May 2018

Selective Compression of Medical Images via Intelligent Segmentation and 3D-SPIHT Coding

Bohan Fan
University of Wisconsin-Milwaukee

SELECTIVE COMPRESSION OF MEDICAL IMAGES VIA INTELLIGENT SEGMENTATION AND 3D-SPIHT CODING

by

Bohan Fan

A Thesis Submitted in

Partial Fulfillment of the

Requirements for the Degree of

Master of Science

in Computer Science

at

The University of Wisconsin-Milwaukee

May 2018

ABSTRACT

SELECTIVE COMPRESSION OF MEDICAL IMAGES VIA INTELLIGENT SEGMENTATION AND 3D-SPIHT CODING

by

Bohan Fan

The University of Wisconsin-Milwaukee, 2018
Under the Supervision of Professor Zeyun Yu

With increasingly high resolutions of 3D volumetric medical images being widely used in clinical patient treatments, efficient image compression techniques have become in great demand due to the cost in storage and time for transmission. While various algorithms are available, the conflicts between high compression rate and the downgraded quality of the images can partially be harmonized by using the region of interest (ROI) coding technique. Instead of compressing the entire image, we can segment the image by critical diagnosis zone (the ROI zone) and background zone, and apply lossless compression or low compression rate to the former and high compression rate to the latter, without losing much clinically important information.

In this thesis, we explore a medical image transmitting process that utilizes a deep learning network, called 3D-Unet to segment the region of interest area of volumetric images and 3D-SPIHT algorithm to encode the images for compression, which can be potentially used in medical data sharing scenario. In our experiments, we train a 3D-Unet on a dataset of spine images with their label ground truth, and use the trained model to extract the vertebral bodies of testing data. The segmented vertebral

regions are dilated to generate the region of interest, which are subject to the 3D-SPIHT algorithm with low compress ratio while the rest of the image (background) is coded with high compress ratio to achieve an excellent balance of image quality in region of interest and high compression ratio elsewhere.

TABLE OF CONTENTS

LIST OF FIGURES vii

LIST OF TABLES viii

ACKNOWLEDGMENTS ix

LIST OF ABBREVIATIONS x

Chapter I Introduction 1

Chapter II ROI extraction 4

2.1 Traditional ROI extraction techniques 4

2.2 Artificial neural network-based techniques 6

2.2.1 Convolution neural network (CNN) 7

2.2.2 Fully convolutional network (FCN) 8

2.3 U-net 10

2.3.1 3D-Unet 11

2.3.2 3D-Unet Implementation 12

Chapter III Compression techniques 14

3.1 ROI coding techniques 14

3.2 Set Partitioning in Hierarchical Trees (SPIHT) 15

3.3 QccPack 17

Chapter IV Data and experiment 19

4.1 Dataset 19

4.2 Process of medical image transmitting 21

4.3 3D-Unet module 23

4.4 Format transformation module 25

4.4.1 From volumetric images to icb files 26

4.4.2 From icb files to volumetric images 31

4.5 The comparison of original image and bit files sizes 35

Chapter V Limitations and future work 37

References 39

LIST OF FIGURES

Figure 1. Example of a convolution neural network structure for classification 7

Figure 2(a). Comparison of convolution neural network and fully convolutional network structures 9

Figure 2(b). FCN process to output the semantic segmentation result 10

Figure 3. Example of U-Net structure 11

Figure 4. Example of 3D-Unet structure 12

Figure 5. The spatial orientation tree structure of SPIHT 16

Figure 6. The spatial orientation tree structure of SPIHT and 3D-SPIHT..........17

Figure 7(a). 3D visualization of opp.nii volumetric image by 3D Slicer 20

Figure 7(b). Vertebral bodies ground truth label of original volumetric image by 3D Slicer 21

Figure 8. Interaction between the medical images user and data provider..........23

Figure 9. 3D visualization of the model predicting label file 25

Figure 10. The 8th slice from opp.nii volumetric image..........26

Figure 11(a). Label slice from 3D-Unet prediction..........28

Figure 11(b). Dilated label image by kernel 7*7 and 1 iteration 28

Figure 12(a). Reconstruction image of slice 21 with 8 bpv for ROI and 0.01 bpv for background..........32

Figure 12(b). Reconstruction image of slice 21 with 8 bpv for ROI and 0.1 bpv for background..........32

Figure 12(c). Reconstruction image of slice 21 with 8 bpv for ROI and 1 bpv for background..........33

Figure 12(d). Reconstruction image of slice 21 with 8 bpv for ROI and 8 bpv for background..........33

Figure 12(e). Original image of slice 21 34

Figure 13. The plot of different information loss measures for different bpv 35

LIST OF TABLES

Table 1. 3DUnetCNN dependencies list 13

Table 2. 3DUetCNN parameter settings 24

Table 3. Bit file size of 3D-SPIHT coding with ROI and background mask for different bpv 30

Table 4. The comparison of different measures of information loss for different bpv 34

Table 5. Total size of bit files required in transmitting and the compression ratio for different bpv choices in background 36

ACKNOWLEDGMENTS

I will always be grateful to my advisor, Professor Zeyun Yu, for his excellent guidance, patience, caring, and inspiring me of working on this thesis. This thesis project also benefited from conversations with many intelligent people in the Biomedical Modeling and Visualization Laboratory at the University of Wisconsin-Milwaukee. I would also like to thank our lab group for sharing their GPU, their knowledge, and their insightful suggestions. This thesis would not have been possible without the help from faculties and friends in UWM, and support from my family.

LIST OF ABBREVIATIONS

CNN	Convolution Neural Network
CT	Computed Tomography
EZW	Embedded Zerotrees of Wavelet transforms
FCN	Fully Convolutional Network
MRC	Mixed Raster Content
MRI	Magnetic Resonance Imaging
Nifti	Neuroimaging Informatics Technology Initiative
RIO	Region of Interest
SPIHT	Set Partitioning in Hierarchical Trees

Chapter I Introduction

Since X-ray was discovered by Wilhelm Röntgen in 1895, people have been able to see through the body by taking images. From the single image of X-ray to Computed Tomography (CT) scan, Medical Ultrasonography and Magnetic Resonance Imaging (MRI), we are increasingly benefiting from the accurate diagnoses and treatments with the assistant of medical imaging.

Medical volumetric images, such as 3D computed tomography (CT) and magnetic resonance (MR) images, are typically a group of 2D image slices through the human body, and through these stacked slices, 3D reconstruction of the anatomy images can be acquired for comprehensive clinical analysis. Because of the huge volume and increasingly use of medical volumetric images, efficient compression techniques are necessary for storage and transmission.

There are always conflicts between high compression rate and the downgraded quality of the images. Their balance can be partially harmonized by region of interest (ROI) coding. Instead of compressing the entire image with single rate, we can divide the image by critical diagnosis zone (the ROI zone) and background zone, and apply lossless compression or low compression rate to the former and high compression rate to the latter, without losing too much clinically important information.

ROI coding relies on image segmentation techniques which extract the ROI from the background. Image segmentation techniques include layer-based segmentation, such as 3-layer MRC (Mixed Raster Content) model [6], and block-based segmentation, including clustering, split and merge, normalized cuts, region growing, threshold, edge detection techniques [8]. Among them, artificial neural network-based techniques, represented by Convolutional Neural Network (CNN), Recurrent neural Network (RNN), Fully Convolutional Network(FCN), have progressed rapidly in recent years. Superior to the traditional image processing techniques, these artificial neural network-based techniques can be fed with raw data, automatically learn the features and make predictions [27]. Therefore, artificial neural network-based techniques are promising in medical images processing context, which has tremendous images and requires extensive and tedious efforts of medical experts [27].

U-net [14] and Set Partitioning in Hierarchical Trees (SPIHT) [1] are the state-of-art medical image segmentation technique and image compression algorithm respectively. Their extension to 3D version-3D-Unet [3] and 3D-SPIHT [2], can be applied to volumetric images.

In this thesis, we explore a medical image transmitting process that utilizes 3D-Unet to segment the region of interest area of volumetric images and 3D-SPIHT algorithm to encode the ROI with low compress ratio while the rest of the image (background) with high compress ratio, and then transmits the generated bit files to the user for decoding and reconstructing. This can be potentially used in medical data sharing scenario.

In Chapter II we review the image segmentation techniques for ROI extraction from traditional ROI extraction techniques to the state-of-art deep learning methods. We choose one of the most efficient techniques in medical image segmentation: 3D-Unet [3], to train and predict on spine volumetric images. In Chapter III we discuss several compression techniques and specifically introduce 3D-SPIHT [2] algorithm and implement it using QccPack [4]. In Chapter IV we use Nifti volumetric images with labeled vertebral bodies dataset to demonstrate the data transmitting process. Chapter V concludes with some limitations and future work.

Chapter II ROI extraction

2.1 Traditional ROI extraction techniques

Region of Interest (ROI) is a subset of an image or a dataset identified for a particular purpose [5]. From the users' perspective, it is the area that contains the needed information that the users are searching for. In image processing context, for example, it may refer to time or frequency interval of a waveform in 1D case, or a rectangle or circle area contains an object in the image [5]. The ROI boundary divides an image into two parts: RIO area and Non-ROI area (or background), both of which are useful for image processing. The ROI area contains the main targeting information for the user, while the background, acting as a map, usually provides the relative location of the ROI area in the image. Due to the differences of relative importance and roles played in image processing, ROI and Non-ROI areas can be treated differently. In compression, we can apply the low compression rate for the former and high compression rate for the latter.

The different treatments of ROI and Non-ROI areas rely on image segmentation techniques. In general, there are two categories of image segmentation techniques: layer-based methods and block-based methods. Layer-based segmentation, most represented by a 3-layer MRC (Mixed Raster Content) model [6], divides images into foreground, background and binary mask. The binary mask determines the assignment of each pixel to the foreground, or background by 1 or 0 [7], and each of the layers is encoded independently with its own encoder.

Block-based segmentation, including clustering, split and merge, normalized cuts, region growing, threshold, edge detection techniques [8], divides images by continuous region or edge of the objects in the images. Clustering is to divide an image into clusters such that the total distance between each point and the center of its cluster is minimized. The drawback of clustering techniques like K-means clustering is that the cluster area may not be continuous [9]. Split and merge technique divides an image into sub regions until they are small enough for segmentation, then merges two regions if they are adjacent and similar until no more merge is possible [10]. Split and merge technique can guarantee the continuity of segmentation area, but the results are position and orientation dependent and also suffer from over segmentation problem [8,10]. Normalized cuts [11] considers image segmentation as a graph partition problem and is a global criterion to be optimized which measures both the dissimilarity between different groups and similarity within each groups. In region growing technique, the user chooses one pixel as a seed and adds other pixels based on similarity [9]. If a pixel does not belong to any current region, then a new region is created. The process is repeated until all the pixels in the image are added to some region [9]. Threshold technique is to partition image pixels into foreground or background according to its intensity value and a specified threshold value. Threshold value can be chosen based on the users' observation and applied to the entire image or local region. Edge detection techniques detect the abrupt changes of intensity value [9] in images which is usually the boundary of two regions. Different methods can be used to extract the edge: Roberts detection, Prewitt detection, Sobel detection, as well as fuzzy logic based approach, genetic algorithm approach and neural network approach.

