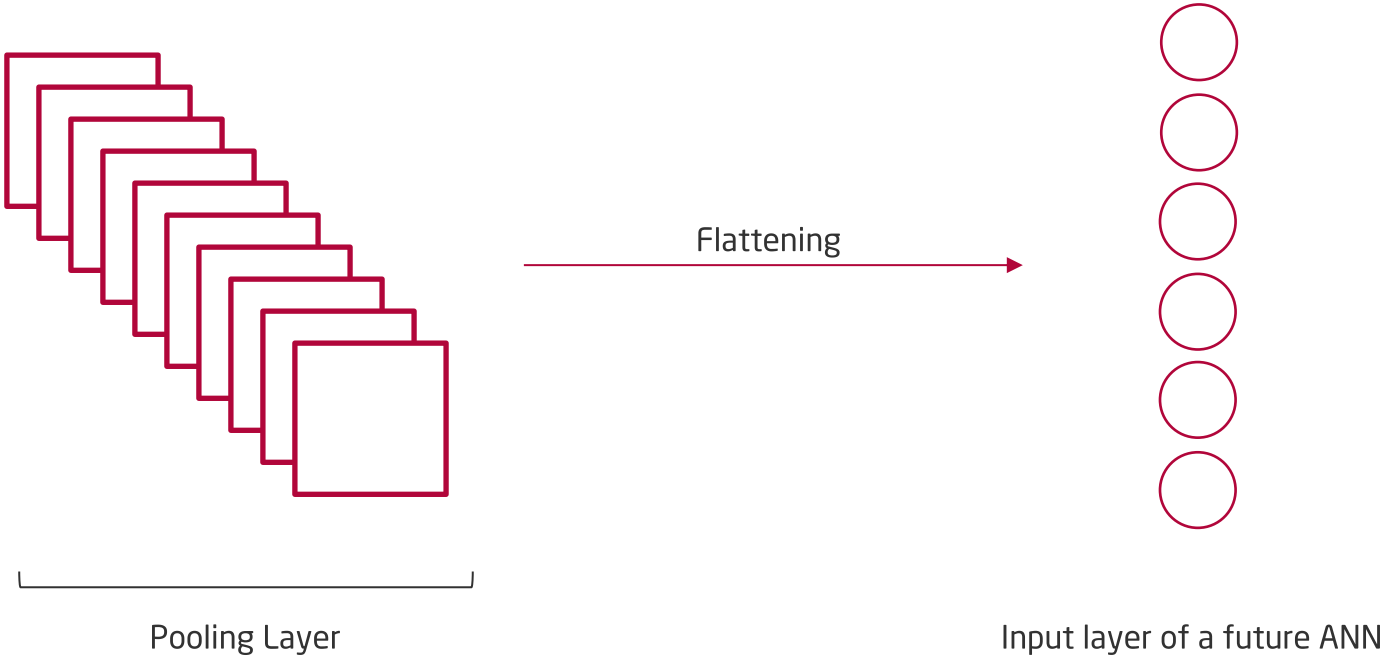
| | | |
|---|---|---|
| 1 | 1 | 0 |
| 4 | 2 | 1 |
| 0 | 2 | 1 |

