# Spotting L3 slice in CT scans using deep convolutional network and transfer learning

Soufiane Belharbi[a], Clément Chatelain[a,d], Romain Hérault[a,d], Sébastien Adam[a,*], Sébastien Thureau[c,a], Mathieu Chastan[b], Romain Modzelewski[a,b]

[a] *Normandie Univ, UNIROUEN, UNIHAVRE, INSA Rouen, LITIS, 76000 Rouen, France*
[b] *Henri Becquerel center, Department of Nuclear Medicine, 76000 Rouen, France.*
[c] *Henri Becquerel center, Department of Radiotherapy, 76000 Rouen, France.*
[d] *These authors contributed equally*

**Abstract**

In this article, we present a complete automated system for spotting a particular slice in a complete 3D Computed Tomography exam (CT scan). Our approach does not require any assumptions on which part of the patient's body is covered by the scan. It relies on an original machine learning regression approach. Our models are learned using the transfer learning trick by exploiting deep architectures that have been pre-trained on imageNet database, and therefore it requires very little annotation for its training. The whole pipeline consists of three steps : i) conversion of the CT scans into Maximum Intensity Projection (MIP) images, ii) prediction from a Convolutional Neural Network (CNN) applied in a sliding window fashion over the MIP image, and iii) robust analysis of the prediction sequence to predict the height of the desired slice within the whole CT scan. Our approach is applied to the detection of the third lumbar vertebra (L3) slice that has been found to be representative to the whole body composition. Our system is evaluated on a database collected in our clinical center, containing 642 CT scans from different patients. We obtained an average localization error of $1.91 \pm 2.69$ slices (less than 5 mm) in an average time of less than 2.5 seconds/CT scan, allowing integration of the proposed system into daily clinical routines.

*Corresponding author
*Email address:* Sebastien.Adam@univ-rouen.fr (Sébastien Adam )

*Keywords:* Convolutional neural networks, deep learning, slice detection, maximum intensity projection, sarcopenia

## 1. Introduction

In recent years, there has been an increasing interest in the analysis of body composition for estimating patient outcomes in many pathologies. For instance, sarcopenia (loss of muscle), visceral and subcutaneous obesity are known prognostic factors in cancers [MBM+13, YDM+15], cardiovascular diseases [AWM+14] and surgical procedures [PVT+11, KOF+13]. Body composition can also be used to improve individual nutritional care and chemotherapy dose calculation [GLC+13, LKTM+14]. It is usually assessed by CT and Magnetic Resonance Imaging (MRI). Moreover, It has been shown that the composition of the third lumbar vertebra (L3) slice is a good estimator of the whole body measurements [MBH+98, SPW+04]. To assess the patient's body composition, radiologists usually have to manually find the corresponding L3 slice in the whole CT exam (spotting step, see Figure 1), and then to segment the fat and muscle on a dedicated software platform (segmentation step). These two operations take more than 5 minutes for an experienced radiologist and are prone to errors. Therefore, there is a need for automating these two tasks.

The segmentation step has been extensively addressed in the literature among the medical imaging community [PXP00, MT96]. Dedicated approaches for L3 slice have been been proposed such as atlas based methods [CCB+09] or deep learning [LHC+15]. On the other hand, to the best of our knowledge, the automatic spotting of a specific slice within the whole CT scan has not been investigated in the literature. The spotting task is particularly challenging since it has to handle:

- The intrinsic variability in the patient's anatomy (genders, ages, morphologies or medical states).

- The various acquisition/reconstruction protocols (low/high X-rays dose, slice thickness, reconstruction filtering, enhanced/non enhanced contrast agent).
- The arbitrary field-of-view scans, displaying various anatomical regions.
- The strong similarities between the L3 slice and other slices, due to the repetitive nature of vertebrae (Fig.2).

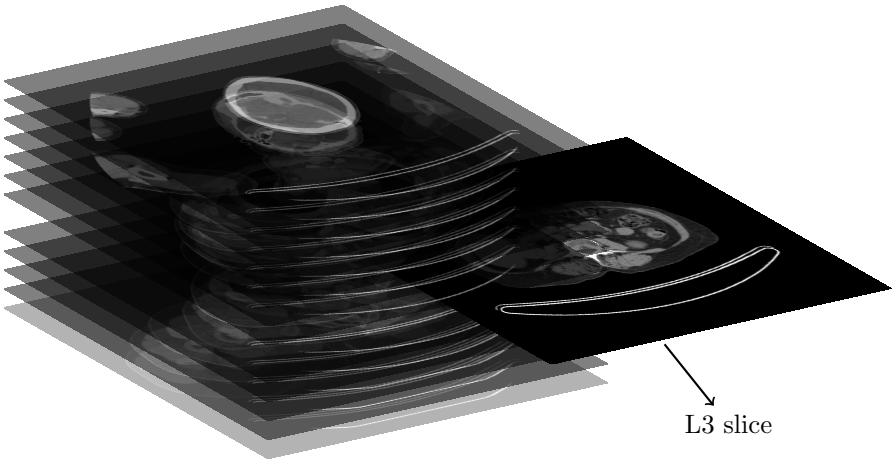


Figure 1: Finding the L3 slice within a whole CT scan.

In the literature, spotting tasks are often achieved using ad hoc approaches such as registration which are not suitable for high variability problems [GZH14, CWJ+15]. In particular, a 3D registration on a whole CT scan would require a large amount of computation at decision time [SEM16]. Here, we suggest a more generic strategy based on machine learning in order to handle high variability context, while maintaining a fast decision process.

In this work, spotting a slice within a CT scan is tackled as a regression problem, where we try to estimate the slice position height. An efficient processing flow is proposed, including a Convolutional Neural Network (CNN) learned using transfer learning. Our approach tackles the classical issues faced in medical image analysis: the data representation issue is addressed using Maximum In-

Figure 2: Two slices from the same patient: a L3 (up) and a non L3 (L2) (down). The similar shapes of both vertebrae prevent from taking a robust decision given a single slice.

tensity Projection (MIP); the variability of the shapes in CT scans is handled using a CNN; and the lack of annotated data is circumvented using transfer learning.

The article is organized as follows: section 2 presents the related work and the general framework for applying machine learning for L3 detection in a CT scan. The third section presents the proposed approach and describes each stage of the whole processing flow. The fourth section describes the experiments and the obtained results.

## 2. Related Work

Machine learning approaches provide generic and flexible systems, provided enough annotated data is available. From a machine learning perspective, the localization of the L3 slice given a whole CT scan can either be considered as a slice-classification problem, a sequence labeling problem or a regression problem. Let us now consider these three options.

