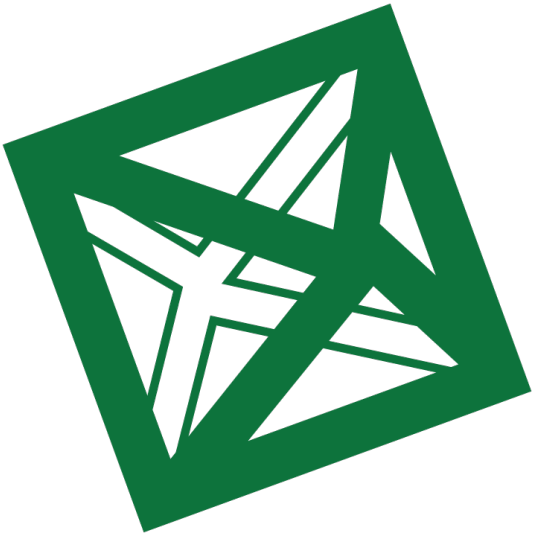
Università degli Studi di Milano-Bicocca

School of Science

Department of Physics



Master Degree in Particle Physics

Overcoming medical data scarcity: transfer learning from synthetic particle jets images to lung CT using deep neural networks

Supervisor:
Prof. Pietro Govoni
Co-Supervisor:
Dott. Simone Gennai

Candidate:
Beatrice Scotti
Student ID: 858535

17 september 2025

Academic Year 2024-2025

Contents

Abstract

Early diagnosis is the most effective weapon in the fight against cancer, with computed tomography and medical imaging playing a crucial role in this process. Recently, artificial intelligence has shown great potential to identify and classify tumor lesions in their early stages. However, the application of these techniques faces a fundamental limitation: the scarcity of annotated data available for training.

As is well known in machine learning, the quality of results depends strictly on the quality and completeness of the training data ("garbage in, garbage out"). To overcome this limitation, this thesis explores the use of transfer learning, a technique that allows the transfer of knowledge from a data-rich source domain to a data-limited target domain.

Specifically, the research focuses on the use of simulated images of high-energy particle jets, constructed specifically for this study, characterized by two distinct classes of physical phenomena: a noisy background with physical characteristics similar to the main jets and morphological properties similar to tumor structures.

The proposed approach involves a pretraining phase on a large high-energy physics dataset, transferring the learned weights to the medical domain, and then fine-tuning them on limited clinical datasets. This method aims to leverage the feature extraction capabilities developed in the particle physics domain, adapting them to the medical context where annotated data are scarce, but diagnostic accuracy is crucial.

Chapter 1

Medical Imaging and Computed Tomography

A medical image is defined as a bidimensional (projection) or a tridimensional (tomography) representation of the values of a certain physical quantity or parameter in every point of the field of view.

Images can be divided in

- **Morphological**: representation of shape and structure of organs
- **Functional**: representation of the spatial distribution of a biological function

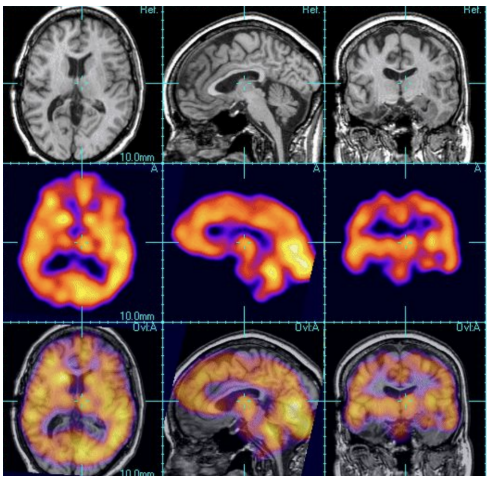


Figure 1.1

Chapter 2

Machine Learning in a nutshell

Machine learning is a subfield of artificial intelligence that focuses on the development of algorithms and statistical models that enable computers to perform tasks without explicit instructions. Instead, these systems learn from data, identifying patterns and making decisions based on the information they have been trained on. Machine learning consists of designing efficient and accurate prediction algorithms. More generally, learning techniques are data-drivem methods combining fundamental concepts in computer science with ideas from statistics, probability and optimization.

Types of tasks

The following are some standard types of tasks in machine learning:

- **Classification:** Assigning labels to data points based on learned patterns (e.g., e-mail spam detection).
- **Regression:** Predicting continuous values based on input features (e.g., predicting house prices). In regression, the penalty for an incorrect prediction depends on the magnitude of the difference between the true and predicted values, in contrast with classification problem, where there is typically no notion of closeness between various categories.
- **Clustering:** Grouping similar data points together without predefined labels.
- **Anomaly detection:** Identifying unusual patterns that do not conform to expected behavior.
- **Ranking**: Ordering items based on their relevance or importance.