Superior to these traditional image segmentation techniques, artificial neural network-based techniques can be fed with raw data, automatically learn the features and make predictions [27]. Therefore, artificial neural network-based techniques are intelligent when dealing with tremendous medical images and performing extensive and tedious work.

2.2 Artificial neural network-based techniques

Artificial neural network has been inspired by the research of biological neural network since 1940s. McCulloch and Pitts [28] proposed a simplified model of neuron as the simulation of biological neuron (or nerve cell). The neuron can be mathematically represented by a weighted sum of n input signals, x_i, i=1, 2, …n,

$$y = \theta(\sum_{i=1}^{n} w_i x_i - u)$$

θ(z)=1, if z>0 and θ(z)=0, otherwise.

A positive weight w_i is the excitatory input, while a negative weight is the inhibitory input [31]. Despite its simplicity, the computational model inspired and paved the way for the neural network research. [32] introduced the first Perceptron, a single layer network that is able to learn to output 0 or 1 classification [33]. However, a rigorous analysis on the limitations of Perceptron in [34] shows that it can not solve the simple linearly inseparable problem like XOR problem, which brings the AI research to winter. The renewed interest in artificial neural network is triggered by backpropagation algorithm [34][35][36], which shows how multilayer neural networks can be trained for complex learning problems [33]. However, multilayer neural networks gradually gave its way to SVM and other machine learning models because of its difficulties in training and the limitation of compute capacity. The latest boom of artificial neural network comes with the success of convolution neural network (CNN).

2.2.1 Convolution neural network (CNN)

The first convolution neural network (CNN) LeNet [20] is proposed by LeCun in 1998, but it did not become popular because of the limitation of compute capacity at that time. The breakthrough in image processing area is widely awarded to AlexNet [12]. With millions of labeled high-resolution images in ImageNet [13], the availability of high performance GPUs and the optimization on CNN network, AlexNet [12] won ImageNet Large Scale Visual Recognition Challenge with an overwhelming lead. Different from the traditional machine learning techniques, in which people need to use their knowledge and experience to extract features, CNN can learn and extract features automatically in training.

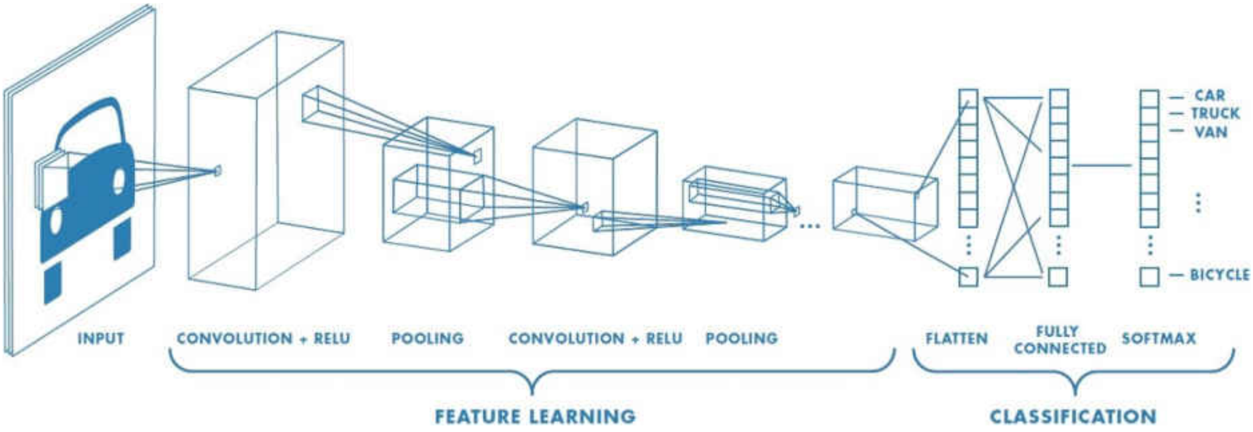

Figure 1. Example of a convolution neural network structure for classification

Source: https://ww2.mathworks.cn/discovery/convolutional-neural-network.html

Figure 1 shows an example of a CNN structure for classification. A CNN typically includes feature learning part and classification part. Features extraction and selection are achieved by alternative of convolution layers and max pooling layers. Using different filters to apply the convolution operation over the original images, we can detect features related to the filter. Simple features go through multiple of convolution layers and max pooling layers and output high-level features. After flattening the high-dimension features, the fully connected layer connects the flattened vector with softmax layer which outputs the probabilities of different classifications. The class corresponding with the highest probability is the predicting classification of the input image.

2.2.2 Fully convolutional network (FCN)

Fully convolutional network (FCN) [21], by its name, is a neural network that contains only convolution layer, instead of ending with a fully connected layer like CNN (see Figure 1). The reason for this network construction is that FCN is designed for the probability prediction of pixels—how likely a pixel belongs to class A or object A.

In Figure 2(a), we can see from the comparison that CNN ends up with a probability distribution in several classes for the main object in the input image through flatting the feature map and fully connected layer. The CNN result shows that tabby cat has the highest probability than any other classes, thus the object in the image is classified as tabby cat. While an FCN generates a heating map of original image after several convolution layers and pooling layers and replacing the fully connected layers by their equivalent convolution layers. The heating map represents the probability that tabby cat appears at different locations in the measure of pixels, where the red pixels has the highest probability to be tabby cat. Based on the heating map and upsampling layers, an FCN generates the segregation of objects by pixels for the original image. Instead of a classification task of CNN, which matches the main object of an image with classes labels, FCN aims at segmenting and labeling all the meaningful objects in an image by pixel (Figure 2(b)), which is called semantic segmentation task.

Since FCN was proposed in 2015, it has inspired a series of neural networks designed for image segmentation, such as SegNet [24], U-net [14], DeconvNet [25]. Among them, U-net [14] works especially well for medical image segmentation.

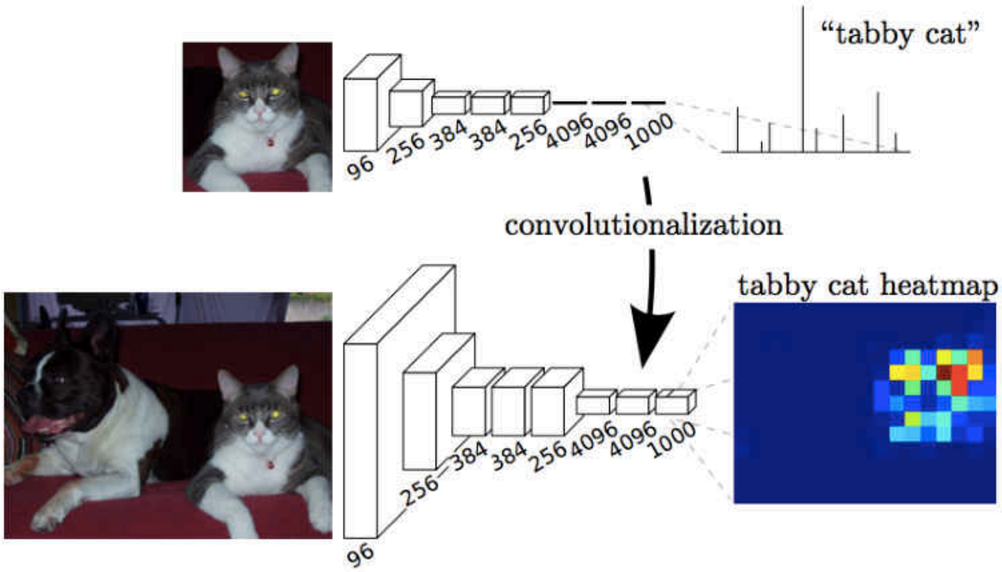

Figure 2(a). Comparison of convolution neural network and fully convolutional network structures

Source: Long, Shelhamer and Darrell (2015)

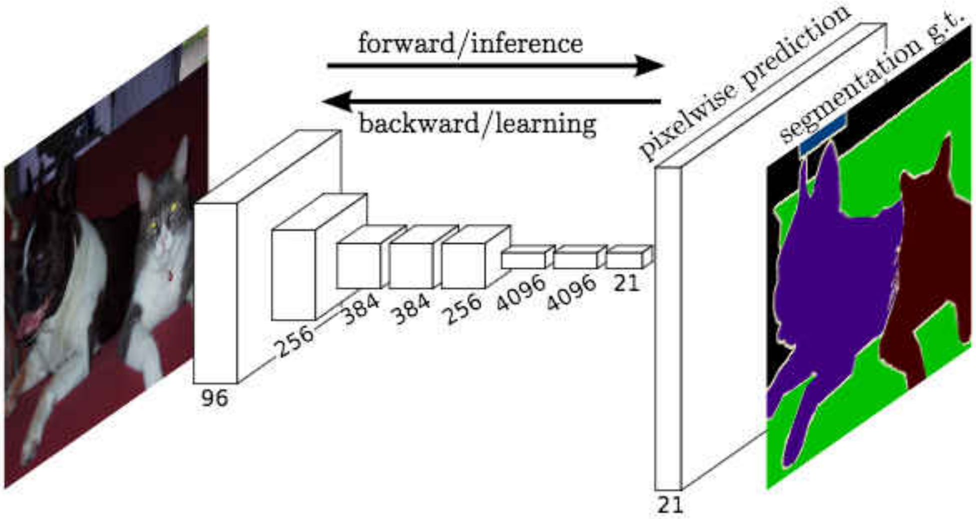

Figure 2(b). FCN process to output the semantic segmentation result

Source: Long, Shelhamer and Darrell (2015)

2.3 U-net

The successful training of a deep network usually requires large training samples with annotation [14], which is especially rare in medical image processing context. By using data augmentation to available training images, [14] gets their elastic deformations, and U-net is proposed to learn the augmented data set. U-net is based on FCN [21], but it extends the FCN architecture by modifying the upsampling part such that it also has many feature channels. Figure 3 shows an example of U-net [14] structure, it has both a contracting path (left) as usual convolution network and expanding path that include upsampling part (right), which form a U shape architecture.