Pooled Feature Map

# CNN
# Flattening

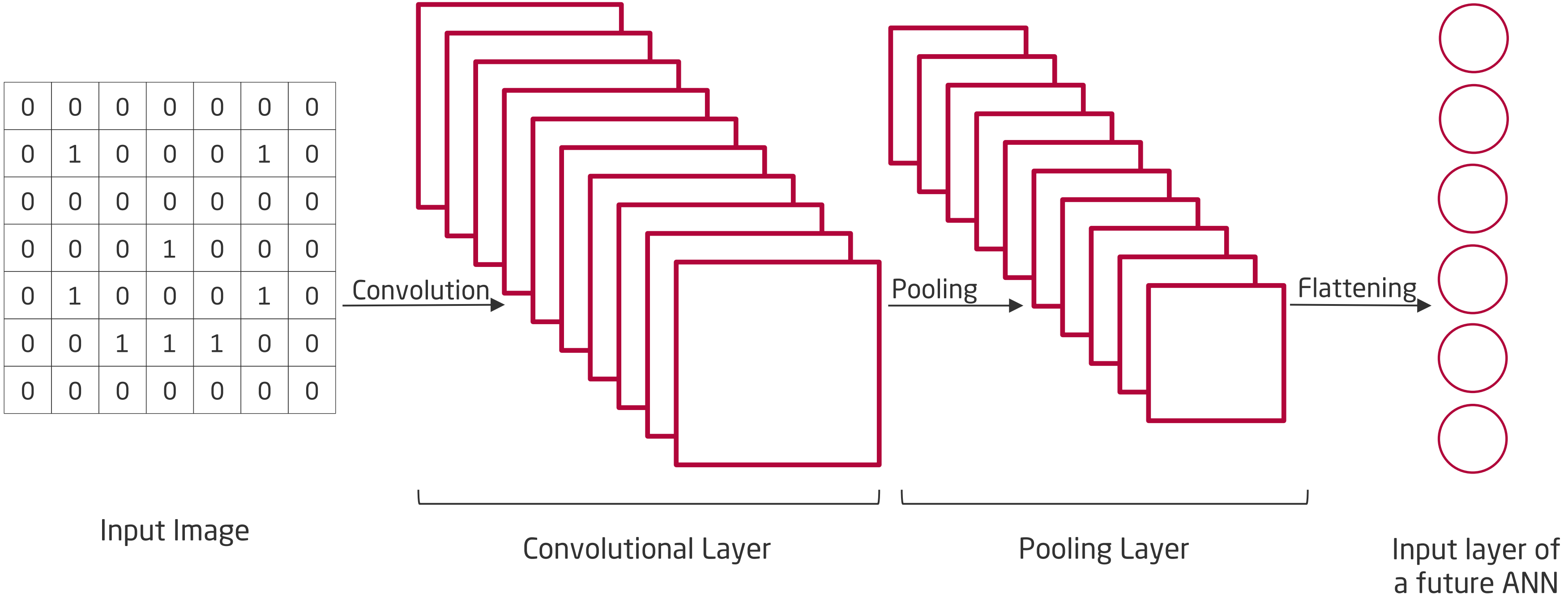


# CNN
# Flattening



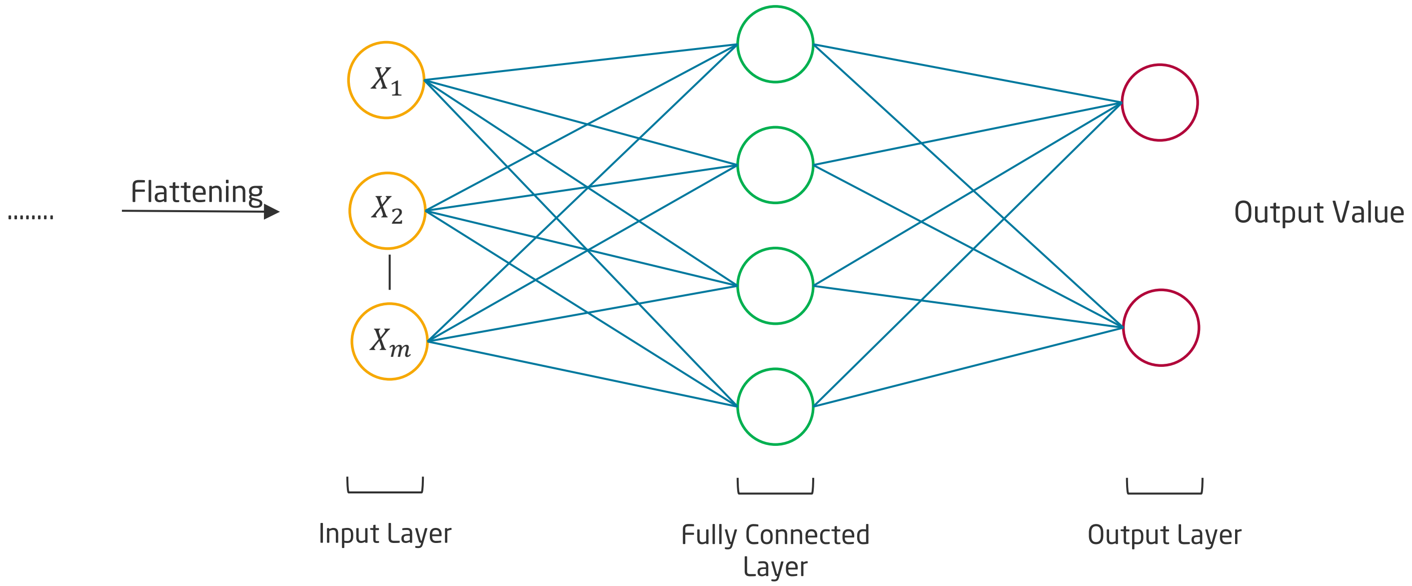
# CNN
# Flattening

Hasso
Plattner
Institut

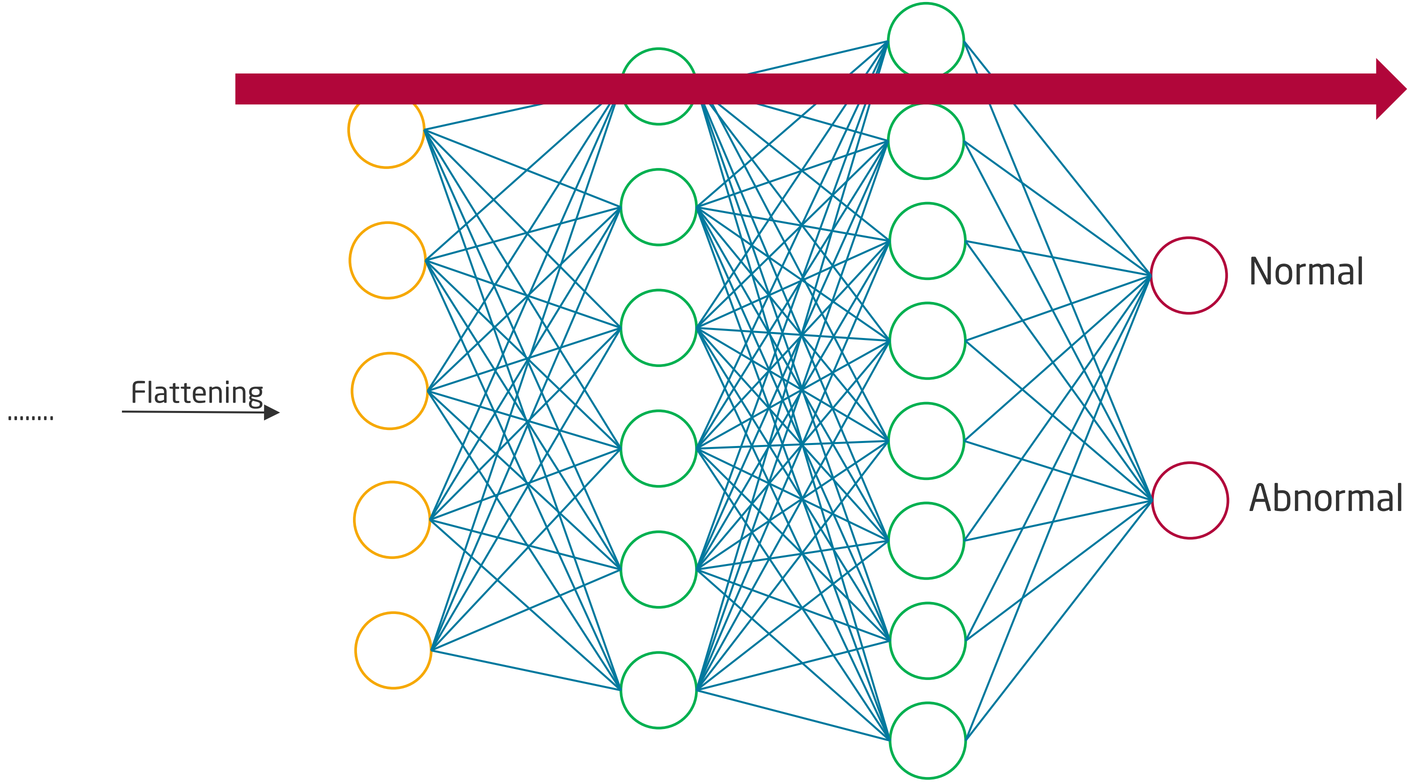