Algorithms that solve a learning task based on semantically annotated historical data are said to operate in a **supervised learning** mode. In contrast, algorithms that use data without any semantic annotation are said to operate in an **unsupervised learning** mode. In the latter case, the algorithm is expected to discover patterns in the data without any prior knowledge of the labels or categories. In this thesis I'll mainly focus on supervised learning.

Label set: We use Y to denote the set of all possible labels for a data point of a given learning problem. Note that the labels can be of two different types: categorical labels, which are discrete and finite and define classification problems, and continuous labels, which can take any value in a continuous range and define regression problems.

2.1 Neural networks and deep learning

Neural networks are a class of machine learning algorithms inspired by the structure and function of the human brain. They consist of interconnected nodes (neurons) that process information in layers. Deep learning refers to the use of neural networks with many layers (deep neural networks) to model complex patterns in large datasets. This approach has led to significant advancements in areas such as image recognition, natural language processing, and game playing.

2.2 YOLO: You Only Look Once

Object detection is a task that involves identifying and classifying objects present in images or videos. Initially, object detection was approached as a pipeline consisting of three main steps: proposal generation, feature extraction and region classification. However, this approach was computationally expensive and often led to suboptimal results. The emergence of deep learning brought a significant change in object detection, with deep convolutional neural networks (CNNs) playing a crucial role in this transformation. CNNs are designed to automatically learn hierarchical features from raw pixel data, eliminating the need for manual feature engineering. This shift allowed for more efficient and accurate object detection systems.

Currently, deep learning-based object detection frameworks can be classified in two families:

- **Two-stage detectors:** These methods first generate region proposals and then classify them. Examples include R-CNNs (Region-based Convolutional Neural Networks), that first generate region proposals using a selective search algorithm and then extracts features from these regions using a CNN; the extracted features are then fed into an SVM for object classification.

- **One-stage detectors:** These methods perform detection in a single pass, directly predicting bounding boxes and class probabilities. Examples include YOLO (You Only Look Once), which exists in eleven versions.The YOLO models are popular for their accuracy and compact size. It is a state-of-the-art model that could be trained on any hardware. YOLOv8, in particular, was developed by Ultralytics and introduced on January 2023. It is used to detect objects in images, classify images and distinguish onjects from each other.

2.2.1 Architecture of YOLOv8

The YOLOv8 architecture is composed of two major parts, namely the **backbone** and the **head**, both of which use a fully convolutional neural network.

- The **backbone** is responsible for extracting features from the input image. It consists of a modified version of the CSPDarknet53 architecture, which has 53 convolutional layers and employs a technique called cross-stage partial connections to enhance the transmission of information across the various levels of the network. The convolutional layers are organized in a sequential manner to extract relevant features from the input image.

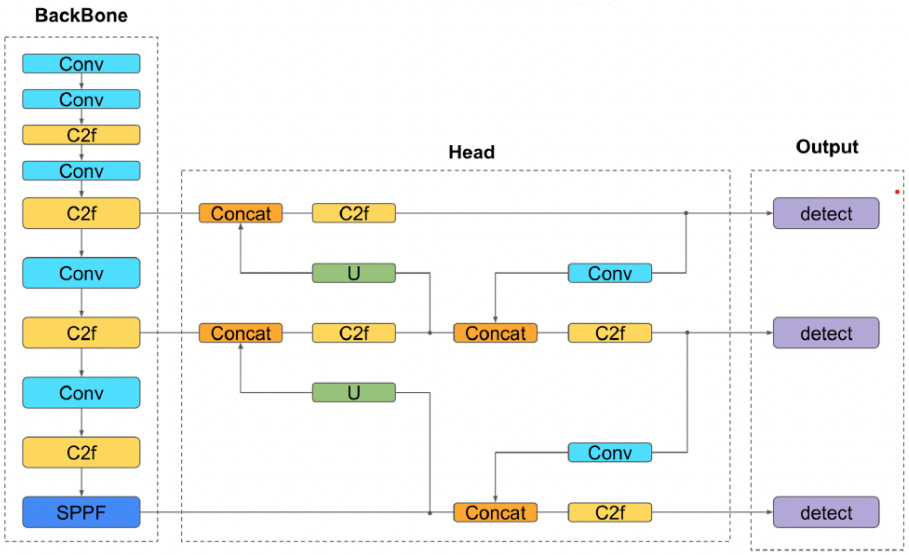


Figure 2.1: Architecture of YOLOv8. The backbone extracts features from the input image, while the head predicts bounding boxes and class probabilities.

