AIMEDIC-08: AI, Med. Info., & Biomed. Eng. Congress, Coruña, Spain-Carleton, Canada-Standford, USA, 2021

DeepLearning-Based Computer Assisted Diagnosis systems (CAD) in Neuroscience

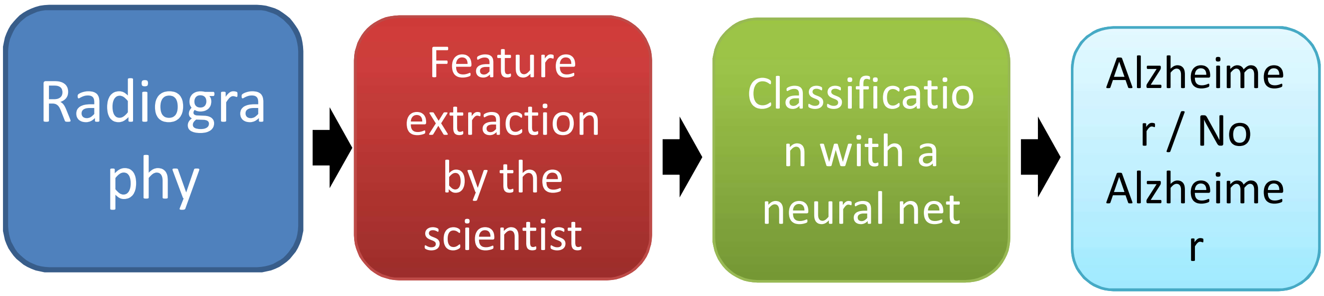
Cristian Robert Munteanu

[1]Computer Science Faculty, University of A Coruna, 15071 A Coruña, Spain.
[2]Centro de Investigación en Tecnologías de la Información y Las Comunicaciones (CITIC), Campus de Elviña s/n, 15071 A Coruña, Spain.
[3]Biomedical Research Institute of A Coruña (INIBIC), University Hospital Complex of A Coruña (CHUAC), 15006 A Coruña, Spain.
[4]IKERDATA S.L., ZITEK, University of Basque Country UPVEHU, Rectorate Building, 48940 Leioa, Spain.

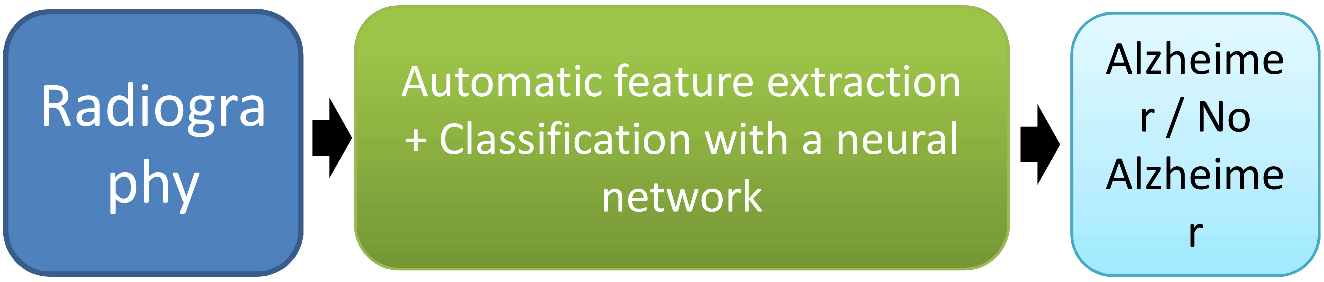
Abstract. *This is the slideshow presentation of talk by Prof. Cristian R. Munteanu presented as part of the NEURODAT'21 training program funded by IBRO-PERC Soft Skills Training call of the International Brain Research Organization (IBRO) and the Pan-Europe Regional Committee (PERC). NEURODAT'21 is devoted to promote soft skills on entry level medicine and also STEMS area students interested on neurosciences. The talk includes two parts, part 1 focuses on Deep Learning models introduction and part 2 focuses on applications to medical diagnosis in Neurosciences. In this second part the talk introduces concepts Computer Assisted Diagnosis systems (CAD), Convolution Neural Network, etc. and also presents a practical case focused on Alzheimer's disease diagnosis. Language note:* ***English-Spanish bi-lingual talk and English text****.*

Computer Assisted Diagnosis systems (CAD), using old-fashioned feature-engineered programmers and supervised learning to highlight abnormalities