# CNN
# Fully Connected Layer

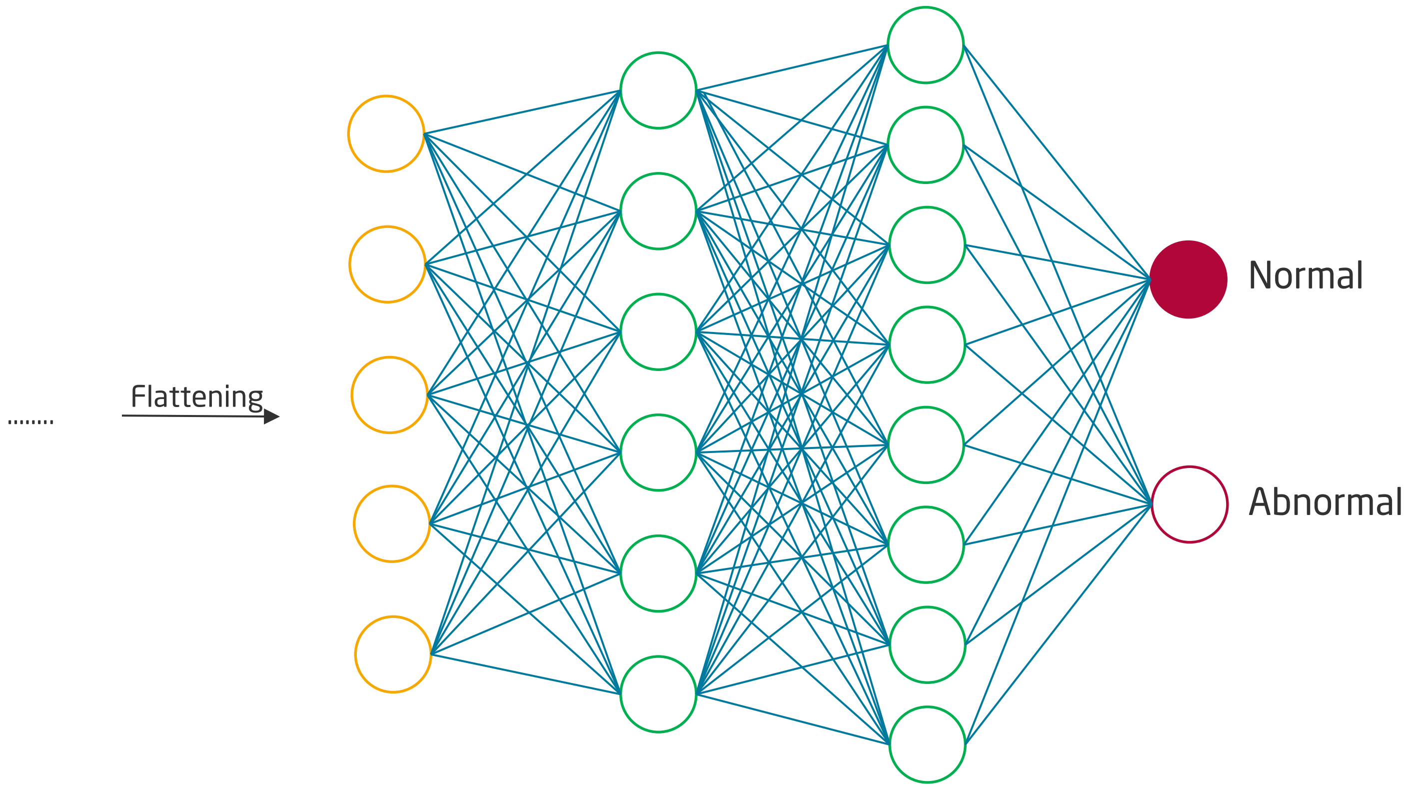
- Neurons in a fully connected layer have full connections to all activations in the previous layer, as seen in regular neural networks