The classification paradigm consists of deciding for each slice of the whole CT scan whether the L3 vertebra is present or not. However, the repetitive nature of individual vertebra induces a similarity between the L3 slice and its neighbors, which prevents to efficiently classify an isolated slice without any context (see Fig. 2). This explains why even experienced radiologists need to browse the CT scan to infer the relative position and precisely identify the L3 slice. To the best of our knowledge, the classification paradigm has not been used in the literature to detect the L3 slice within a whole CT scan.

The sequence labeling paradigm consists of estimating the label (L1, L2, etc.) of every slice of a complete CT scan, then, choose the one that is more likely to correspond to the L3. The advantage of this approach is that the decision is globally taken on the whole CT scan by analyzing the dependencies between the slices. This kind of approach has been recently investigated for labeling the vertebrae of complete spine images [GACD11, GVC09, MWK+13, GDE+13, KLP11, GFC+12, HCLN09, ML13, OA11]. The dependencies are modeled using graphical models, such as Hidden Markov Models (HMMs) [GFC+12] or Markov Random Fields (MRFs) [KLP11]. A full review of the spine labelization methods can be found in [MHSB13]. The major drawback of sequence labeling approaches is that they require a fully annotated learning database where every slice of the CT scan is labeled, which is very time consuming. Such a dataset is

proposed by [GZH14], but this dataset cannot be easily exploited for our problem since i) the data are cropped images of the whole spine, and ii) it contains only 224 CT scan.

The regression problem consists of directly estimating the L3 slice number given the whole CT scan, in a spotting fashion. Like the previous paradigm, it has the advantage of performing a global decision by taking into account the dependencies within the entire exam. Another major advantage of a spotting approach is that it does not require a full labeling of the exams. Indeed, the only annotation needed for learning such a model is the L3 position within the whole exam. For radiologists, this annotation is more lightweight than a full annotation and may lead to creating large datasets easily.

In this work, we retain the third paradigm and propose a machine learning approach for spotting the L3 slice in heterogeneous arbitrary field-of-view CT scans. To the best of our knowledge, this is the first time that slice spotting is addressed as a machine learning regression problem.

Usually, traditional machine learning methods exploit generic hand-designed features which are fed to a learning model with the assumption that they are suitable for describing the image. To achieve high accuracy, usually one ends up combining many types of features which require extensive computation, more time and large memory size. Ideally, it would be better if the model is capable of learning on its own task-dependent features.

Deep neural networks (DNN) are a specific category of models in machine learning which are capable of learning on their own hierarchical features based on the raw image. Convolutional neural networks (CNN) are a particular type of DNN which gained a large reputation in computer vision due to their high performance for many tasks on natural scene images [STE13, ESTA14, RHGS15, KSH12].

In the last years, the use of machine learning, in general, and using CNN, in particular, has grown in various medical domains such as cancer diagnosis [RYL+14, UBHK14], segmentation [HJ13, HDW+15, Lai15] or histological [MMB+08] and drusen identification [CC06]. In all these works, the authors are faced with a common issue which is the lack of annotated data. Although extremely powerful, CNN architectures require a huge amount of data to avoid the "learning by heart" phenomenon, also known as overfitting in machine learning. The classical techniques to limit these issues are dropout, data augmentation or the use of regularization. All these technical tricks are exploited in [Lai15], but the lack of data is still a limitation to train such large models. Recently, a more efficient way has been proposed to circumvent the lack of annotated data in vision. This method consists of exploiting models that have been pre-trained on a huge amount of annotated data on another task and is known as "transfer learning".

In this work, we explore the idea of using a CNN model for the localization of the L3 slice using transfer learning. A full description of our approach is presented in section 3.

## 3. Proposed approach

Using a CNN for solving the L3 detection task formulated as a regression problem (see fig. 1) is not straightforward, and requires the alleviation of some constraints which are inherent to the medical domain and to the data that is being processed (i) Training a CNN on 3D data such as CT scans requires very large computing and memory resources that can even exceed the memory limit of most accelerator cards, while such cards are essential for learning a CNN in a reasonable time; (ii) Training a CNN requires fixed size inputs, while the size of the CT scans can vary from one exam to another because of an arbitrary field of view; (iii) Training a CNN requires a large amount of labeled data.

In this paper, we propose to overcome these limitations by using the approach depicted in figure 3. In this approach, the CT scan is first converted into another representation using Maximum Intensity Projection (MIP), in order to reduce the dimension of the input from 3D to 2D, without loss of important information. Then, the MIP image is processed in a sliding window fashion to be fed to a CNN with a fixed-size input. This CNN is trained with Transfer Learning (TL-CNN) to solve the requirement of a large amount of labeled examples. Once the trained TL-CNN has computed its prediction for each position of a sliding window, the resulting prediction sequence is processed in order to estimate the final L3 position in the full CT scan. The following subsections detail the three important contributions of the proposed system.

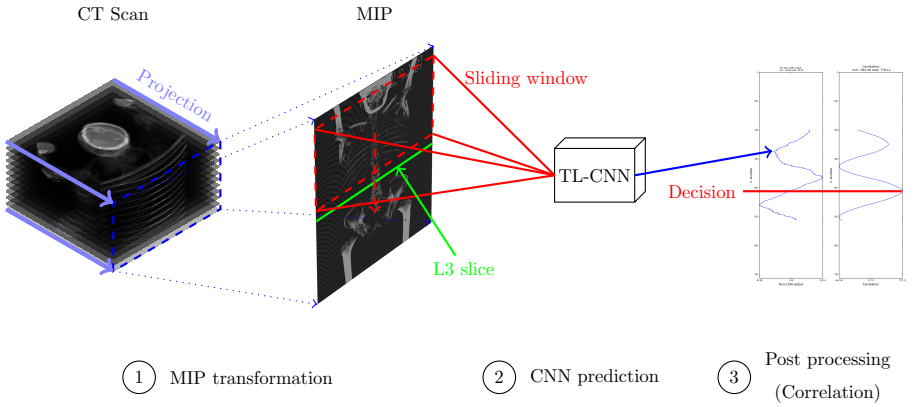


Figure 3: System overview describing the three important stage of our approach : MIP transformation, TL-CNN prediction, and post processing.

### *3.1. MIP transformation*

Ideally, one can use the raw 3D scan image to feed the CNN. If $N$ is the number of slices of the arbitrary field of view CT scan, the input size is $512^2 \times N$. For example, a CT-scan with 1000 slices represents 262M inputs. However, the input size of CNN models strongly impacts their number of parameters. Therefore it would require a very large number of training samples to efficiently

learn the CNN. Thus, in the case of few training samples, using the 3D scan directly as an input is not efficient. We believe that the patient's skeleton carries enough visual information in order to detect the L3.

