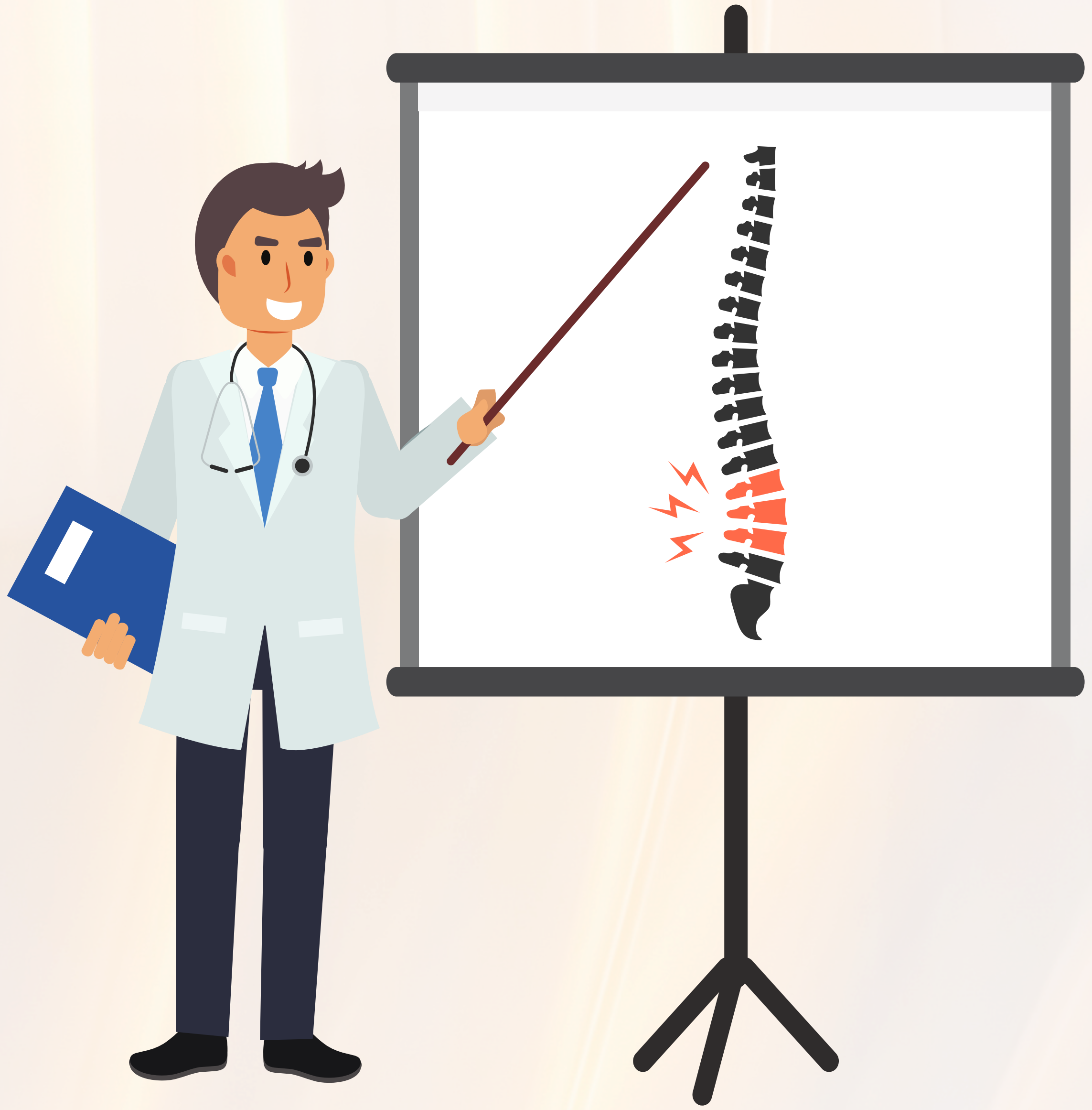
# Lumbar Spine Degenerative Classification using an Optimized CNN (24-2-R-1)

Submitted by: Almog Kadosh - 315699439
Daniel Zada - 318223278
Supervised by: Miri Weiss-Cohen

Department of Software Engineering Braude College of Engineering, Israel
BRAUDE
College of Engineering, karmiel