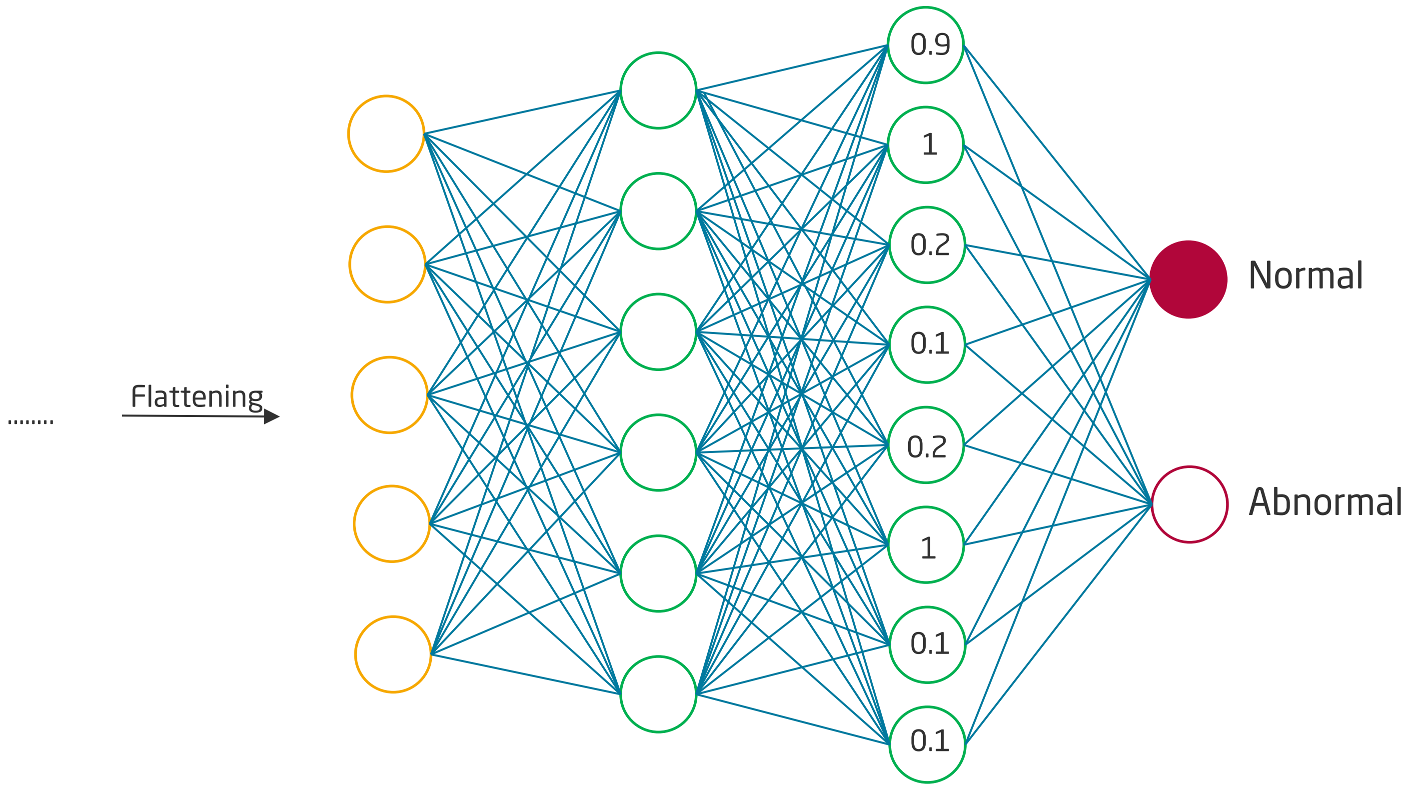
# CNN
# Fully Connected Layer



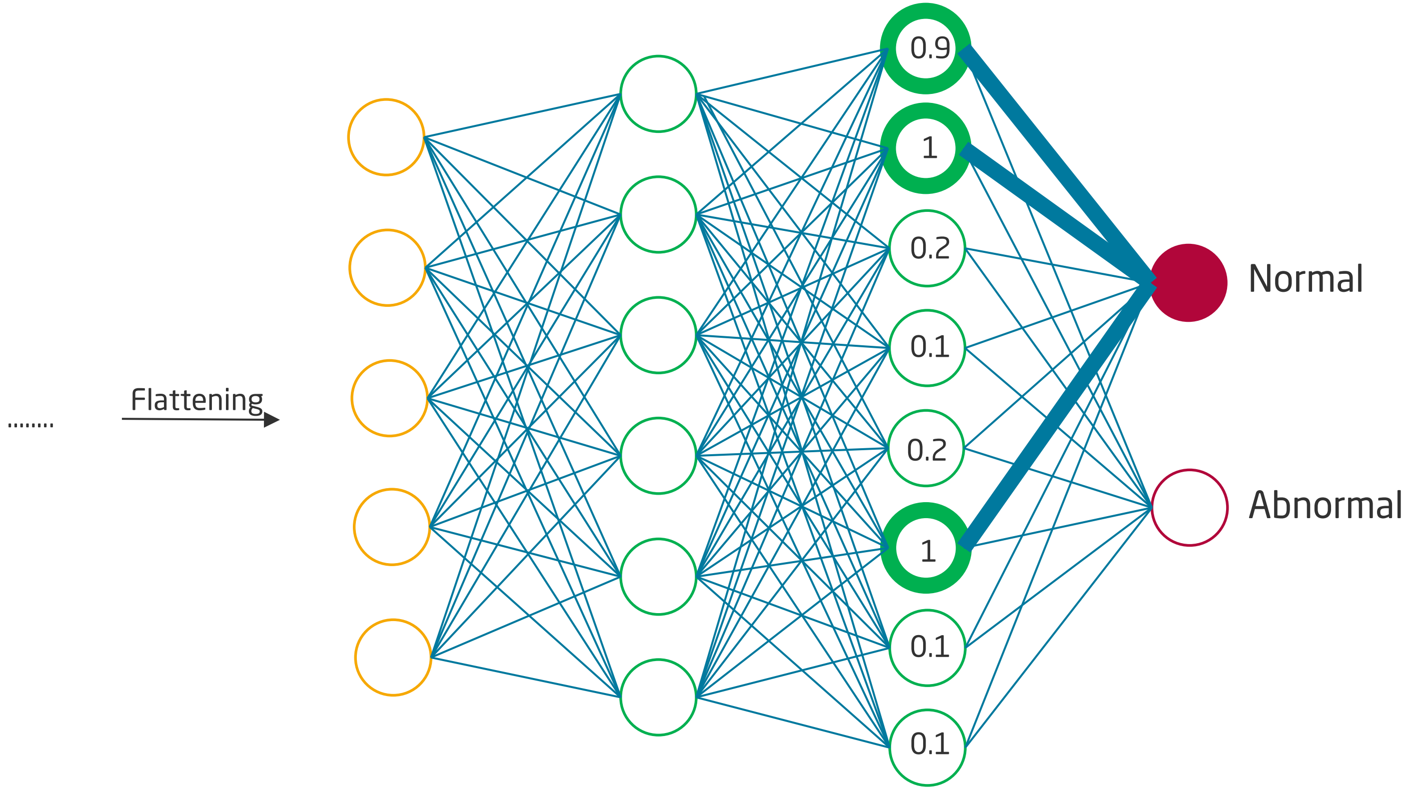
# CNN
# Fully Connected Layer



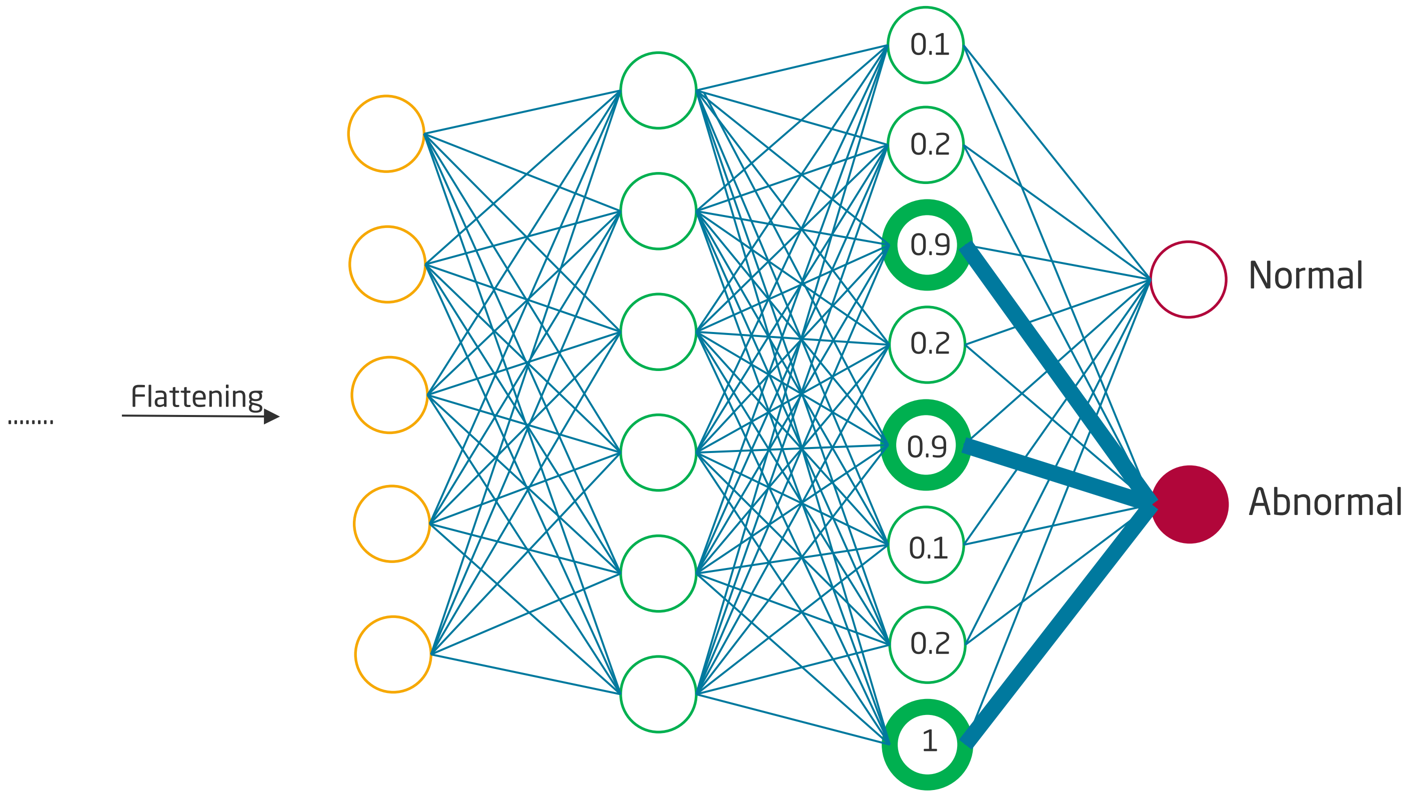
