Introduction to the course MAST31401

"Inverse Problems 1: convolution and deconvolution"

Samuli Siltanen
Luca Ratti
Rashmi Murthy
Heli Virtanen
Salla Latva-Äijö
Vivi Hyttinen

University of Helsinki

September 3, 2019

From Effect to Cause. Robustly and Quickly.

2009: Professor, University of Helsinki, Finland

2006: Professor, Tampere University of Technology, Finland

2005: R&D scientist at Palodex Group

2004: R&D scientist at GE Healthcare

2002: Postdoc at Gunma University, Japan

2000: R&D scientist at Instrumentarium Imaging

1999: PhD, Helsinki University of Technology, Finland

Finnish Centre of Excellence in Inverse Modelling and Imaging

2018-2025

ACADEMY OF FINLAND

OULUN YLIOPISTO UNIVERSITY of OULU

UNIVERSITY OF EASTERN FINLAND

LUT University

UNIVERSITY OF JYVÄSKYLÄ

UNIVERSITY OF HELSINKI

TAMPERE UNIVERSITY OF TECHNOLOGY

FINNISH METEOROLOGICAL INSTITUTE

Aalto University

Outline

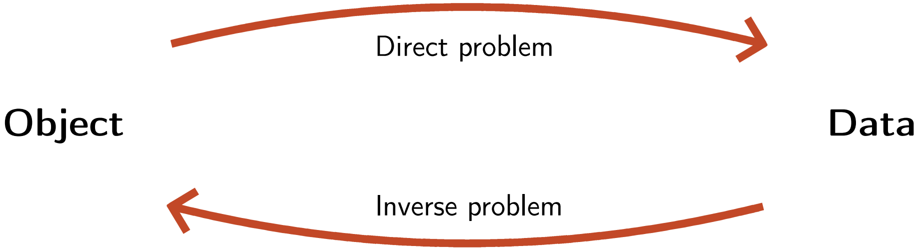
What are inverse problems?

Convolutions and machine learning

Blind deconvolution: Glottal Inverse Filtering

Inverse problems in industry

Direct problem: ***given object, determine data***
Inverse problem: ***given noisy data, recover object***

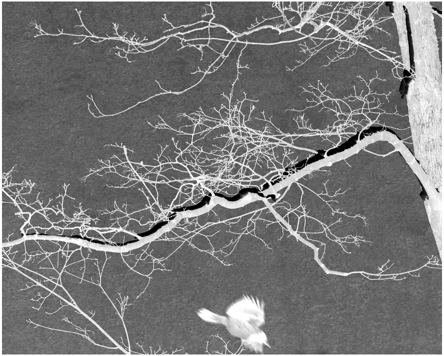


Direct problem: ***given object, determine data***

Inverse problem: ***given noisy data, recover object***

Object (positive photograph)

Data (negative photograph)



Forward map: subtraction from a constant

Direct problem: ***given object, determine data***

Inverse problem: ***given noisy data, recover object***

Object (sharp photograph)

Data (blurred and noisy photo)

Forward map: convolution operator

Direct problem: ***given object, determine data***

Inverse problem: ***given noisy data, recover object***

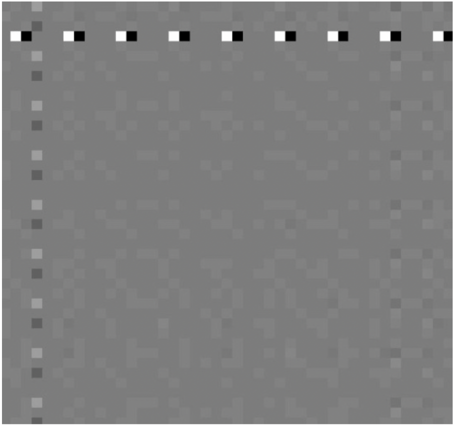
Object (sharp photograph)

Data (blurred and noisy photo)

Forward map: convolution operator

If we just invert the convolution matrix, the result is just numerical garbage

Naïve reconstruction



Data (blurred and noisy photo)

With appropriate regularization, the blurred image can be sharpened to some extent

Reconstruction

Data (blurred and noisy photo)

With appropriate regularization, the blurred image can be sharpened to some extent

Reconstruction (TV)

Data (blurred and noisy photo)

With appropriate regularization, the blurred image can be sharpened to some extent

Reconstruction (TGV)

Data (blurred and noisy photo)

Thanks to Professor Kristian Bredies for the TGV code

With appropriate regularization, the blurred image can be sharpened to some extent

Reconstruction (TGV)

Original

Thanks to Professor Kristian Bredies for the TGV code

The Hubble space telescope, launched in 1990, first gave blurred images due to a flawed mirror

Hubble telescope

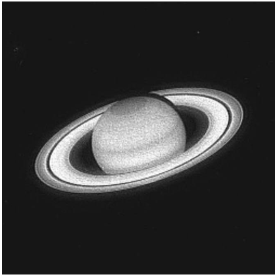
Saturnus (blurred)

Images: NASA, ESA, Quarktet

The mirror flaw was compensated by a deconvolution algorithm

Hubble telescope

Saturnus (blurred)



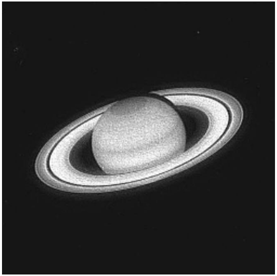
Saturnus (corrected)

Images: NASA, ESA, Quarktet

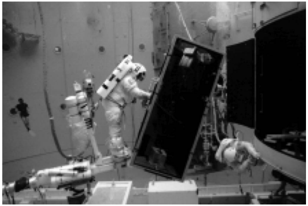
The mirror was replaced in 1993. The new sharp images are further enhanced with deconvolution!