Machine Learning



Deep Learning

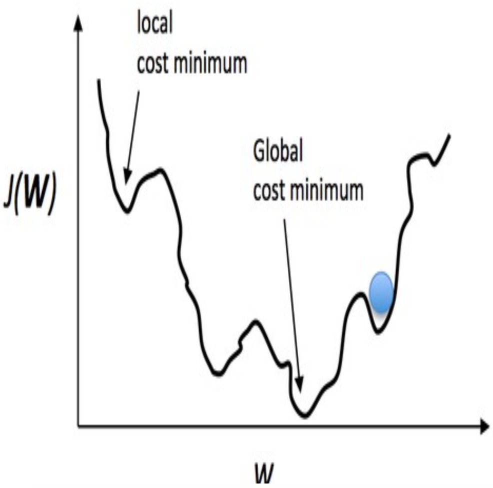


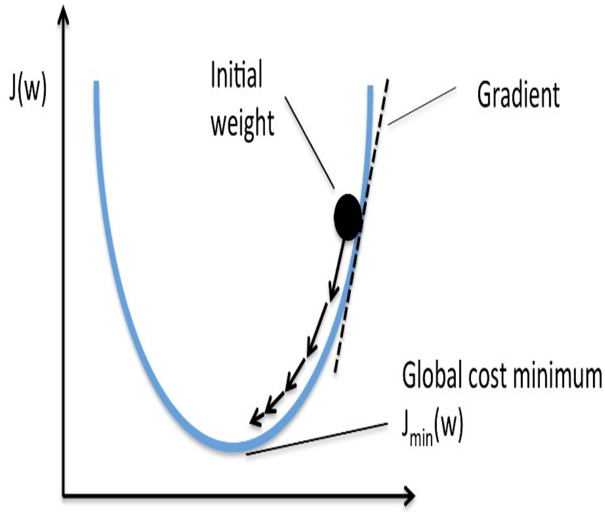
Deep Learning (DL/AI), without old-fashioned feature-engineered programmers, without any knowledge in the field

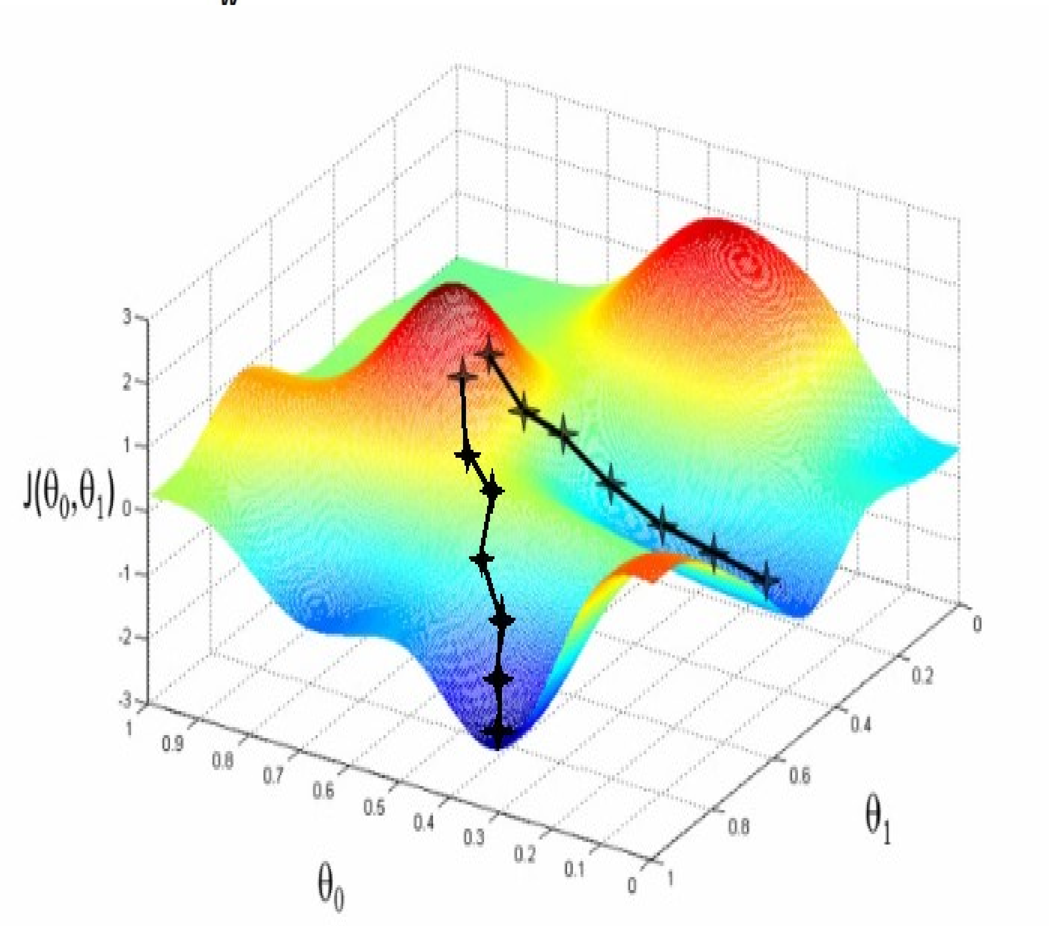
iKER DATA **Cristian Robert Munteanu**
IKERDATA
ai@ikerdata.com