# CNN
# Fully Connected Layer



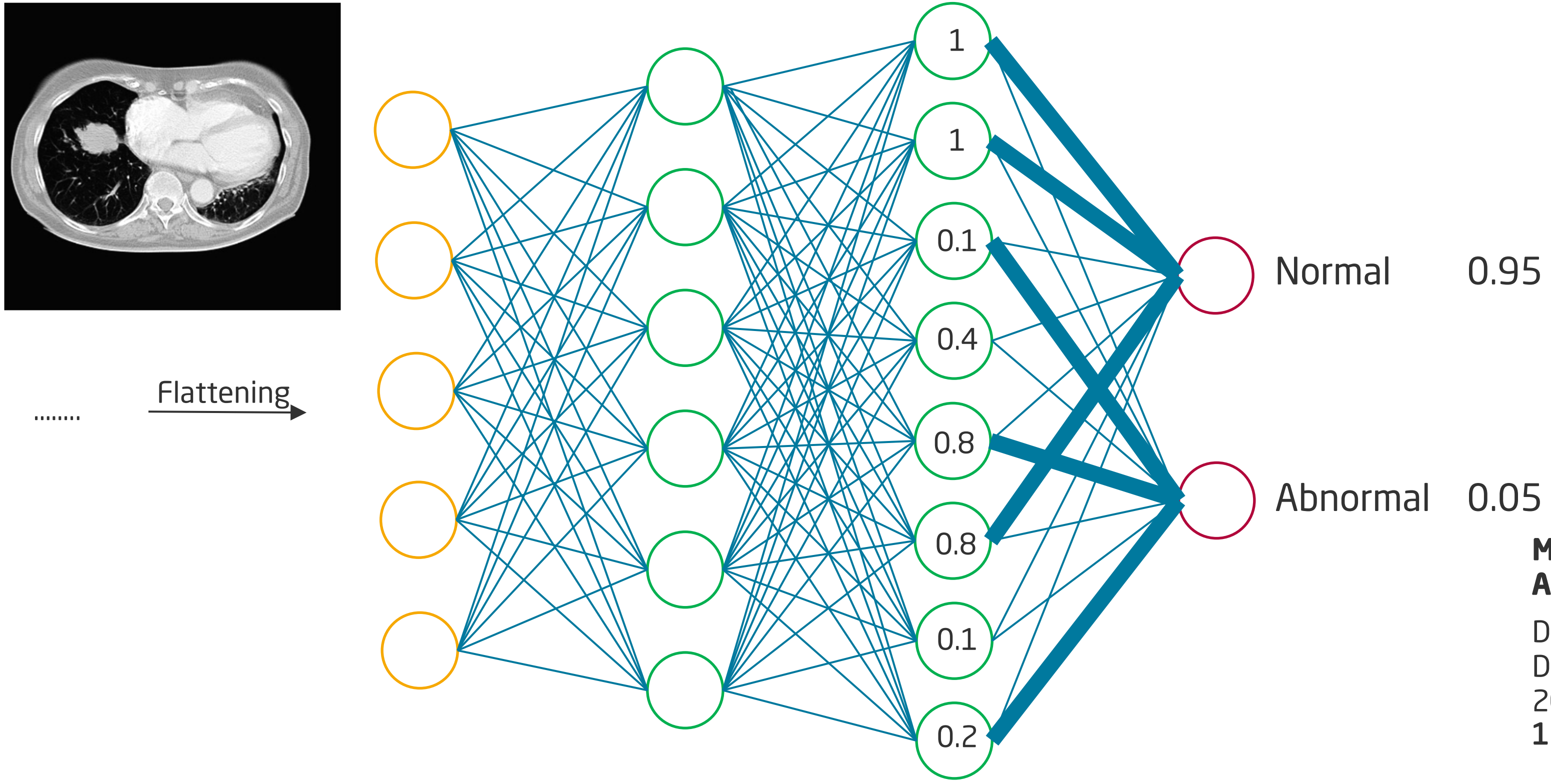
# CNN
# Fully Connected Layer



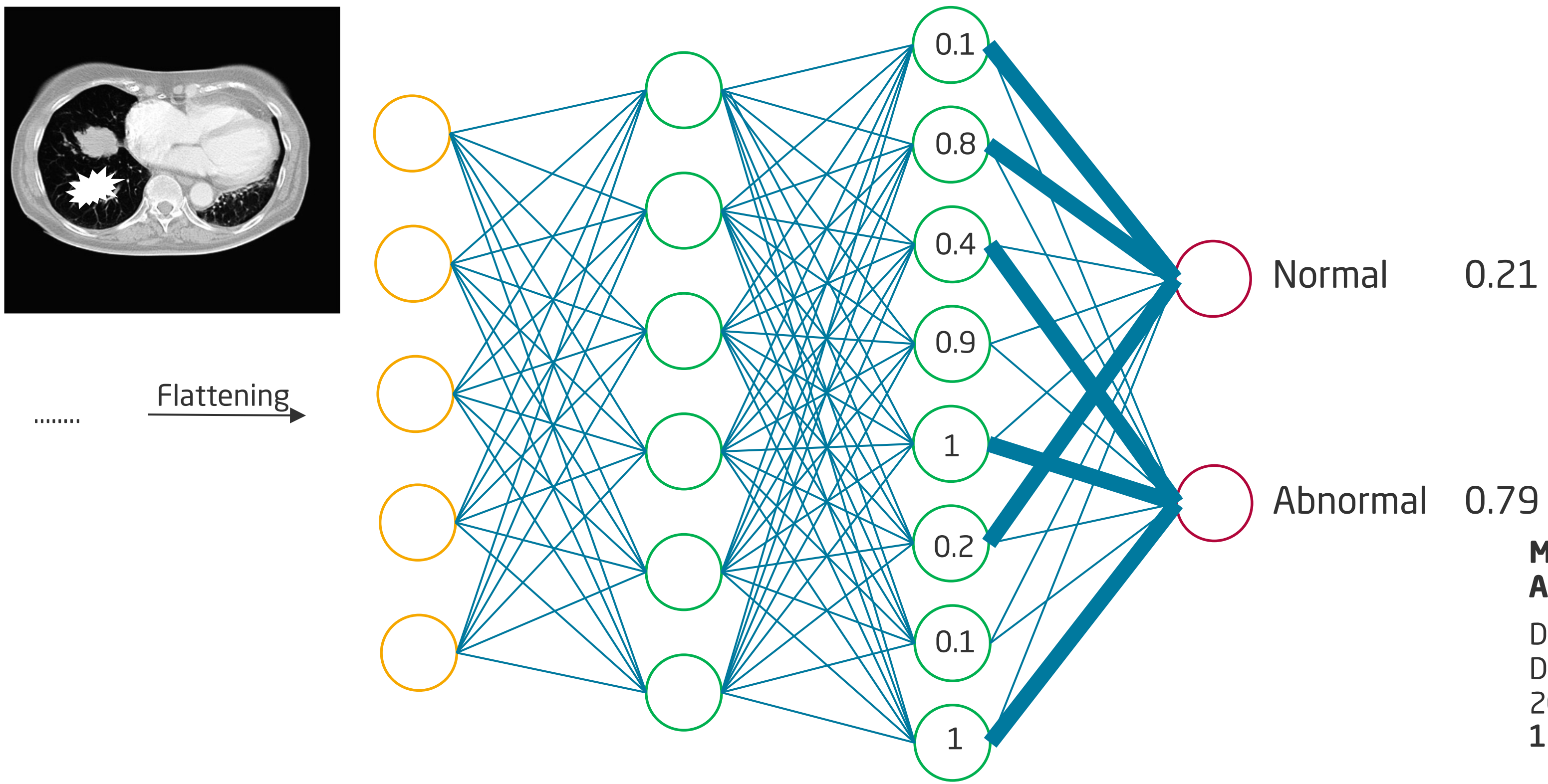