Hubble telescope

Saturnus (blurred)



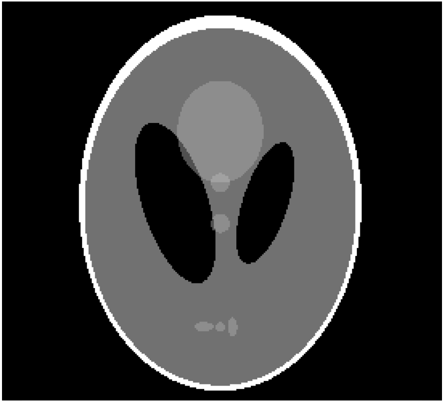
Saturnus (corrected)



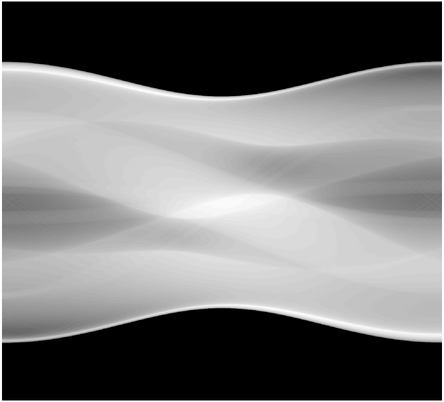
Images: NASA, ESA, Quarktet

Direct problem: *given object, determine data*
Inverse problem: *given noisy data, recover object*

Object (X-ray attenuation)



Data (sinogram)

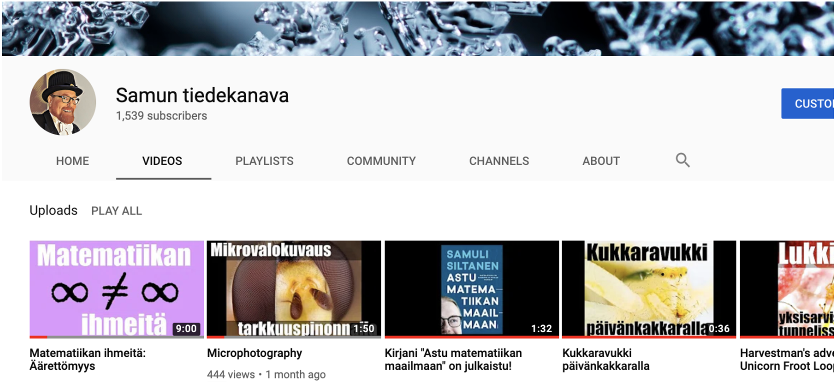


Forward map: discrete Radon transform

Show video!

https://youtu.be/QeXzSUcuX5g

Samun tiedekanava



Ill-posed inverse problems are defined as opposites of well-posed direct problems

Hadamard (1903): a problem is well-posed if the following conditions hold.

1. A solution exists,
2. The solution is unique,
3. The solution depends continuously on the input.

Well-posed direct problem:
Input x, find infinite-precision data $F(x)$.

Ill-posed inverse problem:
Input noisy data $m = F(x) + \varepsilon$, recover x.

The solution of an inverse problem is a ***set of instructions*** for recovering x stably from m

Those instructions need to be

(i) backed up by rigorous mathematical theory, and

(ii) implementable as an effective computational algorithm.

Ill-posed inverse problems are very sensitive to modelling errors and measurement noise. Therefore, the solution needs *a priori* information about the unknown in addition to measurement data.

In this course we incorporate such *a priori* information using regularization. Also, we talk a bit about machine learning in the last week of the course.

Outline

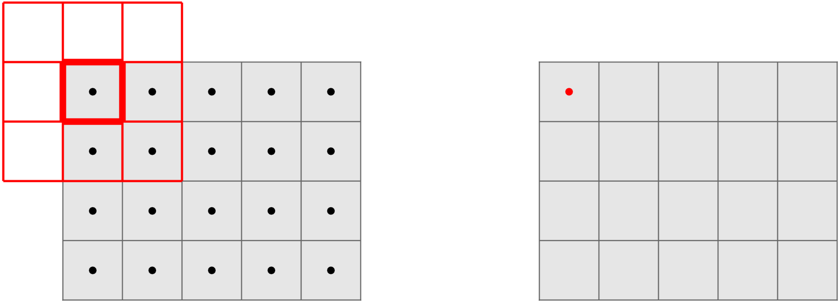
Direct problem: ***given object, determine data***
Inverse problem: ***given noisy data, recover object***

Object (sharp photograph)

Data (blurred and noisy photo)

Forward map: convolution operator

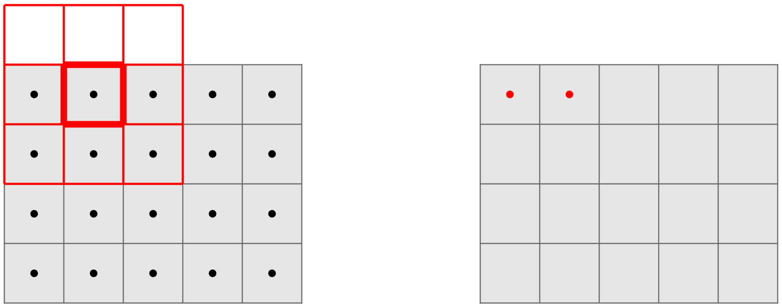
Two-dimensional convolution step-by-step



Original image and
convolution kernel

Result of convolution

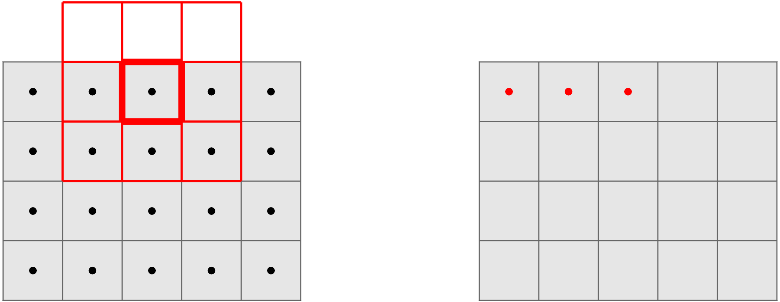
Two-dimensional convolution step-by-step



Original image and
convolution kernel

Result of convolution

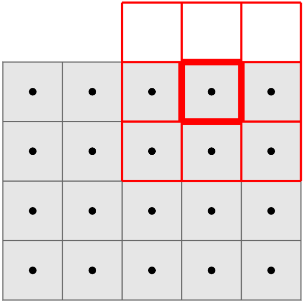
Two-dimensional convolution step-by-step



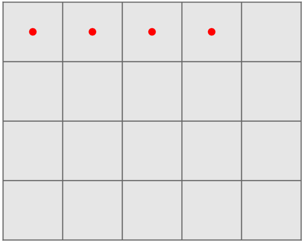
Original image and
convolution kernel

Result of convolution

Two-dimensional convolution step-by-step

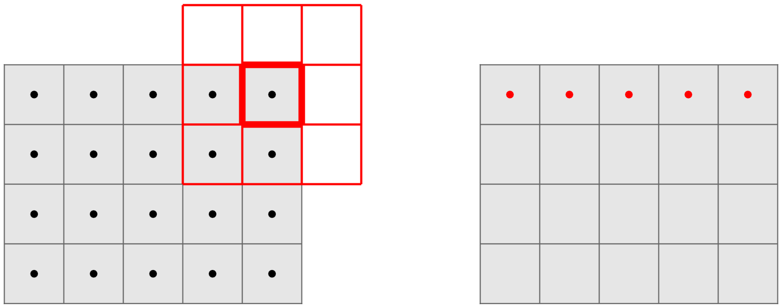


Original image and
convolution kernel



Result of convolution

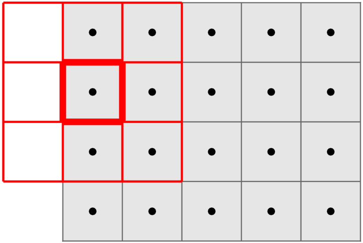
Two-dimensional convolution step-by-step



