

MASTER IN COMPUTING

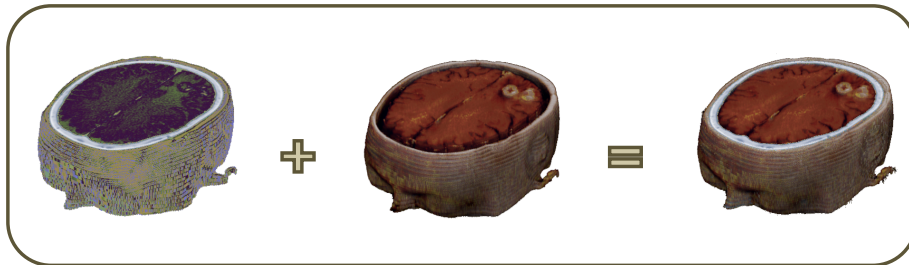
DEPARTAMENT DE LLENGUATGES I SISTEMES INFORMÀTICS

UNIVERSITAT POLITÈCNICA DE CATALUNYA

An Information-Theory Framework for Multi-Modal Visualization

MASTER THESIS

ROGER BRAMON FEIXAS



Advisors:
Dra. IMMA BOADA OLIVERAS
Dr. MIQUEL FEIXAS FEIXAS

September 2009

Contents

1	Introduction	5
1.1	Motivation	5
1.2	Objective	6
1.3	Document Outline	6
2	Background	9
2.1	Volume Visualization	10
2.2	Direct Volume Rendering Techniques	11
2.2.1	Volume Rendering integral	12
2.2.2	Classification of Direct Volume Rendering Techniques	13
2.3	Multi-Volume Visualization	15
2.4	Information Theory	19
2.4.1	Entropy	19
2.4.2	Mutual Information	20
2.4.3	Entropy Rate	21
2.4.4	Erasure Entropy	22
3	A General Framework for Multi-Modal Visualization	23
3.1	Introduction	23
3.2	Registration Module	24
3.3	Segmentation Module	25
3.4	Fusion Module	26
3.5	Visualization Module	27
3.5.1	Ray-casting implementation	29
3.5.2	Color Coded Maps	29
3.6	User Interface Design	30
4	Fusion using the Registration Channel	35
4.1	Introduction	35
4.2	Conditional Entropy Method	35
4.2.1	Conditional Entropy of the Registration Channel . . .	36
4.2.2	Fusion Strategies	37
4.3	Mutual Information Method	45

4.3.1	Mutual Information of the Registration Channel . . .	45
4.3.2	Fusion Strategies	45
4.4	Implementation Details	53
5	Fusion using Higher-Order Entropy Measures	55
5.1	Introduction	55
5.2	Entropy Rate Method	55
5.2.1	Entropy Rate Measure	55
5.2.2	Fusion Strategies	56
5.3	Erasure Entropy Method	65
5.3.1	Erasure Entropy Measure	65
5.3.2	Fusion Strategies	67
5.4	Implementation Details	71
6	Testing of Fusion Methods	77
6.1	Testing Data Sets	77
6.2	Quality of the Fusion	79
6.2.1	Evaluation Criteria	79
6.2.2	Results	82
6.3	Performance Requirements	86
6.3.1	Memory Requirements	86
6.3.2	Computational Time Requirements	87
7	Conclusions	91

Chapter 1

Introduction

1.1 Motivation

Scientific visualization is focused on creating images that convey salient information about underlying data and processes using interactive graphics and imaging techniques. In last decades, it has emerged as a basic tool in different scientific areas, such as medical processes, earth and space sciences, biological processes or fluid dynamics.

Volume visualization is a field within scientific visualization, which is concerned with volume data. Over the years, many techniques have been developed to visualize volumetric data. The majority of them have been used to visualize data from a unique origin, i.e., considering a single model. However, in certain fields, such as medicine, science and engineering, the capability of visualizing different models in a single image will be interesting. For instance, in a medical application, combining the complementary information of a computed tomography and a magnetic resonance will enhance the detection of a pathology.

The idea of multimodal visualization is to provide as many information on the complementary aspects of the different input models as possible. This process is complex and requires the application of different steps: (i) a registration process to align the two or more images or volumes obtained at different times, from different devices or from different viewpoints; (ii) a fusion process which, based on different parameters such original values, gradient magnitude, etc., converts the information of input volumes to a single value and (iii) the visualization of the fused model.

Performing the fusion is one of the most challenging tasks of multimodal visualization. The main difficulty is on determining the information of each input model that has to be represented in the final visualization. In most of the cases, the strategies that have been proposed rely this selection on a user decision which makes difficult the application of the method in real applications, specially when time requirements are imposed and automatic

techniques are desired.

1.2 Objective

The main goal of this master thesis is the development of new fusion strategies that enhance multimodal visualization strategies. We will consider Information Theory (IT) as a main tool to be applied in our approach. This theory deals with the transmission, storage and processing of information [26] and it is used in fields such as physics, computer science, mathematics, statistics, economics, biology, linguistics, neurology, learning, computer graphics, etc. [6, 11]. In particular, IT provides powerful tools for medical image registration [20] and segmentation [3].

Since medical imaging is the major application area of volume rendering we will focus our interest on it.

To reach this objective we aim to:

- Study basic concepts of volume visualization and information theory concepts, and review multimodal visualization strategies focusing on the fusion process.
- Analyze how information theory tools can be applied in the context of multimodal visualization to enhance it.
- Evaluate the proposed method in a real medical scenario.

An important step towards the development of new computer aided diagnosis tools is the validation of proposed techniques in a real environment. Therefore, an essential part of our work is devoted to the evaluation of the techniques in medical practice. Such an evaluation required the implementation of a framework integrating all proposed approaches. To carry out the evaluation process we will collaborate with a team of medical experts.

1.3 Document Outline

This master thesis is organized into seven chapters. In this first chapter, we introduced the project and its context and we presented the initial objectives. In Chapter 2, we describe the background on volume visualization focusing our interest on direct volume rendering techniques. We also analyze the state of the art of multi-modal visualization techniques and introduce the most basic information theory concepts. In Chapter 3, we present the framework that have been developed in this project for multi-modal visualization, we introduce the different modules that have been integrated in the framework, and we propose a classification of fusion strategies. In Chapter 4, we propose our first fusion approach based on the conditional entropy and

mutual information, which we call registration channel measures. In Chapter 5, we present the second fusion approach based on higher-order entropy measures: the entropy rate and erasure entropy. In Chapter 6, we present the different experiments that have been carried out to evaluate the proposed fusion techniques in a medical environment. The memory and time requirements are also tested and analyzed. Finally, in Chapter 7, conclusions and future work are presented.

Chapter 2

Background

Volume visualization is a method of extracting meaningful information from volumetric data using interactive graphics and imaging, and it is concerned with volume data representation, modeling, manipulation, and rendering. In general, a volumetric data set consists of samples (denoted voxels) arranged on a regular grid. Currently, the major application area of volume rendering is medical imaging. In this context, volume data is obtained from medical devices, such as Computer Tomography (CT) scanners and Positron Emission Tomography (PET) scanners. These produce three-dimensional stacks of parallel plane images which can be represented as a volume model and visualized for diagnostic purposes, planning of treatment, or surgery. The development of new techniques to visualize and process images for supporting diagnosis has become a main focus of research. However, despite the advances produced in these fields, there are several challenges that need further development. One of them is multimodal visualization, i.e., the visualization of fused models where the different and, in some cases, complementary information of image modalities is combined in a single image.

The idea of multimodal visualization is to provide as many information on the complementary aspects of the different input models as possible. This process is complex and requires the application of different steps: (i) a registration process to align the two or more images or volumes obtained at different times, from different devices, or from different viewpoints, (ii) a fusion process which, based on different parameters such the original values, the gradient magnitude, etc., converts the information of the input volumes to a single value, and (iii) the visualization of the fused model.

In this chapter, the basic concepts of volume visualization are introduced. Then, multimodal visualization strategies are reviewed, given special interest to the fusion process. Finally, the main information theory concepts that will be used in the scope of this master thesis are introduced.

2.1 Volume Visualization

Visualization, with the generation of visually comprehensible images, has become one of the most important ways of exploring data. It is applied in many scientific fields and application areas, such as material sciences, fluid dynamics, environmental sciences, and medicine. Over the years many techniques have been developed to visualize volumetric data. These methods can be grouped into two different categories:

- Surface Rendering Techniques

These methods involve approximating a surface contained within the data using geometric primitives.

Surface rendering techniques generate the polygonal approximation of an isosurface, i.e., the surface that approximates the volume dataset points with a given property, known as the isovalue. After extracting this intermediate representation, the surface primitives are displayed. This is a good approach for objects with sharply determined borders, like bones in CT, but it is not suitable for amorphous objects which are difficult to represent with surfaces. The most popular approach in this group is the marching cubes algorithm, proposed by Lorensen et al. [19].

In general, these methods require to make a decision for every data sample whether or not the surface passes through it. The main limitation of these methods is that one dimension of information is essentially lost.

- Direct Volume Rendering Techniques

These methods were developed to capture the entire 3D data in a single 2D image. Instead of extracting an intermediate representation, volume rendering provides a method for directly displaying the volumetric data. The original samples are projected onto the image plane in a process which interprets the data as an amorphous cloud of particles. Thus, it is possible to simultaneously visualize information about surfaces and interior structures without making any assumptions about the underlying structure of the data. Direct volume rendering algorithms include approaches such as ray-casting [18], splatting [33], and shear-warp [17]. Volume rendering techniques convey more information than surface rendering methods, but at the cost of increase of the computational complexity, and, consequently, rendering times.

In the scope of this master thesis our interest will be focused on direct volume rendering techniques applied to medical data.

2.2 Direct Volume Rendering Techniques

The volume rendering approach displays data directly as a transparent cloudy object, without any intermediate conversion, assigning attributes such as color, opacity, and gradient vector to each volume element [18, 32, 33]. Volume rendering techniques require an illumination model to determine how the data volume generates, reflects, scatters, and occludes light. An analysis of the main proposed optical models is presented by Max [22].

In Figure 2.1, the main steps of the rendering pipeline are illustrated. The first step is the reconstruction process which constructs a continuous function from the data set. This involves to traverse all data and interpolate values. This step is necessary, since we want to sample the data at any position. The next step is the classification process which assigns a color and an opacity to a data value. This process requires the definition of a transfer function. It is one of the most costly parts of the process. The shading process applies the illumination model. Finally, compositing determines the contribution of a classified and shaded sample to the final image.

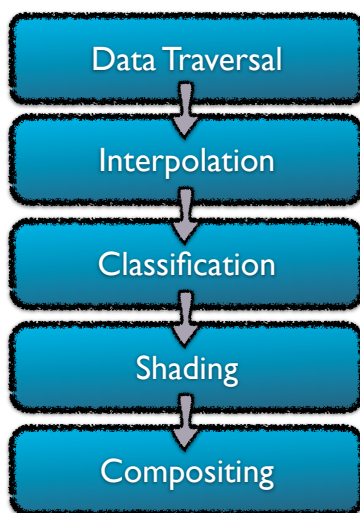


Figure 2.1: Steps of volume-rendering pipeline

There are several possibilities in which these steps can be applied. Both classification and shading can occur before reconstruction (pre-classification, pre-shading) or after reconstruction (post-classification, post-shading). Pre-classification assigns a color and an opacity to the samples before applying a reconstruction filter. On the other hand, post-classification applies the reconstruction filter to the original sample values and then classifies the reconstructed function values. Pre- and post-classification will produce different results whenever the reconstruction does not commute with the transfer functions.

2.2.1 Volume Rendering integral

The basis of most volume rendering techniques is the volume rendering integral in its low-albedo form. This integral is a simplification of the theory of the physics of light transport in the case of neglecting scattering and frequency effects [14, 16]. The volume is seen as a set of particles with certain densities μ , and rays are fired through each pixel on the image plane into the volume. For any ray, the amount of light of wavelength λ is given by

$$I_\lambda = \int_0^L C_\lambda(s)\mu(s)e^{-\int_0^s \mu(t)dt}ds, \quad (2.1)$$

where

- L is the length of the ray
- $C_\lambda(s)$ is the light of wavelength λ reflected at sample s in the direction of the ray. The calculation of $C_\lambda(s)$ can be based on the standard Phong reflection model, given specification of light sources, the material color (assigned by the transfer function), and the normal direction.
- μ is the light extinction coefficient, which defines the rate at which light is occluded per unit length due to scattering or extinction. The weighting by $\mu(s)$ reflects the density at the point.

The integral accumulates the intensity over the length of the ray, attenuated according to the density of the material through which it passes. This attenuation is represented by the exponential term μ [22].

The integral of Equation 2.1 has to be evaluated numerically. If n is the number of steps along the ray at which sample values are taken, using a very simple Riemman sum approximation, we have:

$$I_\lambda = \sum_{i=0}^n C_\lambda(i\Lambda s)\mu(i\Lambda s)\Lambda s \prod_{j=0}^{i-1} \exp(-\mu(j\Lambda s)\Lambda s) \quad (2.2)$$

Replacing the exponential of this sum by the first two terms of its Taylor expansion, i.e.,

$$\exp(-\mu(i\Lambda s)\Lambda s) = 1 - \mu(i\Lambda s)\Lambda s$$

and defining the transparency $t(i\Lambda s)$ as

$$t(i\Lambda s) = \exp(-\mu(i\Lambda s)\Lambda s)$$

so as to give

$$\mu(j\Lambda s)\Lambda s = 1 - t(i\Lambda s) = \alpha(i\Lambda s)$$

where $\alpha = 1 - t$ is the opacity.

This approximation converts Equation 2.2 in the compositing formula that is commonly used in volume rendering:

$$I_\lambda = \sum_{i=0}^n C_\lambda(i\Lambda(s))\alpha(i\Lambda s) \prod_{j=0}^{i-1} (1 - \alpha(j\Lambda s)) \quad (2.3)$$

C and α values are only known at data points. Thus to calculate the values at the sample points $i\Lambda s$, an interpolation process is applied.

Using unit spacing, the Equation 2.3 is further simplified to

$$I_\lambda = \sum_{i=0}^n C_\lambda(i)\alpha(i) \prod_{j=0}^{i-1} (1 - \alpha(j)) \quad (2.4)$$

In practice, this computation is done for R , G , B separately. As it is a sum of intensities of individual samples, each intensity is attenuated by the product of transparencies accumulated as the light passes from sample to observer. The computation can be done recursively by processing one sample at a time, accumulating color for each sample i and opacity separately:

$$C_{out} = C_{in} + (1 - \alpha_{in})\alpha_i C_i, \quad (2.5)$$

and

$$\alpha_{out} = \alpha_{in} + (1 - \alpha_{in})\alpha_i C_i. \quad (2.6)$$

This is a front-to-back ordering, which can be reversed to work back-to-front, in which case only the color needs to be accumulated:

$$C_{out} = C_i\alpha_i + C_{in}(1 - \alpha_i) \quad (2.7)$$

Observe that compositing steps are associative. Thus, groups of samples can be composited and, then, composite the groups, as long as the order is maintained. However, the compositing is not commutative, which means that the order of compositing is important.

2.2.2 Classification of Direct Volume Rendering Techniques

Volume rendering techniques can be classified into three main groups:

- *Object-order volume rendering techniques*, which use a forward mapping scheme where the volume data are mapped onto the image plane.
- *Image-order algorithms*, which use a backward mapping scheme. Rays are cast from each pixel to the image plane through the volume data to determine the final pixel value.

- *Domain-based techniques*, which transform spatial volume data into an alternative domain, such as compression, frequency, or wavelet and, then, a projection is generated directly from that domain.

Some of the most representative volume rendering techniques are:

- *Volume Ray Casting*. This technique can be derived directly from the rendering equation [13]. In this technique, a ray is generated for each desired image pixel. Using a simple camera model, the ray starts at the center of the projection of the camera and passes through the image pixel on the imaginary image plane floating in-between the camera and the volume to be rendered. The ray is clipped by the boundaries of the volume in order to save time. Then, the ray is sampled at regular or adaptive intervals throughout the volume. The data is interpolated at each sample point, the transfer function is applied to form a Red-Green-Blue-Alpha (RGBA) sample, the sample is composited onto the accumulated RGBA of the ray, and the process is repeated until the ray exits the volume. The RGBA color is converted to an RGB color and deposited in the corresponding image pixel. The process is repeated for every pixel on the screen to form the completed image. It provides very high quality results and is usually considered to give the best image quality. Volume ray casting is classified as an image-order algorithm, since the computation emanates from the output image, and not from the input volume data.
- *Splatting*. This technique was proposed by Westover [33] and works by representing the volume as an array of overlapping basis functions, commonly Gaussian kernels with amplitudes scaled by the voxel values. An image is then generated by projecting these basis functions to the screen. The screen projection of these radially symmetric basis functions can be efficiently achieved by the rasterization of a pre-computed footprint lookup table. Here, each footprint table entry stores the analytically integrated kernel function along a traversing ray. A major advantage of splatting is that only voxels relevant to the image must be projected and rasterized. This can reduce the volume data that needs to be both processed and stored [23]. However, depending on the zooming factor, each splat can cover up to hundreds of pixels which need to be processed.
- *Shear Warp*. In this technique, the viewing transformation is transformed such that the nearest face of the volume becomes axis aligned with an off-screen image buffer with a fixed scale of voxels to pixels. The volume is then rendered into this buffer using the far more favorable memory alignment and fixed scaling and blending factors. Once all the slices of the volume have been rendered, the buffer is

then warped into the desired orientation and scaled in the displayed image [17]. This technique is relatively fast in software at the cost of less accurate sampling and potentially worse image quality compared to ray casting. There is a memory overhead for storing multiple copies of the volume, enabling us to have near axis aligned volumes. This overhead can be mitigated using run length encoding.

- *Texture Mapping.* The use of 3D texture mapping was popularized by Cabral [4] for non-shaded volume rendering. The volume is loaded into texture memory and the hardware rasterizes polygonal slices parallel to the viewplane. The slices are then blended back-to-front. Commodity PC graphics cards are fast at texturing and can efficiently render slices of a 3D volume, with realtime interaction capabilities. These slices can either be aligned with the volume and rendered at an angle to the viewer, or aligned with the viewing plane and sampled from unaligned slices through the volume. Volume aligned texturing produces images of reasonable quality, though there is often a noticeable transition when the volume is rotated.

When the visualization involves more than one property per voxel the visualization process requires a fusion process to reduce the data to a single value. In this case specialized fusion techniques are required. In the next section, we review some of the strategies that have been proposed to deal with this situation.

2.3 Multi-Volume Visualization

Multi-volume visualization consists in visualizing the information represented in different volume models in a single image. To carry out this process, a registration process to align the data of the original models is required. Registration is an important focus of research and many different techniques have been proposed to perform it with accuracy and robustness. Once the data is registered, a fusion process is applied to reduce the data to a single value that can be represented in the final image. This fusion process can be done at different steps of the visualization pipeline leading to different strategies. The fusion process is challenging since it is not clear how it can be done in an automate manner while preserving the most relevant information of the models. In addition, it is difficult to determine how the fused model has to be rendered since it is not clear how data has to be classified, i.e., how to define the transfer functions.

In the context of this project, we will focus on the fusion process. For our purposes the registration process will be considered as a pre-processing step.

A general definition of image fusion is given as '*Image fusion is the combination of two or more different images to form a new image by using a certain algorithm*' (Genderen and Pohl 1994). The goal of image fusion is to reduce uncertainty and minimize redundancy in the output while maximizing relevant information particular to an application or task. There are several benefits in using image fusion: wider spatial and temporal coverage, decreased uncertainty, improved reliability, and increased robustness of system performance. However, given the same set of input images, different fused images may be created depending on the algorithm used. Moreover, an algorithm which works fine in some medical imaging modalities could not be used in other area such as meteorological simulation or army industry. The goal is the same in all areas but the properties of each set of input images could be different. Therefore, many aspects have to be looked at before being able to implement and use an image fusion approach. It is important to know what kind of application has to be developed and to study the properties of images that could be taken to be able to decide the most relevant set of input images. The next step is the choice of a suitable fusion algorithm and therefore the fusion level. There are different levels where the fusion can be applied. This choice is important because the pre-processing steps are depending on this.