# CNN
# Fully Connected Layer



# CNN
# Fully Connected Layer

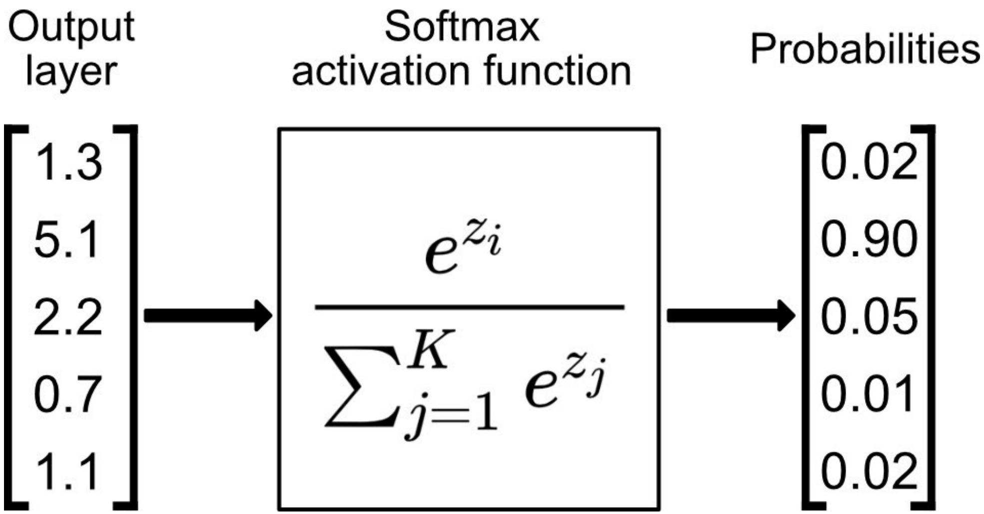


# CNN
## Fully Connected Layer



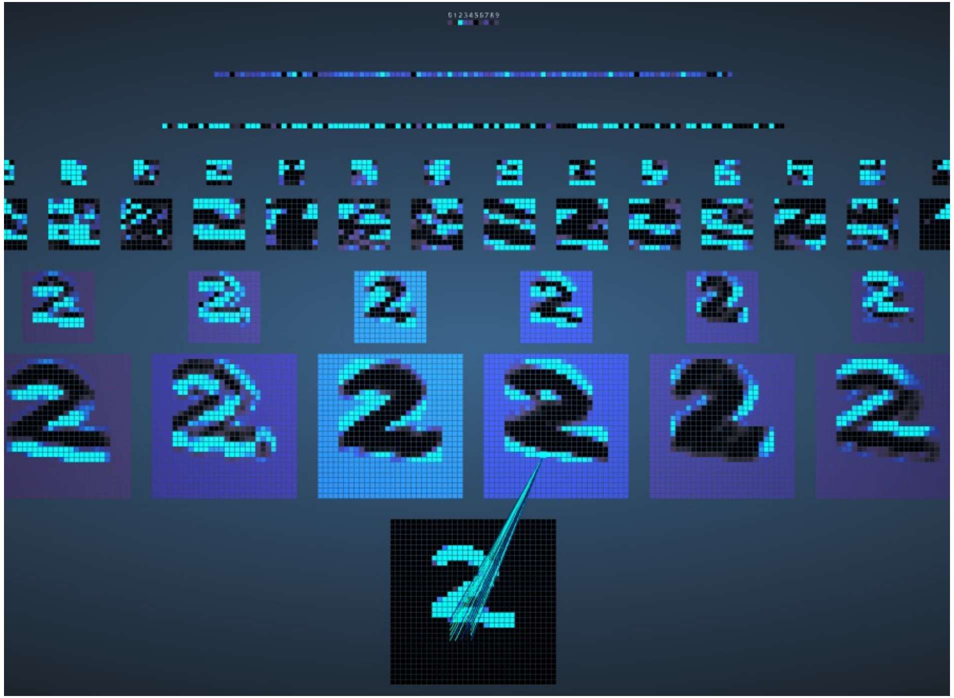
# CNN
# Softmax

- Used as the last activation function of a neural network to normalize the output of a network to a probability distribution over predicted output classes