For these reasons, we propose to use a different data representation which focuses on the patient's skeleton and dramatically reduces the size of the input space. This representation is based on a frontal Maximum Intensity Projection (MIP) [WMLK89, WM91, Wal92]. The idea is to project a line from a frontal view of the CT scan and retain the maximum intensity over all the voxels that fall into that line. We experimented using different views such as frontal and lateral views, as well as their combination but they did not work well as compared to the frontal view alone.

Since the slice thickness can vary within the same scan and the voxels are not squared, the projection often generates a distorted MIP. Visually, this gives an unrealistic image where the skeleton is shrunk or enlarged. The cause of this distortion is that, often, the resulting pixel from the projection does not correspond to one voxel. Often, one voxel can be represented by more than one pixel. In order to obtain an equal correspondence (i.e. one pixel corresponds to one voxel), we resize (normalize) the 2D MIP image using an estimated ratio $r$ and average slice thickness $s$ where $r$ represents the number of pixels corresponding to one voxel (slice).

Fig.4 shows an example of a normalized frontal MIP image. The MIP transformation reduces the input size from $512^2 \times N$ to $512 \times N$.

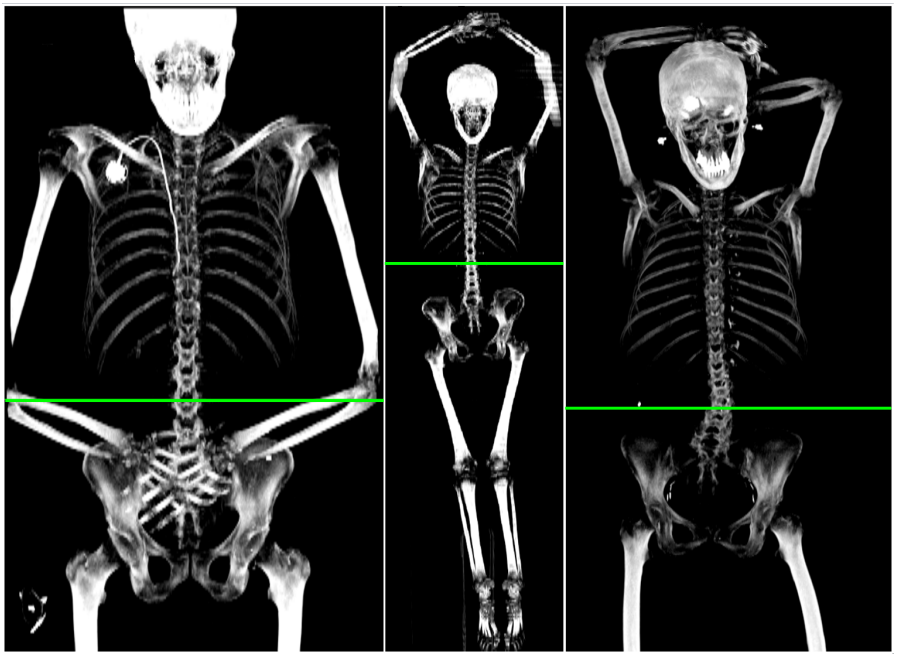

Figure 4: Examples of normalized frontal MIP images with the L3 slice position.

### *3.2. Learning the TL-CNN*

Convolutional neural networks (CNN) are particular architecture of neural networks. Their main building block is a convolution layer that performs a non-linear filtering operation. This convolution can be viewed as a feature extractor applied identically over a plane. The values of the convolution kernel constitute the layer parameters. Several convolution layers can be stacked to extract hierarchical features, where each layer builds a set of features from the previous layer. After the convolutional layers, fully connected layers can be stacked to perform the adequate task such as the classification or the regression.

In the learning phase, both parameters of convolutional layers and fully connected layers are optimized according to a loss function. The optimization of these huge number of parameters is generally performed using stochastic gradient descent method. This process requires a very large number of training samples.

Recently, there has been a growing interest in the exploration of transfer learning methods to overcome the lack of training data. Transfer learning consists in adapting models, trained for different task, to the task in hand (target). It has been applied with success for various applications such as character recognition [Jia15, CMS12], signature identification [HSO16] or medical imaging [BDWG15, SRG+16]. All these contributions exploit CNN architectures which have been pre-trained on computer vision problems, where huge labeled datasets exist. In this framework, the weights of the convolutional layers are initialized with the weights of a pre-trained CNN on another dataset, and then fine-tuned to fit the target application. The fine-tuning starts by transferring only the weights of the convolutional layers from a pre-trained network to the target network. Then, randomly initialized fully connected layers are stacked over the pre-trained convolutional layers and the optimization process is performed on the whole network. This transfer learning framework carried out for our application is illustrated by Figure 5 .

A well-known difficulty when using the transfer learning paradigm is to fit the data to the input size of the pre-trained architecture. Since the size of the normalized MIP images varies from one patient to another, two solutions can be considered. The first one consists of resizing the whole scan to a given fixed size. This solution is straightforward but it dramatically impacts the image quality and the output precision. The second solution consists in decomposing the input MIP into a set of fixed-size windows with a sampling strategy. In this paper, we adopt the second approach which enables to preserve the initial quality of the image data.