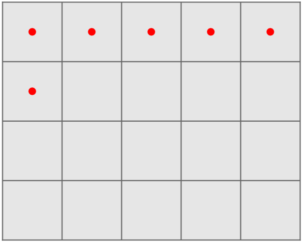
Original image and
convolution kernel

Result of convolution

Two-dimensional convolution step-by-step

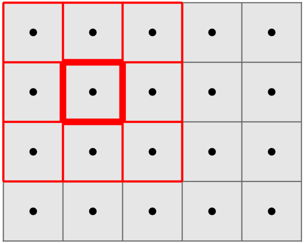


Original image and
convolution kernel

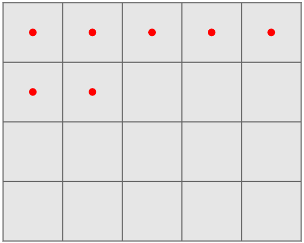


Result of convolution

Two-dimensional convolution step-by-step

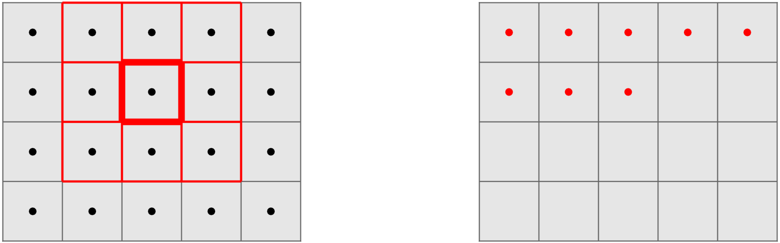


Original image and
convolution kernel



Result of convolution

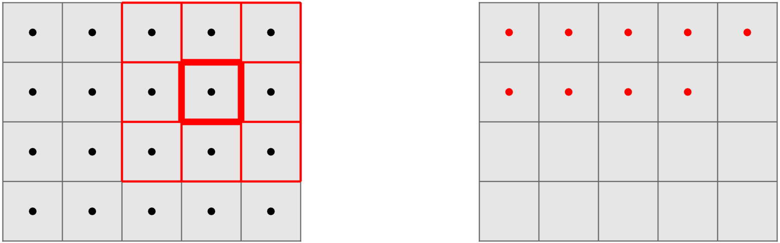
Two-dimensional convolution step-by-step



Original image and
convolution kernel

Result of convolution

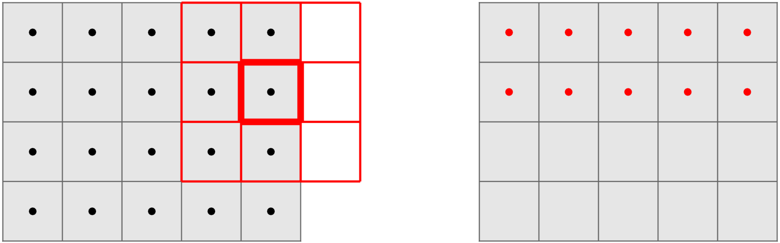
Two-dimensional convolution step-by-step



Original image and
convolution kernel

Result of convolution

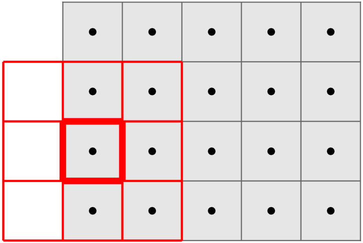
Two-dimensional convolution step-by-step



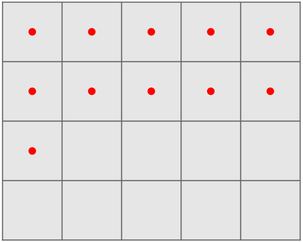
Original image and
convolution kernel

Result of convolution

Two-dimensional convolution step-by-step

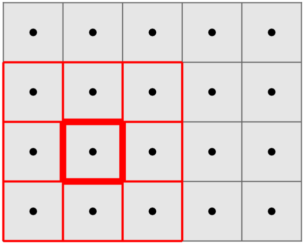


Original image and
convolution kernel

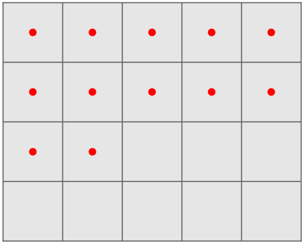


Result of convolution

Two-dimensional convolution step-by-step

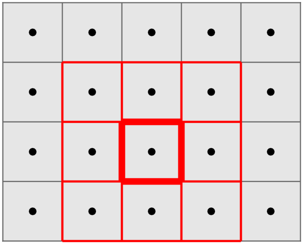


Original image and
convolution kernel

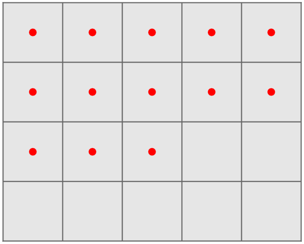


Result of convolution

Two-dimensional convolution step-by-step

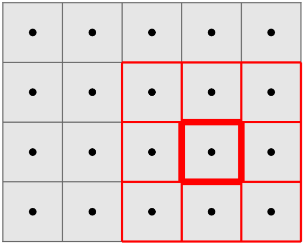


Original image and
convolution kernel

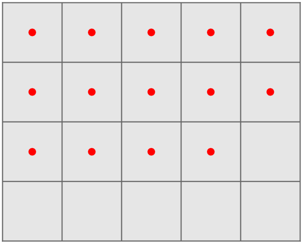


Result of convolution

Two-dimensional convolution step-by-step

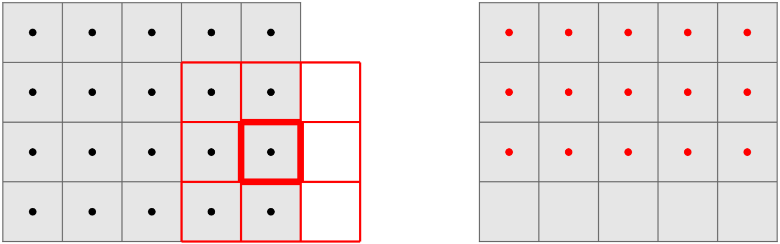


Original image and
convolution kernel



Result of convolution

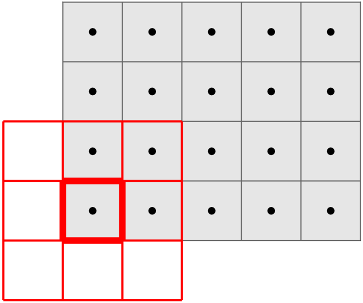
Two-dimensional convolution step-by-step