# Agenda

1. Introduction
2. Explaining the problem
3. Background
4. Solution
5. Process
6. GUI
7. Conclusion



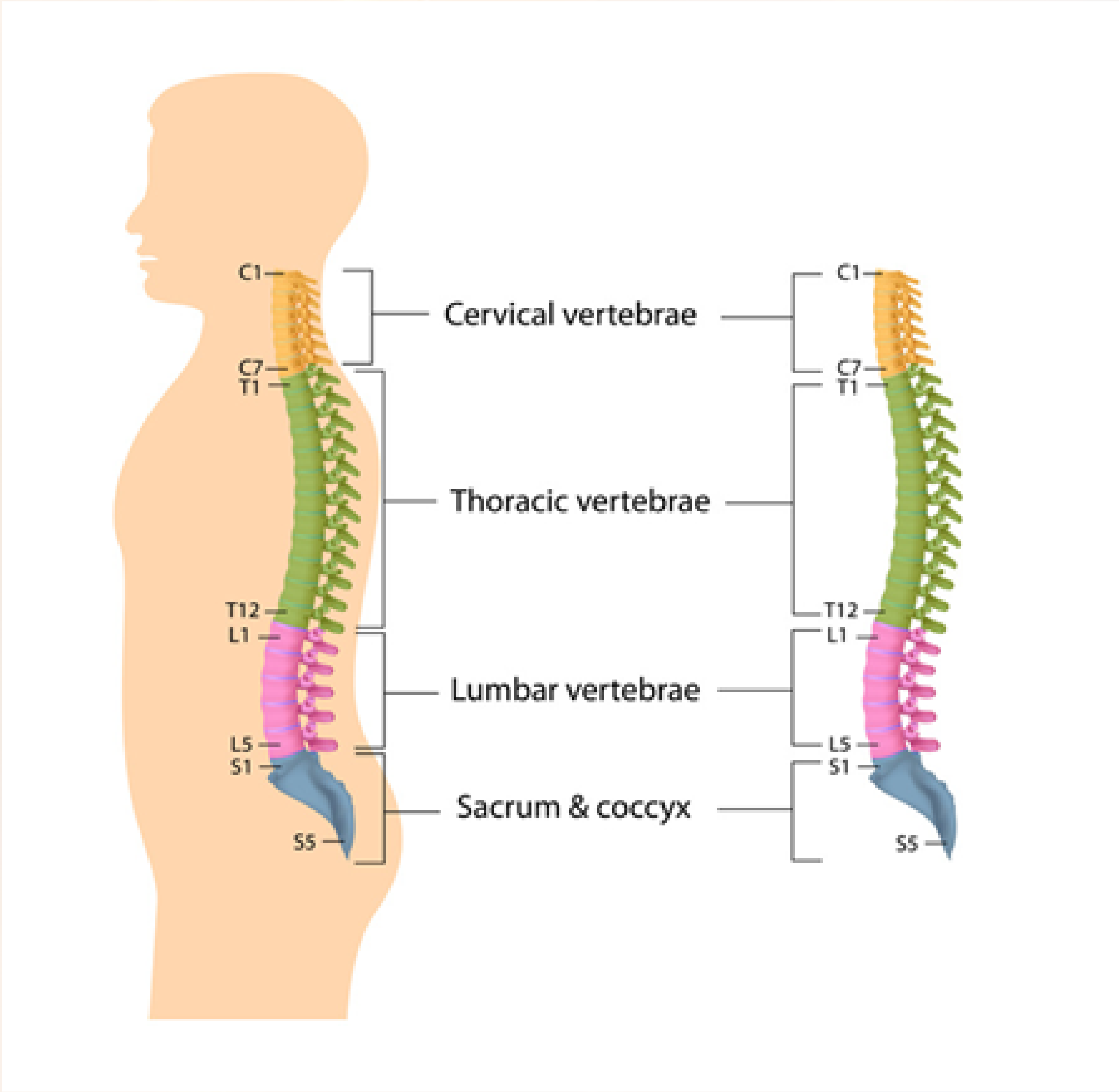
# Introduction-Lower back pain(LBP)

## Causes

- Other medical conditions
- Inflammatory conditions
- Spinal and muscular issues
  - Strains, sprains, and lumbar spine degenerations
- Lifestyle factors
- Aging and genetics
- Improper techniques

# How is the spine built?

- **Spinal Regions:** Cervical (C1-C7), Thoracic (T1-T12), Lumbar (L1-L5), Sacral and Coccygeal (S1-S5).
- **Intervertebral Discs:** Cushion vertebrae, absorb shock, and add flexibility.
- **Muscles:** Erector spinae, trapezius, latissimus dorsi support the spine.
- **Ligaments & Joints:** Ligaments stabilize, facet joints allow limited movement.
- **Spinal Cord:** Runs through vertebrae, connects nerve roots innervate the body.