- The **head** is responsible for predicting bounding boxes and class probabilities. It consists of a series of convolutional layers that take the features extracted by the backbone and apply additional operations to predict the bounding boxes and class probabilities for each object in the image. The head uses a technique called anchor boxes to handle objects of different sizes and aspect ratios.

The YOLOv8 framework can be used to perform computer vision tasks such as detection, segmentation, classification and pose estimation and comes with pre-trained models for each task. For detection, the models are pre trained on the COCO dataset, while for classification on ImageNet dataset. There are different versions of YOLOv8, each designed for different tasks and with different architectures. The most common versions are YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l and YOLOv8x, where the letter indicates the size of the model (n for nano, s for small, m for medium, l for large and x for extra large). The larger the model, the more parameters it has and the more computational resources it requires to train and run.

2.3 Object Detection Metrics

Object detection metrics are used to evaluate the performance of object detection algorithms. The most common metrics are precision, recall and mean Average Precision (mAP).

2.4 Transfer learning

2.5 Curriculum learning

Chapter 3

Particle Jets

Chapter 4

Medical Dataset

4.1 NLST - National Lung Screening Trial Dataset

The aggressive and heterogeneous nature of lung cancer has thwarted efforts to reduce mortality from this cancer through the use of screening. The advent of low-dose helical computed tomography (CT) altered the landscape of lung-cancer screening, with studies indicating that low-dose CT detects many tumors at early stages. The National Lung Screening Trial (NLST) was conducted to determine whether screening with low-dose CT could reduce mortality from lung cancer. The NLST was a large, multicenter, randomized controlled trial that enrolled over 53,000 participants at high risk for lung cancer. Participants were randomly assigned to receive either low-dose CT scans or standard chest X-rays. The primary outcome was lung cancer mortality, with secondary outcomes including the detection of early-stage lung cancer and the impact of screening on overall mortality.

Radiologists at the screening centers reviewed the images obtained at each of the three annual screening exams to check for signs of lung cancer.

The image review was made without reference to any historical images. The radiologist recorded information about all visible abnormalities and assigned a preliminary screening result. A positive screening result (suspicious for lung cancer) was assigned if any non-calcified nodules or masses $\geq 4mm$ in diameter were noted or if any other abnormalities were judged suspicious for lung cancer by the radiologist.

4.1.1 Dataset Pruning

The NLST dataset contains chest CT scans where each scan comprises multiple (~ 143) axial slices. Notably, not all slices include tumor lesions. The dataset was then pruned by filtering slices based on a provided CSV file that identifies slices containing clinically relevant findings. After pruning, approximately 9,000 slices remain with malignant lesions with $\geq 4mm$ in diameter. However, benign tumor slices lack bounding box annotations, which presents a limitation for detection tasks.

The annotations are in the form of a CSV file that contains the following columns:

- **PID**: Unique identifier for each patient.
- **Slice Number**: The specific slice within the CT scan, which is also the name of the file.
- **CT filter**: Indicates the type of filter applied to the CT scan.

- **x, y, width, height**: Coordinates of the bounding box around the lesion, x and y represent the top-left corner of the bounding box, while width and height represent its dimensions.

4.1.2 Extraction of the pixel array

The CT images are in the dicom format, which is a standard format for medical imaging data. Each CT scan consists of multiple slices, and each slice is represented as a 2D image. The pixel values in these images are typically stored in Hounsfield units (HU), which represent the radiodensity of the tissue. A dicom file contains:

- **Pixel Array**: The pixel array is a 2D array of pixel values that represent the intensity of the image. Each pixel value corresponds to a specific location in the image and is typically stored as a 16-bit integer.
- **Metadata**: The metadata contains information about the image, such as the patient's name, the acquisition date, the image orientation, pixel spacing, and slice thickness. This information is essential for interpreting the image correctly.

4.1.3 Image Metadata

The metadata of a CT image contains important information about the image acquisition parameters and the physical characteristics of the image. The following are some key metadata fields commonly found in CT images:

- **Pixel Spacing (mm)**: The physical distance between adjacent pixels in the x and y directions, typically measured in millimeters.
- **Slice Thickness (mm)**: The thickness of each slice in the z direction, also measured in millimeters.
- **Field of View (FOV) (cm)**: The physical dimension of the area captured in the image, typically measured in centimeters.
- **Image Orientation**: The orientation of the image, which can be specified using a combination of row and column vectors.
- **Image Position**: The position of the image in 3D space, typically specified using x, y, and z coordinates.