In the context of medical data, volumes are created from sequences of 2D slices acquired by medical devices. In this way a volume can be seen as a sequence of 2D images or as a 3D volume. Such an interpretation has led to the following classification of fusion strategies:

- *Image-based fusion techniques* which consider the fusion of images. As it is illustrated in Figure 2.2 these methods can be classified in three different categories according to the level at which the fusion is carried out:
 - *Pixel-based fusion techniques* considers the fusion at the lowest processing level, i.e., the fusion of the measured physical parameters represented in the images. Multiresolution analysis is a popular method applied by pixel-level fusion techniques. It consists in generating for each input image a sequence of images that represents information at different levels of detail (or resolution). The main drawbacks of this approach are blurring effects, high sensitivity to noise, and misregistration.
 - *Feature-based techniques* extract features from the images and use them to perform the fusion. This approach may be able to minimize the pixel-based drawbacks. Piella et al. [24] proposes a MR fusion algorithm which combines aspects of region and pixel-level fusion. They use multiresolution recomposition to represent

the input image at different scales and introduce a multiresolution/multimodal segmentation to partition the image domain at these scale. This segmentation is then used to guide the subsequent fusion process. Due to feature-base techniques use the regions of the images, the extraction of the features requires the application of objects recognition process.

- *High-level fusion algorithms* represent a method that uses value-added data where the input images are processed individually for information extraction. The obtained information is then combined applying decision rules to reinforce a common interpretation and resolve differences and furnish a better understanding of the observed objects.

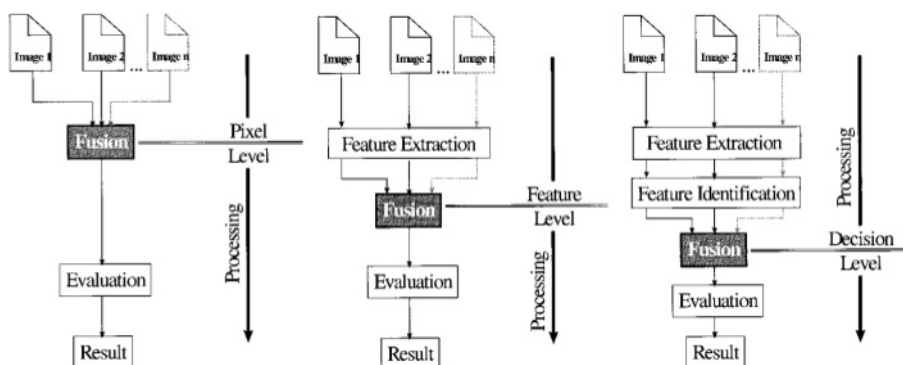


Figure 2.2: An illustration of the three processing levels in 2D image fusion.

- *Volume-based fusion algorithms* which consider the fusion of volumes. Methods of this group can be classified according to the step of the visualization pipeline at which the fusion is carried out [5]:
 - The image-level intermixing is the simplest way for the fusion of two modalities, but it has the disadvantage that the 3D information is lost. Therefore this fusion technique is typically just applied on single slices of the volume. Several techniques have been developed for this purpose, such as alternating pixel display or linked cursor [25, 27].
 - The accumulation-level intermixing fuses the values after optical properties are assigned to each modality individually. In the image-level intermixing, the fusion is done after the 2D images have been rendered.
 - In the illumination-model-level intermixing, optical properties are assigned to a combination of values from the different modalities.

A case study for the rendering of multivariate data, where multiple values are present at each sample point, was done by Kniss et al. [15]. In this work, the idea of multi-dimensional transfer functions to assign optical properties to a combination of values was used. Akiba and Ma [2] used parallel coordinates for the visualization of time-varying multivariate volume data.

Grimm et al. [10] developed methods to efficiently visualize multiple intersecting volumetric objects. They introduced the concept of V-Objects, which represent abstract properties of an object connected to a volumetric data source. Also a method to perform direct volume rendering of a scene composed of an arbitrary number of possibly intersecting V-Objects was presented.

Ferre et al. [9] analyze the rendering of segmented unimodal, hybrid and aligned multimodal voxel models. They propose a data structure that classifies the segmented voxels into categories, so that whenever the model has to be traversed, only the selected categories are visited and the empty and non-selected voxels are skipped. Their strategy is based on: (i) a decision tree, called the rendering decision tree (RDT), which represents the hierarchy of the classification process and (ii) an intermediate run-length encoding (RLE) of the classified voxel model. The traversal of the voxel model given a user query consists of two steps: first, the RDT is traversed and the set of selected categories computed; next, the RLE is visited, but the non-selected runs are skipped and only the voxels of the original model that are codified are accessed in selected runs of the RLE. This strategy has been used to render a voxel model by back-to-front traversal and splatting as well as to construct 3D textures for hardware-driven 3D texture mapping. The results show that the voxel model traversal is significantly accelerated.

Abellan et al. [1] proposed a volume rendering application for multimodal datasets based on 3D texture mapping. The method takes as input two pre-registered voxel models and constructs two 3D textures. It renders the multimodal data by depth compositing view-aligned texture slices of the model. For each texel of a slice, it performs a fetch to each 3D texture and performs fusion and shading using a fragment shader. The application allows users to choose either emission and absorption shading or surface shading for each model. Shading is implemented by using two auxiliary 1D textures for each transfer function. Moreover, data fusion takes into account the presence of surfaces and the specific values that are merged, so that the weight of each modality in fusion is not constant but defined through a 2D transfer function implemented as a 2D texture. This method is very fast and versatile and it provides a good insight into multimodal data.

A common feature of all related methods is that in all the cases the transfer function has to be defined by the user, being a time-consuming and also a non trivial task. Moreover, the user has to determine the information

of the original input models he wants to represent in the fused one. The objective of our project is to automate this process as far as possible. To reach this objective we will use information theory tools.

2.4 Information Theory

In this section, we review some information-theoretic concepts [6, 34] that will be applied to the fusion methods proposed in this work.

2.4.1 Entropy

Let X be a discrete random variable with alphabet \mathcal{X} and probability distribution $\{p(x)\}$, where $p(x) = \Pr\{X = x\}$ and $x \in \mathcal{X}$. This notation is extended to other random variables.

Definition The entropy $H(X)$ of a discrete random variable X is defined by

$$H(X) = - \sum_{x \in \mathcal{X}} p(x) \log p(x), \quad (2.8)$$

where the summation is over the corresponding alphabet and the convention $0 \log 0 = 0$ is taken.

In this work, logarithms are taken in base 2 and, as a consequence, entropy is expressed in bits. The convention $0 \log 0 = 0$ is justified by continuity since $x \log x \rightarrow 0$ as $x \rightarrow 0$. The term $-\log p(x)$ represents the information content (or uncertainty) associated with the result x . Thus, the entropy gives us the average amount of information (or uncertainty) of a random variable. Entropy fulfills that $0 \leq H(X) \leq \log |\mathcal{X}|$.

The definition of entropy is now extended to a pair of random variables.

Definition The joint entropy $H(X, Y)$ of a pair of discrete random variables X and Y with a joint probability distribution $\{p(x, y)\}$ is defined by

$$H(X, Y) = - \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log p(x, y), \quad (2.9)$$

where $p(x, y) = \Pr[X = x, Y = y]$ is the joint probability of x and y .

The conditional entropy of a random variable given another is defined as the expected value of the entropies of the conditional distributions.

Definition The conditional entropy $H(Y|X)$ of a random variable Y given a random variable X is defined by

$$H(Y|X) = - \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log p(y|x), \quad (2.10)$$

where $p(y|x) = \Pr[Y = y|X = x]$ is the conditional probability of y given x . The Bayes theorem relates marginal probabilities $p(x)$ and $p(y)$, conditional probabilities $p(y|x)$ and $p(x|y)$, and joint probabilities $p(x, y)$:

$$p(x, y) = p(x)p(y|x) = p(y)p(x|y). \quad (2.11)$$

If X and Y are independent, then $p(x, y) = p(x)p(y)$. Marginal probabilities can be obtained from $p(x, y)$ by summation: $p(x) = \sum_{y \in \mathcal{Y}} p(x, y)$ and $p(y) = \sum_{x \in \mathcal{X}} p(x, y)$.

The conditional entropy can be thought of in terms of a communication or *information channel* $X \rightarrow Y$ whose output Y depends probabilistically on its input X . This information channel is characterized by a transition probability matrix which determines the conditional distribution of the output given the input [6]. Hence, $H(Y|X)$ corresponds to the uncertainty in the channel output from the sender's point of view, and vice versa for $H(X|Y)$. Note that in general $H(Y|X) \neq H(X|Y)$.

The following properties hold:

- $H(X, Y) = H(X) + H(Y|X) = H(Y) + H(X|Y)$
- $H(X, Y) \leq H(X) + H(Y)$
- $H(X) \geq H(X|Y) \geq 0$

2.4.2 Mutual Information

We now introduce the mutual information between two random variables which expresses the shared information between them:

Definition The mutual information $I(X; Y)$ between two random variables X and Y is defined by

$$\begin{aligned} I(X; Y) &= H(X) - H(X|Y) = H(Y) - H(Y|X) \\ &= \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log \frac{p(x, y)}{p(x)p(y)}. \end{aligned} \quad (2.12)$$

Mutual information represents the amount of information that one random variable, the input of the channel, contains about a second random variable, the output of the channel, and vice versa. That is, mutual information expresses how much the knowledge of Y decreases the uncertainty of X , and vice versa. $I(X; Y)$ is a measure of the correlation or dependence between X and Y . Thus, if X and Y are independent, then $I(X; Y) = 0$.

The following properties hold:

- $I(X; Y) \geq 0$ with equality if and only if X and Y are independent

- $I(X; Y) = I(Y; X)$
- $I(X; Y) = H(X) + H(Y) - H(X, Y)$
- $I(X; Y) \leq \min\{H(X), H(Y)\}$
- $I(X; X) = H(X)$

2.4.3 Entropy Rate

Using the property $H(X_1, X_2) = H(X_1) + H(X_2|X_1)$ (Section 2.4.1) and the induction on n [34], it can be proved that the joint entropy of a collection of n random variables X_1, \dots, X_n is given by

$$H(X_1, \dots, X_n) = \sum_{i=1}^n H(X_i|X_1, \dots, X_{i-1}). \quad (2.13)$$

We now introduce the entropy rate that quantifies how the entropy of a sequence of n random variable increases with n .

Definition The entropy rate or entropy density H_X of a stochastic process¹ $\{X_i\}$ is defined by

$$H_X = \lim_{n \rightarrow \infty} \frac{1}{n} H(X_1, X_2, \dots, X_n) \quad (2.14)$$

when the limit exists.

The entropy rate represents the average information content per symbol in a stochastic process. For a stationary stochastic process², the entropy rate exists and is equal to

$$H_X = \lim_{n \rightarrow \infty} H_X(n), \quad (2.15)$$

where

$$\begin{aligned} H_X(n) &= H(X_1, \dots, X_n) - H(X_1, \dots, X_{n-1}) \\ &= H(X_n|X_{n-1}, \dots, X_1). \end{aligned} \quad (2.16)$$

Entropy rate can be seen as the uncertainty associated with a given symbol if all the preceding symbols are known. It can be also interpreted as the irreducible randomness in sequences produced by an information source [8].

¹A stochastic process or a discrete-time information source $\{X_i\}$ is an indexed sequence of random variables characterized by the joint probability distribution $p(x_1, x_2, \dots, x_n) = \Pr\{(X_1, X_2, \dots, X_n) = (x_1, x_2, \dots, x_n)\}$ with $(x_1, x_2, \dots, x_n) \in \mathcal{X}^n$ for $n \geq 1$ [6, 34].

²A stochastic process $\{X_i\}$ is stationary if two subsets of the sequence, $\{X_1, X_2, \dots, X_n\}$ and $\{X_{1+l}, X_{2+l}, \dots, X_{n+l}\}$, have the same joint probability distribution for any $n, l \geq 1$: $\Pr\{(X_1, \dots, X_n) = (x_1, x_2, \dots, x_n)\} = \Pr\{(X_{1+l}, X_{2+l}, \dots, X_{n+l}) = (x_1, x_2, \dots, x_n)\}$. That is, the statistical properties of the process are invariant to a shift in time. At least, H_X exists for all stationary stochastic processes.

2.4.4 Erasure Entropy

As we have seen in the previous section, the entropy of a sequence X_1, \dots, X_n is equal to the sum of the conditional entropies of each symbol given all preceding (or all succeeding) symbols. Conditioning on the past or the future leads to the same measure of information content. However, conditioning on both the past and the future leads to the following definition [31]:

Definition The erasure entropy of a collection of discrete random variables X_1, \dots, X_n is defined by

$$H^-(X_1, \dots, X_n) = \sum_{i=1}^n H(X_i | X_{\setminus i}) \quad (2.17)$$

$$= nH(X_1, \dots, X_n) - \sum_{i=1}^n H(X_{\setminus i}), \quad (2.18)$$

where

$$X_{\setminus i} = \{X_j, j = 1, \dots, n, j \neq i\}. \quad (2.19)$$

In addition, analogously to the conventional entropy, the erasure entropy rate is defined as the limiting normalized erasure entropy, i.e., the limit of the arithmetic mean of the conditional entropies of each symbol given all preceding and succeeding symbols.

Definition The erasure entropy rate of a process $X = \{X_i\}$ is defined by

$$H_X^- = \lim_{n \rightarrow \infty} \frac{1}{n} H^-(X_1, \dots, X_n). \quad (2.20)$$

Erasure entropy is strictly lower than the conventional entropy (unless the source is memoryless, in which case they are identical)[31].

Chapter 3

A General Framework for Multi-Modal Visualization

Although multi-volume visualization is a valuable tool for diagnosis, the complexity of generating these visualizations hinders their application in clinical practice. To overcome this limitation we decided to develop in the scope of this project a software in which engineers and practitioners combine their knowledge. Motivated by a common interest, the medical imaging group of the University of Girona and medical researchers from the Hospital Josep Trueta of Girona created a working group. We aimed at developing a tool that integrates the operations and tools required for multimodal visualization and also where they can be tested and compared.

In this Chapter, we present the structure and the main functionalities of the designed platform. In Section 1 and 2, we describe the registration and segmentation modules. In Section 3, we present the fusion strategies that have been implemented. In Section 4, we analyze the different visualization strategies that have been implemented to present the results to the user.

3.1 Introduction

The design of the proposed multi-volume visualization platform is presented in Figure 3.1. As can be seen it consists of three different levels:

- Medical Data Input/Output layer which integrates the modules that perform medical image data reading and writing tasks.
- Kernel integrating the main modules of the application, each one designed to support an specific operation.
- Graphical user interface (GUI) which interacts with the user by observing user actions, basically button press and mouse movements. This layer is also responsible for the output display of the program.

The proposed framework has been implemented using the National Library of Medicine Insight Segmentation and Registration Toolkit (ITK) [28], which also guarantees the code quality for the library methods, Visualization Toolkit (VTK) [30], which is a software system for 3D computer graphics, image processing and visualization, and QT [29], which is a cross-platform application and UI framework.

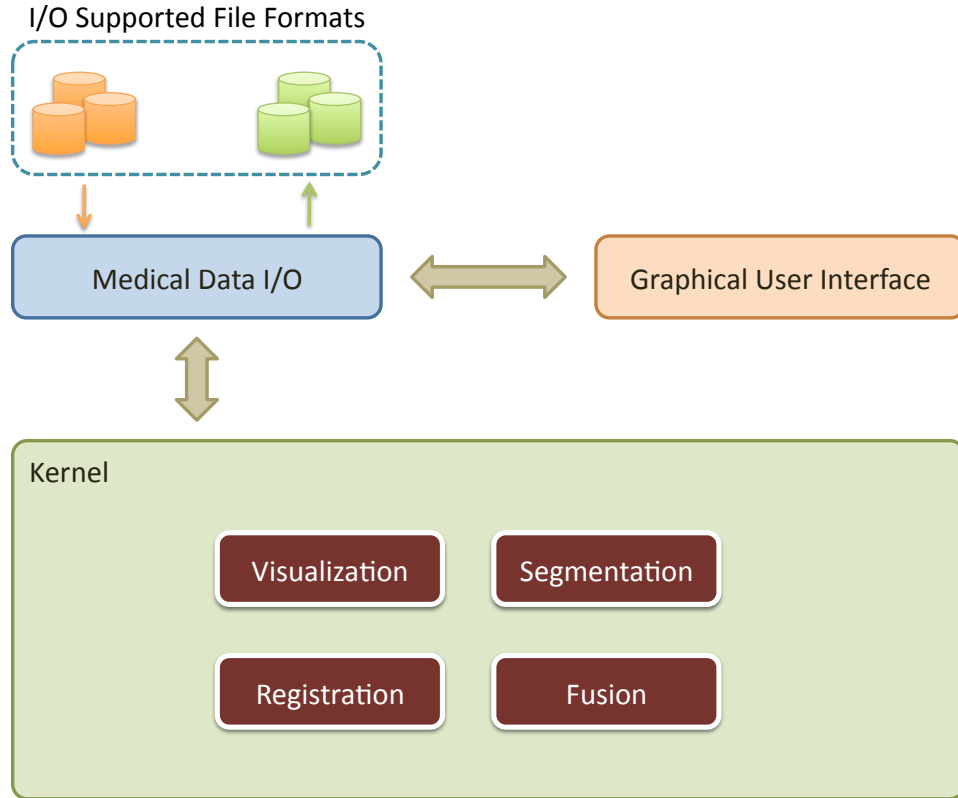


Figure 3.1: Multi-Volume Visualization Platform

3.2 Registration Module

Registration is a fundamental task in image processing used to match two or more images or volumes obtained at different times, from different devices or from different viewpoints. Basically, it consists in finding the geometrical transformation that enables us to align volumes into a unique coordinate space. Hence, it is the previous step for multi-volume visualization.

The image registration pipeline starts with the selection of the two images to be registered. One of the two images is defined as the fixed image and the other one as the moving image. Given these images, registration

is treated as an optimization problem with the goal of finding the spatial mapping that will bring the moving image into alignment with the fixed one. This process can be described as a process composed of four basic elements: the transformation, the interpolator, the metric, and the optimizer. The transformation component represents the spatial mapping of points from the fixed image space to points in the moving image space. The interpolator is used to evaluate moving image intensity at non-grid positions. The metric component provides a measure of how well the fixed image is matched by the transformed moving image. This measure forms the quantitative criterion to be optimized by the optimizer over the search space defined by the parameters of the transformation.

We restricted our study to rigid or affine transformations, i.e., transformations that preserve all distances, the straightness of lines (and the planarity of surfaces), and all nonzero angles between straight lines.

To perform the registration we implemented the *MultiResolutionImageRegistrationMethod* provided by the ITK [28]. This method uses a coarse to fine approximation as it is illustrated in Figure 3.2. For each image, a pyramidal representation is created where each level of the pyramid represents a different level of resolution. The registration is first performed at the coarsest level using the images at the first level of the fixed and moving image pyramids. The transform parameters determined by the registration are then used to initialize the registration at the next finer level using images from the second level of the pyramids. This process is repeated until the last level is reached.

As a transform we used *itkAffineTransform* which allows the definition and manipulation of affine transformations of an n-dimensional affine space (and its associated vector space) onto itself. The used metric is *itkMattesMutualInformationImageToImageMetric* which computes the mutual information between two images to be registered using the method of Mattes et al [21].