# Lumbar spine degenerations

Lumbar spine degenerations can be diagnosed and examined via Magnetic Resonance Imaging (or MRI for short).

## What is MRI?

Magnetic Resonance Imaging (MRI) is a non-invasive imaging technology that produces three dimensional detailed anatomical images.

## How does MRI work?

MRI uses powerful magnets and radiofrequency pulses to align and disturb protons, then detects the energy released as they realign to create detailed images of the body's internal structures.

## Application

MRI is often used for disease detection, diagnosis, and treatment monitoring.

# The problem

The primary issue is the absence of a standardized classification for Lumbar Spine Degeneration in Magnetic Resonance Imaging, coupled with Intra- and Inter-subspecialty Variability.

For example between groups of subspecialty-trained neuroradiologists (NR) and musculoskeletal radiologists (MSK).

( neuroradiologists specializes in imaging the nervous system, including the brain and spinal cord, while musculoskeletal radiologists, focuses on imaging the bones, joints, and soft tissues of the musculoskeletal system.)

# Convolutional neural networks

Convolutional Neural Networks (CNNs) are a sophisticated subset of deep learning frameworks specifically engineered to analyze visual data efficiently

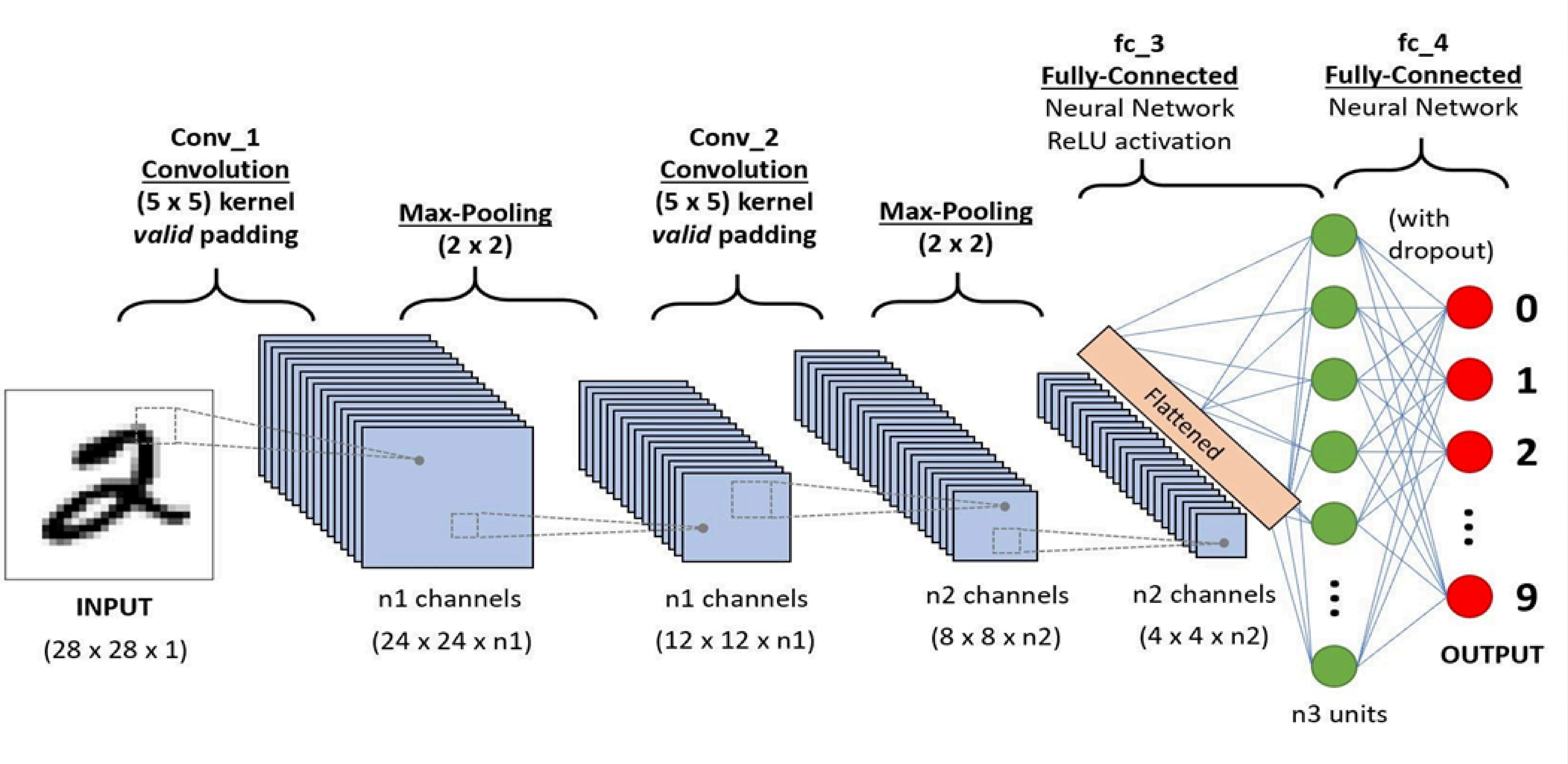
CNNs excel at automatically learning spatial feature hierarchies through multiple layers, making them highly effective for image and video processing tasks