Gradient Descent optimization = Training a network

Find the best weights to minimize the error between predicted and database class



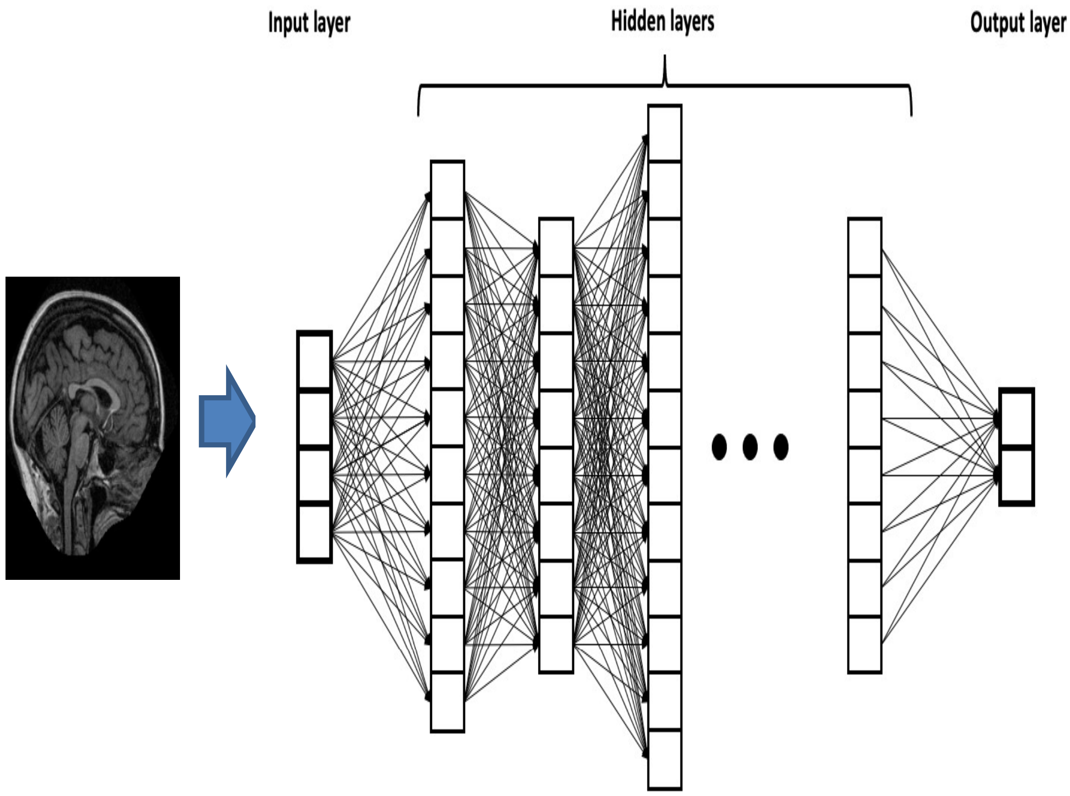




J = error between prediction and observation
w, θ = weights

Cristian Robert Munteanu
IKERDATA
ai@ikerdata.com

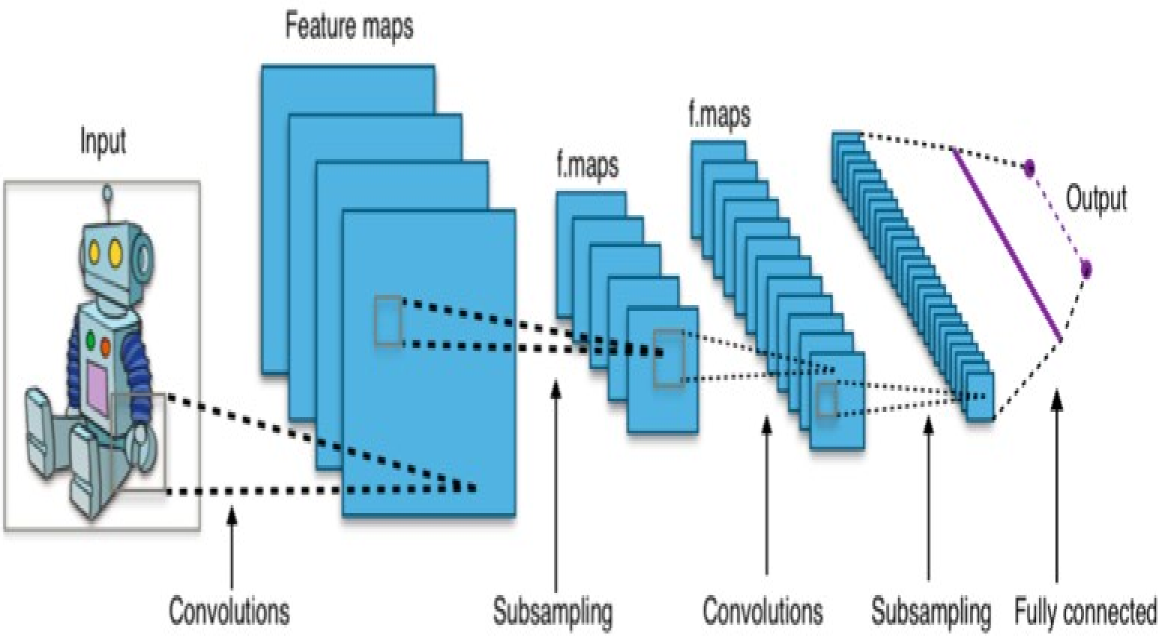
Fully-Connected Artificial Neural Network



Example: if the image has **1000 x 1000 pixels**, we need to use at least a hidden layer with **1 million neurons**! This will create 10^12 parameters (weights) to optimize!

- Extended training (spending more resources)
- Spatial correlation is local

CNN = Convolutional Neural Networks



Automatic feature extraction

Classificatio
n

- ✓ Sharing of features in different parts of the images
- ✓ Less parameters to optimize, faster training
- ✓ No need for domain knowledge for feature extraction

Cristian Robert Munteanu
IKERDATA
ai@ikerdata.com