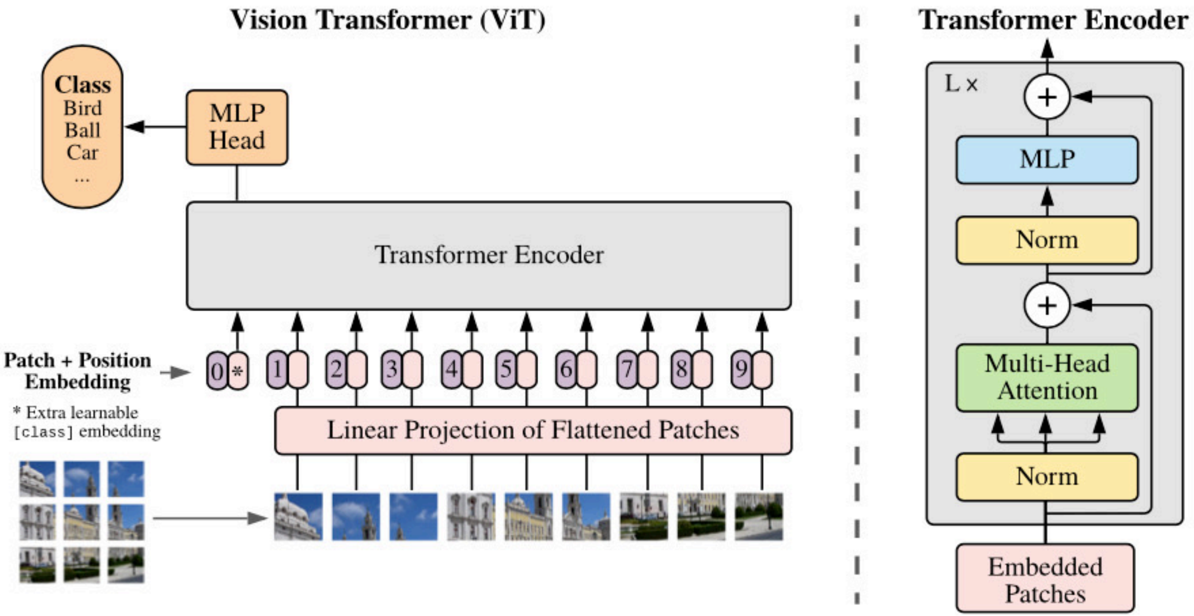


# CNN Visualization

- https://adamharley.com/nn_vis/cnn/2d.html

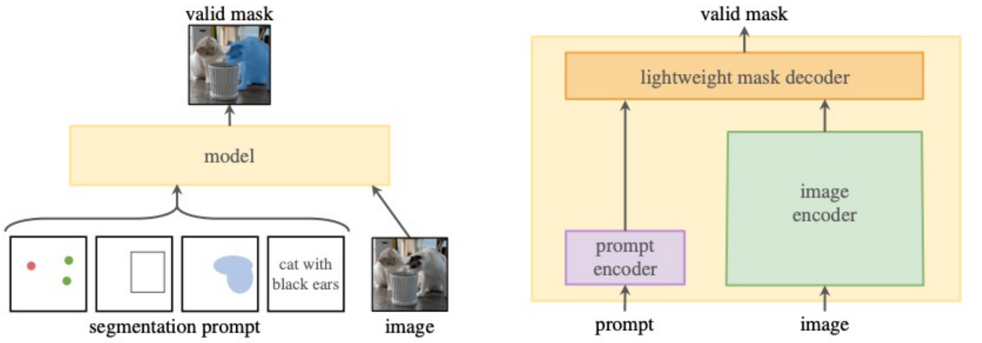


# Vision Transformers (Google, 2021)



# “Prompting” Vision Models?
# segment-anything.com

- Open source “foundation model” for image segmentation
- „11M diverse, high-resolution, licensed, and privacy protecting images and 1.1B high-quality segmentation masks“



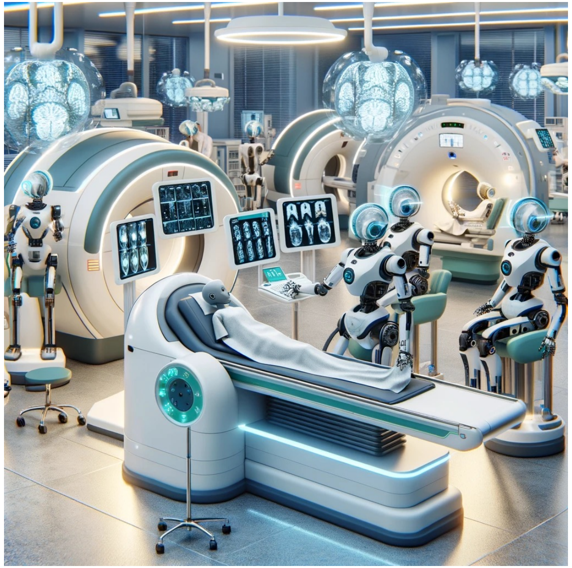
(a) **Task**: promptable segmentation (b) **Model**: Segment Anything Model (**SAM**)

Kirillov, Alexander, et al. "Segment anything." arXiv preprint arXiv:2304.02643 (2023).

# What to Take Home

Hasso Plattner Institut

- Source of medical images / modalities
- DICOM
- PACS
- Computer vision tasks
- Convolutions and CNN architecture



- No notebooks from us this time; if you are curious, there are plenty on the internet, e.g. https://www.kaggle.com/code/kmader/train-simple-xray-cnn

Generated by Dall-E 3