3.3 Segmentation Module

Image segmentation is the process of labeling each voxel in an image dataset according to certain parameter or features. In the case of medical images, the segmentation considers the tissue type or the anatomical structure contained in the voxel. Since a segmented image provides richer information than the original one, it is an essential tool in medical environments. It is used to improve visualization of medical imagery, to perform quantitative measurements of image structures, to build anatomical atlases, to search shapes of anatomical structures, or to track anatomical changes over time. Segmentation is considered a very difficult task and a lot of research is being done to develop automatic segmentation techniques. The main aspects that

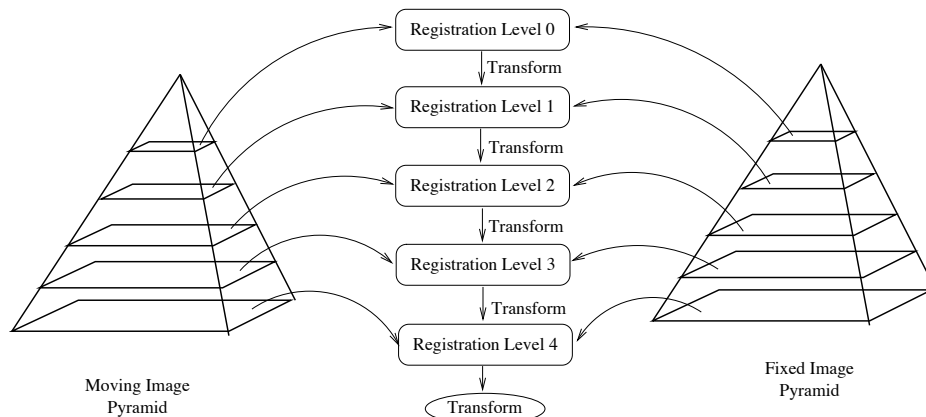


Figure 3.2: Conceptual representation of the multi-resolution registration process.

make this process so difficult are the imaging process itself and the anatomy that is represented in the images. The imaging process is chosen so that its interactions with the tissues of interest will provide clinically relevant information about the tissue in the resulting output image. But this does not mean that the anatomical feature of interest will be particularly separable from its surroundings. The second aspect that makes segmentation difficult is the complexity and variability of the anatomy that is being imaged. It may not be possible to locate or delineate certain structures without detailed anatomical knowledge. For this reason, in most of medical applications, segmentation is carried out manually by an expert radiologist. This is a very time consuming task and also sensitive to subjective errors.

Due to the importance of segmentation in the medical context we have implemented a segmentation module to allow, if it is desired, the fusion of segmented images. In the experiments presented in this master thesis, we have only used a simple method which quantizes the original model in a user defined number of bins.

In Figure 3.3 we can see an input model and its quantizations using 16, 32 and 64 bins, respectively.

3.4 Fusion Module

This module integrates the different methods we propose in this master thesis and we describe in detail in Chapters 4 and 5. All these methods are based on information theory measures.

In our methods we propose two different approaches to carry out the fusion:

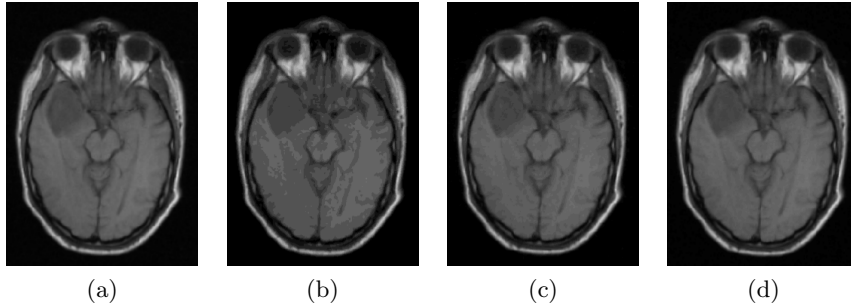


Figure 3.3: (a) Original model. This model is quantized using (b) 16, (c) 32 and (d) 64 bins, respectively.

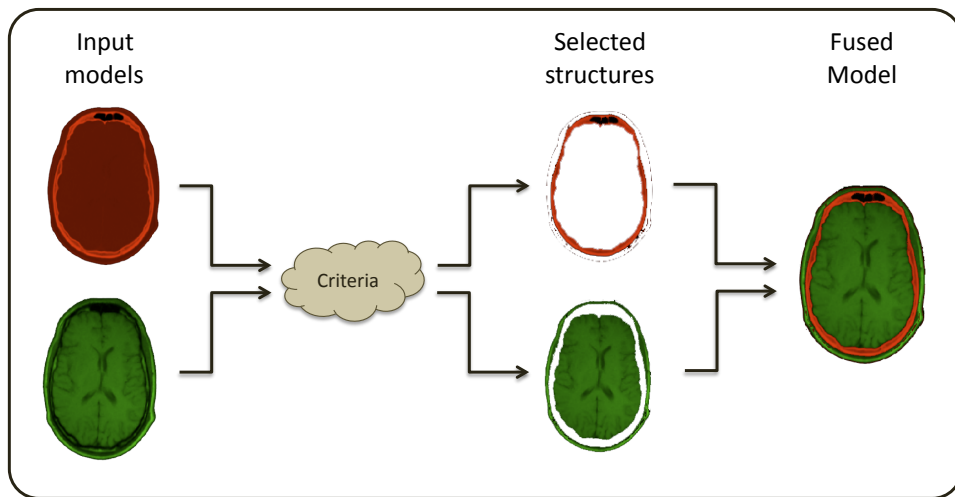
- The symmetric fusion approach considers the two input models to perform the fusion (see Figure 3.4a). According to the applied fusion algorithm, in the fusion model it can appear the voxel of one model, the voxel of the other model, or a combination of both models. The criteria used to determine the contribution of each input model is defined by the applied strategy. In our methods, the criteria are based on the evaluation of the information theory measures.

An interesting property of this type of fusion is its simplicity. Moreover, they can be applied in an automatic way without user interaction. This allows to obtain results in lower times. Some parameters can also be modified.

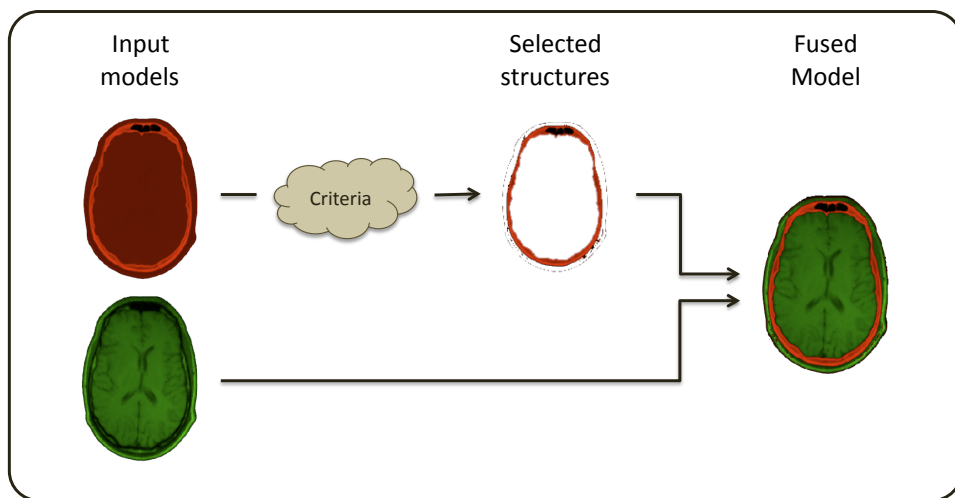
- The asymmetric fusion approach defines one of the input models as the main model and considers it as a reference (see Figure 3.4b). This model is used to perform all the computations that will determine the fusion. In contrast to the symmetric fusion, it is non automatic because you have to decide which part of the model you want to visualize. To enhance this selection process we defined a set of color maps that identify the most representative parts of the volume.

3.5 Visualization Module

The visualization module integrates the algorithms required to present the data to the user. We integrated the Ray Casting 3D visualization technique and also 2D visualization strategies to generate color maps that allows to identify the contribution of each part of the volume according to some information theory measure.



(a)



(b)

Figure 3.4: Concept of (a) symmetric fusion and (b) asymmetric fusion.

3.5.1 Ray-casting implementation

Most of the techniques that have been implemented to visualize fused models are based on ray-casting [40, 56]. An efficient splatting of run-length encoded aligned multimodalities has been proposed by Ferre et al. [54]. The major drawback of these methods is that they are software-based, and therefore, they are not fast enough to provide the interactivity needed by physicians to analyze the data. Texture mapping [57] can provide this speed because it exploits hardware graphics acceleration. Hong et al.[58] use 3D texture-based rendering for multimodality. They use aligned textures in order to use the same 3D coordinates to fetch the texture values in the two models and combine the texel values according to three different operators. Robler et al. [59] describe a visualization framework to handle multi-volumes and render them using 3D textures, but they do not address the fusion of data. Abellan et al. [55] describe a method that takes full profit of graphics hardware capabilities and it proposes a general framework that not only supports multi-volumes, but actually offers different ways of performing fusion. The major novelty of this method is how it lets users specify the fusion parameters.

We designed a parallelized CPU-based ray-casting implementation. For the implementation, we used the VTK libraries [60]. Since VTK methods do not support fused model visualization, we implemented a new *vtkVolumeRayCastFunction* that integrates the *RayCast* method and performs the fusion using a *VoxelShader*, one for each one of the proposed fusion strategies. The *VoxelShader* has as input a voxel p , the two volumes to fuse, their respective transfer functions and the fusion strategy with the corresponding information theory measure. The output of the *VoxelShader* is a colour and an opacity.

The contribution of each input model in the fusion result is done in real-time but, in order to improve the interactivity some calculations are done and some auxiliary structures are generated in a pre-processing step. Every fusion method has its particular auxiliary information as we will see in the next chapter.

3.5.2 Color Coded Maps

Our fusion strategies depend on the value of an information theory measure at a voxel. To represent these values we apply a color coded visualization that allows to represent the measures all over the volume. We apply the color codification illustrated in Figure 3.5. The voxel which has the minimum value of the measure will be visualized using the blue color, the voxel with the maximum value will be red, and the other values will take the intermediate colors. Sometimes, the minimum value or the maximum value, or both, are associated to very few values of the model. In these cases, red or blue colors



Figure 3.5: Color scale used to colorize the information theory measures. The blue color corresponds to the minimum value of the measure and the red color to the maximum value.

are not represented as we can see in Figure 3.6. Therefore, in these cases it is interesting to collapse a certain percent of values in order to balance the map. In Figure 3.6 we can see the map of the conditional entropy measure of a model which is described in Chapter 4. Both images correspond to the same map of the same model but the right one is coloured collapsing the 10 percent of minimum values.

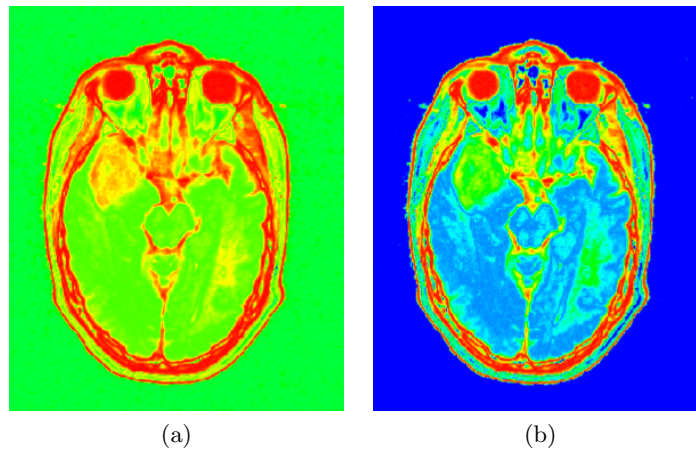


Figure 3.6: Comparison of (a) the original map of the conditional entropy measure of a model and (b) the same colored map collapsing the 10% of minimum values.

3.6 User Interface Design

In Figure 3.7 we illustrate the main areas of the user interface framework. The main area is the *viewer area* which is divided in five viewers (see Figure 3.8). The left viewers are the *input viewers* and they contain the models that will be fused. The model visualized in each viewer can be selected using the menu that appears when the right button is clicked. The viewer of the middle is the *fused viewer*. In this viewer, the fusion of the input models is visualized. The right viewers are the *helper or auxiliary viewers* where certain values computed on the input models are represented in a color coded map. These viewers can be used to visualize complementary

information of the fusion. Using these viewers, the user can analyze the maps of the information theory measures of each input model used in the fusion process, the partitioned models where each level is coloured using a different color, and the contribution of each input model in the fusion result can also be visualized.

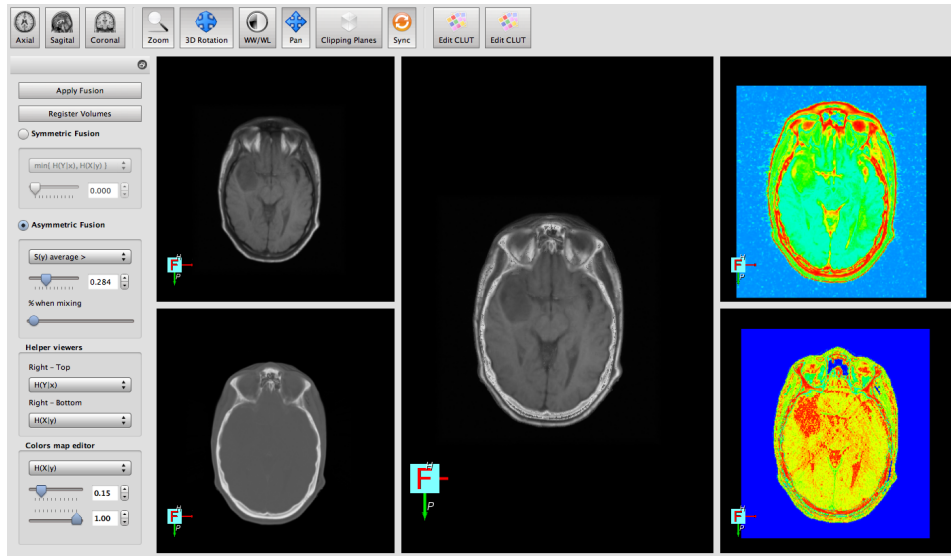


Figure 3.7: Framework User Interface.

The left area is the *fusion option menu* (see Figure 3.9). It controls all the information related with the fusion and the information visualized in the *helper viewers*. This menu can be divided in five parts.

On the top, there are two push buttons which are used to apply the fusion and to register the models. The first time that the *apply fusion button* is clicked, all the information theory measures are computed and the fusion is visualized. If the models don't change, the other clicks only update the transfer functions. The *register volume button* registers the models taking the top input model as the *fixed model* and the bottom as the *moving model*.

After that, we have options which allow us to decide the fusion method. Taking into account the classification described in Section 3.4 we can choose between symmetric or asymmetric fusion methods. The methods of each group are the fusion strategies described in Chapters 4 and 5.

The third part contains options to be able to choose what we want to show in the helper viewers. We can show the maps of the information theory measures and the contribution of each input model in the fusion result can also be visualized using the combo boxes.

The last group of options allows us to manipulate the color maps of the information theory measures presented in Section 3.5.2. The top slider is used to collapse the minimum values and the bottom slider to collapse the

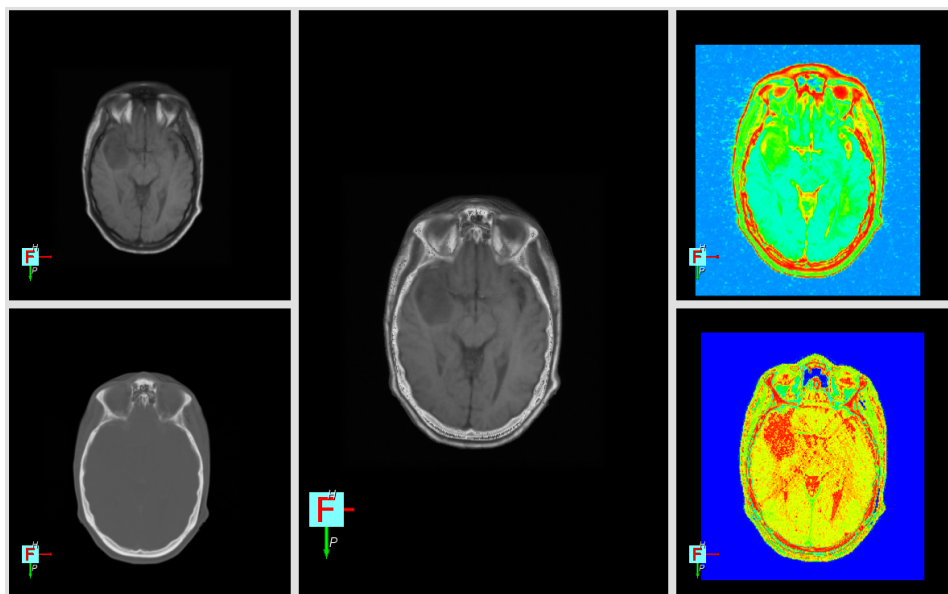


Figure 3.8: Viewer area.

maximum values.

Finally, the top area is the *tool bar* which contains the tools to interact with the viewers (see Figure 3.10). We are able to do basic actions such as applying zoom and modifying the orientation or the window level. Moreover, we added some advanced tools: a transfer function can be defined for each input model, the bounding box of the models can be manipulated in order to visualize the interior, and, finally, a synchronization tool is defined to be able to visualize the same part of the model in all of the viewers.

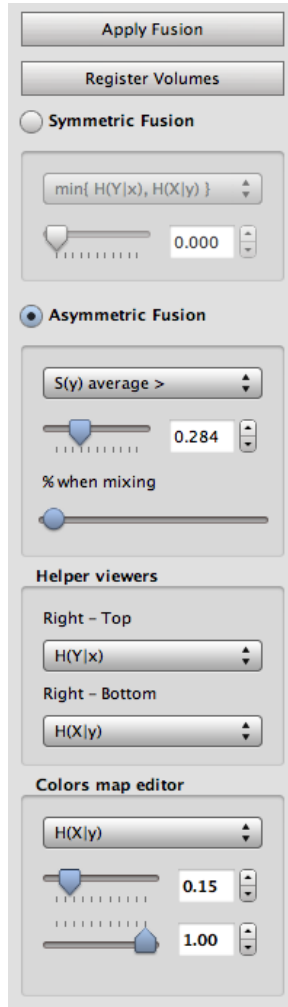


Figure 3.9: Fusion option menu.



Figure 3.10: Framework Tool Bar.

Chapter 4

Fusion using the Registration Channel

4.1 Introduction

In this chapter, we present an information-theoretic approach for fusion visualization which uses the conditional entropy and the mutual information to decide the contribution of each input model for each voxel.

We group these two measures in the same approach because, even though the meanings of the measures are quite different, both measures take into account all input models to compute the information which will be used in the fusion. Therefore, we compute the conditional entropy and the mutual information for each input model taking into account the other input models. If one of the input models is changed, the conditional entropy and the mutual information measures of each model have to be recomputed.

Section 2 presents the conditional entropy method and the fusion strategies associated with that measure. The mutual information method and the corresponding fusion strategies are presented in Section 3. Finally, in Section 4, we discuss about the structures and memory needed for each measure to be able to achieve a good interaction in rendering time.

4.2 Conditional Entropy Method

First of all, 3D images or models A and B are registered using the *MultiResolutionImageRegistrationMethod* provided by ITK [12]. Then, we consider an information channel between A and B , called registration channel.

In the information theory context, the registration of two images A and B is represented by an information channel $X \rightarrow Y$, where random variables X and Y represent the images A and B , respectively. Their marginal probability distributions, $p(x)$ and $p(y)$, and the joint probability distribution, $p(x, y)$, are obtained by simple normalization of the marginal and joint

intensity histograms of the overlapping areas of both images [20].

To compute the conditional entropy and mutual information of this channel, the joint histogram is constructed by counting the number of times a combination of intensity values occurs. For each pair of corresponding points (a, b) , where a is a point in image A and b a point in image B , the entry $(f_A(a), f_B(b))$ in the joint histogram is increased, where $f_A(a)$ and $f_B(b)$ represent the intensity values at points a and b , respectively. From now on, $f_A(a)$ and $f_B(b)$ will be abbreviated as x and y , respectively.