Convolution = matrix element-wise multiplication & sum

Every image can be considered as a matrix of pixel values. Consider a 5 x 5 image whose pixel values are only 0 and 1 (note that for a grayscale image, pixel values range from 0 to 255, the green matrix below is a special case where pixel values are only 0 and 1):

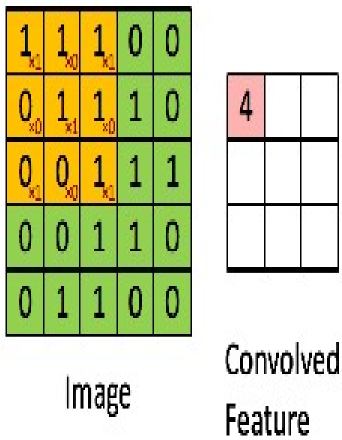
1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

Also, consider another 3 x 3 matrix as shown below:

1	0	1
0	1	0
1	0	1

Then, the Convolution of the 5 x 5 image and the 3 x 3 matrix can be computed as shown in the animation.

In CNN terminology, the 3×3 matrix is called a '**filter**' or '**kernel**' or '**feature detector**' and the matrix formed by sliding the filter over the image and computing the dot product is called the '**Convolved Feature**' or '**Activation Map**' or the '**Feature Map**'. It is important to note that **filters acts as feature detectors** from the original input image.