4.2 CT planes

In computed tomography (CT), the images are typically acquired in three orthogonal planes: axial, coronal, and sagittal. Each plane provides a different perspective of the anatomical structures within the body.

- **Axial**: horizontal plane that divides the body into upper and lower parts.
- **Coronal**: vertical plane that divides the body into anterior (front) and posterior (back) parts.
- **Sagittal**: vertical plane that divides the body into left and right parts.

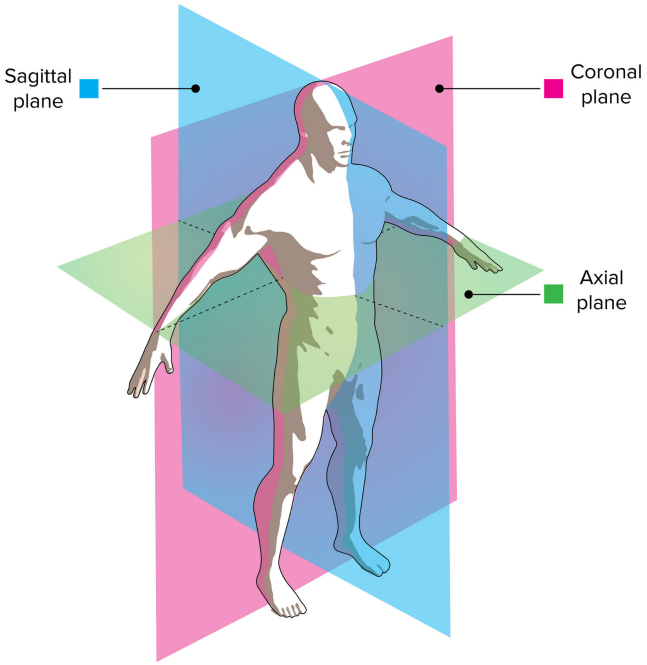


Figure 4.1: CT planes: axial, coronal, and sagittal. The axial plane is horizontal, the coronal plane is vertical and divides the body into front and back, and the sagittal plane is vertical and divides the body into left and right.

4.3 DLCS - Duke Lung Cancer Screening Dataset

4.3.1 Extraction of the 2D slices

4.3.2 Dataset Pruning

4.4 Pre processing the images

4.4.1 Resampling

4.4.2 Windowing

In computed tomography (CT), the intensity values of the pixels are quantified in **Hounsfield units (HU)**, a standardized scale that reflects the radiodensity of the tissue. This scale is defined relative to the attenuation coefficients of water and air:

$$HU_{tissue} = 1000 \times \frac{\mu_{tissue} - \mu_{water}}{\mu_{water} - \mu_{air}} \tag{4.1}$$

where μ represents the linear attenuation coefficient. Key reference points: water = 0 HU for definition, air = -1000 HU.
Windowing is a technique used in computed tomography (CT) to **enhance the visibility of specific tissues or structures within the body.** It involves adjusting the range of Hounsfield units (HU) displayed in the CT image, allowing radiologists to focus on particular types of tissues, such as bones, soft tissues, or air-filled spaces. The window width and center determine the contrast and brightness of the image, respectively.

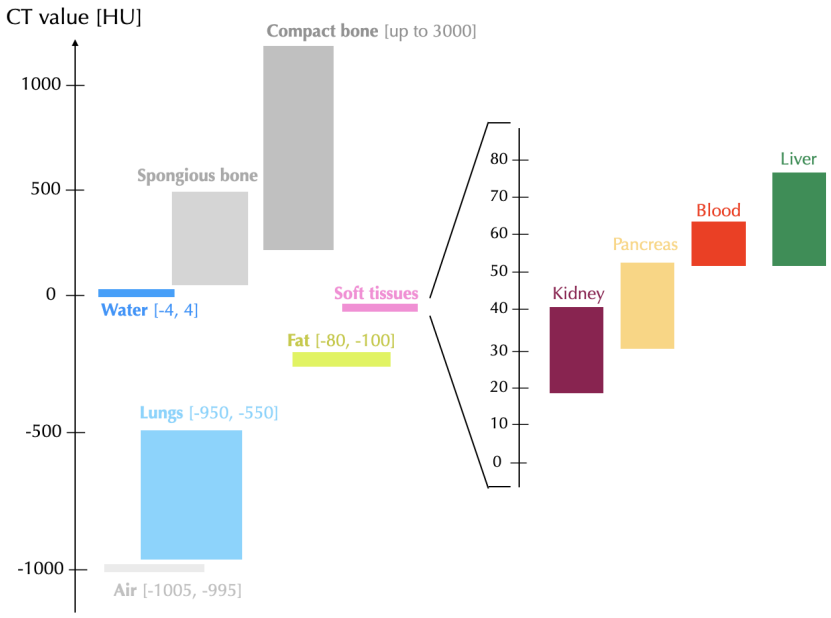


Figure 4.2: CT Hounsfield Unit (HU) ranges for key tissues. Highlighted regions show typical HU values for diagnostic reference, from dense compact bone ($\sim$ 3000 HU) to air (1000 HU), with soft tissues ($20 - 80$ HU) and fluids (blood $\sim$ 45 HU, water 0 HU) in intermediate ranges.