# Convolutional Neural Networks

**Convolutional Layers:** Use filters to detect features in images, like edges and textures. Each filter creates a "feature map" showing where specific patterns appear. Stacking multiple layers helps the network recognize more complex features.

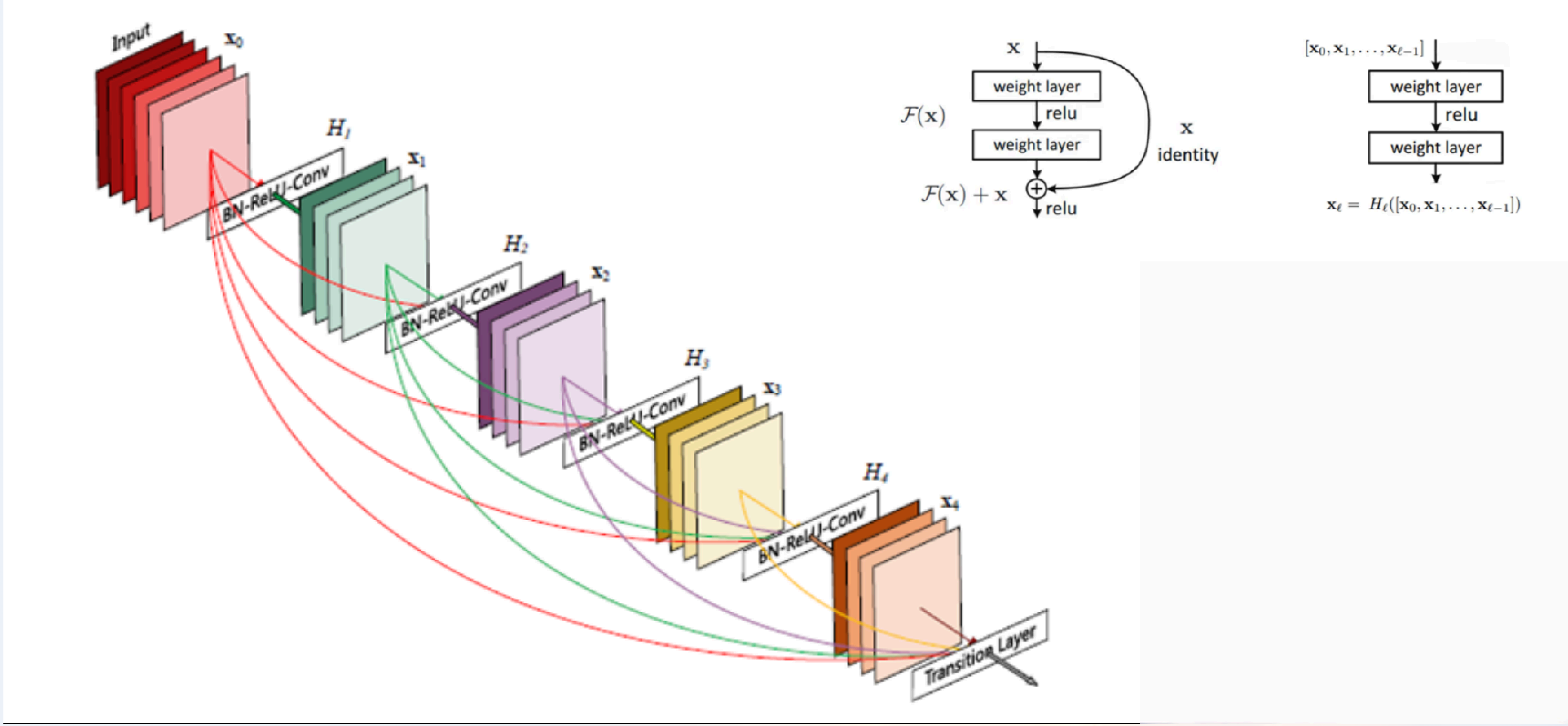
**Pooling Layers:** max pooling is used to reduce feature map size by selecting the highest value from small regions (like 2x2). This keeps important information while making the network faster and less complex.