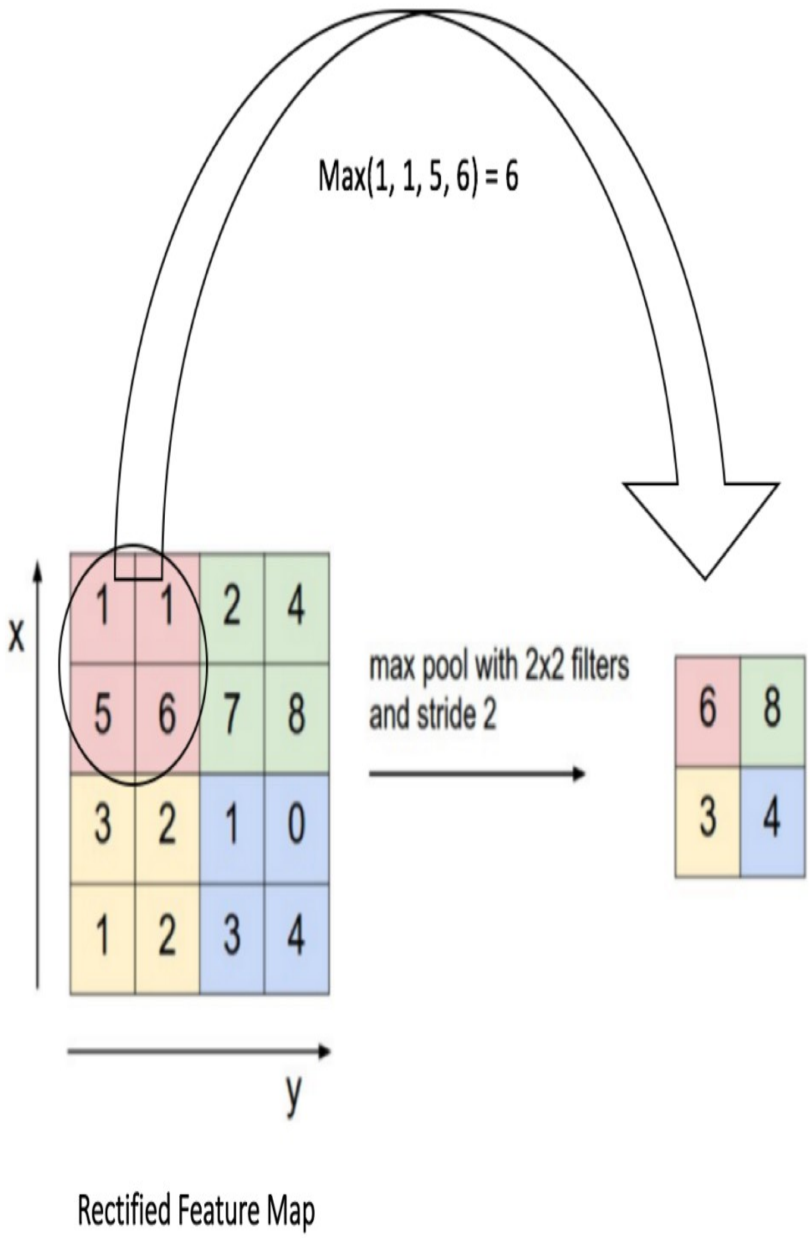


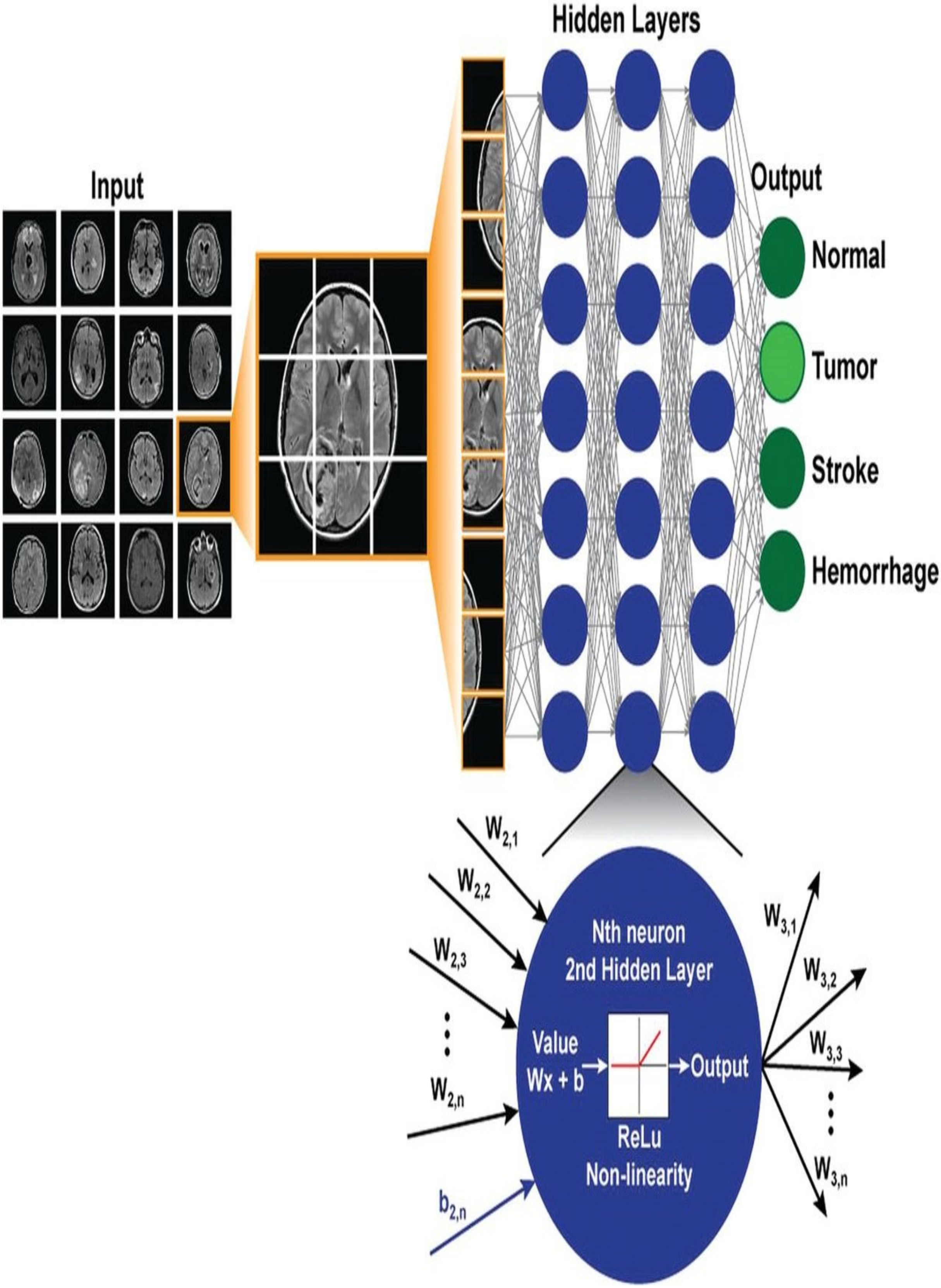
Cristian Robert Munteanu
IKERDATA
ai@ikerdata.com

The Pooling Step

Spatial Pooling (also called subsampling or downsampling) reduces the dimensionality of each feature map but retains the most important information. Spatial Pooling can be of different types: **Max, Average, Sum** etc.

In case of Max Pooling, we define a spatial neighborhood (for example, a 2×2 window) and take the largest element from the rectified feature map within that window. In practice, Max Pooling has been shown to work better.




Hidden Layers
Input
Output
Normal
Tumor
Stroke
Hemorrhage
W2,1
W2,2
W2,3
W2,n
b2,n
Nth neuron
2nd Hidden Layer
Value
Wx + b
Output
ReLu
Non-linearity
W3,1
W3,2
W3,3
W3,n

Access through Universidade da Coruña | Purchase PDF | Access through another institution | Search ScienceDirect

ELSEVIER

Computers in Biology and Medicine

Volume 120, May 2020, 103764

Automatic assessment of Alzheimer's disease diagnosis based on deep learning techniques

Alejandro Puente-Castro [a], Enrique Fernandez-Blanco [a], Alejandro Pazos [a, b], Cristian R. Munteanu [a, b]