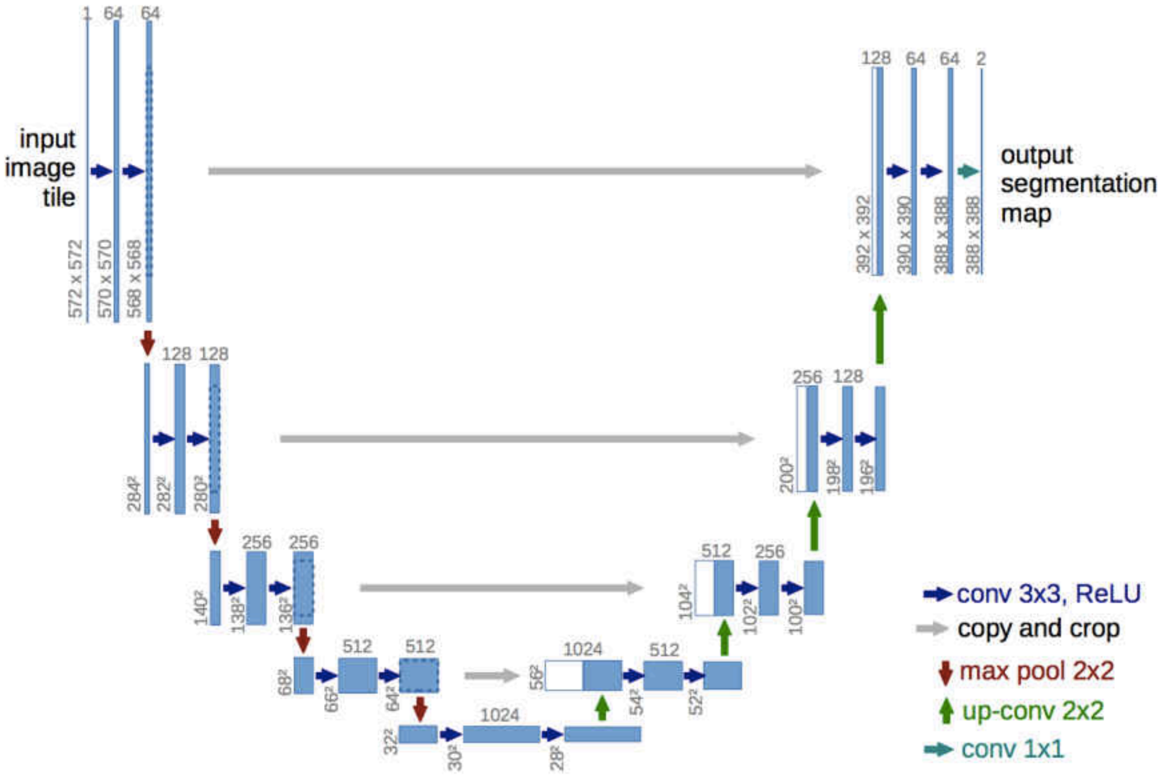

Figure 3. Example of U-Net structure

Source: Ronneberger, Fischer & Brox (2015)

2.3.1 3D-Unet

3D-Unet [3] is the 3D version of original U-net [14], it is proposed for the reason that medical images usually have the form of 3D volumetric type such as Nifti, Dicom, instead of 2D images. The high quality annotated training dataset is even harder to acquire because it involves labeling the region of interest slice by slice for volumetric images, which is both tedious and time consuming [3]. 3D-Unet takes volumetric images as input and can be trained to learn from sparsely annotated images and predict the dense 3D segmentation [3]. A trained 3D-Unet can also apply to a new volumetric image and output its dense 3D segmentation. Figure 4 shows an example of 3D-Unet structure.

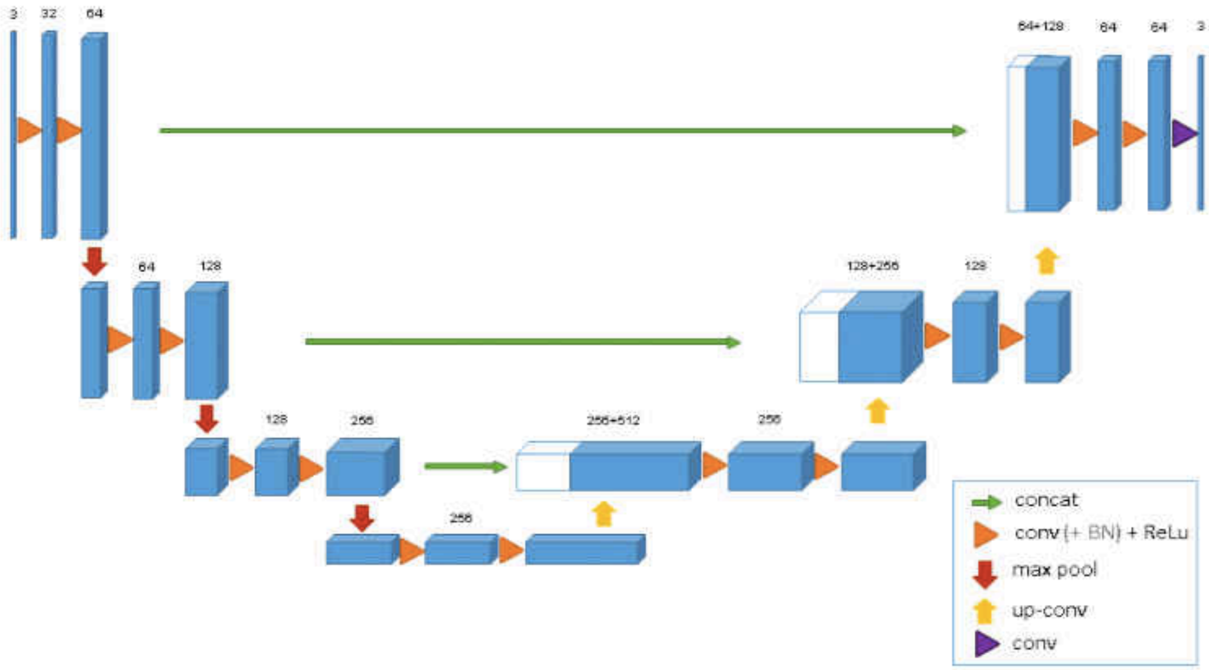

Figure 4. Example of 3D-Unet structure

Source: Çiçek, Abdulkadir, Lienkamp, Brox & Ronneberger (2016)

2.3.2 3D-Unet Implementation

3DUnetCNN is a Github repository developed by David G Ellis and available at [37]. The repository is designed after [14] and written to be trained using BRATS [15, 16] data for segmenting brain tumors. We modify the code for our own application in spine data to segment vertebral bodies. The code is TensorFlow/Keras based so they also need to be installed first.

Tensorflow is an open source software library for high performance numerical computing using data flow graphs. "The graph nodes represent mathematical operations and graph edges represent multidimensional data arrays (tensors) that flow between them" [15]. Since its release in 2015, it has been one of the most popular computing framework for machine learning and deep learning. Keras is a high-level neural network

API written in Python, and can run on top of different backends: Tensorflow, CNTK, Theano, providing a modular way of building neural network.

Our system is Ubuntu 14.04 and Python version is 2.7. We install Tensorflow with CUDA 8 and cuDNN 7, supporting GPU- GeForce GTX 1080 Ti, which has a compute capability 6.1 and 11GB memory. There are also several Python dependencies for the installation of 3DUnetCNN and we list them in Table 1.

Table 1. 3DUnetCNN dependencies list

Dependencies	**Descriptions**
nibabel	Read/write access to some common neuroimaging file formats
keras	high-level neural network API
pytables	manage hierarchical datasets and cope with extremely large amounts of data
nilearn	fast and easy statistical learning on NeuroImaging data
SimpleITK	A simplified, open-source interface to Insight Segmentation and Registration Toolkit (ITK)
matplotlib	2D plotting library

Chapter III Compression techniques

Image compression is the process of reducing the cost for storage or transmission of digital images. This is obtained by exploiting the coding redundancy, interpixel redundancy, and psychovisual redundancy [29]. Coding redundancy comes from the fact that some pixels appear more than others but each pixel value is encoded with a fixed-length code word. The solution is to use a variable-length code to assign more frequent symbols shorter bits [30]. The common methods include Huffman coding and Lempel-Ziv-Welch (LZW) coding. Interpixel redundancy results from the fact that neighboring pixels are similar or correlated, either in space or time. Methods for exploiting spatial redundancy includes Run-Length Coding, Quadtrees, Region Encoding, Predictive Coding, Fractal Image Compression, and Transform Coding [30]. Method for exploiting temporal redundancy includes motion compensation, which is used in MPEG. Psychovisual redundancy comes from the fact that some color differences are imperceptible to humans and is exploited with multimedia standards such as NTSC [30].

Compression ratio is defined by the ratio between uncompressed data size and compressed data size. There are always conflicts between high compression rate and the downgraded quality of the images. ROI coding techniques, which code the ROI and background in different ways, is one of the methods that can harmonize the conflicts.

3.1 ROI coding techniques

Several ROI coding techniques can be used following wavelet transform. Scaling method [17] is to scale the ROI coefficients to place the required bits of ROI in higher bit planes than the Non-ROI area [19]. Depending on the scaling value, some bits of the ROI coefficients may be encoded together with background coefficients, and the shape information about ROI should also be provided in decoding. In MAXSHIFT method [18], the scaling value is chosen such that the minimum coefficients of the ROI are larger than maximum coefficients of the background [19], and no shape information about ROI is needed. In vector quantization (VQ) [24], image is divided into non-overlapping blocks. Each vector of the image is approximated by the nearest element in codebook and the index of the element is transmitted [19]. For ROI-VQ, every region has separate codebook. A large codebook containing small codewords is created for the ROI region while the block size is smaller for non-ROI region [19]. Based on EZW (Embedded Zerotrees of Wavelet transforms) [38], [24] presented the multi rate/resolution control in the progressive transmission for ROI coding. However, all these RIO coding techniques only apply for 2D still images [19].

3.2 Set Partitioning in Hierarchical Trees (SPIHT)

Set Partitioning in Hierarchical Trees (SPIHT) is based on EZW and proposed by [1]. Both of them share the progressive coding feature that the most important wavelet coefficients will be transmitted first. Superior to EZW, SPIHT uses the spatial orientation tree (SOT), three sets O(i,j), D(i,j) and L(i,j) to organize the hierarchical relations and three ordered list: List of insignificant pixels (LIP), List of significant pixels (LSP), List of insignificant set (LIS), to trace the significant test results of the wavelet coefficients. In addition, SPIHT algorithm develops a set partitioning rule (see below) in sorting pass that both the encoder and decoder follow, thus implicitly transmits the ordering information in encoding to decoder and improves the efficiency.

Figure 5 shows the parents-offspring relations in spatial orientation tree structure. Except for the coefficient root node and the leave nodes, all the other nodes (i,j) have four children which form offspring set O(i,j). D(i,j) is the set that includes all the descendants of node (i,j) and L(i,j) = D(i,j)- O(i,j).

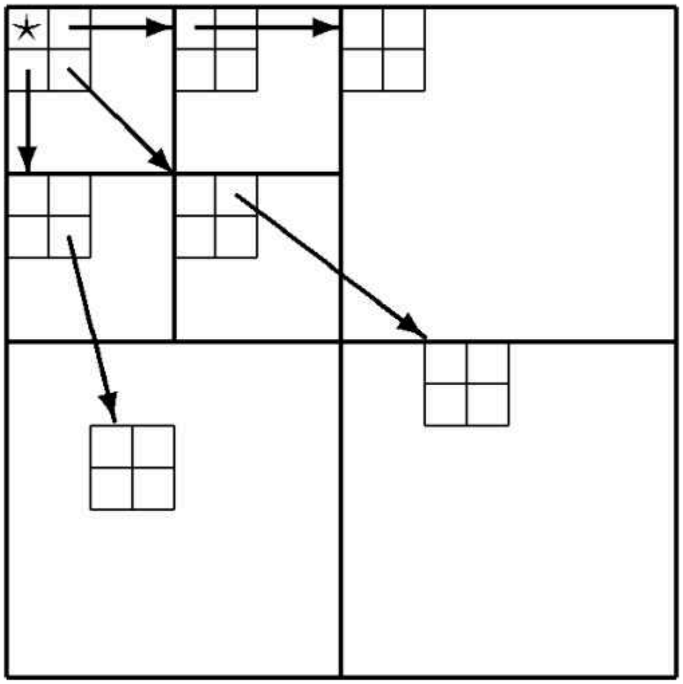

Figure 5. The spatial orientation tree structure of SPIHT

Source: Said, A., & Pearlman, W. A. (1996).