**Fully Connected Layers:** The final layers, which use learned features to classify or make predictions by assigning weights to each feature to find the best match.



Conv_1
Convolution
(5 x 5) kernel
valid padding
Max-Pooling
(2 x 2)
Conv_2
Convolution
(5 x 5) kernel
valid padding
Max-Pooling
(2 x 2)
fc_3
Fully-Connected
Neural Network
ReLU activation
fc_4
Fully-Connected
Neural Network
(with dropout)
Flattened
INPUT
(28 x 28 x 1)
n1 channels
(24 x 24 x n1)
n1 channels
(12 x 12 x n1)
n2 channels
(8 x 8 x n2)
n2 channels
(4 x 4 x n2)
n3 units
0
1
2
9
OUTPUT

# In our project-densenet



# DenseNet Architecture

**Initial Layers:** 7×7 Convolution with stride 2 for initial feature extraction.
This is followed by batch normalization and ReLU activation to stabilize and speed up training
Followed by a 3×3 Max Pooling layer with stride 2 for downsampling.

**Dense block layer:** consists of few layers of 1X1 and 3X3 convolutions.

**Transition layers:** use 1X1 convolution followed by 2X2 average pooling between two adjacent dense blocks.

**Classification layer:** At the end of the last dense block, a 7X7 global average pooling is performed and then a softmax classifier is attached. Uses 1000D fully connected

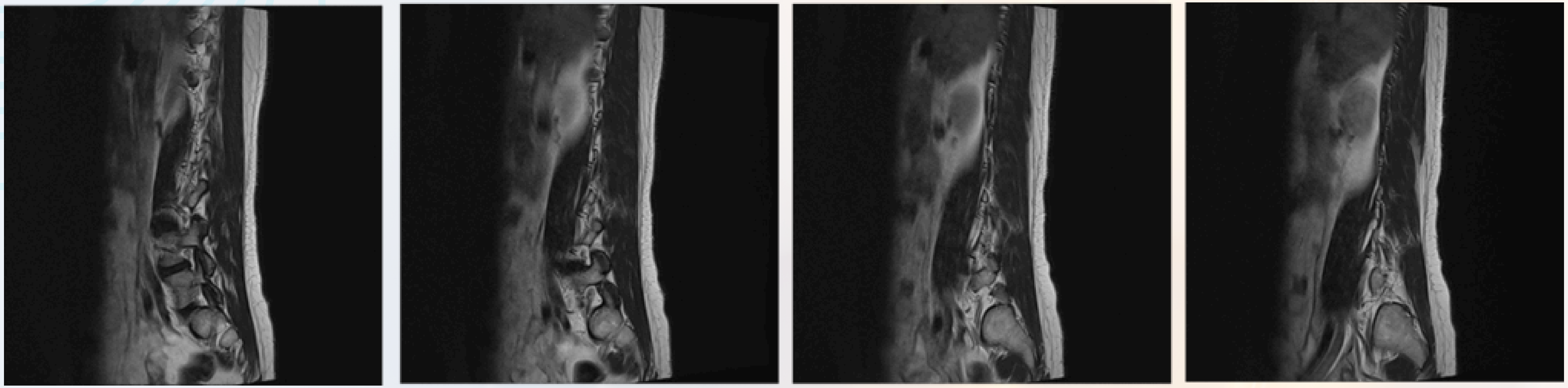
# Key Features of Densenet

DenseNet solves the vanishing gradient problem found in deep neural networks like ResNet. In traditional deep networks, gradients become very small as they are backpropagated, making training slow or ineffective.
DenseNet addresses this by connecting each layer to all previous layers, allowing gradients to flow more easily throughout the network. This improves both gradient flow and feature reuse, enabling more efficient training of deeper networks without losing information.