Show more

+ Add to Mendeley | Share | Cite

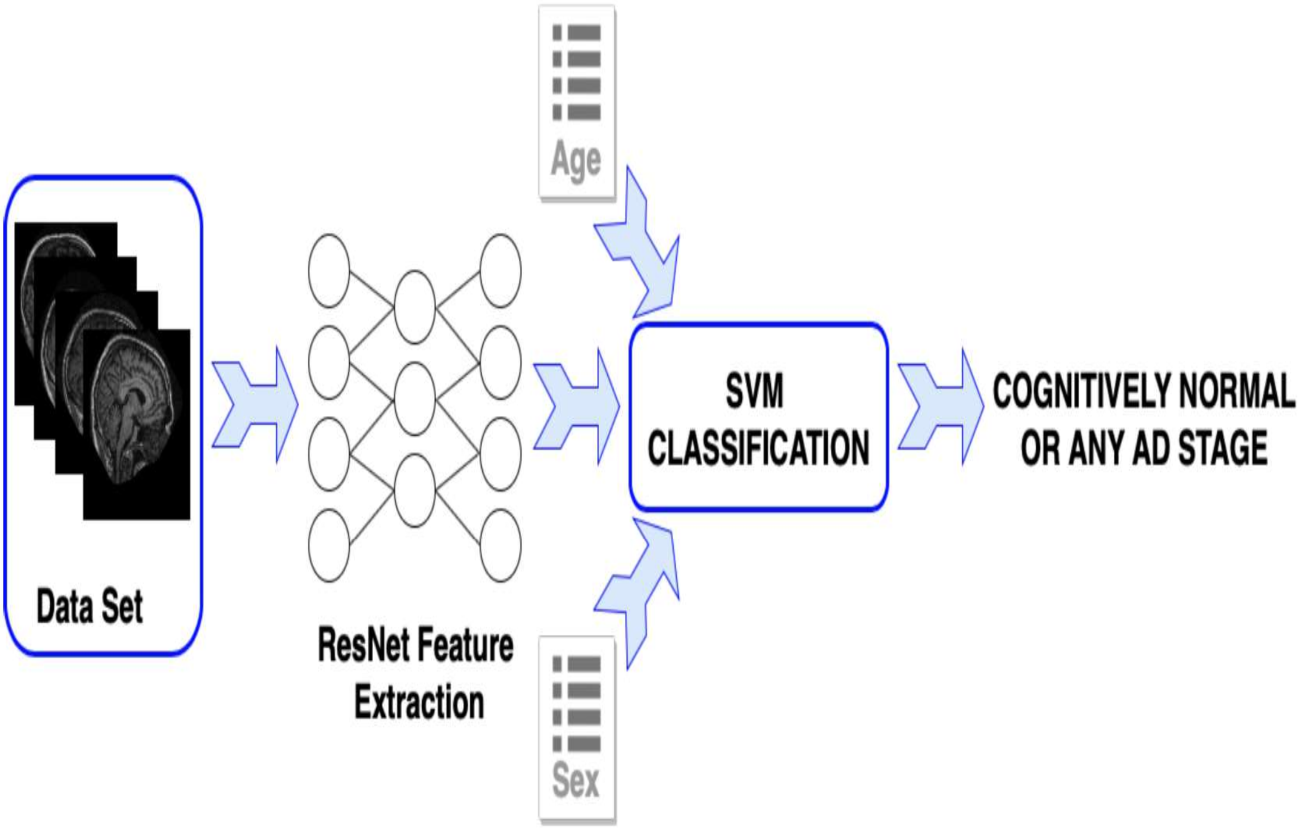
https://doi.org/10.1016/j.compbiomed.2020.103764

Get rights and content

Abstract

Early detection is crucial to prevent the progression of Alzheimer's disease (AD). Thus, specialists can begin preventive treatment as soon as possible. They demand fast and precise assessment in the diagnosis of AD in the earliest and hardest to detect stages. The main objective of this work is to develop a system that automatically detects the presence of the disease in sagittal magnetic resonance images (MRI), which are not generally used. Sagittal MRIs from ADNI and OASIS data sets were employed. Experiments were conducted using Transfer Learning (TL) techniques in order to achieve more accurate results. There are two main conclusions to be drawn from this work: first, the damages related to AD and its stages can be distinguished in sagittal MRI and, second, the results obtained using DL models with sagittal MRIs are similar to the state-of-the-art, which uses the horizontal-plane MRI. Although sagittal-plane MRIs are not commonly used, this work proved that they were, at least, as effective as MRI from other planes at identifying AD in early stages. This could pave the way for further research. Finally, one should bear in mind that in certain fields, obtaining the examples for a data set can be very expensive. This study proved that DL models could be built in these fields, whereas TL is an essential tool for completing the task with fewer examples.

AUTOMATIC ASSESSMENT OF ALZHEIMER'S DISEASE DIAGNOSIS BASED ON DEEP LEARNING TECHNIQUES



The workflow diagram of the study. Patients' MRI scans were fed to a ResNet ANN in order to extract new features vectors and sex and age are concatenated to them. These vectors are separated into training data and test data. Test data is used for training an SVM model. Test data is used for evaluating trained SVM model goodness in order to improve it.

AUTOMATIC ASSESSMENT OF ALZHEIMER'S DISEASE DIAGNOSIS BASED ON DEEP LEARNING TECHNIQUES

Best results with the OASIS data set.