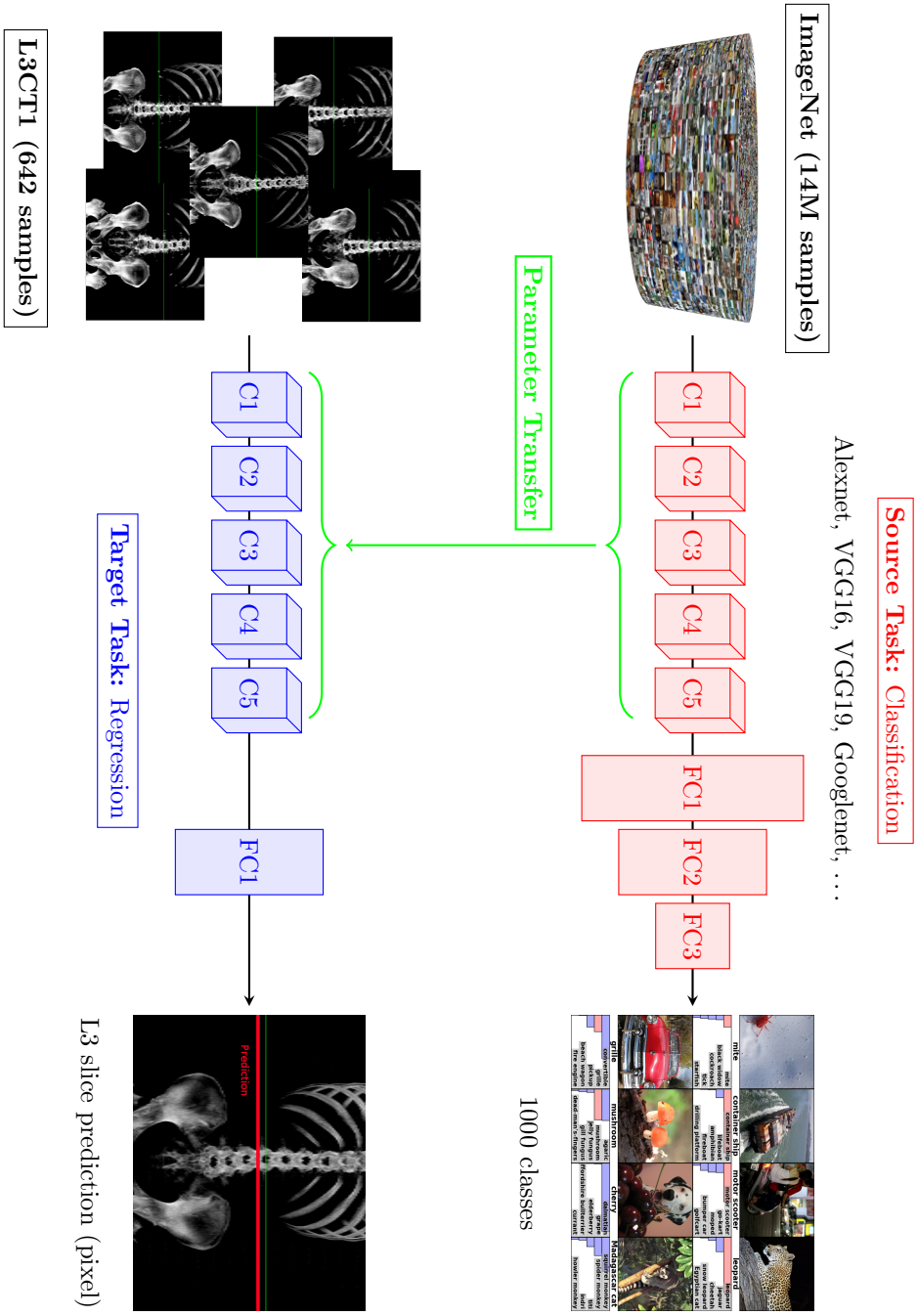


Figure 5: System overview. Layers $C_i$ are Convolutionnal layers, while $FC_i$ denote Full Connected layers. Convolution parameters of previously learnt ImageNet classifier are used as initial values of corresponding L3 regressor layers to overcome the lack of CT examples.

When sampling windows from the MIP image, two sets of window images can be produced. The first one is made of windows containing the L3, and the other one is made of windows without the L3. This raises the question whether the windows without L3 should be present or not in the CNN learning dataset. As we propose a regression approach, adding the non-L3 images in the learning dataset would imply that the CNN learns (and outputs in the decision stage) the offset of the L3 with respect to the current window. Obviously, this offset can be very difficult to learn, particularly if the current window is far from the L3 position. Thus, we have decided to include only the windows containing the L3 in the learning dataset.

Thus, for building the training dataset, we sample all the possible windows of height $H$ such that the L3 position is in the support $[-a, +a]$ where 0 denotes the center of the window. This leads to $2a + 1$ possible windows from each MIP image to be included in the training set. All windows from all MIP are then shuffled: it is highly improbable that two neighboring windows from the same MIP will appear next to each other in the optimization procedure.

### *3.3. Decision process using a sliding window over the MIP images*

A sliding window procedure is applied at the decision phase on the entire MIP image, leading to a sequence of relative L3 position predictions. Such a sequence is illustrated in the left of figure 6.

In this sequence, one can observe two distinct behaviors depending on the presence of the L3 in the corresponding window: i) If the L3 is not in the window, the CNN tends to output random values since it has been trained only on images containing L3. This behavior is illustrated in Figure 6 at the beginning and (less clearly) at the end of the sequence. ii) If the L3 is within the window, the CNN is expected to predict (correctly) the relative L3 position within the window. Since the L3 position is fixed in the MIP and the window slides line by line on the region of interest, the true relative L3 position should decrease one by one. In consequence, the CNN output should evolve linearly along the sequence of windows, leading to a noisy straight line with a slope of

$-1$. The noise may come from local imprecision or error on an individual slide. This behavior can be observed in figure 6 between offset 500 and 600, and it is highlighted with a theoretical green line.

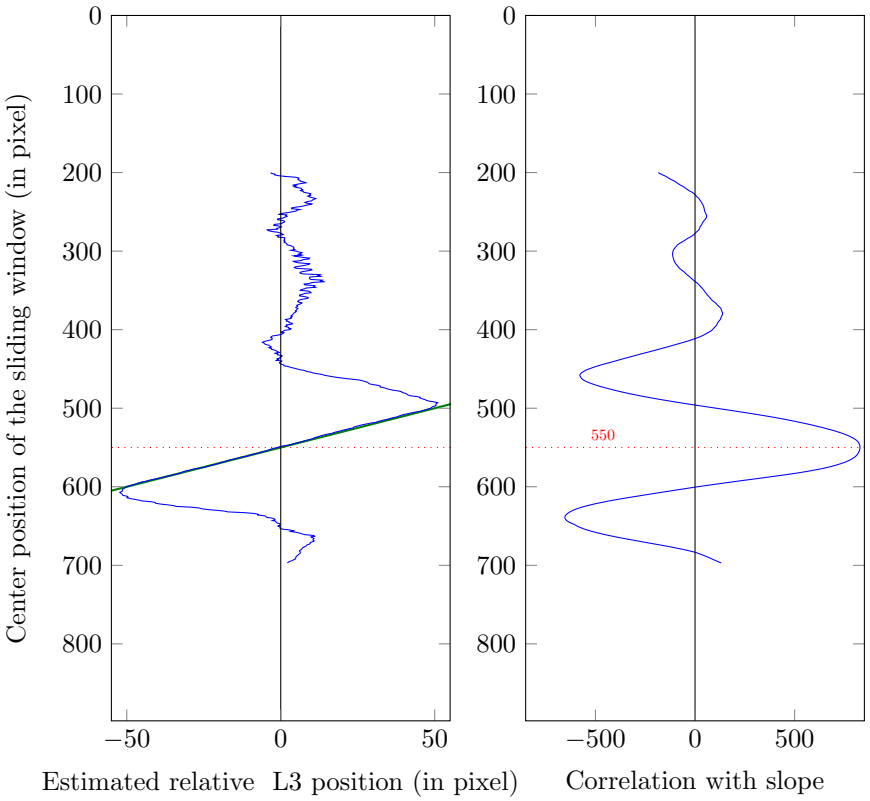


Figure 6: [left]: CNN output sequence obtained for $H = 400$ and $a = 50$ on a test CT scan. The sequence contains the typical straight line of slope $-1$ centered on the L3 (the theoretical line is plotted in green), surrounded by random values. [right]: correlation between the CNN output sequence and the theoretical slope. We retain the maximum of correlation as an estimation of the L3 position.