Following the set partitioning rule:

1. Initial partition is sets {(i,j)} and D(i,j), where (i,j) is the root node,
2. If D(i,j) is significant then it is partitioned into L(i,j) and four single-element sets each has one child of node (i,j),

3. If L(i,j) is significant then it is partitioned into four sets D(k,l), with (k,l) being the child of node (i,j),

the algorithm starts with an empty LSP set. Then the comparison of the coefficients at nodes in spatial orientation trees and the threshold value will determine how the sets partition and how the three ordered lists are updated. The bit streams file will also be generated in this process (more details at [1]). Figure 6 shows the spatial orientation tree structure of SPIHT and 3D-SPIHT. All the coefficients nodes in 3D-SPIHT either have no child or have 8 children (more details at [2]).

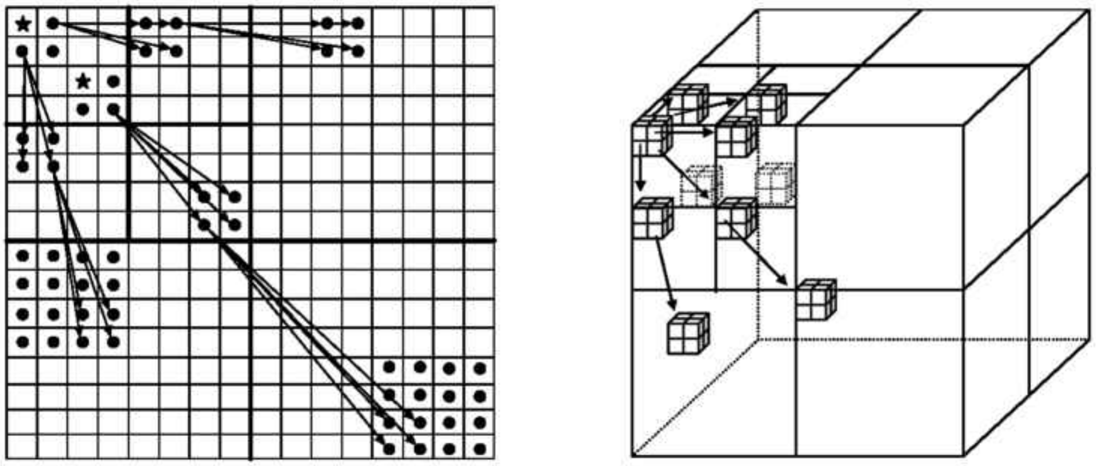

Figure 6. The spatial orientation tree structure of SPIHT and 3D-SPIHT

Source: www.spiedigitallibrary.org

3.3 QccPack

QccPack software package is an "open-source collection of library routines and utility programs for quantization, compression, and coding data" written by [4]. It implements procedures commonly used in coding and compression such as entropy coding, vector quantization, wavelet transforms, wavelet-based subband coding as well as SPIHT and SPECK coding.

The utility programs for SPIHT and 3D-SPIHT algorithms which respectively involve 2D and 3D DWT followed by a progressive "bit-plane" coding of the wavelet coefficients are used in this thesis project. The package also supports the mask parameter which is a transparent area specified by the user for shape-adaptive coding. The mask must be determined before the encoding begins and is also needed in decoding. The area except for region covered by the mask will be dropped. This property is not desired in many medical diagnosis contexts if we use the mask feature to encode the ROI. Because the background can provide the relative location information of the ROI area for accurate diagnosis. In this thesis project, we separately encode the images by its ROI area and background area, and combine the images for reconstruction after transmitting and decoding.

Chapter IV Data and experiment

4.1 Dataset

In this thesis, we use MRI images of human spine and its label files for vertebral bodies. The image format is Neuroimaging Informatics Technology Initiative (nifti). We have 8 image sample sets and each sample set includes wat.nii, opp.nii, fat.nii, inn,nii and truth.nii files. The dataset is provided by [22] in Institute for Surgical Technology and Biomechanics at University of Bern. The 3D visualization of Nifti images can be shown by 3D Slicer [23], an open source software for medical image processing and visualization. Figure 7(a) shows the 3D visualization of an opp.nii file and Figure 7(b) shows its vertebral bodies ground truth which is manually labeled by medical experts.

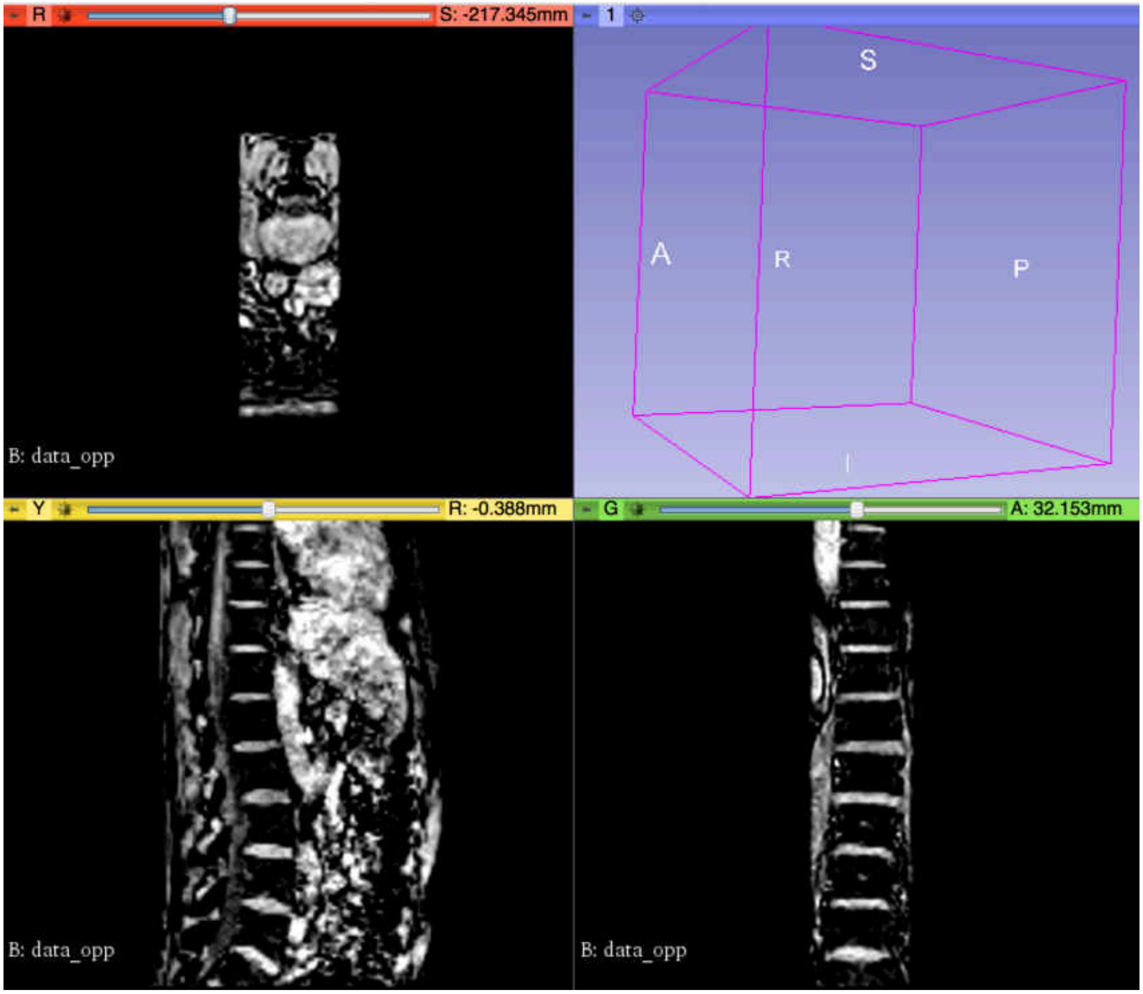

Figure 7(a). 3D visualization of opp.nii volumetric image by 3D Slicer

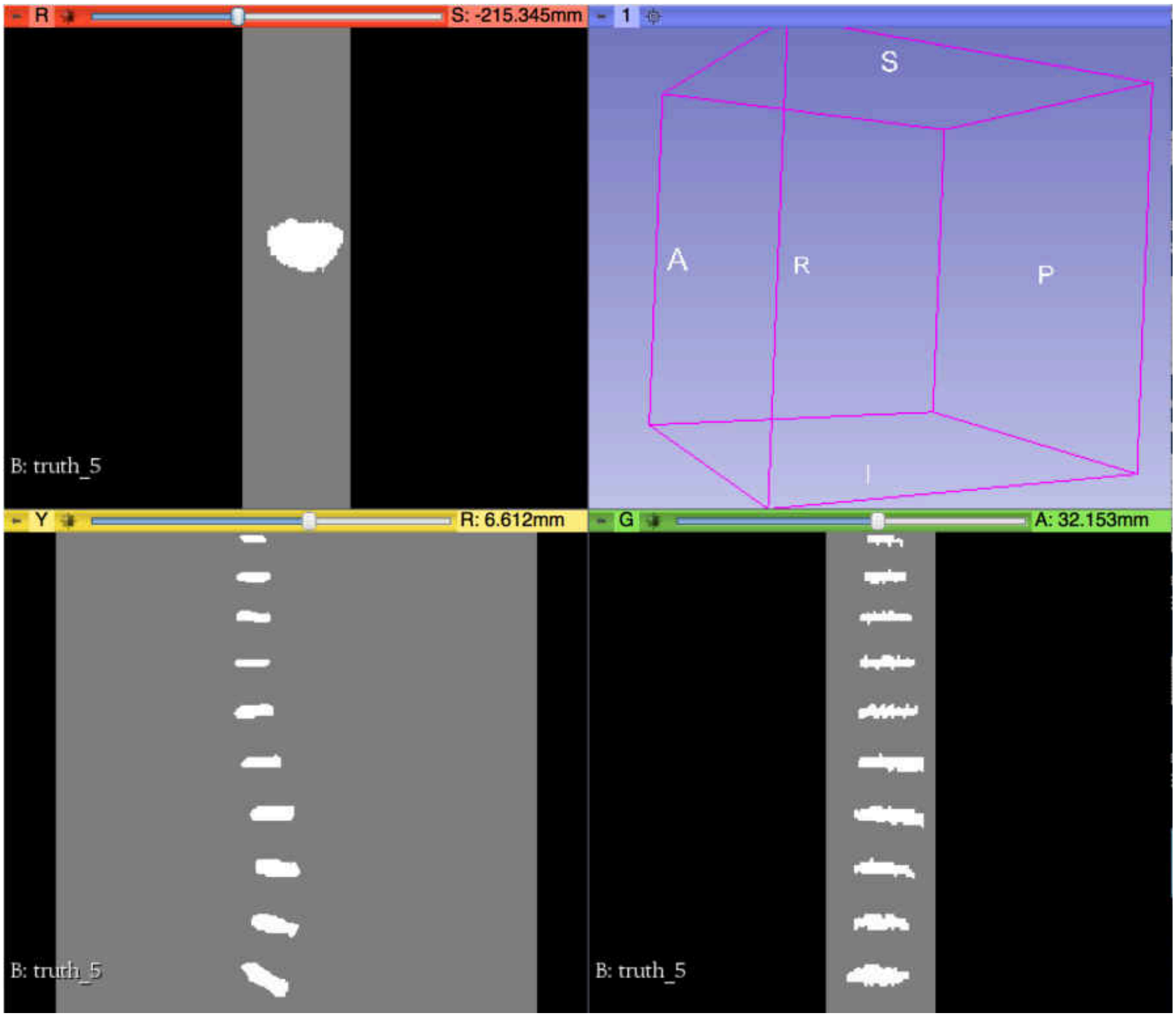

Figure 7(b). Vertebral bodies ground truth label of original volumetric image by 3D Slicer

4.2 Process of medical image transmitting

In our dataset, Nifti volumetric image have a dimension of 36*256*256. We can see it as a stack of 36 slices of 256*256 greyscale images and the size of the volumetric image is 4.7MB. Transmitting the original volumetric image is costly in the sense of time and bandwidth, especially for remote usage. Figure 8 shows our proposed flowchart how a data receiver (user) interacts with the data provider to get the encoded data. The flowchart is composed of the data provider on the upper side and the user at the bottom.