- **Window width** (WW): This parameter controls the range of Hounsfield units displayed in the image. A narrow window width enhances the contrast between tissues with similar densities, while a wider window width allows a broader range of densities to be visualized.
- **Window center** (WC): This parameter sets the midpoint of the Hounsfield unit range displayed in the image. Adjusting the window level changes the brightness of the image, allowing radiologists to focus on specific tissue types.

The choice of window width and level depends on the specific clinical question and the type of tissue being examined.

```
    window_center = -600
    window_width = 1700
    window_min = window_center - window_width // 2
    window_max = window_center + window_width // 2

    array = np.load(array_path)
    array_windowed = np.clip(array, window_min, window_max)
    array_normalized = 255*(array_windowed - window_min)/(window_width)
    array_uint8 = array_normalized.astype(np.uint8)

    img = Image.fromarray(array_uint8)
```

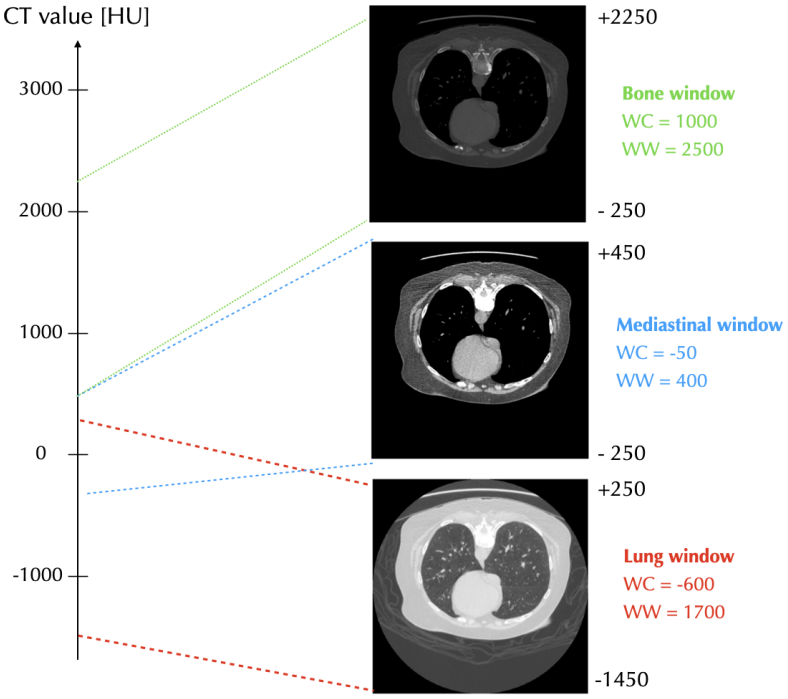


Figure 4.3: Standard CT windowing settings for different anatomical structures. Full CT value range ([-1000, 3000] HU). Clinical presets with window center (WC) and width (WW) values for bone (WC=1000, WW=2500), mediastinum (WC=-50, WW=400), and lung (WC=-600, WW=1700) visualization.

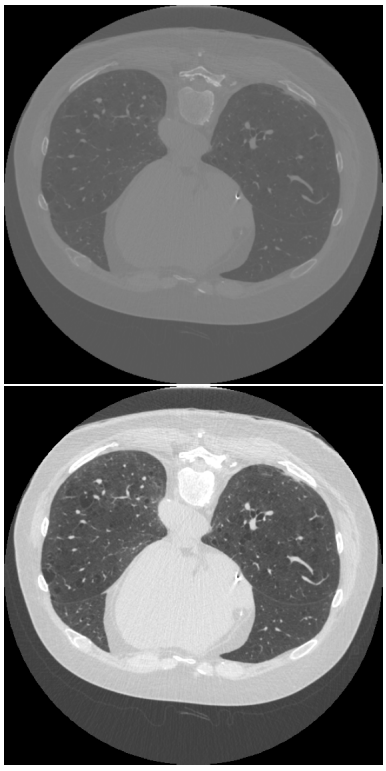


Figure 4.4: Didascalia principale delle due immagini affiancate

RBG Windowing

4.4.3 Padding

4.4.4 Masking