	Class	Accuracy	Precision	Recall	Specificity	F_1
[23] (Horizontal MRI)	Cognitively Normal	-	99.00%	99.00%	-	99.00%
	Very Mild Dementia	-	75.00%	50.00%	-	60.00%
	Mild Dementia	-	63.00%	71.00%	-	67.00%
	Moderate AD	-	33.00%	50.00%	-	40.00%
	Average	-	67.50%	67.50%	-	66.50%
Proposed Model without considering sex and age (Sagittal MRI)	Cognitively Normal	79.36%	89.94%	82.44%	69.00%	86.02%
	Very Mild Dementia	74.31%	33.06%	58.57%	77.32%	42.27%
	Mild Dementia	92.66%	0.00%	0.00%	99.02%	0.00%
	Moderate AD	99.54%	0.00%	0.00%	100.00%	0.00%
	Average	86.47%	30.75%	35.25%	86.34%	32.07%
Proposed Model (Sagittal MRI)	Cognitively Normal	80.05%	92.54%	81.25%	78.00%	86.53%
	Very Mild Dementia	75.00%	35.77%	70.00%	75.96%	47.34%
	Mild Dementia	92.66%	0.00%	0.00%	99.02%	0.00%
	Moderate AD	99.54%	0.00%	0.00%	100.00%	0.00%
	Average	86.81%	32.08%	37.81%	88.25%	33.47%

Comparison of the sagittal plane (436 cases) against the horizontal plane (436 cases + data augmentation cases). Model compared against the proposed model may be learning non-real cases because of the artificially created cases.

Automatic Assessment of Alzheimer's Disease
Diagnosis Based on Deep Learning Techniques

Best results with the ADNI data set.

	Class	Accuracy	Precision	Recall	Specificity	F_1
[15] (Horizontal MRI)	Cognitively Normal	-	100.00%	100.00%	-	100.00%
	Mild Cognitive Dementia	-	60.00%	80.00%	-	69.00%
	AD	-	70.00%	47.00%	-	56.00%
	Average	-	76.67%	75.67%	-	75.00%
Proposed Model without considering sex and age (Sagittal MRI)	Cognitively Normal	78.25%	64.44%	59.80%	86.05%	62.03%
	Mild Cognitive Dementia	71.51%	69.02%	84.32%	56.95%	75.91%
	AD	86.40%	73.42%	31.88%	97.62%	44.45%
	Average	78.72%	68.96%	58.66%	80.21%	60.79%
Proposed Model (Sagittal MRI)	Cognitively Normal	78.36%	63.69%	59.60%	85.97%	61.58%
	Mild Cognitive Dementia	71.50%	69.00%	84.61%	56.50%	76.01%
	AD	86.05%	73.93%	30.62%	97.72%	43.31%
	Average	78.64%	68.87%	58.28%	80.06%	60.30%

Comparison of the sagittal plane (1743 cases) against the horizontal plane (210 cases). Model compared against the proposed model needs to learn fewer cases, having better results but being more overfit.

Source code: https://github.com/TheMVS/DL_AD_mri_sex_age_stages
Docker image:
https://hub.docker.com/r/themvs/dl_ad_mri_sex_age_stages

Cristian Robert Munteanu
IKERDATA
ai@ikerdata.com

Free cloud computing
DL with Fastai and Google Colaboratory

Cristian Robert Munteanu
IKERDATA
ai@ikerdata.com

Code example: Medical images classifier using DL on Google Colaboratory

Shortest-Fastai-Colon-Polyps.ipynb

File Edit View Insert Runtime Tools Help

Colon Polyp Classification using FastAI

```
[1] %reload_ext autoreload
    %autoreload 2
    %matplotlib inline

    from fastai.vision import *
    from fastai.metrics import error_rate

[2] from google.colab import drive
    drive.mount('/content/drive')

[3] path = "/content/drive/My Drive/myAI-Projects/Colonoscopy-polyps-detection-with-CNNs/data_polyps"

[4] data = ImageDataBunch.from_folder(path, ds_tfms=get_transforms(), size=224, bs=64).normalize(imagenet_stats)

[5] learn = cnn_learner(data, models.resnet34, metrics=[accuracy])

[6] learn.fit_one_cycle(5)
```

In **couple of minutes** you can obtain a model classifier using DL for your medical images with an **accuracy > 80%**. The code is **general** for any image, you just create a folder for any class you need and change the path to this folder.

This code is using a **transfer learning** using a pre-trained **resnet34** network (trained by Google) and it is training only the fully-connected part (the convolutional blocks are frozen).

IKERDATA
Spain

Cristian R. Munteanu
AI expert
ai@ikerdata.com
ORCID: 0000-0002-5628-2268

Deep Learning in Neuroscience

2021

Cristian Robert Munteanu
IKERDATA
ai@ikerdata.com