4.2.1 Conditional Entropy of the Registration Channel

Given the information channel $X \rightarrow Y$, the conditional entropy can be computed. From Equation 2.10, the conditional entropy can be rewritten as

$$H(Y|X) = \sum_{x \in \mathcal{X}} p(x) \left(- \sum_{y \in \mathcal{Y}} p(y|x) \log p(y|x) \right) \quad (4.1)$$

$$= \sum_{x \in \mathcal{X}} p(x) H(Y|x), \quad (4.2)$$

where

$$H(Y|x) = - \sum_{y \in \mathcal{Y}} p(y|x) \log p(y|x), \quad (4.3)$$

and \mathcal{X} and \mathcal{Y} represent the set of intensity values (alphabet) of images A and B , respectively. For the registration channel, $H(Y|X)$ represents the uncertainty in Y known X , and $H(Y|x)$, called color entropy, represents the uncertainty in Y known the intensity value x . That is, $H(Y|x)$ quantifies the uncertainty we have on the intensity value y in image B when we have a given value x in image A .

Figure 4.1 illustrates the conditional entropy (CE) maps of two different images (CT and MR) which have been previously registered. Each conditional entropy map shows the values $H(Y|x)$ (Figure 4.1b) and $H(X|y)$ (Figure 4.1d) for each voxel, when X and Y represent, respectively, the CT and MR images. These maps are colorized using the color scale described in Section 3.5.2, where the blue color represents the minimum conditional entropy value and the red color the maximum conditional entropy value.

Low values of CE are obtained when a value x in image A has a low number of corresponding values y in image B . The color entropy $H(Y|x)$ is equal to zero when all the corresponding values of x in B have the same intensity (not necessarily equal to x). The maximum entropy $H(Y|x)$ is obtained when all the corresponding values of x in B have different intensity values.

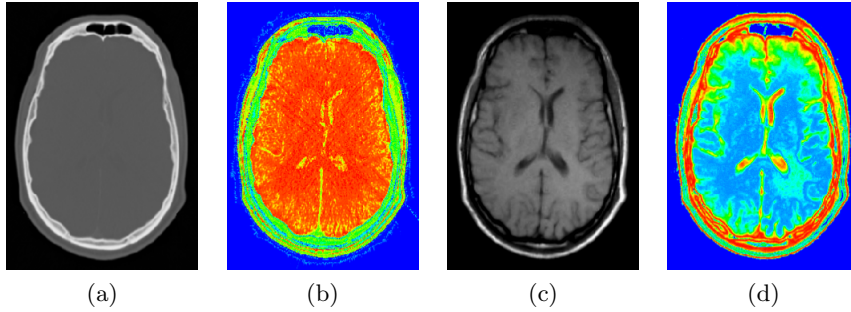


Figure 4.1: Original (a) CT and (c) MR models and (b) and (d) are their corresponding CE colored maps.

4.2.2 Fusion Strategies

As we described in Section 3.4, we divide the fusion strategies into two groups: *symmetric* and *asymmetric* fusion. The *symmetric fusion* compares for each voxel the conditional entropy of the input models and computes, using a given criterion, the contribution of each of them in the fusion visualization. On the other hand, the *asymmetric fusion* takes one of the two models as the reference and, for each voxel, whether its conditional entropy fulfills a given criterion the information of this model is used for the fusion visualization, otherwise the information of other model is used.

In this section we describe the fusion methods implemented using this measure and we illustrate the results in some figures. In Chapter 6 we evaluate the performance of the proposed fusion methods taking into account the opinion of medical experts. In all images we are working with a CT and MR-T1 studies of *training_001* patient which was obtained from the Retrospective Image Registration Evaluation Project database [7]. This project was designed to compare retrospective CT-MR and PET-MR registration techniques and has been used by a number of groups. The models provided by this database is also known as the “Vanderbilt Database”.

In order to know exactly which is the origin of the information of each pixel of the fusion model, the CT study is visualized using a red scale and the MR-T1 is visualized using a green scale.

Using this measure we implemented two symmetric fusion methods and two asymmetric fusion methods.

Symmetric fusions

The first symmetric fusion is called *MinConditionalEntropy (MCE) fusion*. For each pair of voxels, the resulting $F_p(X, Y)$ intensity for voxel p is obtained from the voxel of the input model which has the minimum

conditional entropy:

$$F_p(X, Y) = \begin{cases} x & \text{if } H(Y|x) \leq H(X|y) \\ y & \text{if } H(Y|x) > H(X|y) \end{cases} \quad (4.4)$$

where X and Y represent the input models A and B , $F(X, Y)$ represents the fused model, $x = f_X(a)$ and $y = f_Y(b)$ are, respectively, the intensity values associated with points a and b of images A and B and the subindex p represents a 3D point in the fused model which matches with points a and b .

The main drawback of this strategy is that it is non flexible to allow us to modify the fusion result to try to improve it. Figure 4.2 shows some result of this method. The original CT and MR models and these models quantized using 16, 32 and 64 bins are used to illustrated it. A 3D multi-modal visualization using MCE fusion is shown in Figure 4.3.

NormalizedMinConditionalEntropy (NMCE) fusion is the second symmetric fusion strategy implemented:

$$F_p(X, Y) = \begin{cases} x & \text{if } \hat{H}(Y|x) \leq \hat{H}(X|y) \\ y & \text{if } \hat{H}(Y|x) > \hat{H}(X|y) \end{cases} \quad (4.5)$$

where \hat{H} represents the normalized conditional entropy with values between 0 and 1:

$$\hat{H}(Y|x) = \frac{H(Y|x) - H(Y|min)}{H(Y|max) - H(Y|min)} \quad (4.6)$$

where :

- $x, min, max \in X$, and
- $\forall u \in X \quad H(Y|u) \geq H(Y|min)$, and
- $\forall u \in X \quad H(Y|u) \leq H(Y|max)$

This method is similar to the *MCE fusion* but the conditional entropy has been normalized for each input model individually. We implemented it in order to relax the rigidity of the *MCE fusion*.

The mechanism to interact with the fusion works like the mechanism to modify the colour map described in Section 3.6. If we collapse a percent of the minimum values of $\hat{H}(Y|X)$ or $\hat{H}(X|Y)$, their value are replaced by 0. Otherwise, if we collapse a percent of the maximum values of $\hat{H}(Y|X)$ or $\hat{H}(X|Y)$, their values are replaced by 1. Therefore, if we collapse a percent of the minimum values of $\hat{H}(Y|X)$, these values will be more relevant in the fusion because the minimum value is selected. On the other hand, if we

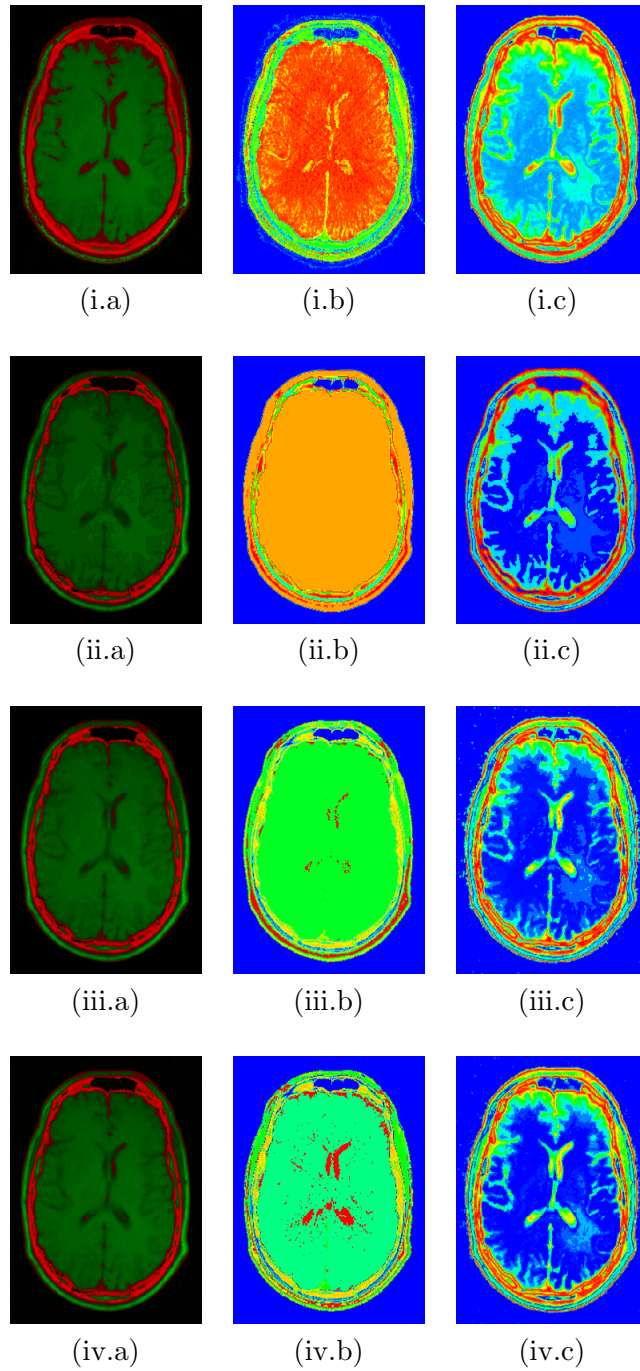


Figure 4.2: (a) Fusion results of the CT and MR model shown in Figure 4.1 using *MCE fusion*. (b - c) Conditional Entropy maps of the CT and MR-T1 models, respectively. (i) Original resolution models. These models have been quantized using (ii) 16, (iii) 32, (iv) 64 bins, respectively.

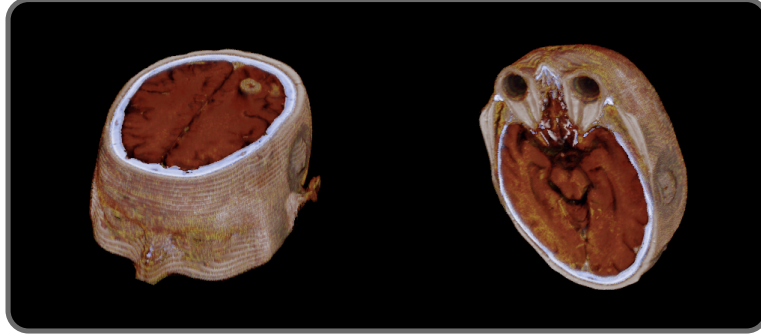


Figure 4.3: 3D fusion visualization using MCE strategy.

collapse a percent of the maximum values of $\hat{H}(Y|X)$, $\hat{H}(Y|X)$ will be less relevant.

Figure 4.4 shows some fusion results using *NMCE fusion* without any collapse and applying some collapse. In order to show the collapse applied, we include the percent in the colored maps where *min* or *max* means that low or high values have been collapsed. Figure 4.5 shows a 3D multi-modal visualization using NMCE fusion.

Asymmetric fusions

The asymmetric strategies implemented are based on the visualization of the reference model if its conditional entropy is less than a threshold or if it is greater than a threshold. Otherwise, in both strategies, the other model is visualized.

- *ConditionalEntropyMoreThanThreshold-based (CEMTT) fusion*

$$F_p(X, Y) = \begin{cases} x & \text{if } H(Y|x) > T \\ y & \text{if } H(Y|x) \leq T \end{cases} \quad (4.7)$$

where T is a threshold value defined by the user and X is the reference model.

- *ConditionalEntropyLessThanThreshold-based (CELTT) fusion*

$$F_p(X, Y) = \begin{cases} x & \text{if } H(Y|x) < T \\ y & \text{if } H(Y|x) \geq T \end{cases} \quad (4.8)$$

These strategies are opposed but it is interesting to have both in order to cover all the possibilities. In Figure 4.6 we can see an illustration of the *CEMTT* and *CELTT* fusions using the CT and the MR-T1 as a reference

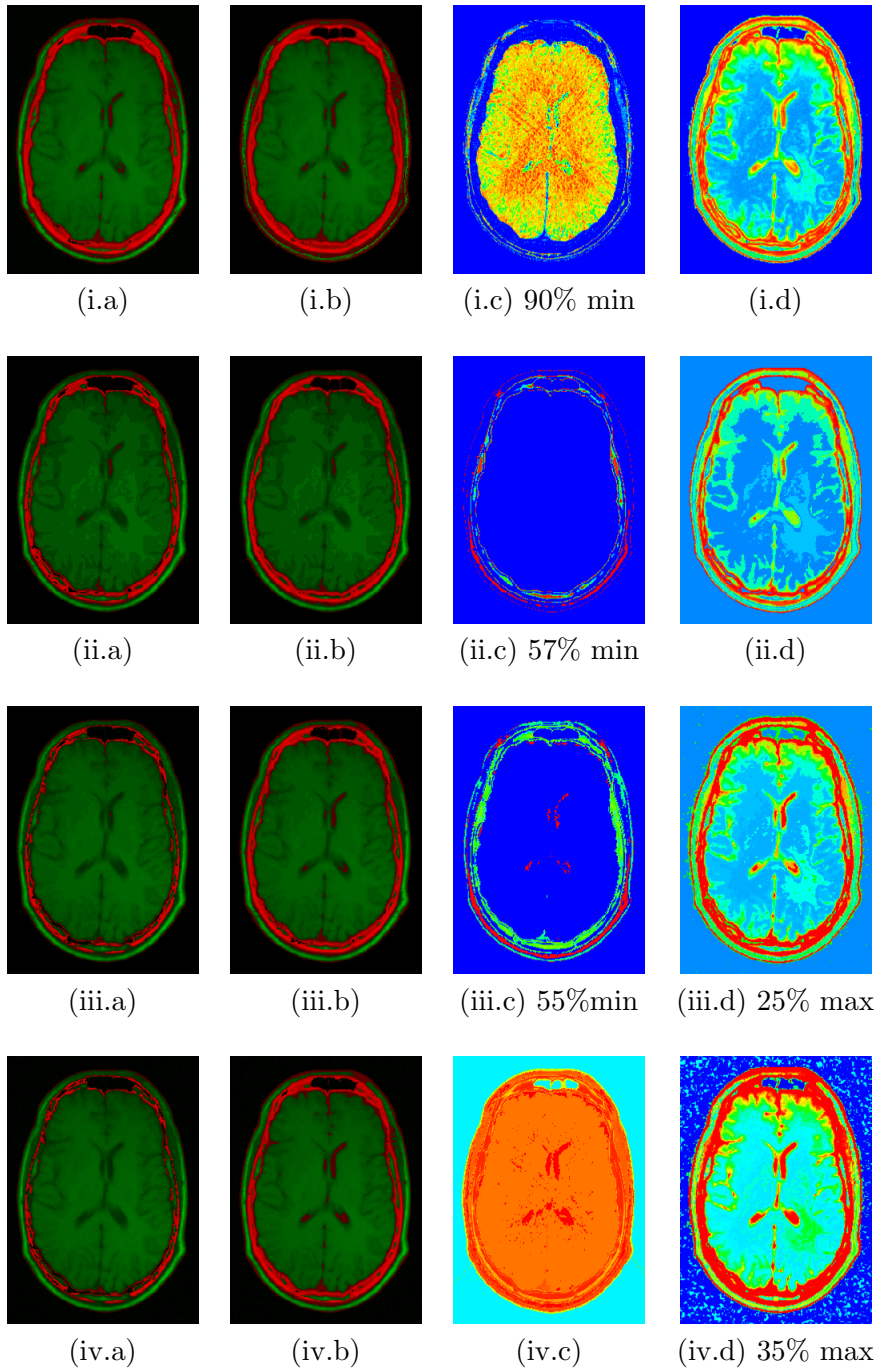


Figure 4.4: (a) Fusion results of the CT and MR model shown in Figure 4.1 using *NMCE fusion* without applying any collapse and (b) applying some collapse. (c - d) Conditional Entropy collapsed maps of the CT and MR-T1 models respectively. (i) Original resolution models. These models have been quantized using (ii) 16, (iii) 32, (iv) 64 bins, respectively.

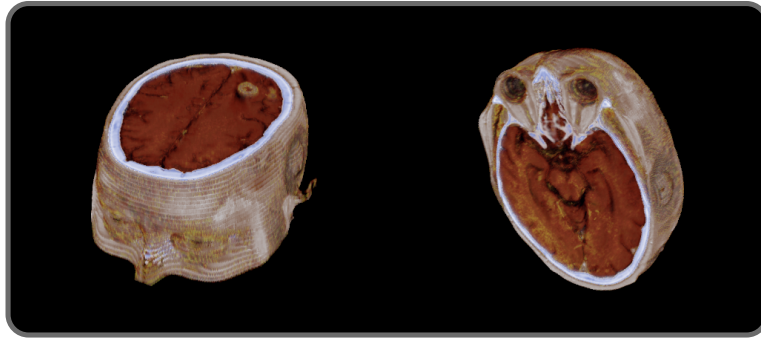


Figure 4.5: 3D fusion visualization using NMCE strategy.

model, respectively. Some results using CT and MR-T1 models quantized using 16, 32 and 64 bins are presented in Figure 4.7. In Figure 4.8, we show a 3D multi-modal visualization using CELTT fusion strategy.

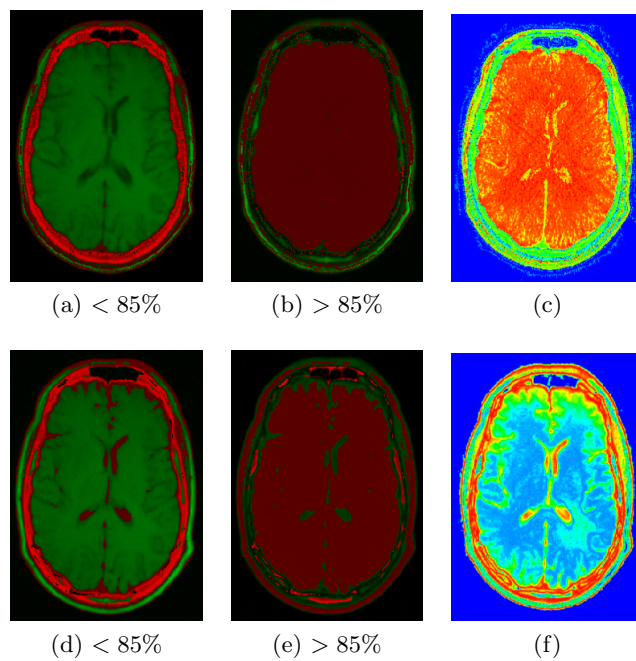


Figure 4.6: Fusion results of the CT and MR model shown in Figure 4.1 using (a) *CELTT fusion* and (b) *CEMTT fusion* where the reference model is the CT study. Fusion result of (b) *CELTT fusion* and (d) *CEMTT fusion* where the reference model is the MR-T1 study. (c , f) Conditional Entropy maps of the CT and MR-T1 models respectively.

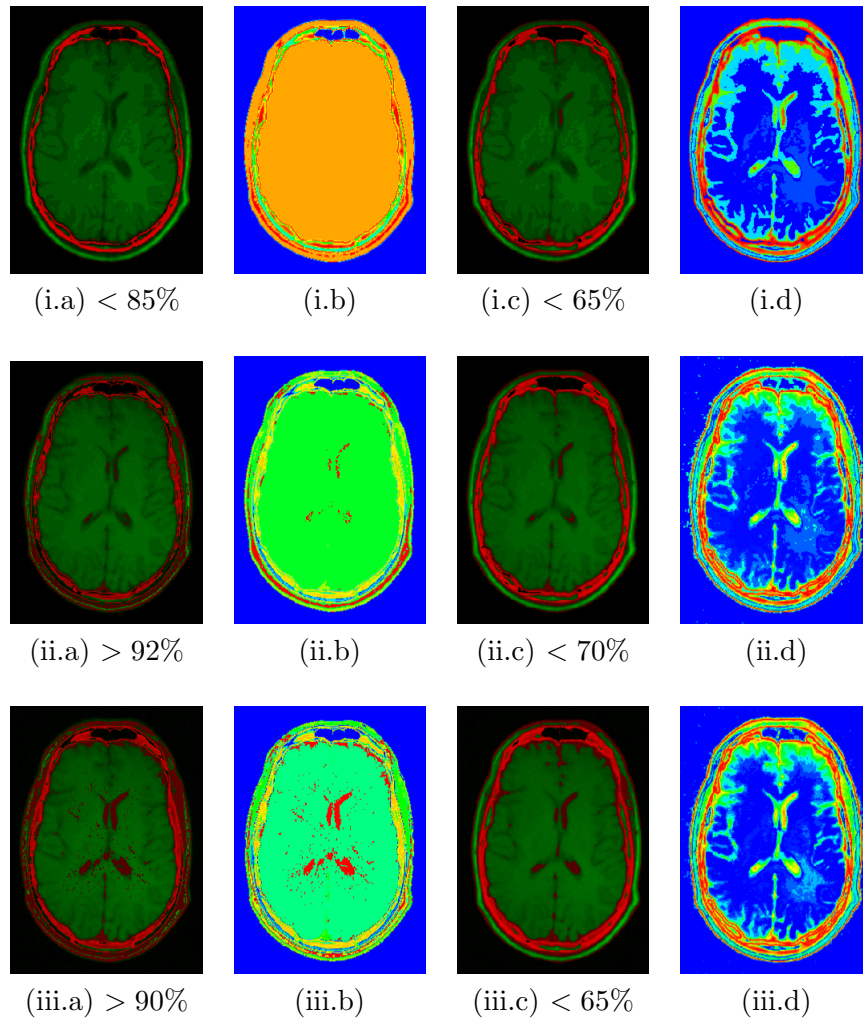


Figure 4.7: Fusion results of the CT and MR model shown in Figure 4.1 using *CELTT fusion* or *CEMTT fusion* using (a) CT and (c) MR-T1 models as a reference. The models are the original models quantized using (i) 16, (ii) 32, (iii) 64 bins, respectively.