When there is a need in volumetric image data, the user can interact with the data provider in following steps:

1. The user makes a request with specific region of interest and its corresponding bit per voxel (usually high) as well as the desired bpv (usually low) for the background. After receiving the request and parameters, volumetric image is sent to 3D-Unet module and its prediction of ROI and background segmentation is generated. The output is also a volumetric image but each slice is a binary image with ROI (label) being 1 and background being 0, which will be used to generate volumetric mask.
2. The original volumetric image and its segmentation predicting image are transformed to icb files, which are special volumetric image format defined by QccPack and can be used by 3D-SPIHT encoding utilities. Volumetric mask is generated from the transformation.
3. The icb file of original volumetric image and the ROI mask are used by the 3D-SPIHT module, and the output is the ROI bit file and mask bit file. With the background mask, background bit file is also generated. Then the ROI bit file, background bit file, and mask bit file are transmitted to the user.
4. After receiving the encoded bit files, the user can decode them to get icb files.
5. Through format transformation and reconstruction, a volumetric image with separate desired resolutions is acquired.

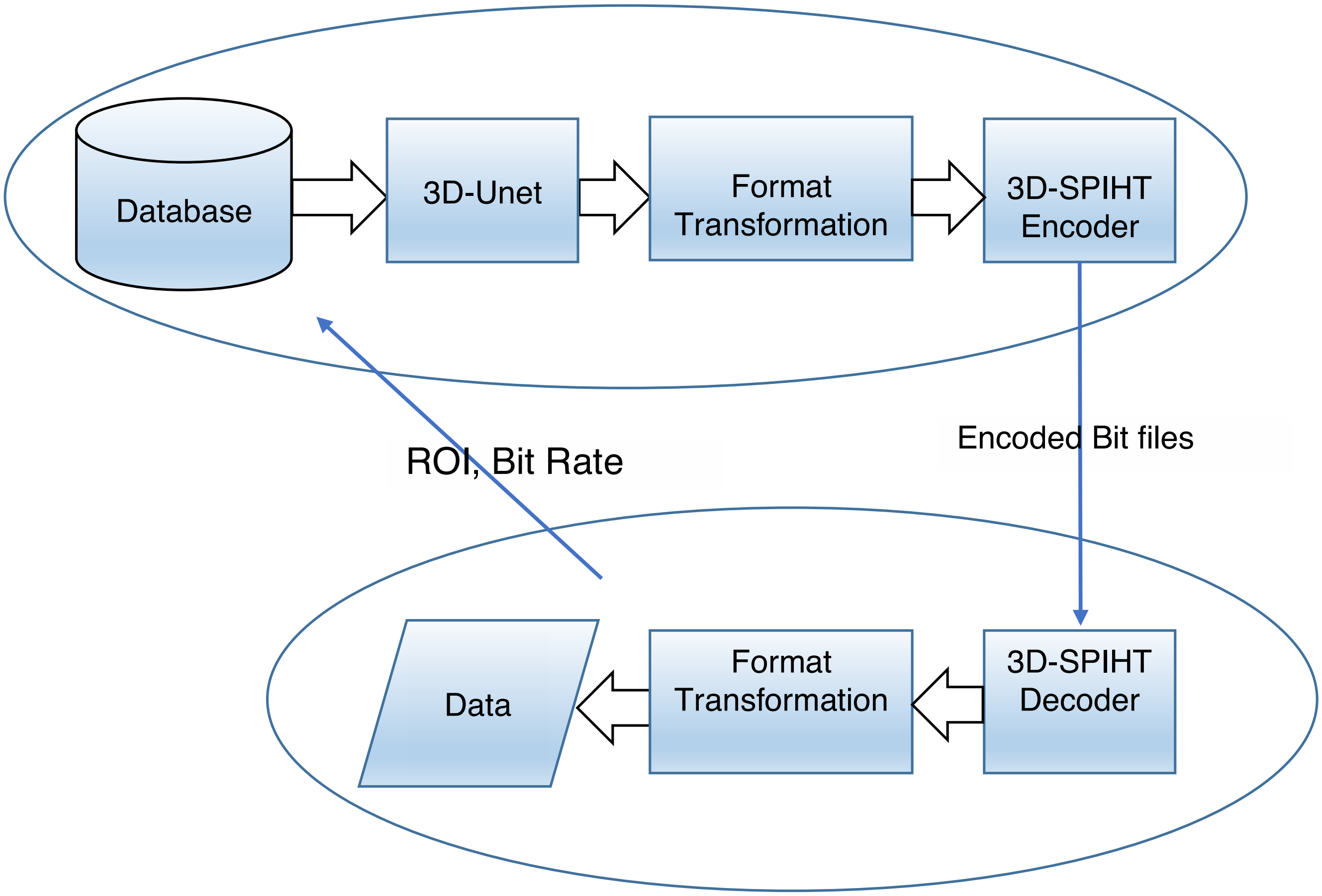

Figure 8. Interaction between the medical images user and data provider

4.3 3D-Unet module

3D-Unet Module has two usage scenarios: the first is when the data provider receives a region of interest request for the first time, which means there is no trained network available, and the user should provide sample volumetric images and corresponding label images to train the neural network. The second (more general) is that there is a trained network available and it only reads the input volumetric image together with the saved parameters of the trained network to get the region of interest prediction. As more users request data from the data provider, more trained networks corresponding to the common ROIs will be available.

In this thesis, however, we implement the former case because it demonstrates an end-to-end scenario to train the 3D-Unet and make prediction. Table 2 shows the parameter settings we use to train the 3D-Unet.

Table 2. 3DUetCNN parameter settings

config["labels"] = 1	the label numbers on the input image
config["all_modalities"] = ["opp", "wat"]	
config["training_modalities"] = config["all_modalities"]	use opp.nii, wat.nii file for training the model
config["n_epochs"] = 500	cutoff the training after 500 epochs
config["patience"] = 10	learning rate will be reduced after 10 epochs if the validation loss is not improving
config["early_stop"] = 50	training will be stopped after 50 epochs without the validation loss improving
config["initial_learning_rate"] = 0.00001	
config["learning_rate_drop"] = 0.5	factor by which the learning rate will be reduced
config["flip"] = False	do not use flip to augment the data
config["permute"] = False	do not augment the data by permuting in various directions

After setting all the parameters, we can run train.py to train the model and predict.py to write the predicting label file. The model receives Nifti images of 6 patients for training and the remaining 2 sets of images for prediction. Figure 9 is the 3D visualization of the model predicting label file. The result shows that comparing to the ground truth label file, the model predicting area is smaller and also has some noisy points.

The prediction of vertebral bodies is not good enough, however, for spine segmentation task, what the user cares most is the whole area near the spine, instead of only the vertebral bodies. Therefore, we dilate the predicting label areas to get a connected region that includes the spine and nearby area. The size of the region is determined by the kernel size of dilating operator.

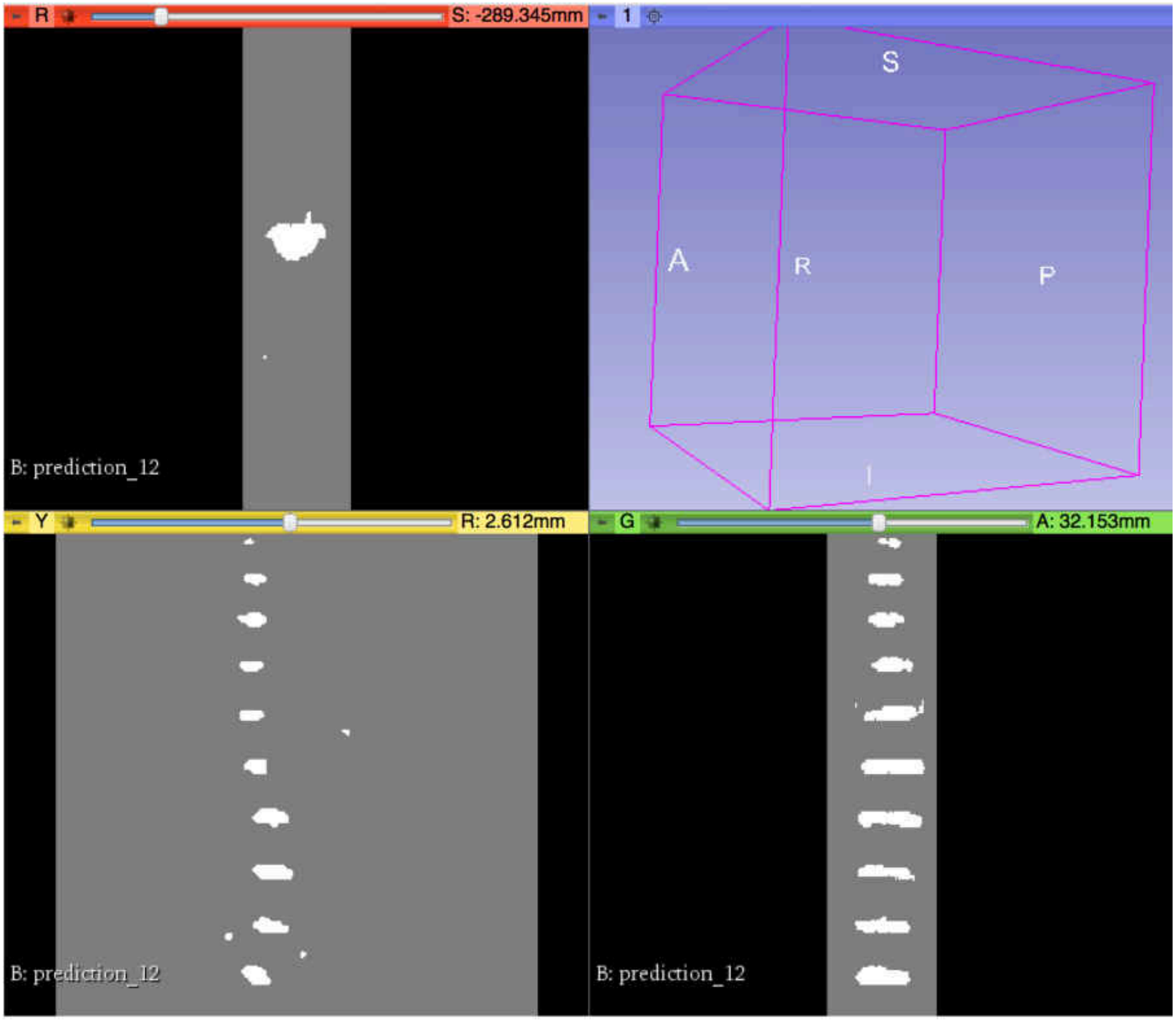

Figure 9. 3D visualization of the model predicting label file

4.4 Format transformation module

There are two format transformation module in the whole process: the data provider side that transfers volumetric images to icb files for encoding and the user side that recovers the volumetric images from icb files.