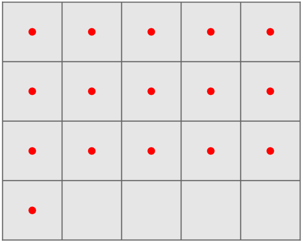
Original image and
convolution kernel

Result of convolution

Two-dimensional convolution step-by-step

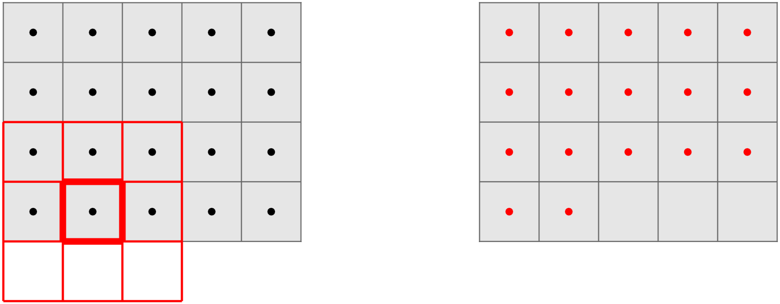


Original image and
convolution kernel



Result of convolution

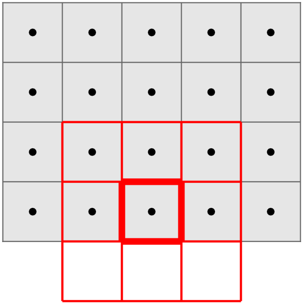
Two-dimensional convolution step-by-step



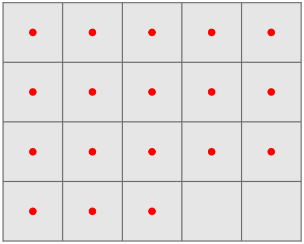
Original image and
convolution kernel

Result of convolution

Two-dimensional convolution step-by-step

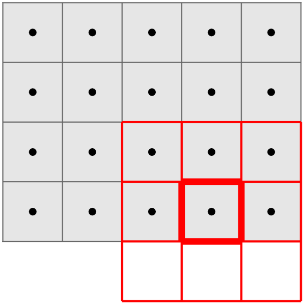


Original image and
convolution kernel

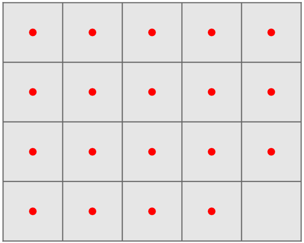


Result of convolution

Two-dimensional convolution step-by-step

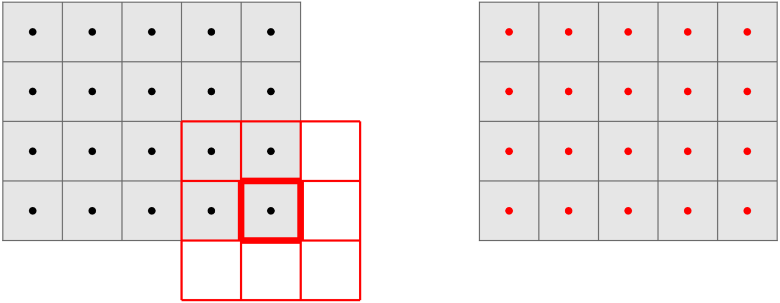


Original image and
convolution kernel



Result of convolution

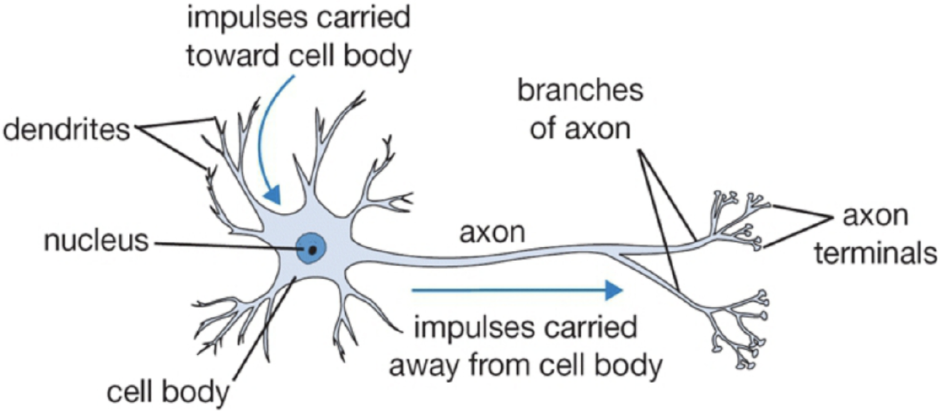
Two-dimensional convolution step-by-step