Therefore, at decision stage, the L3 position can be estimated through the localization of the middle of this particular straight segment. This estimation can easily be achieved by searching the maximum of a simple correlation between

the sequence and the expected slope. This procedure, illustrated at the bottom of Fig. 6, easily filters out boundary windows which do not contain the L3, and shows robustness by averaging several predictions of the CNN.

## 4. Experimental protocol

### *4.1. CT exams database description*

In order to validate the proposed approach, a database named L3CT1 has been collected[1]. The main part of the dataset is composed of 642 CT exams from different patients. All patients were included in this study after being informed of the possible use of their images in a retrospective research. The institutional ethical board of the Rouen Henri Becquerel Center approved this study [2]. The CT exams show a high heterogeneity of patients in terms of anatomy, sex, cancer pathologies, position and properties of the reconstructed CT images: 4 scanner models (PET/CT modalities) and 2 manufacturer, acquisition protocols (low dose acquisition (100 to 120 kV) and modulated mAs along the body) axial field of view (FOV) (400 to 500 mm), reconstruction algorithms (Filtered Back Projection (FBP) or iterative reconstruction) and slice thickness (2 to 5 mm).

On each CT scan, the L3 slice was located by an expert radiologist on a dedicated software [LKTM+14], providing the annotation for the position of the L3 through its distance in (mm) from the first slice in the scan (top).

Moreover, 43 supplementary CT scans have been annotated by the same radiologist and 3 other experts, in order to evaluate the variability of annotations among experts.

To be as reproducible and precise as possible, detailed guidelines were given to all radiologists for annotation.

[1] This dataset is available on demand, please contact the corresponding author

[2] IRB Number 1604B.

From all the scans, frontal MIP images have been computed using the process described in 3.1. This results in a set of 642 images of constant width (512 pixels) and variable height, varying from 659 to 1862 pixels. Fig 4 shows some examples of frontal MIP images extracted from three patients of the L3CT1 database.

### *4.2. Datasets preparation*

The first step consists in splitting the dataset into 5 folds, in order to allow a cross-validation procedure. The split is applied at the patient level, in order to prevent that a given CT-scan provides windows in different sets (learning, validation, test), what should lead to biased results. Moreover, due to variable slice thickness in the dataset, we make sure when dividing the dataset to obtain stratified folds. Thus, we end up with the same number of samples from each slice thickness in each set.

Once the MIP images folds have been generated, learning, validation and test windows are sampled as explained in section 3.3, where the value of $a$ has been experimentally set to $a = 50$ using a cross validation procedure. For the validation set, in order to speed up the training, we take only 300 random windows from different patients.

### *4.3. Neural networks models*

In order to conduct our experiments, two types of convolutional neural networks have been compared:

- **Homemade CNN (CNN4):** We have designed and trained a CNN from scratch, with specific architecture of four convolutional layers followed by a fully connected output layer. In each convolution layer, a horizontal max-pooling is performed. We found in practice that vertical max-pooling distorts the target position. The number of kernels that we used in the four convolution layers are $[10, 3, 3, 5]$, with respective sizes $[5, 7, 9, 3]$. The hyper-parameters of our CNN were tuned on the validation set [Ben12]. We refer to our model as $CNN4$.

- **Pre-trained CNNs:** In our study, we have collected a set of pre-trained convolutional neural networks over ImageNet dataset [DDS+09]: Alexnet [KSH12], VGG16 [SZ14], VGG19[SZ14], Googlenet (Inception V1) [SLJ+14][3]. The models are created using the library Keras [Cho15]. For each model, we keep only the convolutional layers which are considered as shared perception layers that may be used for different tasks. On top of that, we add one fully connected layer to be specialized in our specific task (i.e. L3 detection). Our experiments have shown that adding more fully connected layers does not improve the results.

  The input of pre-trained models is supposed to be an RGB image (i.e. a 3D matrix), while in the other hand, our sampled windows are 2D matrix. In order to match the required input, we duplicate the 2D matrix in each color channel. Then, each channel is normalized using its mean from the ImageNet Dataset.

  We use $L_2$ regularization for training all the models with value of $\lambda = 10^{-3}$, except for Googlenet where we used the original regularization values.

## 5. Results

### *5.1. Data view: Frontal Vs. Lateral*

The use of the MIP representation allows us to access to different views of the CT scan, such as the frontal and lateral views (other views with different angles are possible). In order to choose the best view, we re-train a VGG16 model with one fully connected layer using different input views. We recall that the input of the VGG16 is an image with 3 plans. We experimented three configurations. In the first and second cases, we repeat the frontal and lateral views, respectively, in the three input channels. In the last case, we mixed

[3]The weights of Googlenet were obtained from: `https://gist.github.com/joelouismarino/a2ede9ab3928f999575423b9887abd14`, and the weights of the rest of the models were obtained from `https://github.com/heuritech/convnets-keras`

the frontal and the lateral view. The motivation behind the combination of the views is that each view will provide an additional information (hopefully complementary) that will help the model to decide. The sampling margin of the windows is done over the range $[-50, +50]$. Tab.1 shows that using frontal view alone is more suitable. One possible explanation of this results is that the frontal view contains more structural context (ribs, pelvis) which helps to locate the L3 slice, in the opposite of the lateral view. Combining lateral and frontal views gave better results than lateral alone but worse than frontal alone. One may think that lateral view adds noise to the frontal view.

| View | VGG16 |
|---|---|
| | Error $m_e$ (slices) |
| Frontal | $\mathbf{1.71 \pm 1.59}$ |
| Lateral | $4.29 \pm 14.90$ |
| Frontal Lateral Frontal | $1.89 \pm 2.05$ |

Table 1: Test error (mean $\pm$ standard deviation) over the test set of fold 0, expressed in slices, using VGG16 model with frontal and lateral views.

### *5.2. Detection performance*

All the models described in section 4.3 have been evaluated in a cross validation procedure on the L3CT1 dataset by computing the prediction error. The prediction error for one CT scan is computed as the absolute difference between the prediction $y_{pred}$ and the target $y$: $e = |y - y_{pred}|$. The error is expressed in slices. We report the mean and the standard deviation of the test error $(\mu_e, \sigma_e)$, respectively in the form $\mu_e \pm \sigma_e$, over the entire test set. Obtained results are reported in Tab.2.