4.4.1 From volumetric images to icb files

Both the original volumetric medical image and the predicted volumetric label file need to be transformed, and the transformation of label file is different from the former. Take our data for example, opp.nii is the original volumetric image with dimension 36*256*256, we use nib.load() in nibabel package to read the file and get_data() to get the 36 slices of pgm files. Each slice is a greyscale pgm image with 256*256 dimension. Figure 10 shows the 8th slice from opp.nii file. Then we combine the series of pgm images to icb file using seqtoicb utility program provided by QccPack. Since 3D-SPIHT algorithm we used only allows an icb file with frame number being the power of 2, we drop the first 2 and last 2 frames which do not contain any spine pixels, generating an icb file with a dimension of 32*256*256.

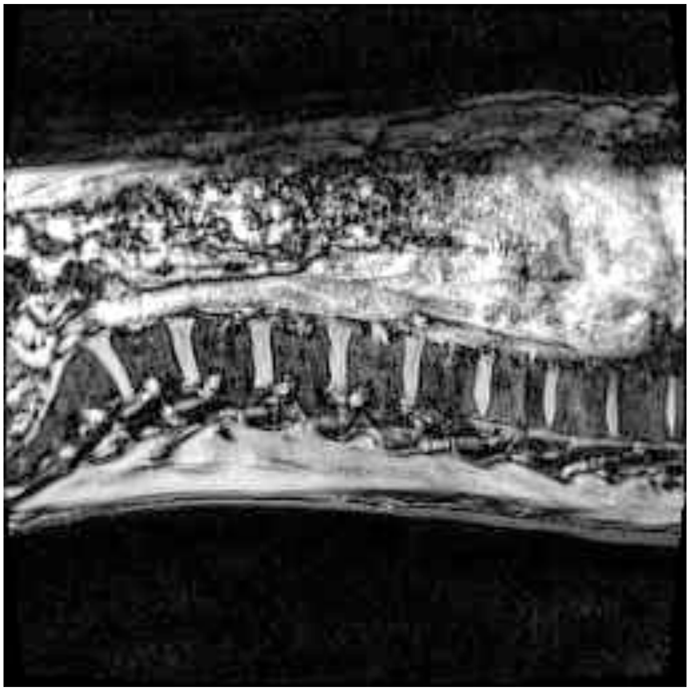

Figure 10. The 8th slice from opp.nii volumetric image

For the predicted volumetric label image, whose 3D visualization is shown in Figure 9, we apply the dilate operation after getting its slices. Figure 11(a) shows a slice from original predicting label image given by 3D-Unet, and Figure 11(b) is our dilated result using 7*7 kernel and 1 iteration.

We continue to increase the kernel size until 25*25, for the reason that all the vertebral bodies' label will joint together, thus forming a cavity that covers the entire spine. Based on the dilated label files, we build our mask for region of interest. Figure 12 shows the dilated label slice 8,16, 24 and 31 by kernel 25*25 and 1 iteration.

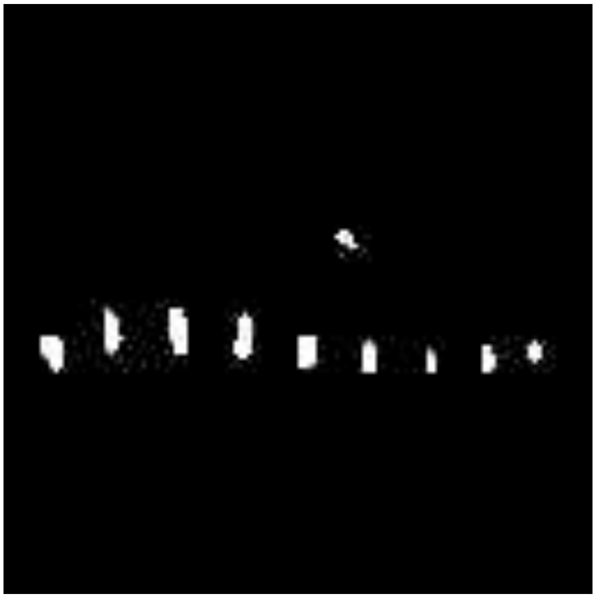

Figure 11(a). Label slice from 3D-Unet prediction

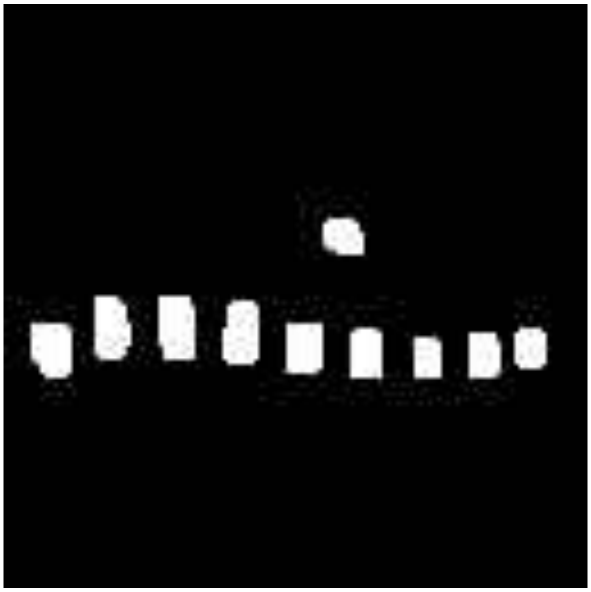

Figure 11(b). Dilated label image by kernel 7*7 and 1 iteration

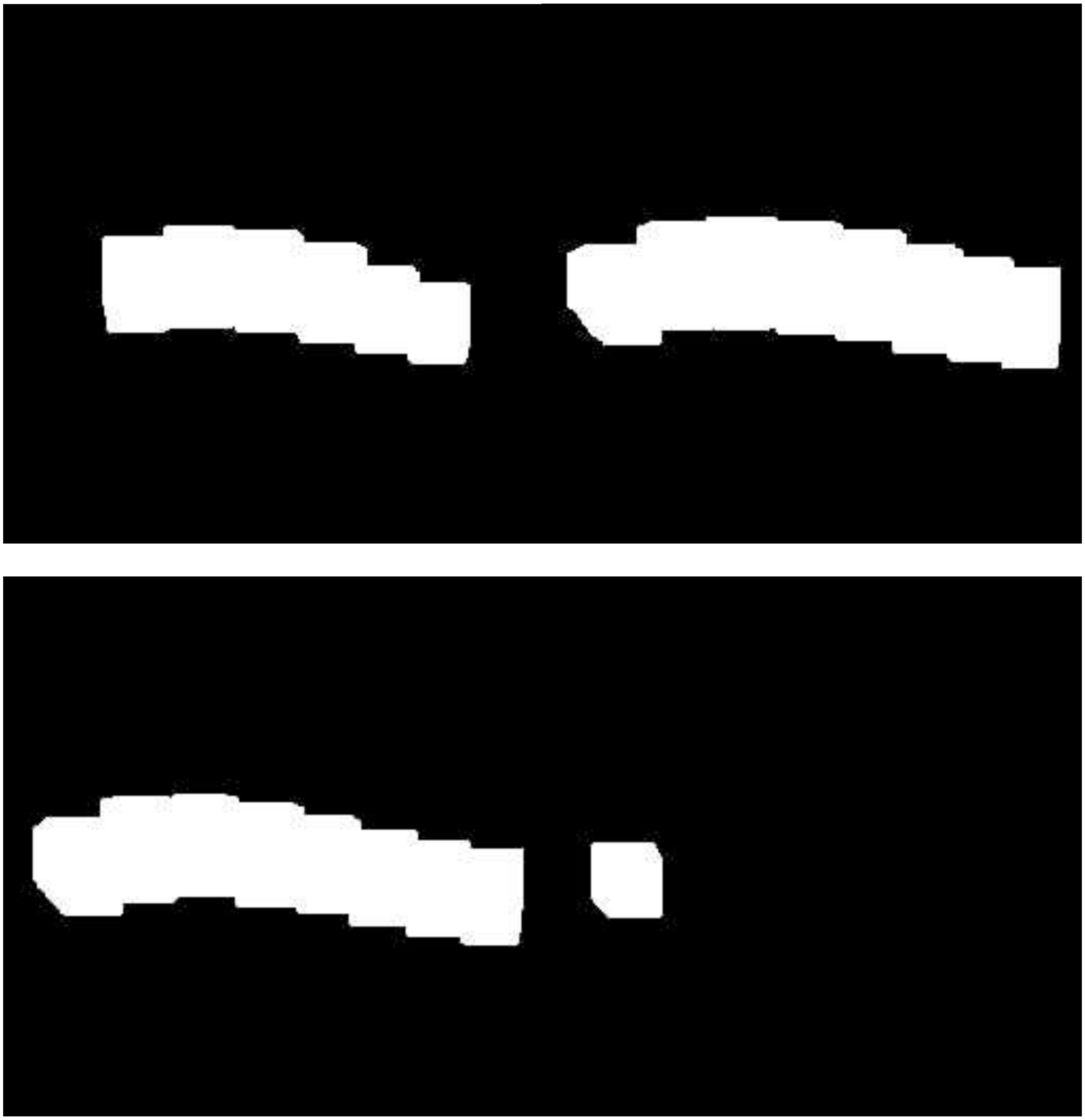

Figure 12. Dilated label image by kernel 25*25 and 1 iteration (a) slice 8 (left top) (b) slice 16 (right top) (c) slice 24 (left bottom) (d) slice 31 (right bottom)

After performing dilate operation for all the prediction slices, we use seqtoicb again to get the ROI mask icb file. Also, we create sequences of inverse image of the prediction slices to get the background mask icb file. Then the icb file of volumetric image can be encoded with ROI mask and corresponding bit rate to get ROI bit file. With background mask and corresponding bit rate, background bit file is also generated, and the ROI mask can be encoded with 1 bpv losslessly. Table 3 shows the different size of bit files with respect to different bit per voxel.