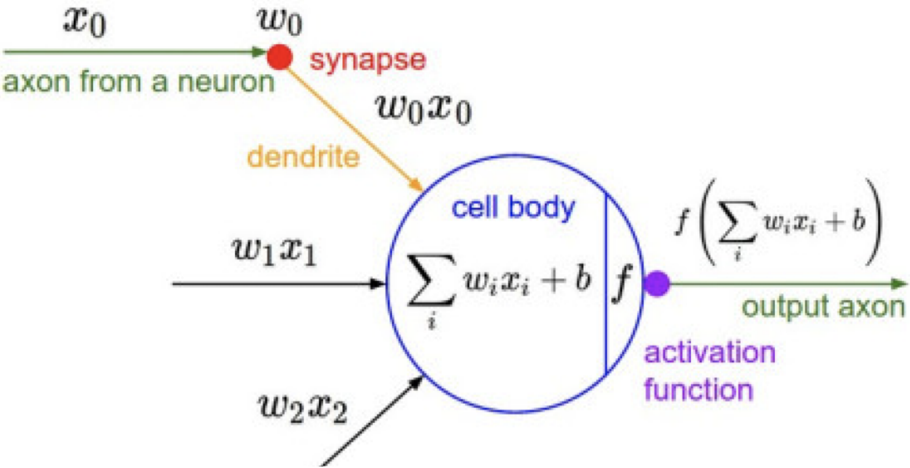
Original image and
convolution kernel

Result of convolution

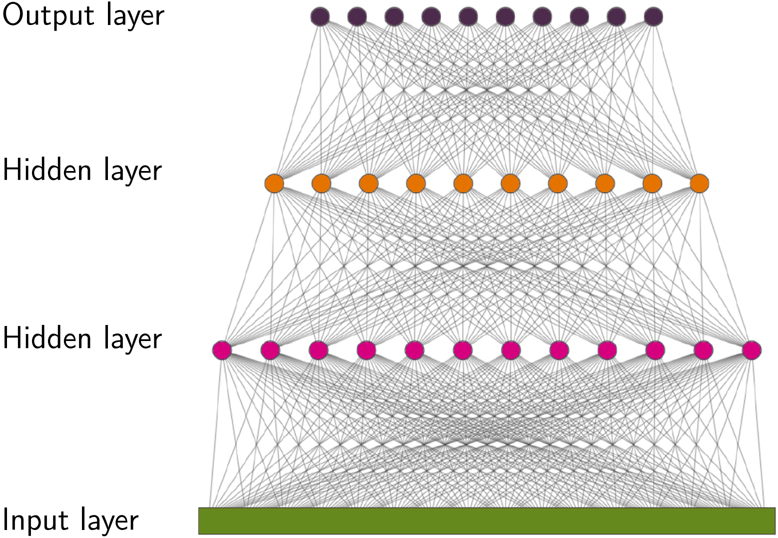
Neural network models are based on physiology of nerve cells



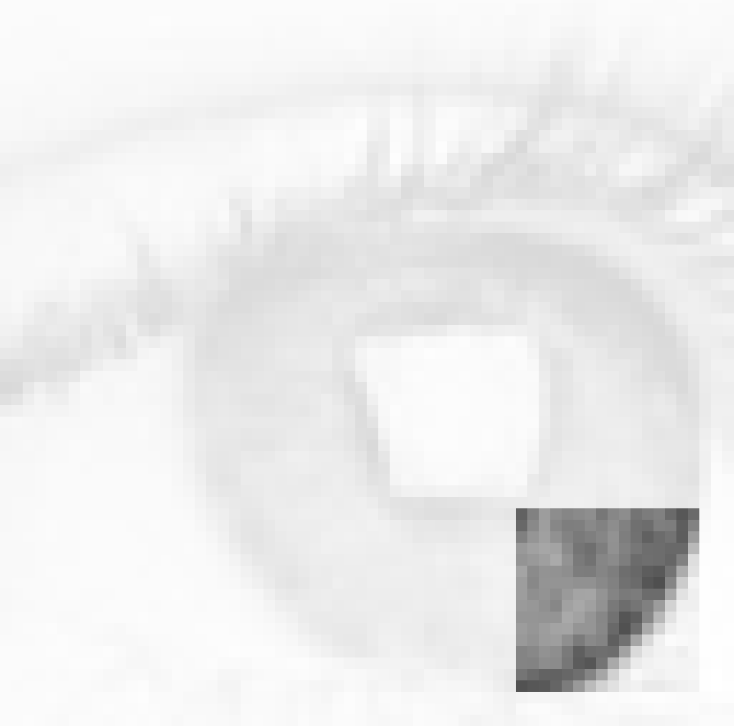
Neuron model is a simple formula that takes in numbers and produces a weighted sum (plus bias)

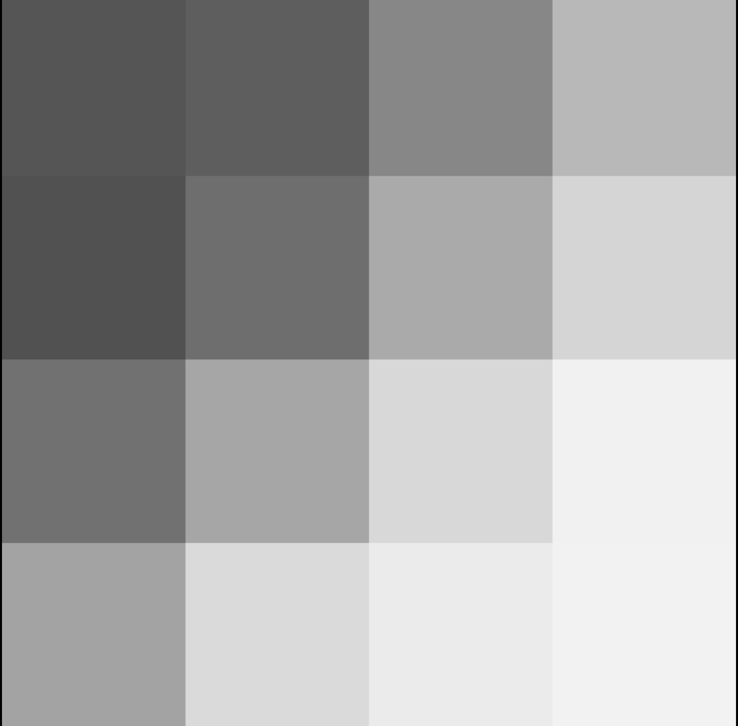


A traditional approach from the 1990's is to organize neurons into fully connected layers

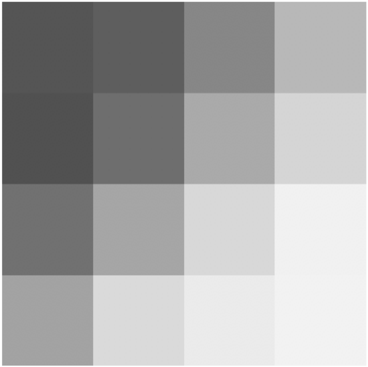


Artificial intelligence related to images benefits from convolutional neural networks





Gray colors are represented as numbers.
Here black is 0 and white is 255



85	94	135	184
81	110	170	213
113	166	216	241
163	218	235	242

This grayscale image has a vertical boundary. How does a convolutional network detect it?