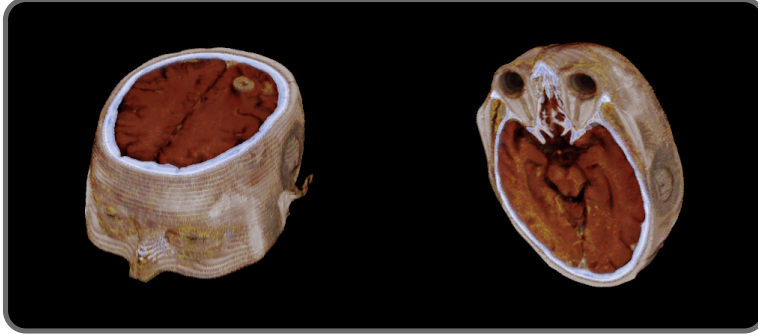


Figure 4.8: 3D fusion visualization using CELTT strategy.

4.3 Mutual Information Method

4.3.1 Mutual Information of the Registration Channel

Given the information channel $X \rightarrow Y$, mutual information can be computed. From Equation 2.12, the mutual information can be rewritten as

$$I(X; Y) = \sum_{x \in \mathcal{X}} p(x) I(x; Y), \quad (4.9)$$

where

$$I(x; Y) = \sum_{y \in Y} p(y|x) \log \frac{p(y|x)}{p(y)}. \quad (4.10)$$

For the registration channel, $I(X; Y)$ represents the shared information or dependence between 3D images A and B and $I(x; Y)$ gives as the degree of correlation between the intensity values x in image A and the corresponding intensity values in image B . We can also consider that $I(x; Y)$ represents the (shared) information associated with the intensity value x .

Figure 4.9 illustrates the mutual information (MI) maps of two different images (CT and MR) which have been previously registered. Each mutual information map shows the $I(x; Y)$ (Figure 4.9b) and $I(y; X)$ (Figure 4.9d), respectively, for each voxel. The maps are colorized using the color scale described in Section 3.5.2 where the blue color represents the minimum mutual information value and the red color the maximum mutual information value.

4.3.2 Fusion Strategies

Following the same structure used in Section 4.2.2, we describe the fusion strategies implemented using the mutual information measure and we illustrate the results in some figures. In all images we are working with the CT and MR-T1 studies of *training_001* patient and the CT study is visualized using a red scale and the MR-T1 is visualized using a green scale.

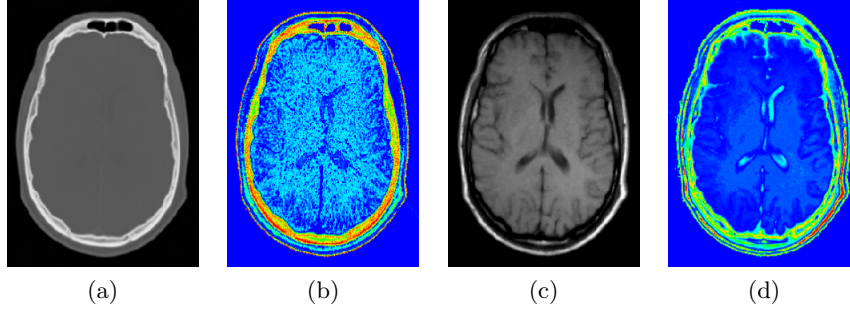


Figure 4.9: Original (a) CT and (c) MR models and (b) and (d) are their corresponding MI colored maps.

Using this measure we implemented two symmetric fusion methods and two asymmetric fusion methods.

Symmetric fusions

The first symmetric fusion called *MaxMutualInformation (MMI) fusion* compares, for each pair of voxels, the voxel of the input model and visualize the voxel which has the maximum mutual information:

$$F_p(X, Y) = \begin{cases} x & \text{if } I(x; Y) \leq I(y; X) \\ y & \text{if } I(x; Y) > I(y; X) \end{cases} \quad (4.11)$$

where X and Y are the input model and $F(X, Y)$ is the fused model.

Some results can be seen in Figure 4.10 and Figure 4.11 shows a 3D multi-modal visualization using this fusion strategy. A drawback of this method is that it has the same problem related with the flexibility that the *MCE fusion* described in Section 4.2.2 has. In order to avoid that, we implemented the *NormalizedMaxMutualInformation (NMMI) fusion*.

NMMI fusion, as can be seen in Equation 4.12, works in the same way as the *MMI fusion* but with the only difference that NMMI works with the mutual information of each input model normalized between 0 and 1:

$$F_p(X, Y) = \begin{cases} x & \text{if } \hat{I}(x; Y) \leq \hat{I}(y; X) \\ y & \text{if } \hat{I}(x; Y) > \hat{I}(y; X) \end{cases} \quad (4.12)$$

where \hat{I} represents the normalized mutual information with values between 0 and 1:

$$\hat{I}(x; Y) = \frac{I(x; Y) - I(\min; Y)}{I(\max; Y) - I(\min; Y)} \quad (4.13)$$

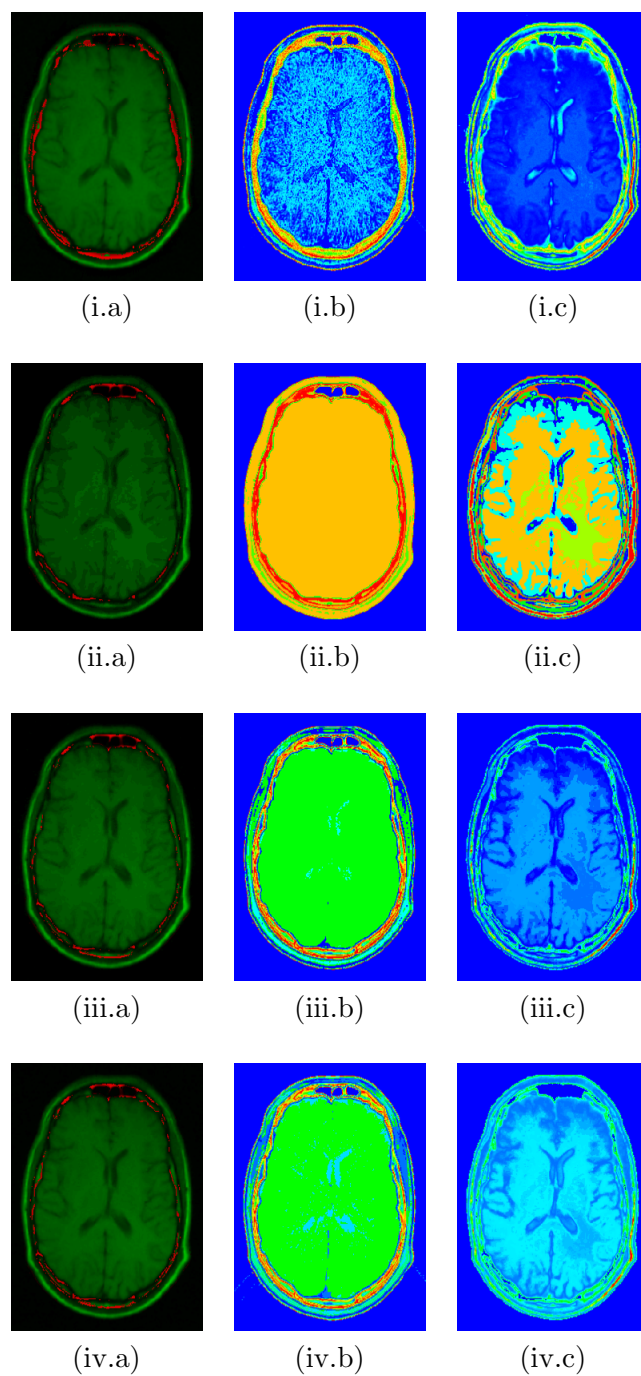


Figure 4.10: (a) Fusion results of the CT and MR model shown in Figure 4.9 using *MMI fusion*. (b - c) Mutual Information maps of the CT and MR-T1 models, respectively. (i) Original resolution models. These models have been quantized using (ii) 16, (iii) 32, (iv) 64 bins, respectively.

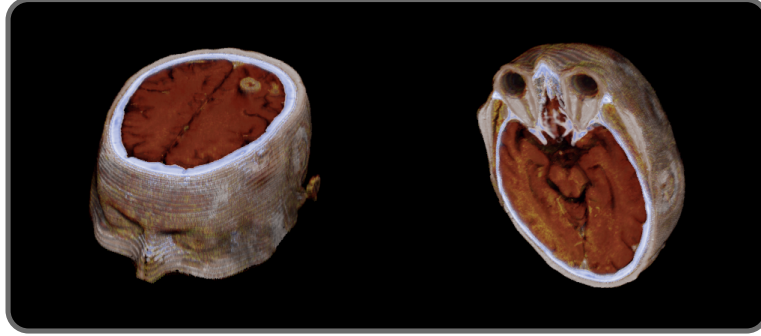


Figure 4.11: 3D fusion visualization using MMI strategy.

where :

- $x, min, max \in X$, and
- $\forall u \in X \quad I(u; Y) \geq I(min; Y)$, and
- $\forall u \in X \quad I(u; Y) \leq I(max; Y)$

The interaction with the fusion works in the same way as the *NMCE fusion* described in Section 4.2.2. If we collapse a percent of the minimum values of $\hat{I}(X; Y)$ or $\hat{I}(X; Y)$, their value is replaced by 0. Otherwise, if we collapse a percent of the maximum values of $\hat{I}(X; Y)$ or $\hat{I}(X; Y)$, their values are replaced by 1. Therefore, if we collapse a percent of the minimum values of $\hat{I}(X; Y)$, these values will rarely be used in the fusion because the maximum value of the pair of values is selected. On the other hand, if we collapse a percent of the maximum values of $\hat{I}(X; Y)$, these collapsed values will be more relevant.

Figure 4.12 shows some fusion result using *NMMI fusion* without any collapse and applying some collapse included in each MI map. Figure 4.13 is an illustration of a 3D multi-modal visualization using this strategy.

Asymmetric fusions

The asymmetric strategies are based on the visualization of the reference model whether its mutual information is less than a threshold or it is greater than a threshold. Otherwise, in both strategies, the other model is visualized.

- *MutualInformationMoreThanThreshold-based (MIMTT) fusion*

$$F_p(X, Y) = \begin{cases} x & \text{if } I(x; Y) > T \\ y & \text{if } I(x; Y) \leq T \end{cases} \quad (4.14)$$

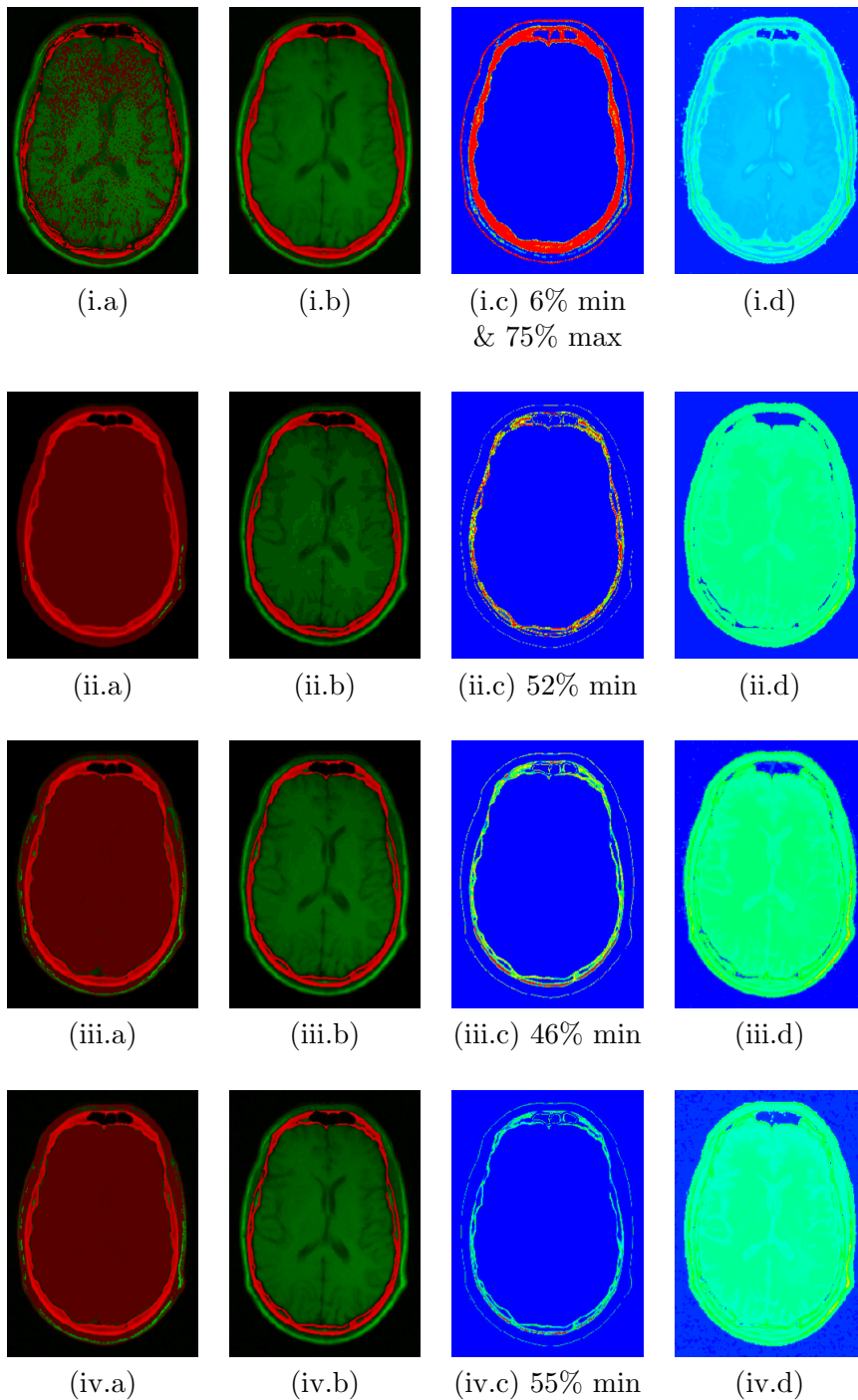


Figure 4.12: (a) Fusion results of the CT and MR model shown in Figure 4.9 using *NMMI fusion* without applying any collapse and (b) applying some collapse. (c - d) Mutual Information collapsed maps of the CT and MR-T1 models respectively. (i) Original resolution models. These models have been quantized using (ii) 16, (iii) 32, (iv) 64 bins, respectively.

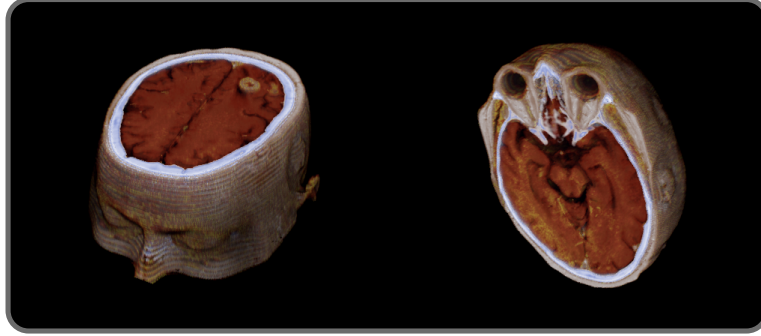


Figure 4.13: 3D fusion visualization using NMMI strategy.

where T is a threshold value defined by the user and X is the reference model.

- *MutualInformationLessThanThreshold-based (MILTT) fusion*

$$F_p(X, Y) = \begin{cases} x & \text{if } I(x; Y) < T \\ y & \text{if } I(x; Y) \geq T \end{cases} \quad (4.15)$$

In Figure 4.14 we can see an illustration of the *MIMTT fusion* and *MILTT fusion* using the CT and the MR-T1 as a reference model. It is an interesting example in order to justify the implementation of both strategies even though they are opposite. Considering a good fusion the result obtained using the *MIMTT fusion* when the reference model is the CT study, if we want to generate a similar fusion using the MR-T1 as the reference model the *MILTT fusion* has to be applied. Therefore, we need both strategies in order to cover all the possibilities, due to the most useful method may vary depending on the reference model used.

Figure 4.15 shows some results of these methods using CT and MR-T1 models quantized using 16, 32 and 64 bins. In Figure 4.8, we show a 3D multi-modal visualization using MIMTT fusion strategy.

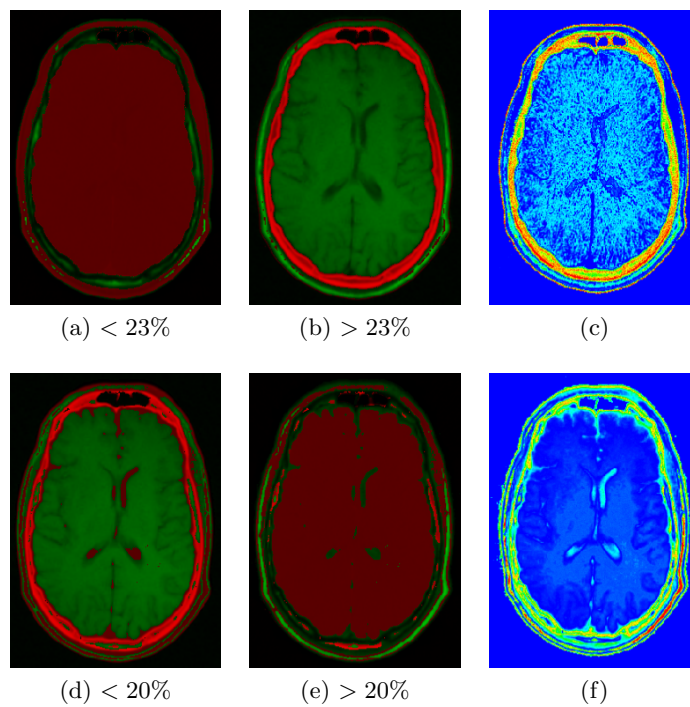


Figure 4.14: Fusion results of the CT and MR model shown in Figure 4.9 using (a) *MILTT fusion* and (b) *MIMTT fusion* where the reference model is the CT study. Fusion result of (b) *MILTT fusion* and (d) *MIMTT fusion* where the reference model is the MR-T1 study. (c, f) MI maps of the CT and MR-T1 models respectively.