For the sake of comparison, we used Random Forest Regression (RF) [Bre01, Ho95] as a regressor instead of our CNN. As in most pattern recognition problems, we need to extract input features to train our Random Forest Regression. Local Binary Patterns (LBP) features have shown to be very efficient in many computer vision tasks [OPM02], especially in medical imaging [NLB10]. Therefore, we have retained this feature descriptor. To extract the LBP features we

| | RF500 | CNN4 | Alexnet | VGG16 | VGG19 | Googlenet |
|---|---|---|---|---|---|---|
| fold 0 | $7.31 \pm 6.52$ | $2.85 \pm 2.37$ | $2.21 \pm 2.11$ | $2.06 \pm 4.39$ | $1.89 \pm 1.77$ | $1.81 \pm 1.74$ |
| fold 1 | $11.07 \pm 11.42$ | $3.12 \pm 2.90$ | $2.44 \pm 2.41$ | $1.78 \pm 2.09$ | $1.96 \pm 2.10$ | $3.84 \pm 12.86$ |
| fold 2 | $13.10 \pm 13.90$ | $3.12 \pm 3.20$ | $2.47 \pm 2.38$ | $1.54 \pm 1.54$ | $1.65 \pm 1.73$ | $2.62 \pm 2.52$ |
| fold 3 | $12.03 \pm 14.34$ | $2.98 \pm 2.38$ | $2.42 \pm 2.23$ | $1.96 \pm 1.62$ | $1.76 \pm 1.75$ | $2.22 \pm 1.79$ |
| fold 4 | $8.99 \pm 7.83$ | $1.87 \pm 1.58$ | $2.69 \pm 2.41$ | $1.74 \pm 1.96$ | $1.90 \pm 1.83$ | $2.20 \pm 2.20$ |
| Average | $10.50 \pm 10.80$ | $2.78 \pm 2.48$ | $2.45 \pm 2.42$ | $\mathbf{1.82 \pm 2.32}$ | $1.83 \pm 1.83$ | $2.54 \pm 4.22$ |

Table 2: Error expressed in slice over all the folds using different models: RF500, CNN4 (Homemade model), and Alexnet/VGG16/VGG19/GoogleNet (Pre-trained models).

used a number of neighbors of 8 and a radius of 3 which creates an input feature vector with dimension of $2^8 = 256$. From each sampled window, we extract LBP features. We investigated different number of trees: 10, 100 and 500. The obtained results showed that random forests do not perform well over this task. We report in Tab.2 the results using 500 ($RF500$) trees which are in the same order of performance compared the other cases (i.e. 10 and 100 trees).

From Tab.2, one can see that pre-trained models perform better than our homemade CNN4 with an improvement of about 35%. In particular, VGG16 showed the best results by an average error of $1.82 \pm 2.32$ followed by VGG19 with $1.83 \pm 1.83$. This result confirms the strong benefit of transfer learning between two different tasks. Moreover, it shows that the convolutional layers can be shared as a perception tool between different tasks with slight adaptation. On the other hand, this illustrates the capability for modeling such task using the pre-trained models.

### *5.3. Processing time issues*

One must mention that the price we paid in order to reach the performance mentioned above is to increase the complexity of the model. In Table 3, we present the number of parameters of each model and the average required time for the prediction of the L3 slice. We observe that VGG16 contains approximately 264 times more parameters than CNN4. Beside the required memory

for such models, the real paid cost is the evaluation time during the test phase. Computed on a GPU (Tesla K40), VGG16 requires an average of 13.28 seconds per CT scan while our CNN4 only needs 4.46 second per CT scan.

| | Number of parameters | Average forward pass time (seconds/CT scan) |
|---|---|---|
| CNN4 | 55,806 | 04.46 |
| Alexnet | 2,343,297 | 06.37 |
| VGG16 | 14,739,777 | 13.28 |
| VGG19 | 20,049,473 | 16.02 |
| Googlenet | 6,112,051 | 17.75 |

Table 3: Number of parameters for different models and average forward pass time per CT scan.

An important factor which affects the evaluation time in these experiments is the number of windows processed by the CNN for a given CT scan. Thus, it is possible to dramatically reduce the computation time by shifting the window by a bigger value than 1 pixel. An experimental evaluation of this strategy with VGG16 has shown that a good compromise between processing time and performance could be obtained for a shift value up to 6 pixels without affecting the localization precision. This sub-sampling reduces the evaluation time from 13.28 seconds/CT scan to 2.36 seconds/CT scan and moved the average localization error from $1.82 \pm 2.32$ slices to $1.91 \pm 2.69$ slices, respectively. This shows the robustness of the proposed correlation post-processing.

### *5.4. Comparison with radiologists*

In order to further assess the performance of the proposed approach, an extra set of 43 CT scans was used for test. This particular dataset was annotated by the same radiologist who annotated L3CT1 dataset and also by three other experts. Each annotation was performed at two different times, in order to evaluate the intra-annotator variability. We refer to both annotations by the same expert by *Review 1* and *Review 2*.

Obtained results are illustrated in Tab.4. It compares the error made by CNN models with those made by the radiologists, using the radiologist who annotated the L3CT1 dataset as reference. These results corroborate the results

provided in Table 2 since VGG16 is better than CNN4 with an improvement of about 35% in average for both reviews. The results also demonstrate that radiologists are in average more precise than automatic models with an improvement of about 50%. However, they also show that there exists some variabilities among radiologist annotations and even an intra-annotator variability. This latter is visible in Tab. 4 since computed errors for automatic systems vary between both reviews while the automatic system gives the same output, showing that reference values have changed. This illustrates the difficulty of the task of precisely locating the L3 slice and the interest of CNN which does not change its prediction.

| Errors (slices) / operator | CNN4 | VGG16 | Ragiologist #1 | Radiologist #2 | Radiologist #3 |
|---|---|---|---|---|---|
| Review1 | $2.37 \pm 2.30$ | $1.70 \pm 1.65$ | $0.81 \pm 0.97$ | $0.72 \pm 1.51$ | $0.51 \pm 0.62$ |
| Review2 | $2.53 \pm 2.27$ | $1.58 \pm 1.83$ | $0.77 \pm 0.68$ | $0.95 \pm 1.61$ | $0.86 \pm 1.30$ |

Table 4: Comparison of the performance of both the automatic systems and radiologists. The L3 annotations given by the reference radiologist vary between the two reviews.

## 6. Conclusion

In this paper, we proposed a new and generic pipeline for spotting a particular slice in a CT scan. In our work, we applied our approach to the L3 slice, but it can easily be generalized to other slices, provided a labeled dataset is available.