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

6x6

A filter is needed for detecting vertical boundaries. We consider filters of size 3×3

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

6x6

*

1	0	-1
1	0	-1
1	0	-1

3x3 filter

*

Convolution will produce an image of size $4{\times}4$ when we take the borders into account

$$\underbrace{\begin{bmatrix} 10 & 10 & 10 & 0 & 0 & 0 \\ 10 & 10 & 10 & 0 & 0 & 0 \\ 10 & 10 & 10 & 0 & 0 & 0 \\ 10 & 10 & 10 & 0 & 0 & 0 \\ 10 & 10 & 10 & 0 & 0 & 0 \\ 10 & 10 & 10 & 0 & 0 & 0 \end{bmatrix}}_{\text{6x6}} * \underbrace{\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}}_{\text{3x3 filter}} = \underbrace{\begin{bmatrix} \; & \; & \; & \; \\ \; & \; & \; & \; \\ \; & \; & \; & \; \\ \; & \; & \; & \; \end{bmatrix}}_{\text{4x4}}$$

Let's compute the value of the corner pixel in the convolution image. We get zero.

10^{1}	10^{0}	10^{-1}	0	0	0
10^{1}	10^{0}	10^{-1}	0	0	0
10^{1}	10^{0}	10^{-1}	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

6x6

*

1	0	-1
1	0	-1
1	0	-1

3x3 filter

=

0			

4x4

Let's calculate the value of the next pixel in the convolution image. We get 30.

10	10^{1}	10^{0}	0^{-1}	0	0
10	10^{1}	10^{0}	0^{-1}	0	0
10	10^{1}	10^{0}	0^{-1}	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

6x6

*

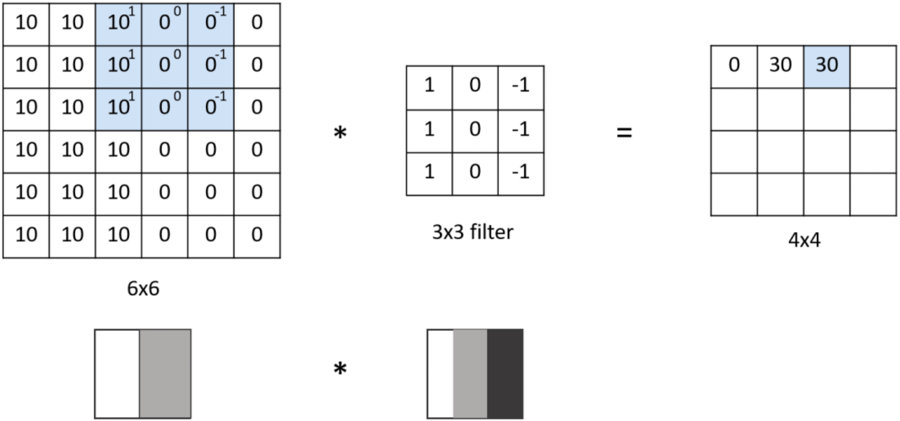
1	0	-1
1	0	-1
1	0	-1

3x3 filter

=

0	30		

4x4


10 10 10 0 0 0
10 10 10 0 0 0
10 10 10 0 0 0
10 10 10 0 0 0
10 10 10 0 0 0
10 10 10 0 0 0
6x6
*
1 0 -1
1 0 -1
1 0 -1
3x3 filter
=
0 30 30
4x4
*

10	10	10	0^{1}	0^{0}	0^{-1}
10	10	10	0^{1}	0^{0}	0^{-1}
10	10	10	0^{1}	0^{0}	0^{-1}
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

6x6

*

1	0	-1
1	0	-1
1	0	-1

3x3 filter

=

0	30	30	0

4x4

*

10	10	10	0	0	0
10^{1}	10^{0}	10^{-1}	0	0	0
10^{1}	10^{0}	10^{-1}	0	0	0
10^{1}	10^{0}	10^{-1}	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

6x6

*

1	0	-1
1	0	-1
1	0	-1

3x3 filter

=

0	30	30	0
0			

4x4

*

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0^{1}	0^{0}	0^{-1}
10	10	10	0^{1}	0^{0}	0^{-1}
10	10	10	0^{1}	0^{0}	0^{-1}

6x6

*

1	0	-1
1	0	-1
1	0	-1

3x3 filter

=

0	30	30	0
0	30	30	0
0	30	30	0
0	30	30	0

4x4

An important part of the network's function is to "learn" a set of filters that fit to the training data

$$
\underbrace{\begin{bmatrix} 10 & 10 & 10 & 0 & 0 & 0 \\ 10 & 10 & 10 & 0 & 0 & 0 \\ 10 & 10 & 10 & 0 & 0 & 0 \\ 10 & 10 & 10 & 0 & 0 & 0 \\ 10 & 10 & 10 & 0 & 0 & 0 \\ 10 & 10 & 10 & 0 & 0 & 0 \end{bmatrix}}_{\text{6x6}} * \underbrace{\begin{bmatrix} w_1 & w_2 & w_3 \\ w_4 & w_5 & w_6 \\ w_7 & w_8 & w_9 \end{bmatrix}}_{\text{3x3 filter}} = \underbrace{\begin{bmatrix} \; & \; & \; & \; \\ \; & \; & \; & \; \\ \; & \; & \; & \; \\ \; & \; & \; & \; \end{bmatrix}}_{\text{4x4}}
$$

Pooling layers collect and summarize localized information in a flexible way

0

0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0

Pooling layers collect and summarize localized information in a flexible way

0	0

0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0

Pooling layers collect and summarize localized information in a flexible way

0 0 30

0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0

Pooling layers collect and summarize localized information in a flexible way

0 0 30 30

0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0

Pooling layers collect and summarize localized information in a flexible way

0 0 30 30 30

0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0

Pooling layers collect and summarize localized information in a flexible way

0 0 30 30 30 0

0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0

Pooling layers collect and summarize localized information in a flexible way

	0	0	30	30	30	0	0	
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0

Pooling layers collect and summarize localized information in a flexible way

0	0	30	**30**	30	0	**0**

0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0
0	0	0	0	30	0	0	0	0

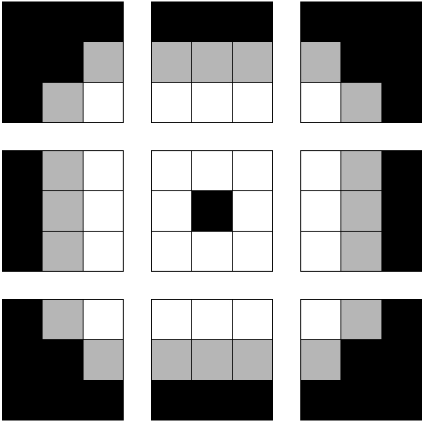
Pooling layers collect and summarize localized information in a flexible way