**Efficient**: Reuses features, avoiding duplication and reducing parameters. (This reduces the need for each layer to learn redundant features, as it can directly use features computed by earlier layers)

**Compact**: Reduces information loss and network size.(By concatenating feature maps instead of recomputing them, DenseNet achieves better performance with fewer parameters compared to other deep architectures like ResNet.)

# Lumbar spine degenerations- Dataset



Kaggle Dataset-
https://www.kaggle.com/competitions/rsna-2024-lumbar-spine-degenerative-classification/data

The dataset contains 147,320 files in the size of 35.34GB, and contains both CSV files for results and classification, and DCM files of the MRI images.
In order to use the images we wrote a short python script to converte from .dcm to .jpeg

.

# Integrating optimized CNN into the classification of lumbar spine degenerations during the mri analysis stage

We aim to develop a system that takes a DICOM image as input to our CNN model and outputs a classification of lumbar spine degeneration based on our pretrained model. We will implement a DenseNet architecture and train it using Kaggle's dataset. Additionally, to reduce training time, we plan to leverage transfer learning by fine-tuning a pretrained classification model.

# Research process-hyperparameters

- **Learning rate:** Ranges from $1 \times 10^{-3} - 1 \times 10^{-7}$ This determines the size of the steps taken during gradient descent to minimize the loss function. We'll adopt a variable learning rate approach, testing different values to see which minimizes error most effectively based on our results graphs (loss and accuracy graphs).

- **Batch size:** We will experiment with batch sizes of 32 and 64, balancing between computational efficiency and the stability of the learning updates.

- **Epochs:** To fine-tune our model, we will compare the effects of running 50 and 100 epochs, observing the trade-off between learning performance and the risk of overfitting.

- **Dropout rate:** As a regularization method, dropout randomly disables a percentage of neurons during training to prevent the model from relying too heavily on any small set of neurons. We will test dropout rates of 0.2 and 0.5 to identify the best option for mitigating overfitting.

# Expected Achievements

Our goal is to develop a reliable and accurate system that simplifies this process as much as possible. We aim for the system to achieve at least 80% accuracy, ensuring it is both user-friendly and efficient. Additionally, we hope our system will reduce the time doctors spend analyzing MRI scans, helping to speed up diagnosis so patients can receive their results faster than the current methods.

**Expected Outcome:**

Build a working DenseNet-based network.
Achieve classification accuracy greater than 80%.
If possible, grade the severity of lumbar spine degeneration in MRI scans.

**Impact:** Simplify the diagnostic process, reduce analysis time, and help doctors deliver faster, more consistent diagnoses, improving patient outcomes.