First, the CT scan is converted into a frontal Maximum Intensity Projection (MIP) image. Afterwards, this representation is processed in a sliding window fashion to be fed to a CNN which is trained using Transfer Learning. In the test phase, all the predictions concerning the position of the L3 within the sliding windows are merged into a robust post-processing stage to take the final decision about the position of the L3 slice in the full CT scan.

Obtained results show that the approach is efficient to precisely detect the target slice. Using a fine-tuned VGG16 network coupled with an adequate decision strategy, the average error is under 2 slices where experienced radi-

ologists can provide annotations that differ of about 1 slice. The computing time is within an acceptable range for clinical applications, and can be further reduced by (i) increasing the shift value (ii) adapting the network architecture by pre-training smaller networks over ImageNet, for example, which has not been studied in this work (iii) and prun the final trained CNN by dropping the less important filters. Recently, prunning CNNs has seen a lot of attention in order to deploy large CNNs on devices with less computation resource. We are currently working on this idea to speedup more the computation.

This contribution confirms the interest of using machine learning and more particularly deep learning in medical problems. One of the main reasons deep learning is not popular in medical domain is the lack of training data. Pre-training the networks over other large dataset will strongly alleviate this problem and encourage the use of such efficient models.

## References


[AWM+14] Janice L Atkins, Peter H Whincup, Richard W Morris, Lucy T Lennon, Olia Papacosta, and S Goya Wannamethee. Sarcopenic obesity and risk of cardiovascular disease and mortality: a population-based cohort study of older men. *Journal of the American Geriatrics Society*, 62(2):253–60, February 2014.

[BDWG15] Yaniv Bar, Idit Diamant, Lior Wolf, and Hayit Greenspan. Deep learning with non-medical training used for chest pathology identification. *Proc. SPIE, Medical Imaging: Computer-Aided Diagnosis*, 9414:94140V–7, 2015.

[Ben12] Yoshua Bengio. Practical recommendations for gradient-based training of deep architectures. In Grégoire Montavon, Genevieve B. Orr, and Klaus-Robert Müller, editors, *Neural Networks: Tricks of the Trade (2nd ed.)*, volume 7700 of *Lecture Notes in Computer Science*, pages 437–478. Springer, 2012.

[Bre01] Leo Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, October 2001.

[CC06] P. Checco and F. Corinto. Cnn-based algorithm for drusen identification. In *International Symposium on Circuits and Systems*, 2006.

[CCB+09] Howard Chung, Dana Cobzas, Laura Birdsell, Jessica Lieffers, and Vickie Baracos. Automated segmentation of muscle and adipose tissue on CT images for human body composition analysis. *Proceedings of SPIE*, 7261:72610K–72610K–8, 2009.

[Cho15] François Chollet. Keras. `https://github.com/fchollet/keras`, 2015.

[CMS12] D. C. Cireşan, U. Meier, and J. Schmidhuber. Transfer learning for latin and chinese characters with deep neural networks. In *International Joint Conference on Neural Networks*, pages 1–6, 2012.

[CWJ+15] A.R. Cunliffe, B. White, J. Justusson, C. Straus, R. Malik, Al-H.A. Hallaq, and S.G. Armato. Comparison of Two Deformable Registration Algorithms in the Presence of Radiologic Change Between Serial Lung CT Scans. *Journal of Digital Imaging*, 28(6):755–760, 2015.

[DDS+09] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.

[ESTA14] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. In *CVPR*, pages 2155–2162, 2014.

[GACD11] Subarna Ghosh, Raja' S. Alomari, Vipin Chaudhary, and Gurmeet Dhillon. Automatic lumbar vertebra segmentation from clinical CT for wedge compression fracture diagnosis. *Proceedings of the SPIE*, 3:796303–9, 2011.

[GDE+13] B. Glocker, D.Zikic, E.Konukoglu, D.R. Haynor, and A. Criminisi. Vertebrae localization in pathological spine CT via dense classification from sparse annotations. *MICCAI*, 16(Pt 2):262–70, 2013.

[GFC+12] Ben Glocker, J. Feulner, Antonio Criminisi, D. R. Haynor, and E. Konukoglu. *Automatic Localization and Identification of Vertebrae in Arbitrary Field-of-View CT Scans*, pages 590–598. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[GLC+13] S Gouérant, M Leheurteur, M Chaker, R Modzelewski, O Rigal, C Veyret, G Lauridant, and F Clatot. A higher body mass index and fat mass are factors predictive of docetaxel dose intensity. *Anticancer research*, 33(12):5655, 2013.

[GVC09] S. Golodetz, I. Voiculescu, and S. Cameron. Automatic spine identification in abdominal CT slices using image partition forests. *International Symposium on Image and Signal Processing and Analysis*, 2009.

[GZH14] Ben Glocker, Darko Zikic, and David R. Haynor. *Robust Registration of Longitudinal Spine CT*, pages 251–258. Springer International Publishing, 2014.

[HCLN09] Szu H. Huang, Yi Hong Chu, Shang Hong Lai, and Carol L. Novak. Learning-Based Vertebra Detection and Iterative Normalized-Cut Segmentation for Spinal MRI. *IEEE Transactions on Medical Imaging*, 28(10):1595–1605, 2009.

[HDW+15] M. Havaei, A. Davy, D. Warde-Farley, A. Biard, A.C. Courville, Y. Bengio, C. Pal, P.-M. Jodoin, and H. Larochelle. Brain tumor segmentation with deep neural networks. *CoRR*, abs/1505.03540, 2015.

[HJ13] Gary B. Huang and Viren Jain. Deep and wide multiscale recursive networks for robust image labeling. *CoRR*, abs/1310.0354, 2013.

[Ho95] Tin Kam Ho. Random decision forests. In *Proceedings of the Third International Conference on Document Analysis and Recognition (Volume 1) - Volume 1*, ICDAR '95, pages 278–, Washington, DC, USA, 1995. IEEE Computer Society.

[HSO16] Luiz G. Hafemann, Robert Sabourin, and Luiz S. Oliveira. Writer-independent feature learning for offline signature verification using deep convolutional neural networks. *CoRR*, abs/1604.00974, 2016.

[Jia15] Xiang Jiang. Representational transfer in deep belief networks. In *28th Canadian Conference on Artificial Intelligence*, pages 338–342, 2015.

[KLP11] Samuel Kadoury, Hubert Labelle, and Nikos Paragios. Automatic inference of articulated spine models in CT images using high-order markov random fields. *Medical Image Analysis*, 15(4):426–437, 2011.

[KOF+13] Toshimi Kaido, Kohei Ogawa, Yasuhiro Fujimoto, Y Ogura, K Hata, T Ito, K Tomiyama, S Yagi, A Mori, and S Uemoto. Impact of sarcopenia on survival in patients undergoing living donor liver transplantation. *American Journal of Transplantation*, 13(6):1549–1556, 2013.