0	0	0	**30**	30	30	**0**

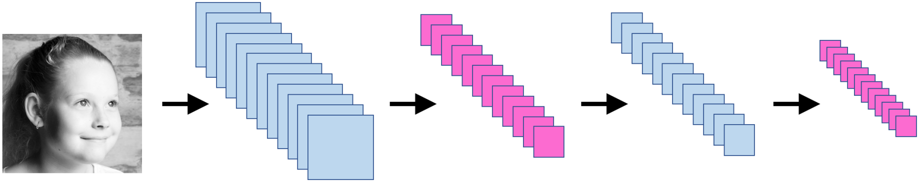
0	0	0	0	0	30	0	0	0
0	0	0	0	0	30	0	0	0
0	0	0	0	0	30	0	0	0
0	0	0	0	0	30	0	0	0
0	0	0	0	0	30	0	0	0

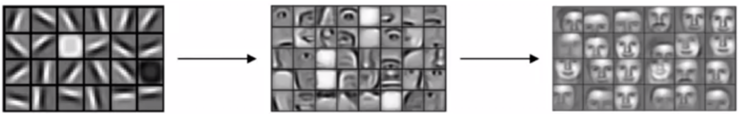
Images of full faces and parts of faces

Representing an eye with $3{\times}3$ patches can be done with a $3{\times}3$ arrangement of patches

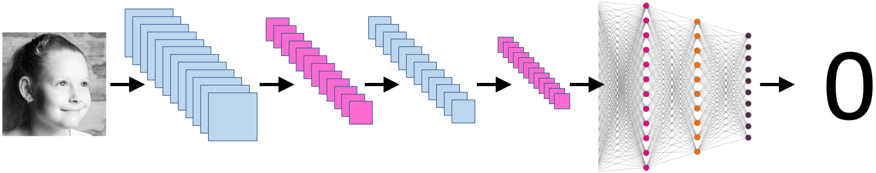


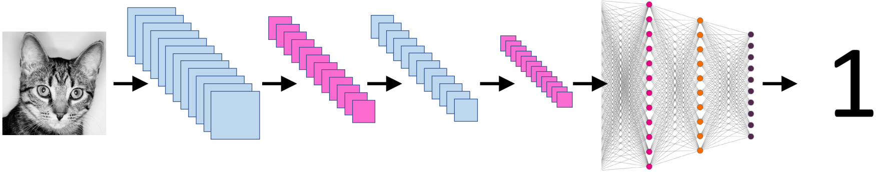
Stacked convolution layers and pooling layers: pieces of the puzzle and how they fit together



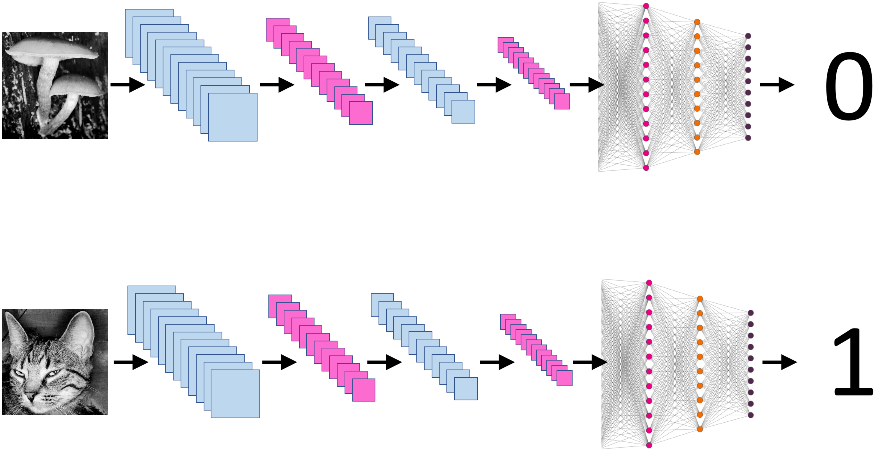


How does a convolutional neural network learn?
It needs a training data set and optimization.



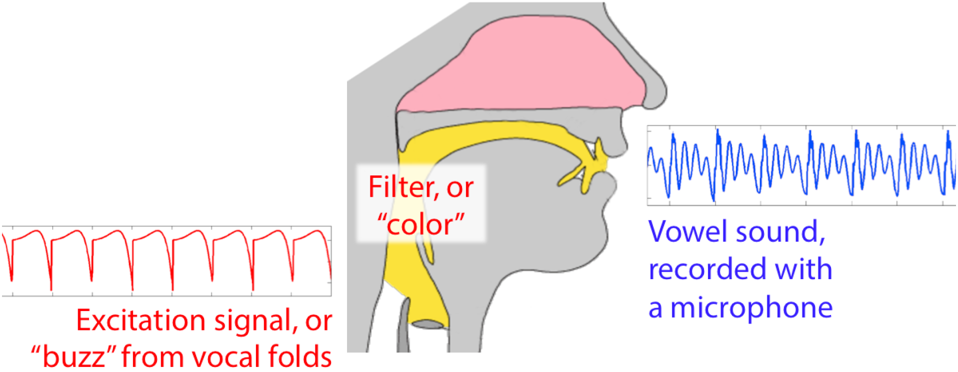


How does a convolutional neural network learn? It needs a training data set and optimization.



Outline

A vowel sound consists of two structural parts: excitation and vocal tract filter



Direct problem: *given object, determine data*
Inverse problem: *given noisy data, recover object*

In glottal inverse filtering (GIF), the data is a vowel sound recorded using a microphone. The aim is to reconstruct the excitation signal and the filter.

GIF has important applications in

- Computer-generated speech (think Stephen Hawking),
- Speech recognition (think Apple's Siri).

Forward map: bilinear convolution operator

The main application of GIF is synthetic speech

Here are examples of low-quality and high-quality speech signals generated by a computer (it is not a person speaking).

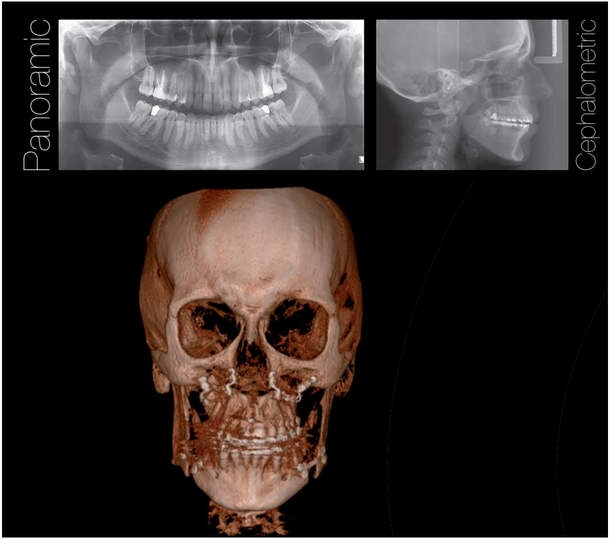
Sample 1, low quality: high quality:

Sample 2, low quality: high quality:

The high-quality samples, developed in **Raitio, Suni, Yamagishi, Pulakka, Nurminen, Vainio** & **Alku** (2010), are based on a Hidden Markov Model (HMM).