Table 3. Bit file size of 3D-SPIHT coding with ROI
and background mask for different bpv

	ROI mask	Background mask
0.01 bpv		2.3 KB
0.1 bpv		23.1 KB
1 bpv	31.3 KB	230.9 KB
2 bpv	62.5 KB	461.7 KB
3 bpv	93.8 KB	692.6 KB
4 bpv	125.1 KB	923.5 KB
5 bpv	156.4 KB	1.2 MB
8 bpv	250.2 KB	1.8 MB

As we can see from Table 3, the size of 3D-SPIHT coding bit file is related to the mask and bit per voxel. Background contains the information that can suffer from high compression rate. With the user's choices of high bpv for ROI area and low bpv for background area, decoding and combining them will recover a desired image that has high resolution in region of interest and low resolution in background.

Since ROI area is relatively small comparing to the background, we choose 8 bpv for ROI and match with different bpv for background.

4.4.2 From icb files to volumetric images

After decoding the bit files and getting the icb files for both ROI and the background, we can convert them to the pgm sequences for ROI and background separately, and then combine them to get pgm sequences of complete image, where the ROI area has high resolution and background has low resolution.

Figure 12 shows the reconstruction images of slice 21, with different bpv for background. Figure 12(a) is from 8 bpv for ROI and 0.01 bpv for background, Figure 12(b) is from 8 bpv for ROI and 0.1 bpv for background, Figure 12(c) is from 8 bpv for ROI and 1 bpv for background, Figure 12(d) is from 8 bpv for ROI and 8 bpv for background and Figure 12(e) shows the original image slice. A comparison of different information loss measures is reported in Table 4 and plotted in Figure 13.

Except for the slice encoded with 0.01 bpv for background, which loses most details, all the other reconstruction images still keep background information enough to provide the relative location of the ROI. The reconstruction information loss measured by MSE (the smaller the better), SNR (the larger the better), PSNR or MAE decreases with the increase of bpv for background. After reconstructing all the slices, we can use ImageJ [25] plugin nifti_io.jar [26] to convert the pgm slices to nii volumetric image.

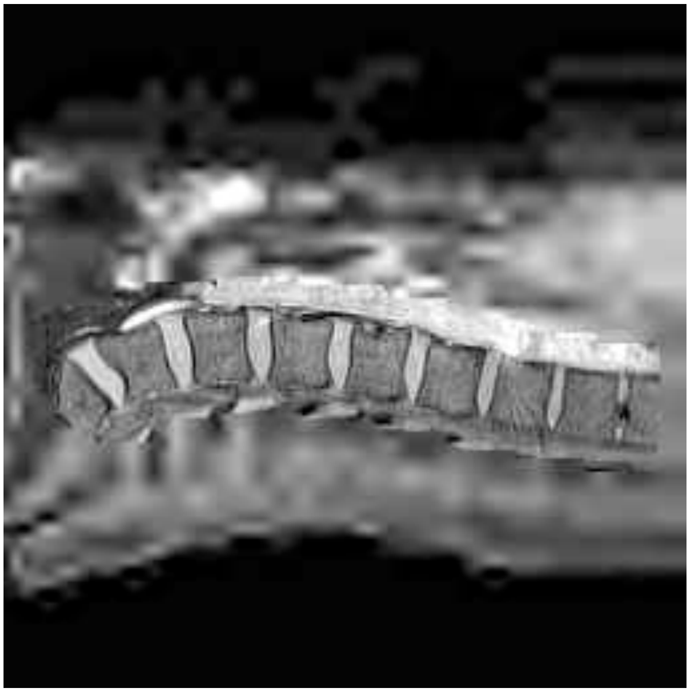

Figure 12(a). Reconstruction image of slice 21 with 8 bpv for ROI and 0.01 bpv for background

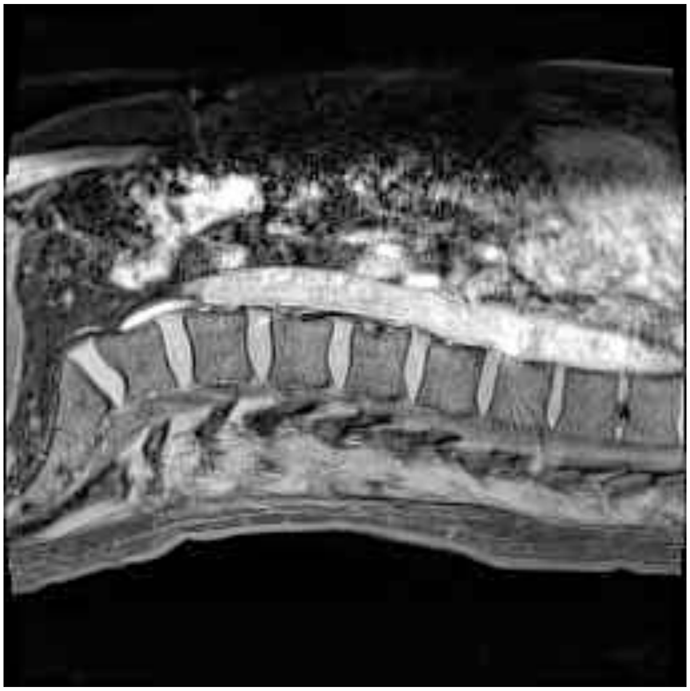

Figure 12(b). Reconstruction image of slice 21 with 8 bpv for ROI and 0.1 bpv for background

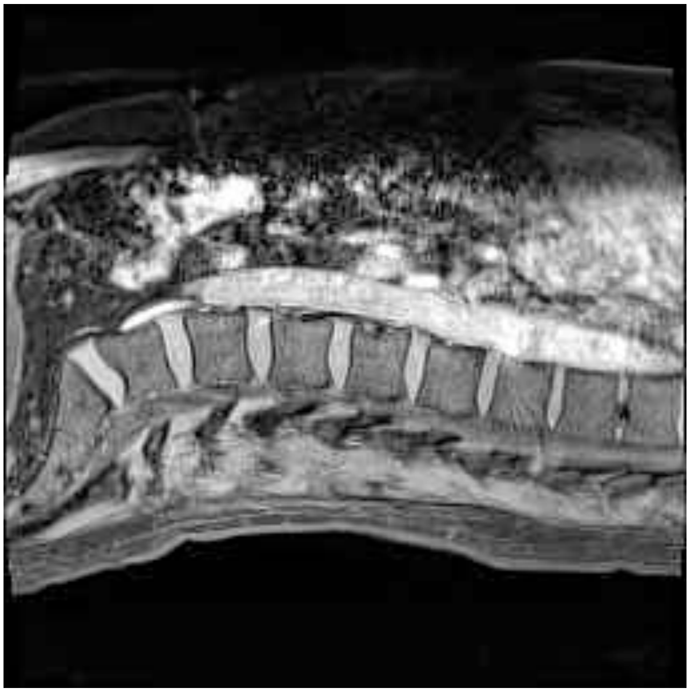

Figure 12(c). Reconstruction image of slice 21 with 8 bpv for ROI and 1 bpv for background

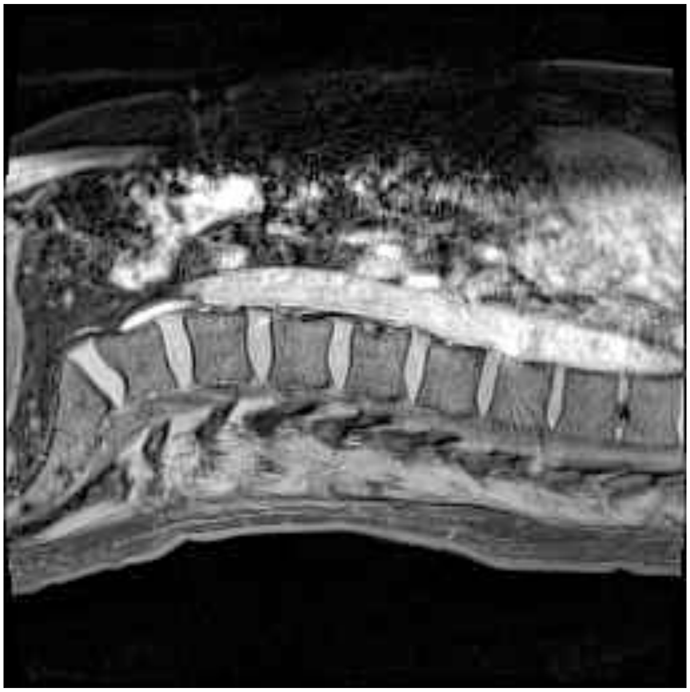

Figure 12(d). Reconstruction image of slice 21 with 8 bpv for ROI and 8 bpv for background

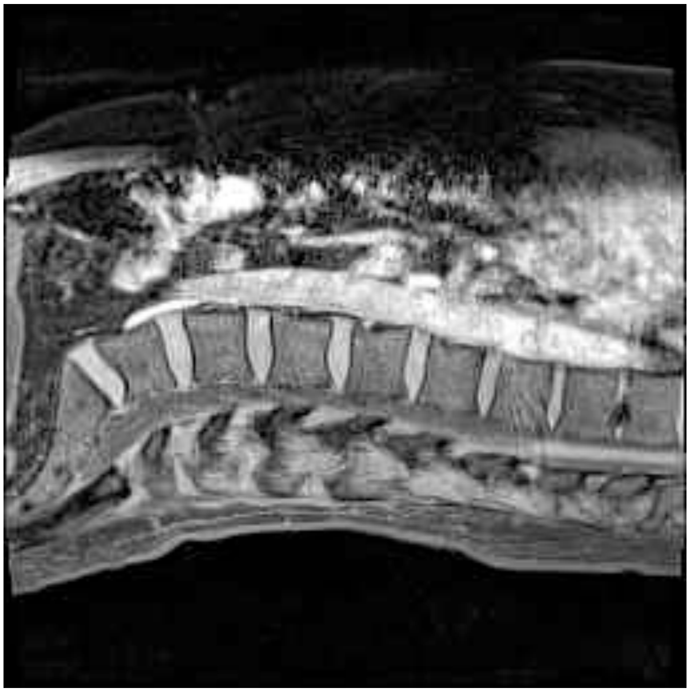

Figure 12(e). Original image of slice 21

Table 4. The comparison of different measures of information loss for different bpv

bpv	MSE	SNR(dB)	PSNR(dB)	MAE
1	14.11	24.81	36.63	23
2	3.44	30.94	42.76	10
3	1.03	36.19	48.01	7
4	0.31	41.38	53.21	3
5	0.05	49.15	60.97	1
8	0	-1	-1	0