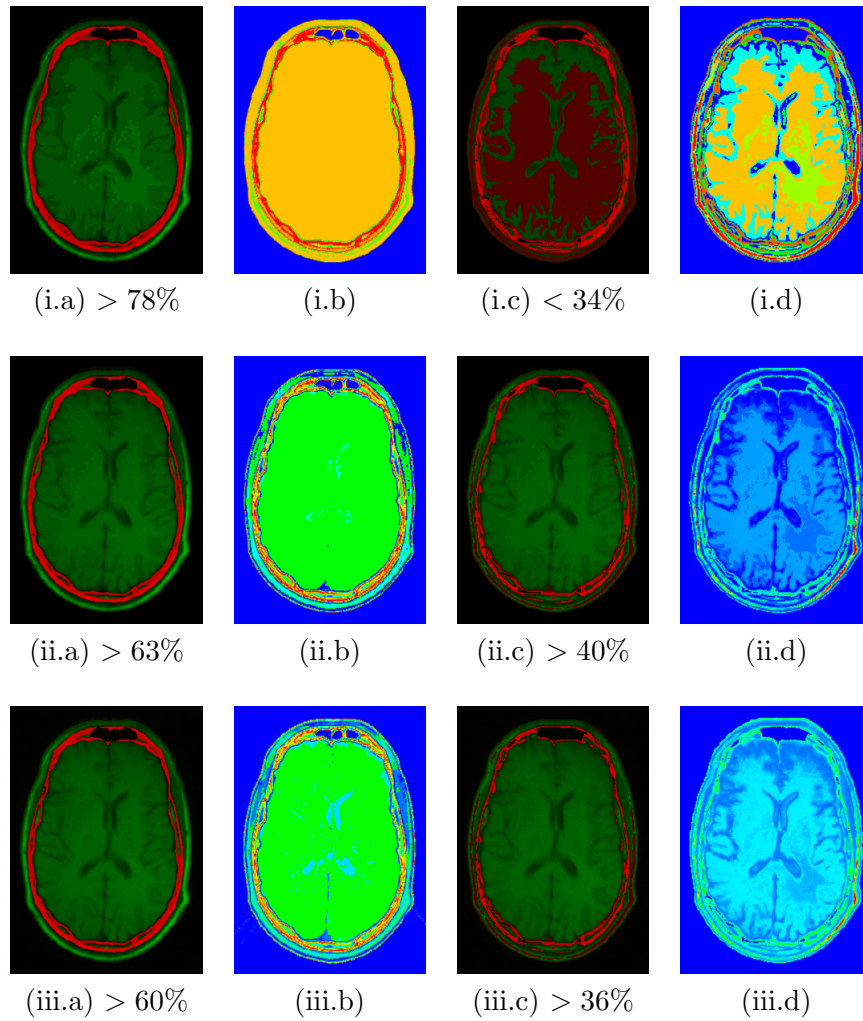


Figure 4.15: (a) Fusion results of the CT and MR model shown in Figure 4.9 using *MILTT fusion* or *MIMTT fusion* using (a) CT and (c) MR-T1 models as a reference. The models are the original resolution models quantized using (i) 16, (ii) 32, (iii) 64 bins, respectively. (b,d) MI maps of the CT and MR-T1 models respectively.

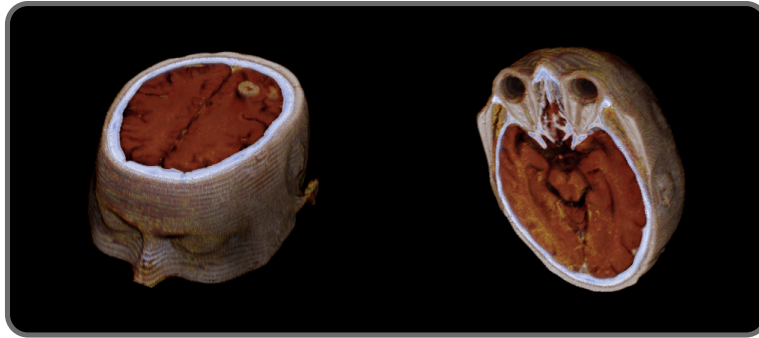


Figure 4.16: 3D fusion visualization using MIMTT strategy.

4.4 Implementation Details

In order to achieve an interactive frame rate, the conditional entropy and the mutual information measures have to be precomputed. The memory needed depends on the intensity range of the input models taking into account that the range can be different for each input model. Therefore, it means that if we increase the intensity range the needed memory will be greater.

For each model, we have to compute the conditional entropy and the mutual information of each intensity value and we store these computed values in an array (one for each measure). Moreover, some auxiliary information is needed in order to compute these measures. We have to compute:

- The histogram of each input model, which is a 1D array where its length depends on the intensity range values.
- The joint histogram, which is a 2D array and the length of each dimension also depends on the intensity range of the models.

However, the auxiliary information needed is common for both measures. Therefore, we can minimize the computation time if we compute both measures at the same time. Moreover, this is only needed during the preprocessing.

Figure 4.17 shows the preprocessing and rendering steps of the process.

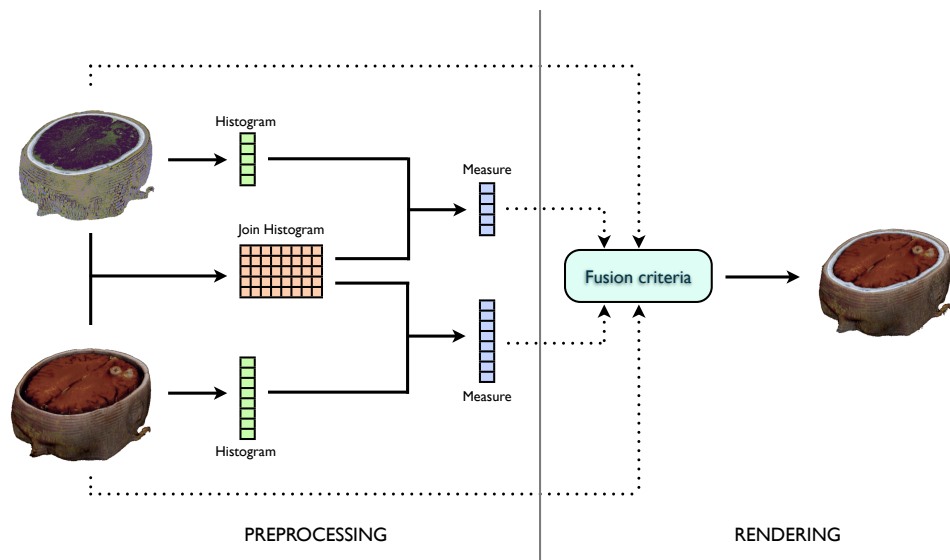


Figure 4.17: Main steps of fusion using registration channel measures.

Chapter 5

Fusion using Higher-Order Entropy Measures

5.1 Introduction

In this chapter, we present an information-theoretic approach for fusion visualization that uses the entropy rate and the erasure entropy as criteria for the fusion process. In contrast to the measures presented in the previous chapter, these measures are computed for each input model without considering the other models. Therefore, only the properties of the model that we are working with are considered.

This chapter is divided in three sections. In Section 2 we introduce the entropy rate and the fusion strategies associated with that measure. Erasure entropy measure and the fusion strategies implemented using it are presented in Section 3. In Section 4, we discuss about the structures and memory needed for each measure to be able to achieve a good interaction in rendering time.

5.2 Entropy Rate Method

5.2.1 Entropy Rate Measure

In this section we use the concept of entropy rate to analyze the contribution of each voxel to the entropy rate of an image. The computation of entropy rate is approximated taking blocks of three intensity values on the three tetrad axes of a cube.

As we have seen in Section 2.4.3 the entropy rate or information density is given by

$$H_X = \lim_{n \rightarrow \infty} H_X(n), \quad (5.1)$$

where

$$\begin{aligned}
H_X(n) &= H(X_1, \dots, X_n) - H(X_1, \dots, X_{n-1}) & (5.2) \\
&= H(X_n | X_{n-1}, \dots, X_1). & (5.3)
\end{aligned}$$

Entropy rate can be seen as the uncertainty associated with a given symbol if all the preceding symbols are known.

Due to the high dimensionality of the intensity alphabet, we calculate the finite version of entropy rate (Equation 5.3) using a quantization of an image and short blocks of n consecutive intensity values. In particular, in our computations we use quantizations of 16, 32, 64 and 256 intensity values and blocks of $n = 3$ consecutive values.

For $n = 3$, Equation 5.3 can be written as

$$\begin{aligned}
H_X(3) &= H(X_3 | X_2, X_1) & (5.4) \\
&= - \sum_{x_1, x_2, x_3} p(x_1, x_2, x_3) \log p(x_3 | x_2, x_1). & (5.5)
\end{aligned}$$

where $-\log p(x_3 | x_2, x_1)$ is interpreted as the information density or unpredictability of x_3 known the rest of the block, i.e, x_1, x_2 . Hence, we can compute the information density of a given voxel a , called color information density, as the average of the information densities of the six blocks associated to this voxel (see Figure 5.1):

$$H_a = -\frac{1}{6} \sum_{k=1}^6 \log p(x_3 | x_2^k, x_1^k) = -\frac{1}{6} \sum_{k=1}^6 \log \frac{p(x_1^k, x_2^k, x_3)}{p(x_1^k, x_2^k)}, \quad (5.6)$$

where k stands for the axis on which the block is taken. In Figure 5.1, we can observe that the voxel a , which intensity is x_3 , is taken as the center of a cube and the six blocks around it are taken on the 3 tetrad axes of the cube.

Figure 5.2 illustrates the entropy rate (ER) maps of two different images (CT and MR) which have been previously registered. Each entropy rate map shows the H_a (Figure 5.2b) and H_b (Figure 5.2d), respectively, for each voxel. The maps are colorized using the color scale described in Section 3.5.2 where the blue color represents the minimum entropy rate value and the red color the maximum entropy rate value.

High values of H_a correspond to the most informative or unpredictable voxels. Frequently, these voxels correspond to edges or singularities inside the most homogeneous regions.

5.2.2 Fusion Strategies

Using the classification of the fusion methods presented in Section 3.4, we developed two symmetric and two asymmetric fusion methods where the

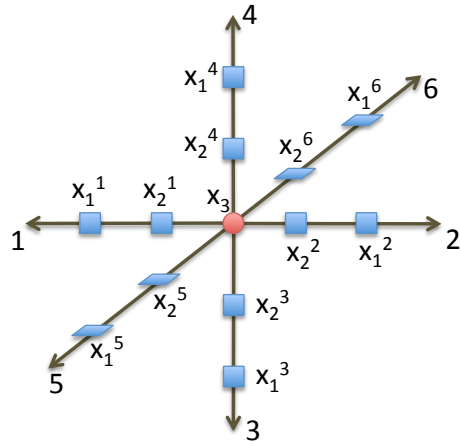


Figure 5.1: Representation of the six blocks associated to a voxel.

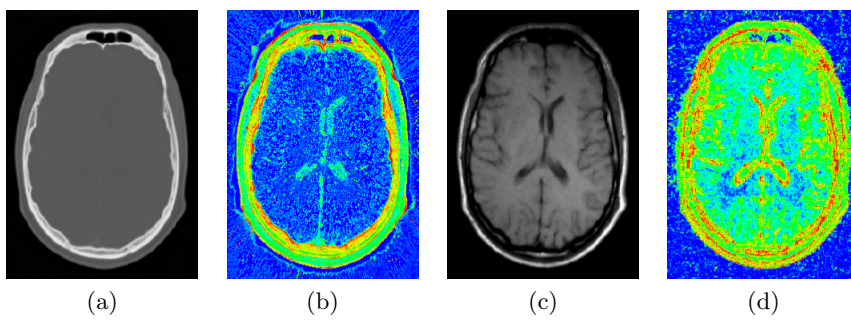


Figure 5.2: Original (a) CT and (c) MR models and (b) and (d) are their corresponding ER colored maps.

entropy rate is the measure used in order to determine the contribution of the input models in the fusion visualization.

In this section we describe the fusion methods implemented using this measure and we illustrate the results in some figures. In all images we use the CT and MR-T1 studies of *training_001* patient and the CT study is visualized using a red scale and the MR-T1 is visualized using a green scale.

Symmetric fusions

MaxEntropyRate (MER) fusion is the first method implemented.

$$F_p(X, Y) = \begin{cases} x & \text{if } H_a > H_b \\ y & \text{if } H_a \leq H_b \end{cases} \quad (5.7)$$

where X and Y are the input models and $F(X, Y)$ is the fused model.

For each pair of voxels, we visualize the voxel of the input model which has the maximum entropy rate. It produces useful and nice results but it is non flexible and it does not allow us to interact with the fusion result. Figure 5.3 shows some results of this method and Figure 5.4 a 3D multi-modal visualization.

NormalizedMaxEntropyRate (NMER) fusion is the second symmetric fusion strategy developed. The goal of this method is to allow the user to be able to interact with the fusion:

$$F_p(X, Y) = \begin{cases} x & \text{if } \hat{H}_a > \hat{H}_b \\ y & \text{if } \hat{H}_a \leq \hat{H}_b \end{cases} \quad (5.8)$$

where \hat{H} represents the normalized entropy rate with values between 0 and 1:

$$\hat{H}_a = \frac{H_a - H_{min}}{H_{max} - H_{min}} \quad (5.9)$$

where :

- $a, min, max \in X$, and
- $\forall i \in X \quad H_i \geq H_{min}$, and
- $\forall i \in X \quad H_i \leq H_{max}$

In this case, the interaction with the fusion works in the same way as the *NMCE* and *NMMI fusion* strategies described in Chapter 4. If we collapse a percent of the minimum values of \hat{H}_a or \hat{H}_b , their values are replaced

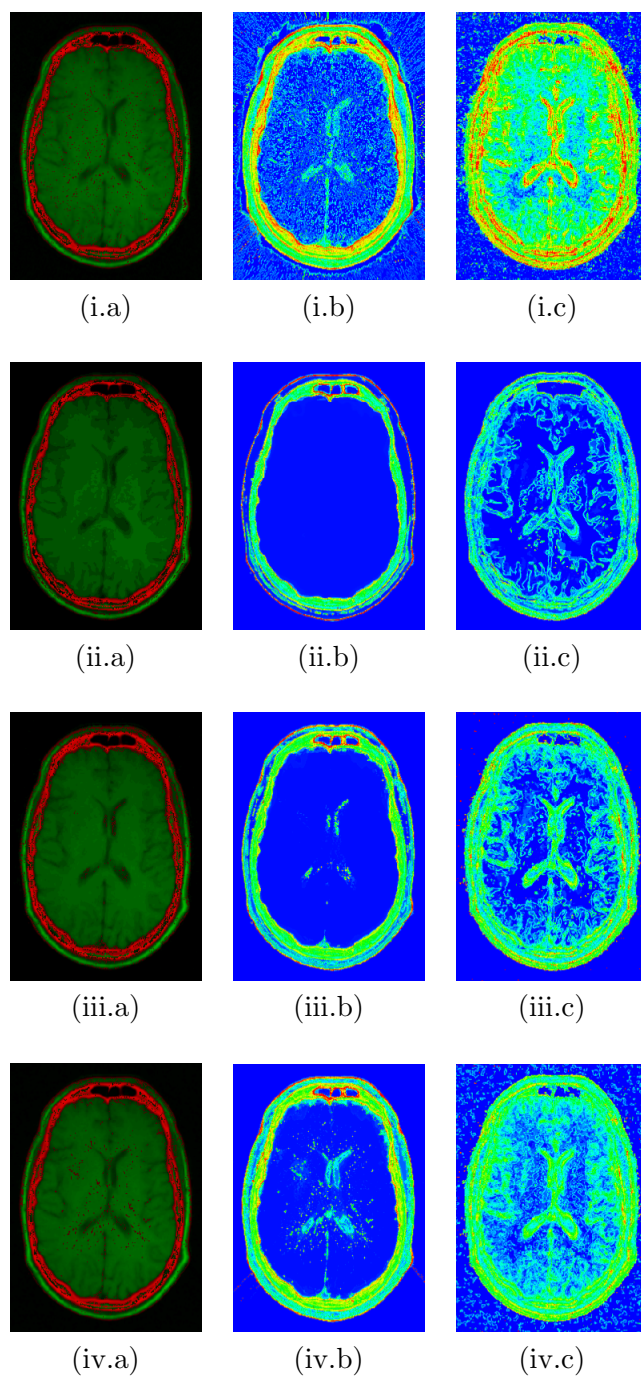


Figure 5.3: (a) Fusion results of the CT and MR model shown in Figure 5.2 using *MER fusion*. (b - c) Entropy Rate maps of the CT and MR-T1 models respectively. These models have been quantized using (i) 256, (ii) 16, (iii) 32, (iv) 64 bins, respectively.

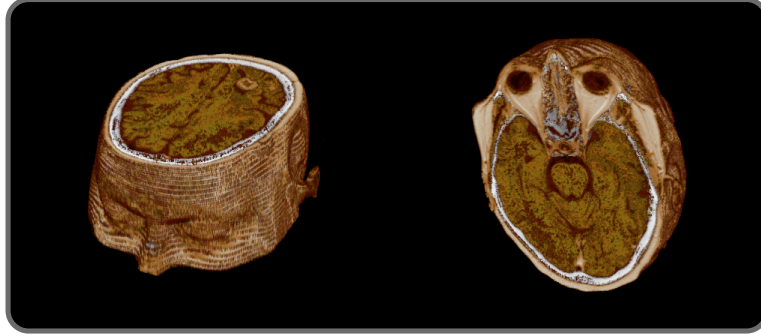


Figure 5.4: 3D fusion visualization using MER strategy.

by 0. Otherwise, if we collapse a percent of the maximum values of \hat{H}_a or \hat{H}_b , their values are replaced by 1. Therefore, if we collapse a percent of the minimum values of \hat{H}_a , these values will rarely be used in the fusion because the maximum value of the pair of values is selected. Otherwise, if we collapse a percent of the maximum values of \hat{H}_b , these collapsed values will be more relevant.

Figure 5.5 shows some fusion results using *NMER fusion* without any collapse and applying a collapse included in each ER map. Figure 5.6 shows a 3D multi-modal visualization using this strategy.

Asymmetric fusions

The asymmetric strategies are based on the visualization of the reference model whether its mutual information is less than a threshold or it is greater than a threshold. Otherwise, in both strategies, the other model is visualized.

- *EntropyRateMoreThanThreshold-based (ERMTT) fusion*

$$F_p(X, Y) = \begin{cases} x & \text{if } H_a > T \\ y & \text{if } H_a \leq T \end{cases} \quad (5.10)$$

where T is a threshold value defined by the user and X is the reference model.