Outline

Dental X-ray imaging: www.palodexgroup.com, www.planmeca.com, ajat.fi



Medical technology: www.varian.com

Edge™ Radiosurgery System

Using technology designed for radiosurgical ablation, the Edge™ radiosurgery system represents an evolution in the way advanced radiosurgery is delivered.

TrueBeam™ Radiotherapy System

TrueBeam™ system brings leading edge cancer care to communities by positioning clinics at the forefront in the fight against cancer.

VitalBeam™ Radiotherapy System

The VitalBeam™ radiotherapy system is an advanced option for clinics that want sophisticated functionality on a scalable platform.

VARIAN
medical systems

Environmental monitoring: www.vaisala.com

Vaisala Weather Radar WRM200

WRM200 is a dual polarization Doppler weather radar with real time operational hydrometeor classification software.

Vaisala Ceilometer CL51

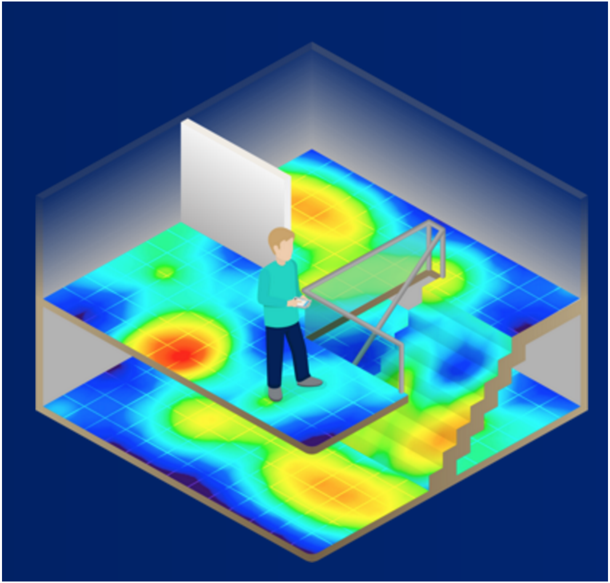
CL51 is designed to measure high range cirrus clouds without unsurpassing the low and middle layer clouds, or vertical visibility in harsh conditions. Cloud reporting range up to 13 km (43,000 ft) and backscatter profiling over full measurement range up to 15 km (49,200 ft). Advanced single-lens design provides excellent performance also at low altitudes.

Vaisala Remote Surface Condition Sensor DSC111

Provides an accurate reading of surface conditions, including water, ice and snow amounts and a high resolution value for grip/friction level. Pole mounted roadside set-up eliminates the disruption caused by slot-cutting, making installation and maintenance relatively easy. When used with a infrared temperature sensor it provides a complete view of the road conditions.

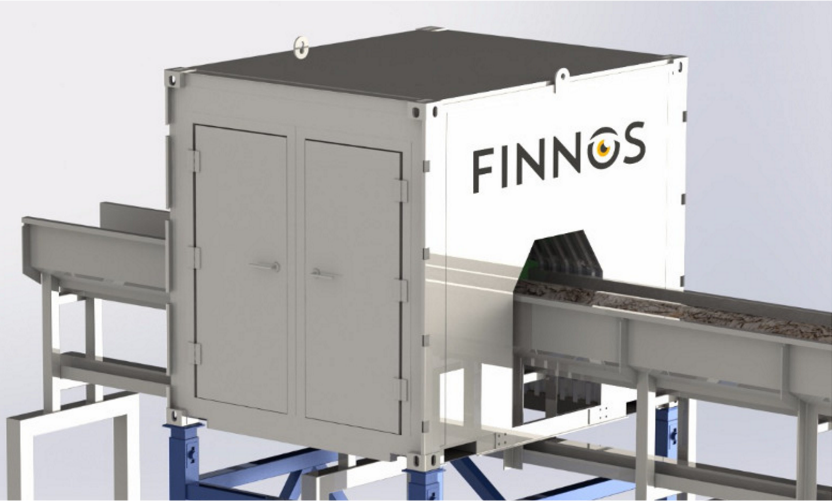
VAISALA

Navigation technology: www.indooratlas.com

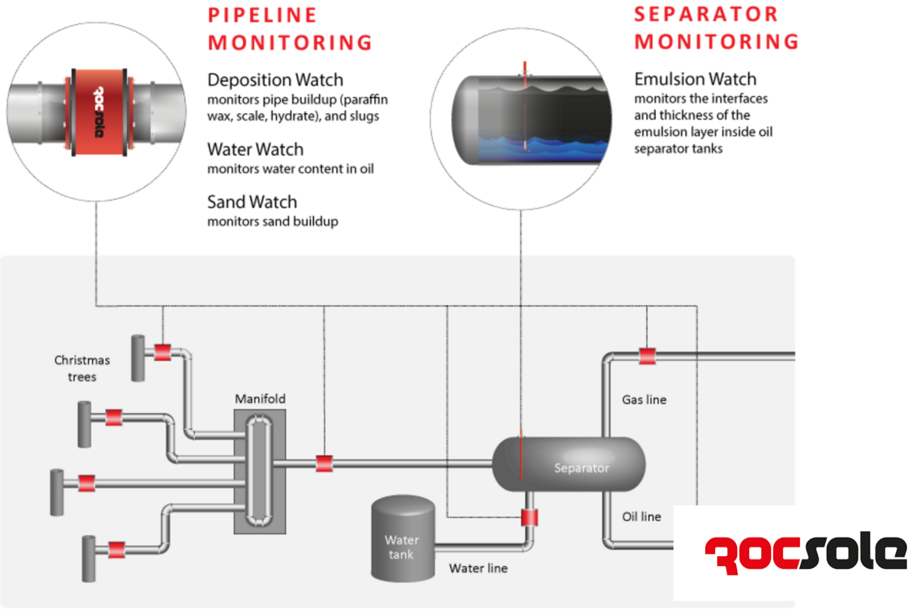


IndoorAtlas

Optimal log cutting: finnos.fi



Process pipeline monitoring: www.rocsole.com



Thank you for your attention!