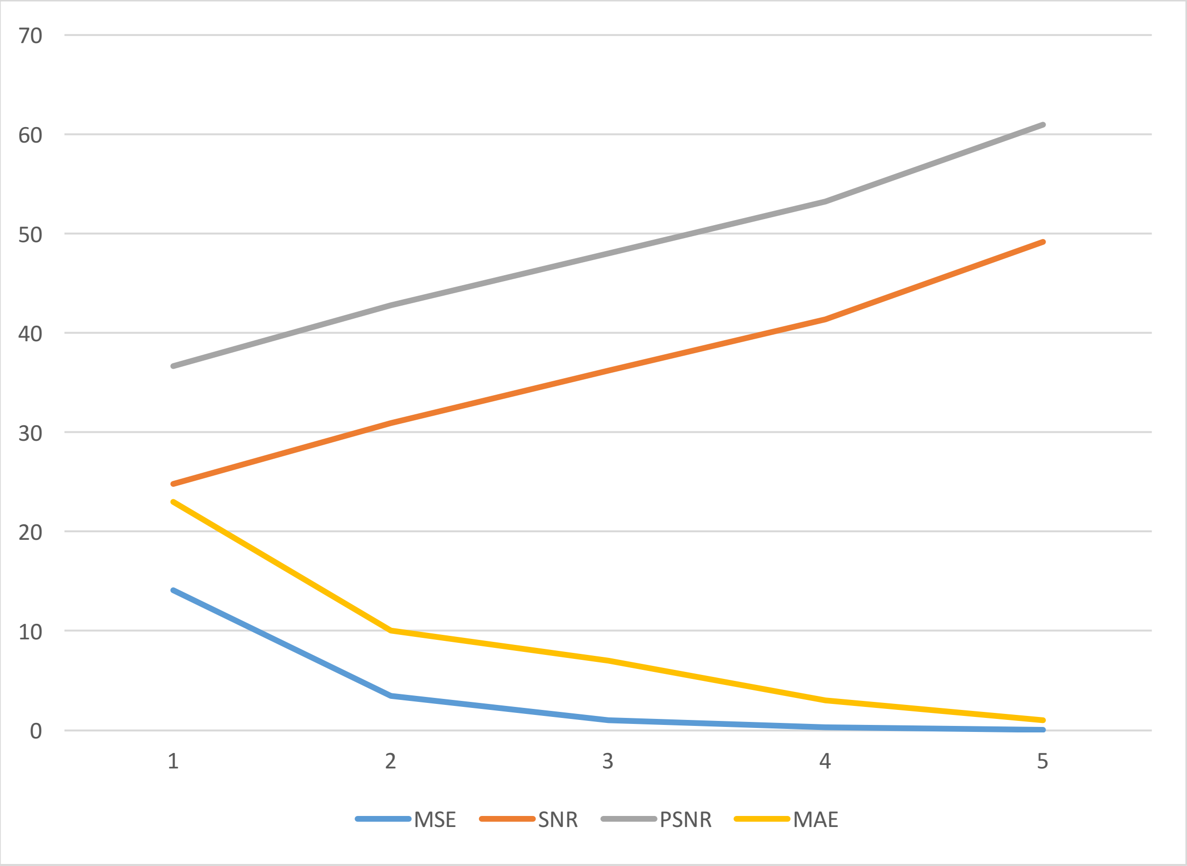

Figure 13. The plot of different information loss measures for different bpv

4.5 The comparison of original image and bit files sizes

The original images have 32 slices and each takes 65.6KB. After encoding, the data provider need to send three bit files to the user: ROI bit file, background bit file and mask bit file. Table 5 reports the total size of bit files that is required in transmitting and the compression ratio for different bpv choices in background.

Table 5. Total size of bit files required in transmitting and the compression ratio for different bpv choices in background

bpv choices	total size of bit files	Compression ratio
Background 1bpv ROI 8bpv	743.2KB	2.82
Background 2bpv ROI 8bpv	974KB	2.16
Background 3bpv ROI 8bpv	1204.9KB	1.74
Background 4bpv ROI 8bpv	1435.8KB	1.46
Background 5bpv ROI 8bpv	1712.3KB	1.23

Chapter V Limitations and future work

There are several limitations in this thesis: First, the segmentation of vertebral bodies predicted by the 3D-Unet model is not as good as the brain tumor example shown on the package developers' website. The GPU used in this project is GeForce GTX 1080 Ti, which has compute capability 6.1 with 11G memory, but the program suffers from out of memory error and we have to reduce the modalities number and use less data augmentation option. This may be one potential reason of the bad prediction issue. Using GPUs with larger memory or optimizing the source code may resolve the problem and get more accurate prediction.

Second, for its importance in medical diagnosis, medical volumetric images usually do not afford to lossy compression, in which the information of critical diagnosis zone may lose due to image artifacts. The original 3D-SPIHT algorithm supports lossless compression, but the QccPack only provides lossy transform. In our experiment, we decode the bit-file and restore the original images with different information losses depending on the different bpv chosen. These losses may not acceptable in some other cases, therefore, the modification to the package to enable lossless compression is another direction of work in the future.

Except for extracting the ROI area and applying 3D-SPIHT encoding to ROI and background separately. ROI coding can also be achieved through multiplying the DWT coefficients in the desired ROI spatial area by some constant greater than 1 and less than or equal to $2^{\text{QCCSPIHT\_MAXBITPLANES}} – 1$, after the DWT but before the first coding pass. This is a potential way to simplify the image transmitting process, where only one bit-file is needed to be generated and transmitted.

References

[1] Said, A., & Pearlman, W. A. (1996). A new, fast, and efficient image codec based on set partitioning in hierarchical trees. *IEEE Transactions on circuits and systems for video technology*, *6*(3), 243-250.

[2] Kim, B. J., & Pearlman, W. A. (1997, March). An embedded wavelet video coder using three-dimensional set partitioning in hierarchical trees (SPIHT). In *Data Compression Conference, 1997. DCC'97. Proceedings* (pp. 251-260). IEEE.

[3] Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., & Ronneberger, O. (2016, October). 3D U-Net: learning dense volumetric segmentation from sparse annotation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 424-432). Springer, Cham.

[4] Fowler, J. E. (2000, December). QccPack: An open-source software library for quantization, compression, and coding. In *Applications of Digital Image Processing XXIII* (Vol. 4115, pp. 294-302). International Society for Optics and Photonics.

[5] Region of Interest (ROI) Retrieved March 25, 2018, from https://www.techopedia.com/definition/339/region-of-interest-roi

[6] Maheswari, D., & Radha, V. (2011). Comparison of layer and block based classification in compound image compression. *International Journal of Computer Science and Information Technologies*, *2*(2), 888-890.

[7] Haneda, E., & Bouman, C. A. (2011). Text segmentation for MRC document compression. *IEEE transactions on Image Processing*, *20*(6), 1611-1626.

[8] Zaitoun, N. M., & Aqel, M. J. (2015). Survey on image segmentation techniques. *Procedia Computer Science*, *65*, 797-806.

[9] Sharma, N., Mishra, M., & Shrivastava, M. (2012). Colour image segmentation techniques and issues: an approach. *International Journal of Scientific & Technology Research*, *1*(4), 9-12.

[10] Bala, A., & Sharma, A. K. Split and Merge: A Region Based Image Segmentation.

[11] Shi, J., & Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, *22*(8), 888-905.

[12] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

[13] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In CVPR09, 2009.

[14] Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (pp. 234-241). Springer, Cham.

[15] Spyridon Bakas, Hamed Akbari, Aristeidis Sotiras, Michel Bilello, Martin Rozycki, Justin Kirby, John Freymann, Keyvan Farahani, and Christos Davatzikos. (2017) Segmentation Labels and Radiomic Features for the Pre-operative Scans of the TCGA-GBM collection. The Cancer Imaging Archive. https://doi.org/10.7937/K9/TCIA.2017.KLXWJJ1Q

[16] Spyridon Bakas, Hamed Akbari, Aristeidis Sotiras, Michel Bilello, Martin Rozycki, Justin Kirby, John Freymann, Keyvan Farahani, and Christos Davatzikos. (2017) Segmentation Labels and Radiomic Features for the Pre-operative Scans of the TCGA-LGG collection. The Cancer Imaging Archive. https://doi.org/10.7937/K9/TCIA.2017.GJQ7R0EF

[17] ISO/IEC JTC 1/SC 29/WG 1 (ITU-T SG8), JPEG 2000 Part II Final Committee Draft, Dec. 2000.

[18] ISO/IEC JTC 1/SC 29/WG 1 (ITU-T SG8), JPEG 2000 Part I Final Committee Draft, Version 1.0,Mar. 2000.

[19] Jangbari, P., & Patel, D. (2016). Review on Region of Interest Coding Techniques for Medical Image Compression. International Journal of Computer Applications (0975–8887) Volume.

[20] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

[21] Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3431-3440).

[22] Zheng, G., Chu, C., Belavý, D. L., Ibragimov, B., Korez, R., Vrtovec, T., ... & Glocker, B. (2017). Evaluation and comparison of 3D intervertebral disc localization and segmentation methods for 3D T2 MR data: A grand challenge. Medical image analysis, 35, 327-344.

[23] Fedorov, A., Beichel, R., Kalpathy-Cramer, J., Finet, J., Fillion-Robin, J. C., Pujol, S., ... & Buatti, J. (2012). 3D Slicer as an image computing platform for the Quantitative Imaging Network. Magnetic resonance imaging, 30(9), 1323-1341.

[24] Dilmaghani, R. S., Ahmadian, A., Ghavami, M., Oghabian, M., & Aghvani, H. (2003, September). Multi rate/resolution control in progressive medical image transmission for the region of interest (ROI) using EZW. In Engineering in Medicine and Biology Society, 2003. Proceedings of the 25th Annual International Conference of the IEEE (Vol. 1, pp. 818-820). IEEE.

[25] Schneider, C. A., Rasband, W. S., & Eliceiri, K. W. (2012). NIH Image to ImageJ: 25 years of image analysis. Nature methods, 9(7), 671.

[26] Williams G. NIfTi Input/Output. https://imagej.nih.gov/ij/plugins/nifti.html. Accessed 25 March 2018

[27] Razzak, M. I., Naz, S., & Zaib, A. (2018). Deep Learning for Medical Image Processing: Overview, Challenges and the Future. In Classification in BioApps (pp. 323-350). Springer, Cham.

[28] McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics, 5(4), 115-133.

[29] Qi, X. Proposal from Utah State University. Retrieved March 25, 2018, from http://digital.cs.usu.edu/~xqi/Proposal/Chapter2.pdf

[30] Carnegie Mellon University. Image Compression. Retrieved March 25, 2018, from www.cs.cmu.edu/afs/andrew/scs/cs/15-463/99/pub/www/notes/compression.ps.gz

[31] Jain, A. K., Mao, J., & Mohiuddin, K. M. (1996). Artificial neural networks: A tutorial. Computer, 29(3), 31-44.

[32] Rosenblatt, F. (1957). The perceptron, a perceiving and recognizing automaton Project Para. Cornell Aeronautical Laboratory.

[33] Dharanikota, D. (July 23, 2017). History of Neural Networks. Retrieved April 25, 2018, from https://www.linkedin.com/pulse/history-neural-networks-datta-dharanikota

[34] Linnainmaa, S. (1970). The representation of the cumulative rounding error of an algorithm as a Taylor expansion of the local rounding errors. Master's Thesis (in Finnish), Univ. Helsinki, 6-7.

[35] Werbos, P. (1974). Beyond regression: new fools for prediction and analysis in the behavioral sciences. PhD thesis, Harvard University.

[36] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985). Learning internal representations by error propagation (No. ICS-8506). California Univ San Diego La Jolla Inst for Cognitive Science.

[37] David, G. E. 3D U-Net Convolution Neural Network with Keras. Retrieved April 25, 2018, from https://github.com/ellisdg/3DUnetCNN

[38] Shapiro, J. M. (1993). Embedded image coding using zerotrees of wavelet coefficients. IEEE Transactions on signal processing, 41(12), 3445-3462.