- *EntropyRateLessThanThreshold-based (ERLTT) fusion*

$$F_p(X, Y) = \begin{cases} x & \text{if } H_a < T \\ y & \text{if } H_a \geq T \end{cases} \quad (5.11)$$

In Figure 5.7 we can see an illustration of the *MIMTT fusion* and *MILTT fusion* using the CT and the MR-T1 as a reference model. In order to show

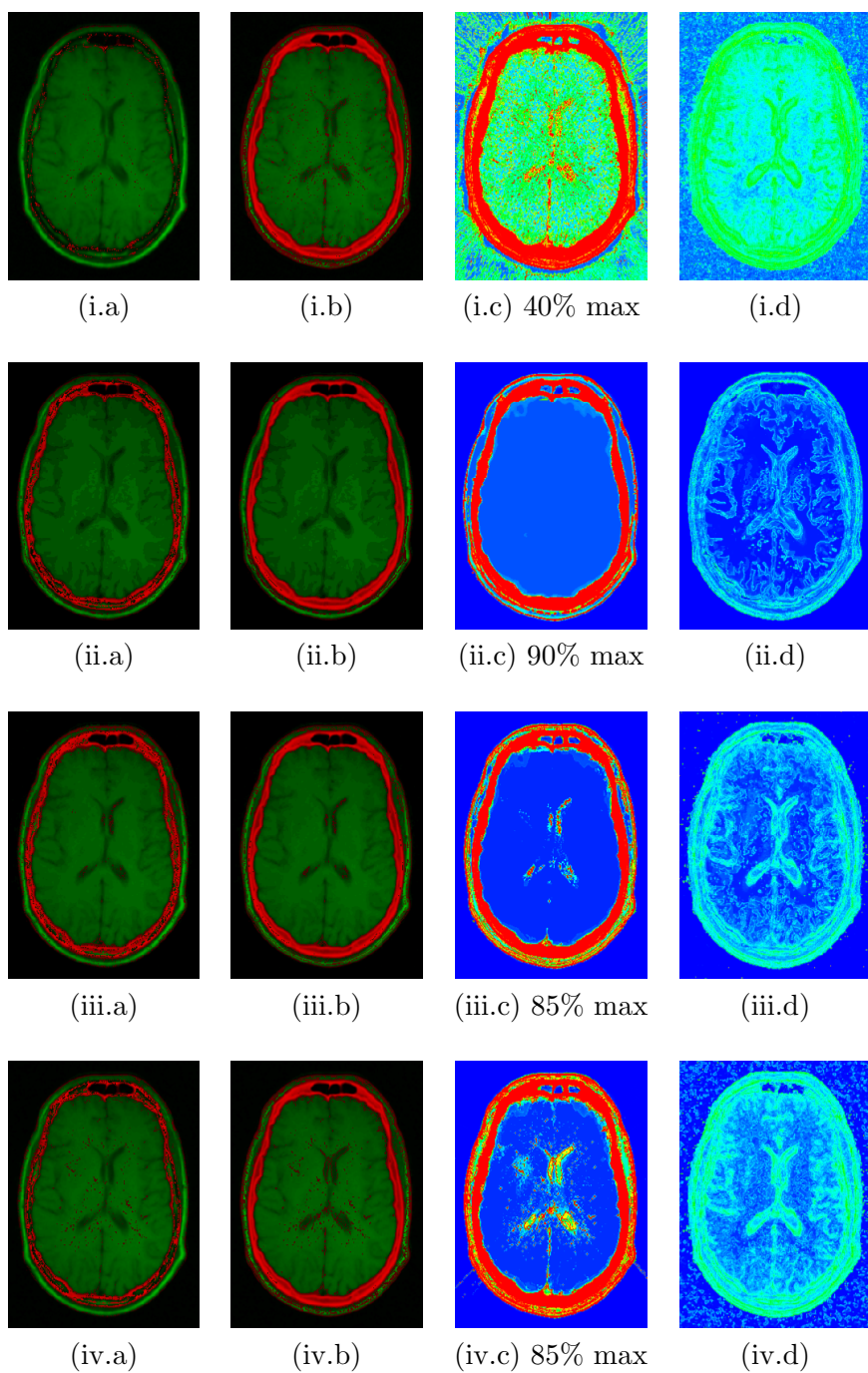


Figure 5.5: (a) Fusion results of the CT and MR model shown in Figure 5.2 using *NMER fusion* without applying any collapse and (b) applying some collapse. (c - d) Entropy Rate collapsed maps of the CT and MR-T1 models respectively. These models have been quantized using (i) 256, (ii) 16, (iii) 32, (iv) 64 bins, respectively.

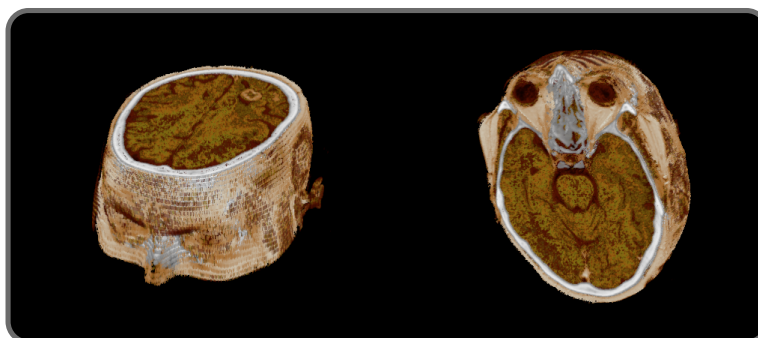


Figure 5.6: 3D fusion visualization using NMER strategy.

the collapse applied, we include the percent in the colored maps where *min* or *max* means that low or high values have been collapsed. Some results using CT and MR-T1 models quantized using 16, 32 and 64 bins are presented in Figure 5.8. Figure 5.9 shows a 3D multi-modal visualization using ERMTT fusion.

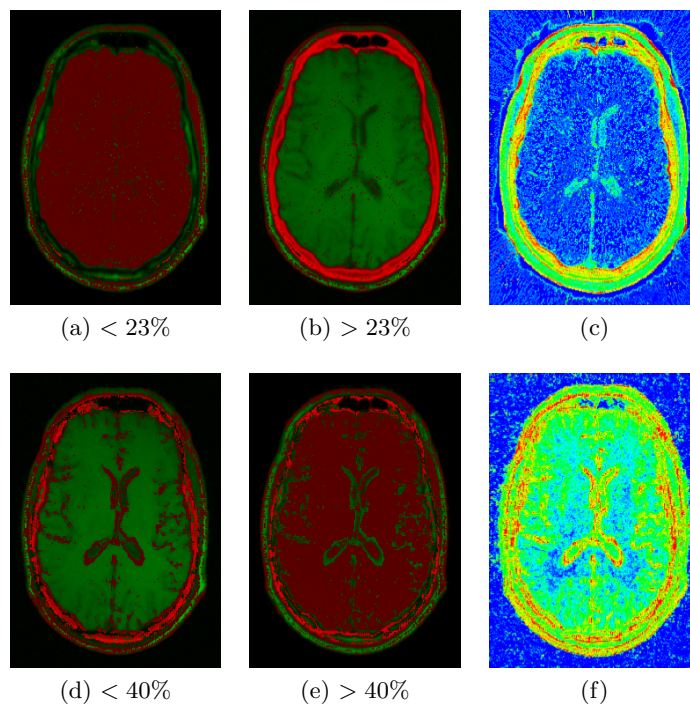


Figure 5.7: Fusion results of the CT and MR model shown in Figure 5.2 using (a) *ERLTT fusion* and (b) *ERMTT fusion* where the reference model is the CT study. Fusion result of (b) *ERLTT fusion* and (d) *ERMTT fusion* where the reference model is the MR-T1 study. (c, f) ER maps of the CT and MR-T1 models respectively.

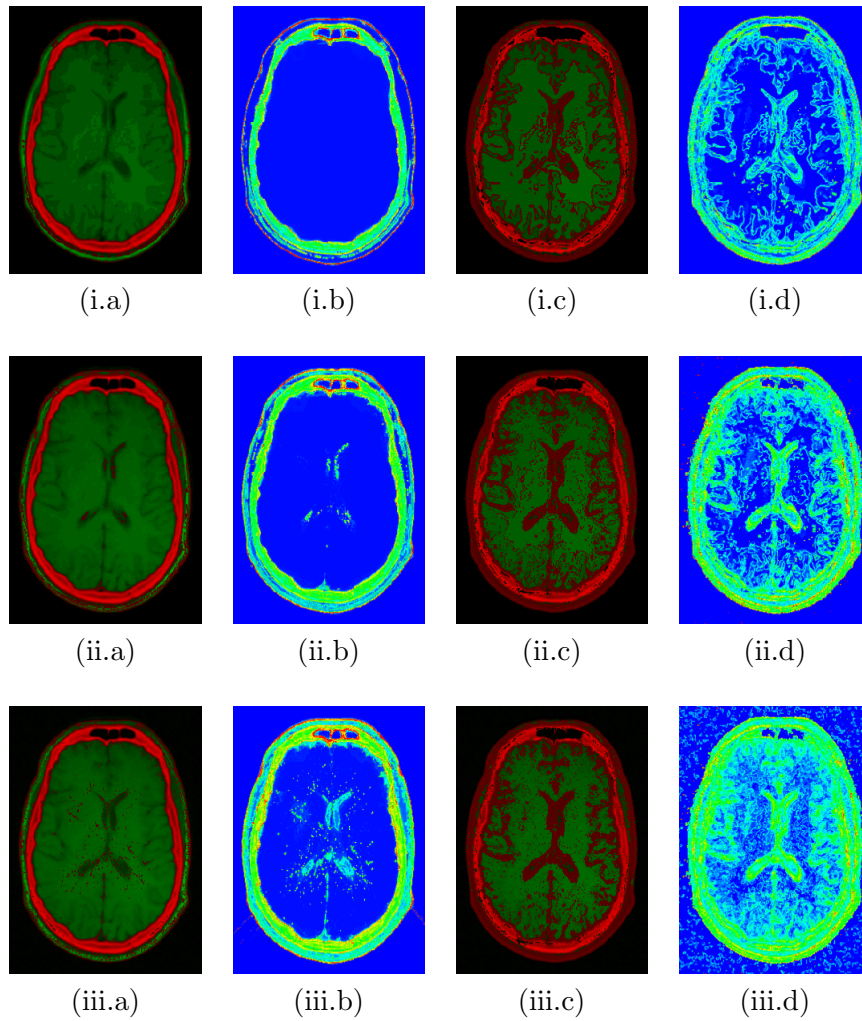


Figure 5.8: Fusion results of the CT and MR model shown in Figure 5.2 using *ERLTT fusion* or *ERMTT fusion* using (a) CT and (c) MR-T1 models as a reference. The models are the original resolution models quantized using (i) 16, (ii) 32, (iii) 64 bins, respectively. (b,d) ER maps of the CT and MR-T1 models respectively.

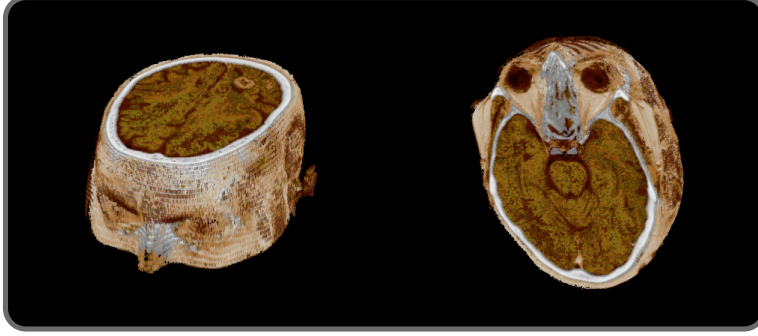


Figure 5.9: 3D fusion visualization using ERMTT strategy.

5.3 Erasure Entropy Method

5.3.1 Erasure Entropy Measure

As we have seen in Section 2.4.4, the erasure entropy of a collection of discrete random variables $\{X_1, \dots, X_n\}$ is given by

$$H^-(X_1, \dots, X_n) = \sum_{i=1}^n H(X_i | X_{\setminus i}), \quad (5.12)$$

where

$$X_{\setminus i} = \{X_j, j = 1, \dots, n, j \neq i\}. \quad (5.13)$$

Similarly to the entropy rate case, in our computation we use the finite version of the erasure entropy (Equation 5.12) with quantizations of 16, 32, 64 and 256 intensity values and blocks of $n = 3$ consecutive values.

For $n = 3$, Equation 5.12 can be written as

$$\begin{aligned} H^-(X_1, X_2, X_3) &= \sum_{i=1}^3 H(X_i | X_{\setminus i}) \\ &= - \sum_{x_1, x_2, x_3} p(x_1, x_2, x_3) \log p(x_3 | x_2, x_1) \\ &\quad - \sum_{x_1, x_2, x_3} p(x_1, x_2, x_3) \log p(x_2 | x_3, x_1) \\ &\quad - \sum_{x_1, x_2, x_3} p(x_1, x_2, x_3) \log p(x_1 | x_3, x_2). \end{aligned} \quad (5.14)$$

where $-\log p(x_3 | x_2, x_1)$ is interpreted as the erasure entropy associated with block $\{x_1, x_2, x_3\}$. Hence, we can compute the erasure entropy of a given voxel a , called color erasure entropy, as the average of the erasure entropy of the three blocks associated to this voxel (see Figure 5.10):

$$H_a^- = -\frac{1}{3} \sum_{k=1}^3 \log \frac{p(x_1^k, x_2, x_3^k)^3}{p(x_1^k, x_2) p(x_1^k, x_3^k) p(x_2, x_3^k)}, \quad (5.16)$$

where k stands for the axis on which the three blocks are taken. In Figure 5.10 we can observe that the voxel a which intensity is x_2 is taken as the center of a cube and the three blocks around it are taken on the 3 tetrad axes of the cube. Observe how the erasure entropy (Figure 5.10) considers less information around the voxel a than the entropy rate (Figure 5.1). While the entropy rate takes two neighbor values of a in the six directions around it, the erasure entropy is computed with only one neighbor value.

Figure 5.11 illustrates the erasure entropy (EE) maps of two different images (CT and MR) which have been previously registered. Each erasure entropy map shows the H_a^- (Figure 5.11b) and H_b^- (Figure 5.11d), respectively, for each voxel. The maps are colorized using the color scale described in Section 3.5.2 where the blue color represents the minimum erasure entropy value and the red color the maximum erasure entropy value.

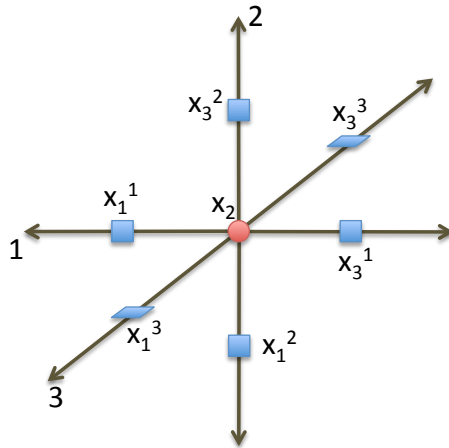


Figure 5.10: Representation of the six blocks associated to a voxel.

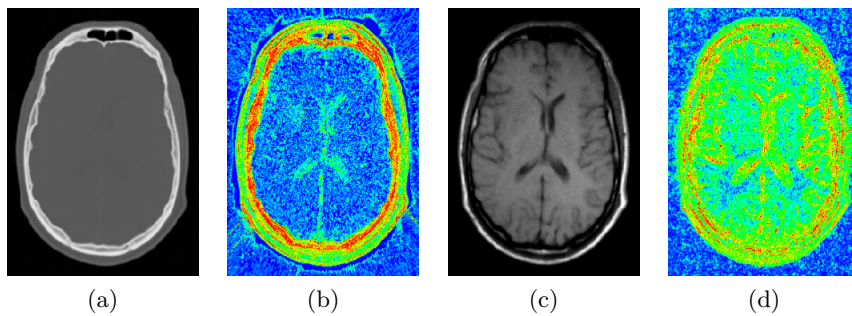


Figure 5.11: Original (a) CT and (c) MR models and (b) and (d) are their corresponding EE colored maps.

5.3.2 Fusion Strategies

Using the classification of the fusion methods presented in Section 3.4, we developed two symmetric and two asymmetric fusion methods where the Entropy Rate is the measure used in order to determine the contribution of the input models in the fusion visualization.

In this section we describe the fusion methods implemented using this measure and we illustrate the results in some figures. In all images we use the CT and MR-T1 studies of *training_001* patient and the CT study is visualized using a red scale and the MR-T1 is visualized using a green scale.

Symmetric fusions

The symmetric fusion strategies implemented are the following:

- *MaxErasureEntropy (MEE) fusion.*

$$F_p(X, Y) = \begin{cases} x & \text{if } H_a^- > H_b^- \\ y & \text{if } H_a^- \leq H_b^- \end{cases} \quad (5.17)$$

where X and Y are the input model and $F(X, Y)$ is the fused model.

For each pair of voxels, we visualize the voxel of the input model which has the maximum Erasure Entropy. It is an automatic fusion strategy which means that any parameter has to be given by the user. Figure 5.12 shows some results of this method and Figure 5.13 is a 3D multi-modal visualization.

- *NormalizedMaxErasureEntropy (NMEE) fusion* is the second symmetric fusion strategy developed. The goal of this method is to allow the user to be able to interact with the fusion:

$$F_p(X, Y) = \begin{cases} x & \text{if } \hat{H}_a^- > \hat{H}_b^- \\ y & \text{if } \hat{H}_a^- \leq \hat{H}_b^- \end{cases} \quad (5.18)$$

where \hat{H}^- represents the normalized erasure entropy with values between 0 and 1:

$$\hat{H}_a^- = \frac{H_a^- - H_{min}^-}{H_{max}^- - H_{min}^-} \quad (5.19)$$

where :

- $x, min, max \in X$, and
- $\forall i \in X \quad H_i^- \geq H_{min}^-$, and

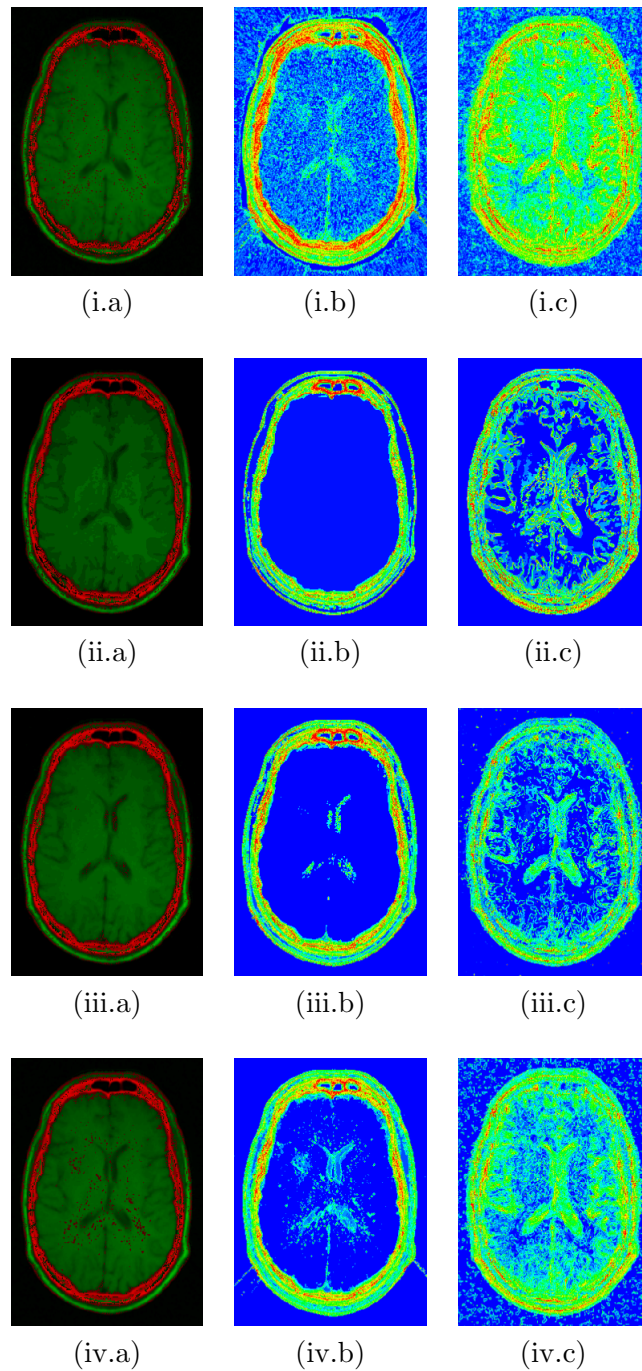


Figure 5.12: (a) Fusion results of the CT and MR model shown in Figure 5.11 using *MEE fusion*. (b - c) Erasure Entropy maps of the CT and MR-T1 models respectively. These models have been quantized using (i) 256, (ii) 16, (iii) 32, (iv) 64 bins, respectively.

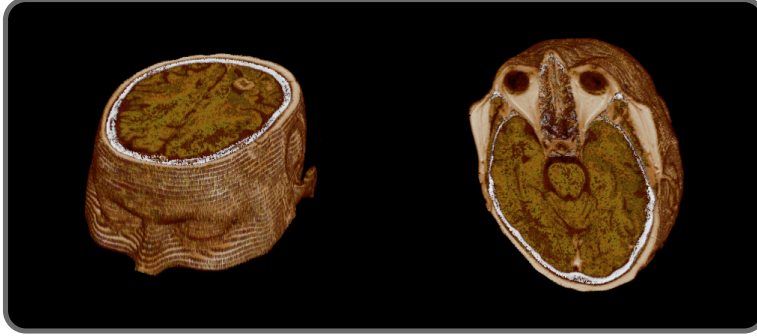


Figure 5.13: 3D fusion visualization using MEE strategy.

$$- \forall i \in X \quad H_i^- \leq H_{max}^-$$

In this method, the interaction with the fusion works in the same way as the *NMER* presented in Section 5.2. If we collapse a percent of the minimum values of \hat{H}_a^- or \hat{H}_b^- , their values are replaced by 0. Otherwise, if we collapse a percent of the maximum values of \hat{H}_a^- or \hat{H}_b^- , their values are replaced by 1. Therefore, if we collapse a percent of the minimum values of \hat{H}_a^- , these values will rarely be used in the fusion because the maximum value of the pair of values is selected. Otherwise, if we collapse a percent of the maximum values of \hat{H}_a^- , these collapsed values will be more relevant.