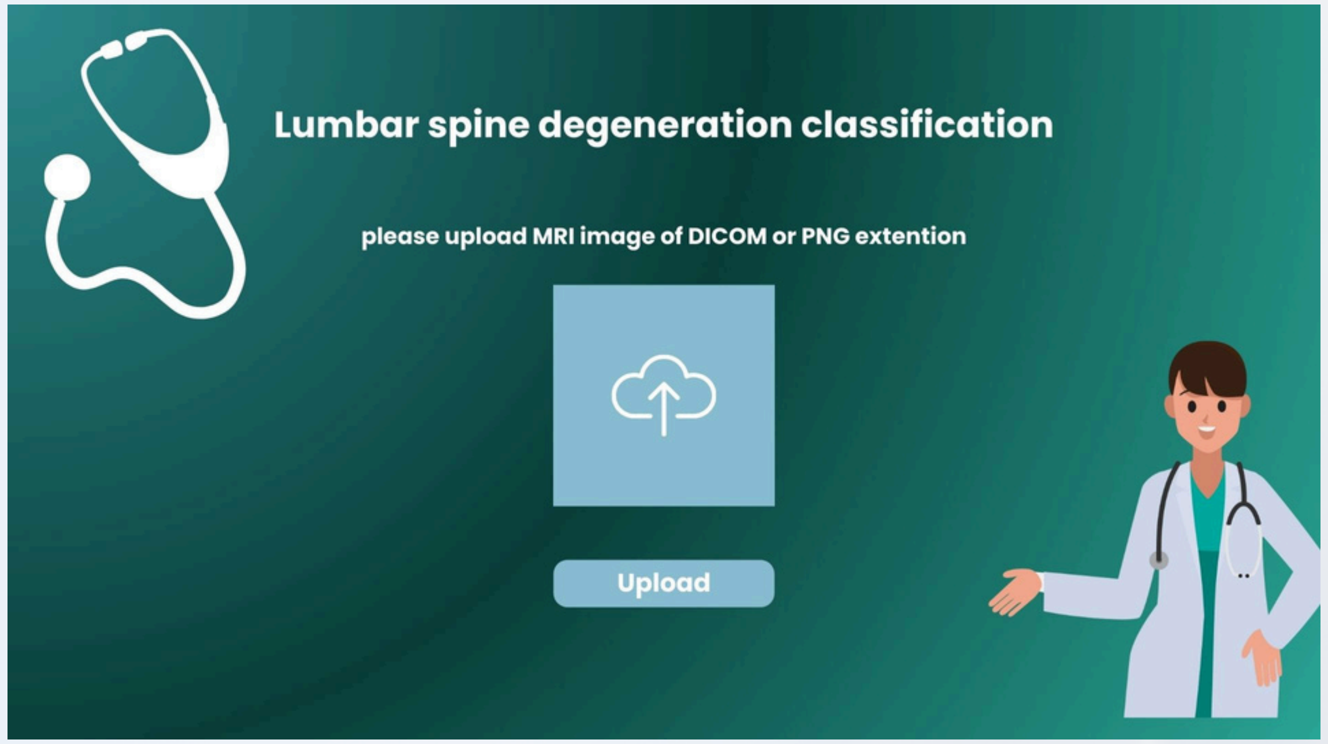
# Testing

**5.3 system testing**

| **Case number** | **Test case** | **Expected result** |
|---|---|---|
| 1 | Insert wrong image format | Error message: "Wrong format" |
| 2 | Insert DICOM or PNG format | The system will allow to press the "upload" button. |
| 3 | Insert expected picture format with no lumbar spine degeneration | The system will allow to press the "upload" button and prompting "no degeneration" in the result screen. |
| 4 | Press the "upload" button without an image uploaded | No action since the "upload" button will not be available to press |
| 5 | Upload viable image, with lumbar spine degeneration | The system will allow to press "upload" and show the classification result |
| 6 | Press "back" button on the last GUI screen | The system will go to the first GUI screen |
| 7 | Press "save result" button on the last GUI screen | The system will prompt local save option |