[KSH12] A. Krizhevsky, I. Sutskever, and G.E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *NIPS 25*, pages 1097–1105. 2012.

[Lai15] Matthew Lai. Deep learning for medical image segmentation. *CoRR*, abs/1505.02000, 2015.

[LHC+15] J Lerouge, R Herault, C Chatelain, F Jardin, and R Modzelewski. IODA : An input / output deep architecture for image labeling. *Pattern Recognition*, 48(9):2847–2858, 2015.

[LKTM+14] Hélène Lanic, Jerôme Kraut-Tauzia, Romain Modzelewski, Florian Clatot, Sylvain Mareschal, Jean Michel Picquenot, Aspasia Stamatoullas, Stéphane Leprêtre, Hervé Tilly, and Fabrice Jardin. Sarcopenia is an independent prognostic factor in elderly patients with diffuse large b-cell lymphoma treated with immunochemotherapy. *Leukemia & Lymphoma*, 55(4):817–823, 2014.

[MBH+98] N Mitsiopoulos, R N Baumgartner, S B Heymsfield, W Lyons, D Gallagher, and R Ross. Cadaver validation of skeletal muscle measurement by magnetic resonance imaging and computerized tomography. *Journal of applied physiology*, 85(1):115–122, 1998.

[MBM+13] Lisa Martin, Laura Birdsell, Neil MacDonald, Tony Reiman, M. Thomas Clandinin, Linda J. McCargar, Rachel Murphy, Sunita Ghosh, Michael B. Sawyer, and Vickie E. Baracos. Cancer cachexia in the age of obesity: Skeletal muscle depletion is a powerful prognostic factor, independent of body mass index. *Journal of Clinical Oncology*, 31(12):1539–1547, 2013.

[MHSB13] David Major, Jiří Hladůvka, Florian Schulze, and Katja Bühler. Automated landmarking and labeling of fully and partially scanned spinal columns in CT images. *Medical Image Analysis*, 17(8):1151–1163, 2013.

[ML13] Jun Ma and Le Lu. Hierarchical segmentation and identification of thoracic vertebra using learning-based edge detection and coarse-to-fine deformable model. *Computer Vision and Image Understanding*, 117(9):1072–1083, 2013.

[MMB+08] Christopher Malon, Matthew Miller, Harold Christopher Burger, Eric Cosatto, and Hans Peter Graf. Identifying histological elements with convolutional neural networks. In *Int. Conf. on Soft Computing As Transdisciplinary Science and Technology*, pages 450–456, 2008.

[MT96] T. McInerney and D. Terzopoulos. Deformable models in medical image analysis: a survey. *Medical image analysis*, 1(2):91–108, 1996.

[MWK+13] B. Michael Kelm, Michael Wels, S. Kevin Zhou, Sascha Seifert, Michael Suehling, Yefeng Zheng, and Dorin Comaniciu. Spine detection in CT and MR using iterated marginal space learning. *Medical Image Analysis*, 17(8):1283–1292, 2013.

[NLB10] Loris Nanni, Alessandra Lumini, and Sheryl Brahnam. Local binary patterns variants as texture descriptors for medical image analysis. *Artificial Intelligence in Medicine*, 49(2):117 – 125, 2010.

[OA11] Ayse Betul Oktay and Yusuf Sinan Akgul. Localization of the lumbar discs using machine learning and exact probabilistic inference. In *Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics*, pages 158–165, 2011.

[OPM02] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, Jul 2002.

[PVT+11] Peter D. Peng, Mark G. Van Vledder, Susan Tsai, Mechteld C. De Jong, Martin Makary, Julie Ng, Barish H. Edil, Christopher L. Wolfgang, Richard D. Schulick, Michael A. Choti, Ihab Kamel, and Timothy M. Pawlik. Sarcopenia negatively impacts short-term outcomes in patients undergoing hepatic resection for colorectal liver metastasis. *HPB*, 13(7):439–446, 7 2011.

[PXP00] Dzung L Pham, Chenyang Xu, and Jerry L Prince. Current methods in medical image segmentation 1. *Annual review of biomedical engineering*, 2(1):315–337, 2000.

[RHGS15] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *NIPS 28*, pages 91–99, 2015.

[RYL+14] Holger R. Roth, Jianhua Yao, Le Lu, James Stieger, Joseph E. Burns, and Ronald M. Summers. Detection of sclerotic spine metastases via random aggregation of deep convolutional neural network classifications. *CoRR*, abs/1407.5976, 2014.

[SEM16] Antonis D. Savva, Theodore L. Economopoulos, and George K. Matsopoulos. Geometry-based vs. intensity-based medical image registration: A comparative study on 3D CT data. *Computers in Biology and Medicine*, 69:120–133, 2016.

[SLJ+14] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going Deeper with Convolutions. *CoRR*, abs/1409.4842, 2014.

[SPW+04] Wei Shen, Mark Punyanitya, ZiMian Wang, Dympna Gallagher, Marie-Pierre St-Onge, Jeanine Albu, Steven B Heymsfield, and Stanley Heshka. Total body skeletal muscle and adipose tissue volumes: estimation from a single abdominal cross-sectional image. *Journal of applied physiology*, 97(6):2333–2338, 2004.

[SRG+16] H. C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers. Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning. *IEEE Transactions on Medical Imaging*, 35(5):1285–1298, 2016.

[STE13] C. Szegedy, A. Toshev, and D. Erhan. Deep neural networks for object detection. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *NIPS 26*, pages 2553–2561. 2013.

[SZ14] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[UBHK14] G. Urban, M. Bendszus, Fred A. Hamprecht, and J. Kleesiek. Multi-modal brain tumor segmentation using deep convolutional neural networks. In *MICCAI BraTS Challenge Proceedings*, pages 31–35, 2014.

[Wal92] Jerold W. Wallis. *Cardiovascular Nuclear Medicine and MRI: Quantitation and Clinical Applications*, pages 89–100. Springer Netherlands, 1992.

[WM91] JW Wallis and TR Miller. Three-dimensional display in nuclear medicine and radiology. *Journal of nuclear medicine : official publication, Society of Nuclear Medicine*, 32(3):534–546, March 1991.

[WMLK89] J. W. Wallis, T. R. Miller, C. A. Lerner, and E. C. Kleerup. Three-dimensional display in nuclear medicine. *IEEE Trans. on Medical Imaging*, 8(4):297–230, Dec 1989.

[YDM+15] Connie Yip, Charlotte Dinkel, Abhishek Mahajan, Musib Siddique, Gary Cook, and Vicky Goh. Imaging body composition in cancer patients: visceral obesity, sarcopenia and sarcopenic obesity may impact on clinical outcome. *Insights into Imaging*, pages 489–497, 2015.