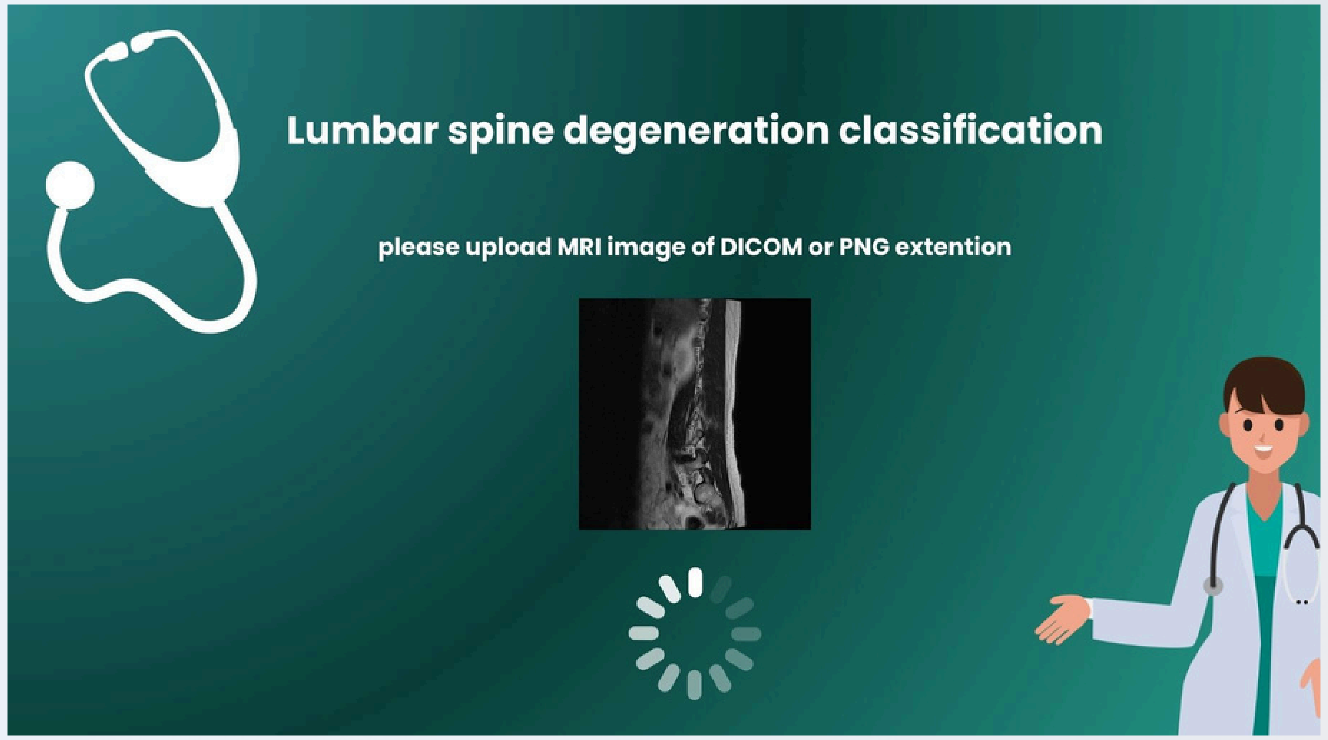
# GUI

## 1.First page

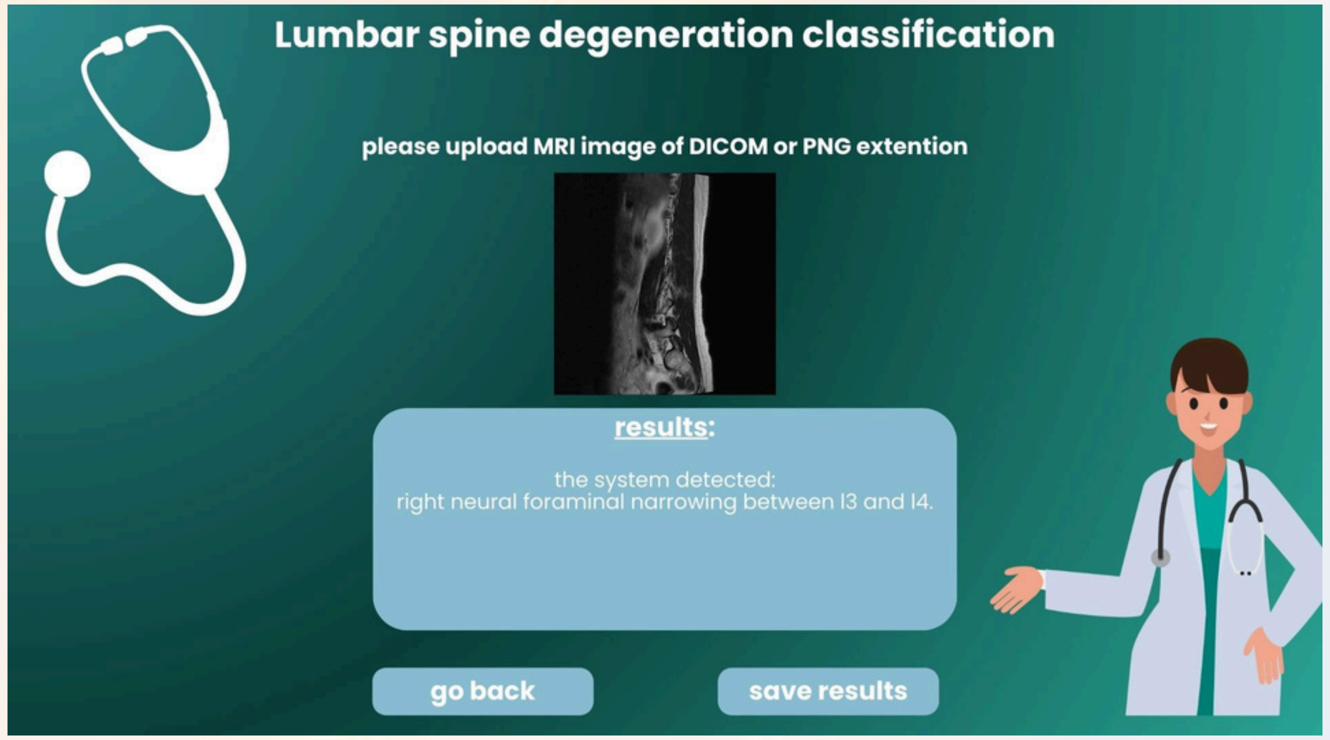



## 2. Uploded image


## 3. Proccessing




## 4. Results



Thank you