Figure 5.14 shows some fusion results using *NMEE fusion* without any collapse and applying a collapse included in each EE map. Figure 5.15 shows a 3D multi-modal visualization using this strategy.

Asymmetric fusions

The asymmetric strategies are based on the visualization of the reference model whether its erasure entropy is less than a threshold or it is greater than a threshold. Otherwise, in both strategies, the other model is visualized.

- *ErasureEntropyMoreThanThreshold-based (EEMTT) fusion*

$$F_p(X, Y) = \begin{cases} x & \text{if } H_a^- > T \\ y & \text{if } H_a^- \leq T \end{cases} \quad (5.20)$$

where T is a threshold value defined by the user and X is the reference model.

- *ErasureEntropyLessThanThreshold-based (EELTT) fusion*

$$F_p(X, Y) = \begin{cases} x & \text{if } H_a^- < T \\ y & \text{if } H_a^- \geq T \end{cases} \quad (5.21)$$

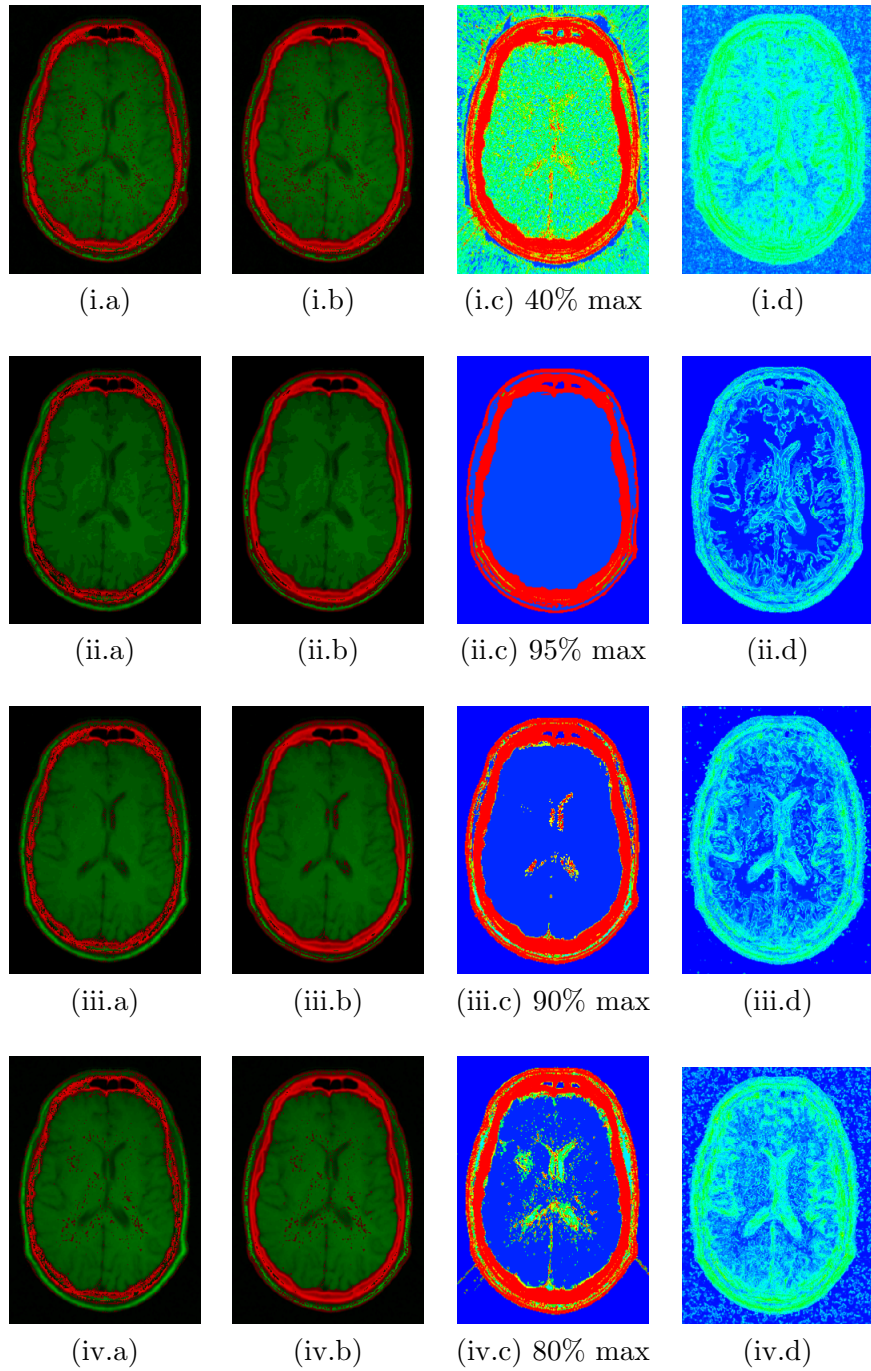


Figure 5.14: (a) Fusion results of the CT and MR model shown in Figure 5.11 using *NMEE fusion* without applying any collapse and (b) applying a collapse. (c - d) Erasure Entropy collapsed maps of the CT and MR-T1 models respectively. These models have been quantized using (i) 256, (ii) 16, (iii) 32, (iv) 64 bins, respectively.

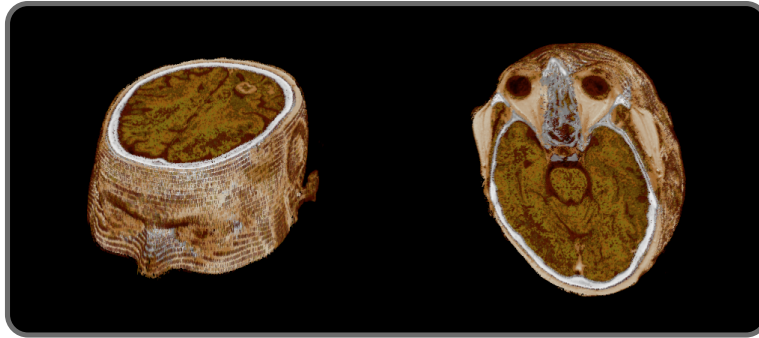


Figure 5.15: 3D fusion visualization using NMEE strategy.

In Figure 5.16 we can see an illustration of the *EEMTT fusion* and *EELTT fusion* using the CT and the MR-T1 as a reference model. Some results using CT and MR-T1 models quantized using 16, 32 and 64 bins are presented in Figure 5.17. A 3D multi-modal visualization using EEMTT fusion is shown in Figure 5.18.

5.4 Implementation Details

As you can see in Chapter 6, where we discuss the memory requirements of the fusion approaches presented, the approaches presented in this chapter need more memory than the approaches presented in Chapter 4.

The higher-order entropy measures are computed for each model without taking into account the other input models and they have to be stored in a 3D array because they have to have the input model measurements. Moreover, in order to compute them, we need some auxiliary information which depends on the intensity range of the input models and it varies depending on the measure.

On one hand, the entropy rate needs:

- A 2D histogram which depends on the intensity range of the model.
- A 3D histogram which also depends on the intensity range of the model.

In Figure 5.19, the workflow can be seen to compute the entropy rate. On the other hand, the erasure entropy needs:

- Three 2D histograms which depend on the intensity range of the model.
- 3D histogram which also depends on the intensity range of the model.

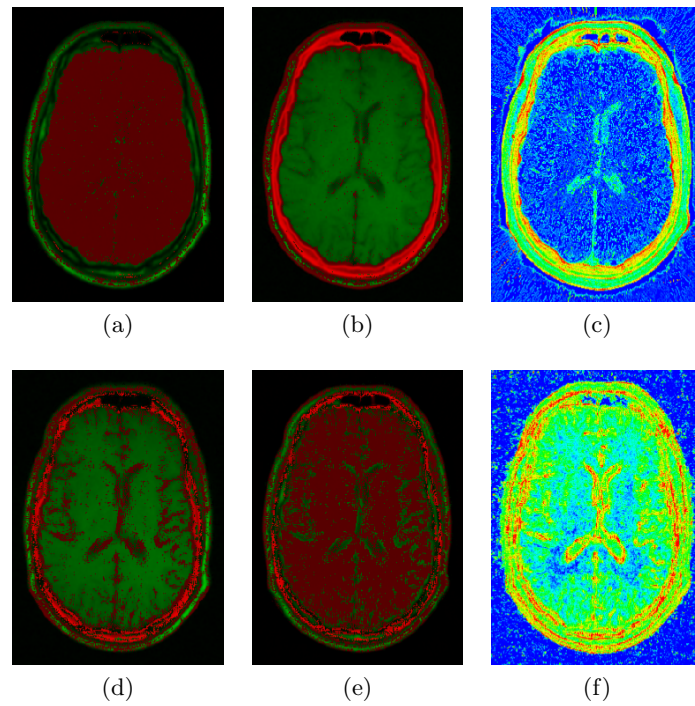


Figure 5.16: Fusion results of the CT and MR model shown in Figure 5.11 using (a) *EELTT fusion* and (b) *EEMTT fusion* where the reference model is the CT study. Fusion result of (b) *EELTT fusion* and (d) *EEMTT fusion* where the reference model is the MR-T1 study. (c, f) EE maps of the CT and MR-T1 models respectively.

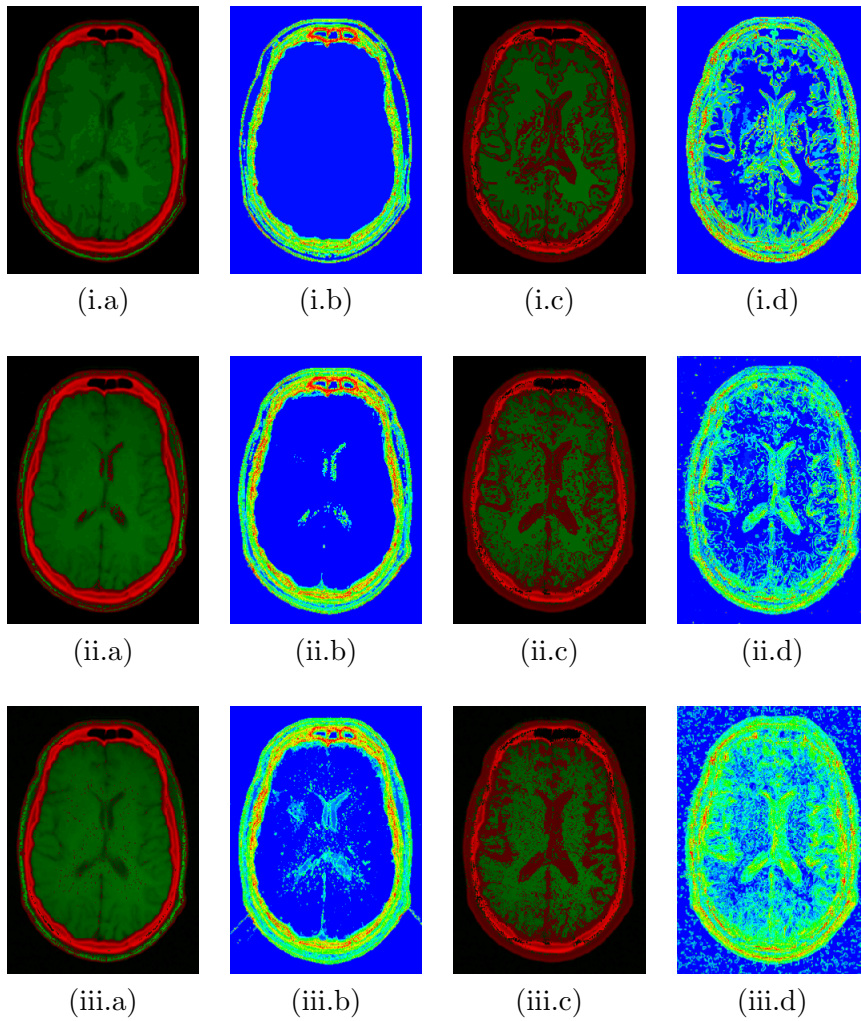


Figure 5.17: Fusion results of the CT and MR model shown in Figure 5.11 using *EELTT fusion* or *EEMTT fusion* using (a) CT and (c) MR-T1 models as a reference. The models are the original models quantized using (i) 16, (ii) 32, (iii) 64 bins, respectively. (b,d) EE maps of the CT and MR-T1 models respectively.

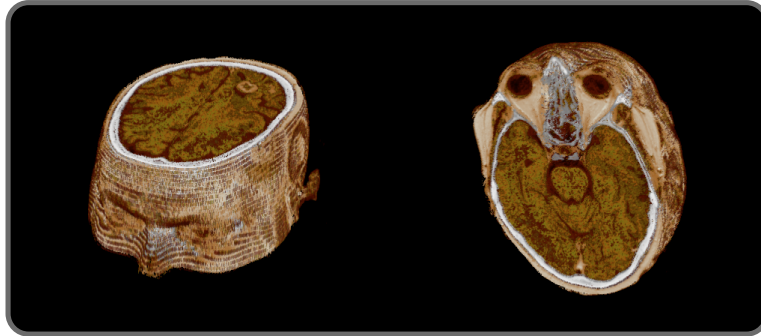


Figure 5.18: 3D fusion visualization using EEMTT strategy.

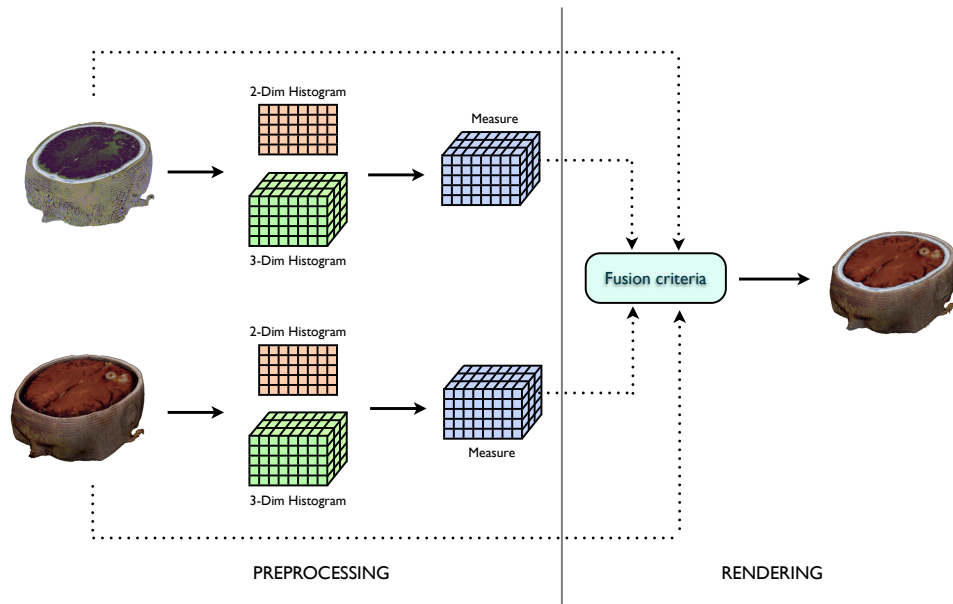


Figure 5.19: Main steps of fusion using entropy rate measure.

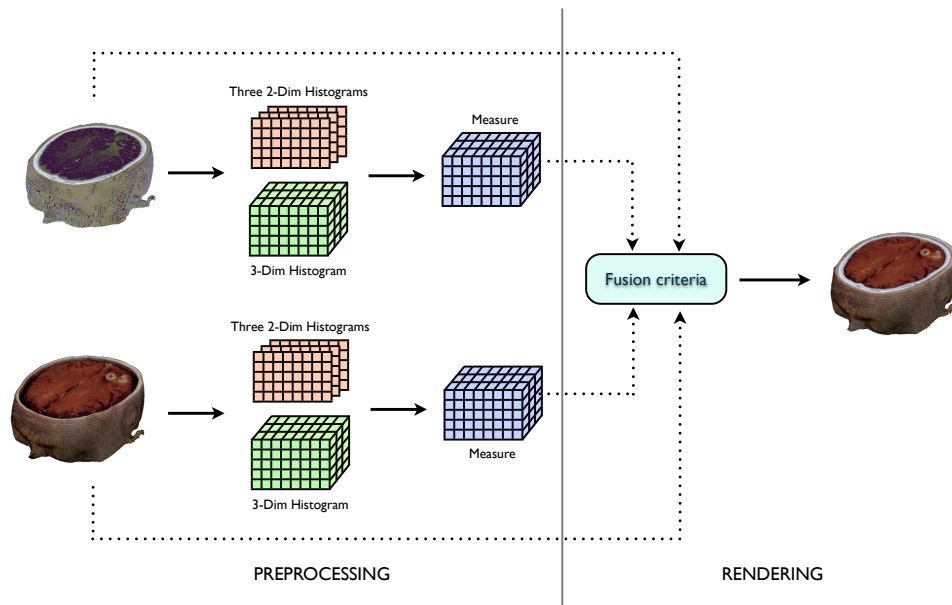


Figure 5.20: Main steps of fusion using erasure entropy measure.

Chapter 6

Testing of Fusion Methods

In this chapter we evaluate and discuss the performance of the proposed fusion strategies. We consider the fusion strategies based on conditional entropy, mutual information, entropy rate and erasure entropy presented in previous chapters. To carry out the experiments we consider the real and synthetic data sets described in Section 1. The methods were applied to these models and a group of experts (an anatomist, a neuroradiologist and a radiologic technician) evaluated them considering the parameters described in Section 2. The results of the experiment are also presented in Section 2. Moreover, a study about the requirements of the fusion methods proposed is described in Section 3.

6.1 Testing Data Sets

To evaluate the performance of the proposed fusion methods two different sets of models were used. The first one, was obtained from the Retrospective Image Registration Evaluation Project database. This project was designed to compare retrospective CT-MR and PET-MR registration techniques and has been used by a number of groups. This database is also known as the “Vanderbilt Database”. From this database we select for our tests the models represented in Table 6.1.

The second group was composed of real-data sets obtained from different patients from the Hospital Josep Trueta of Girona. For all images, an informed consent was obtained in accordance with the guidelines of the institutional review board for human subject studies. For each patient, a CT and a MR was acquired. First of all, we analyzed each one of the patients to determine if they were suitable for our study. Since the resolution of the CT and MR models in some of the cases differs considerably we perform a selection process to determine which models can be used for the test. During the selection process, the quality of the CT-MR registration is one of the parameters considered. Note that in the case of the Vanderbilt data this

process is not required since the quality of the registration can be ensured. In Table 6.1, we illustrate the real models used for the tests describing their features.


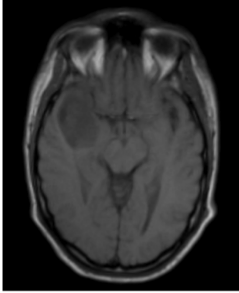
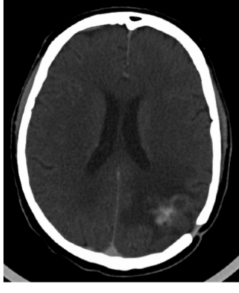
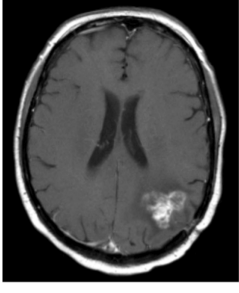
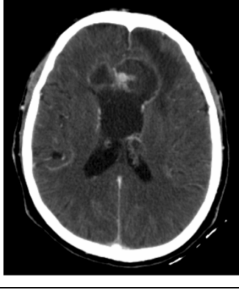
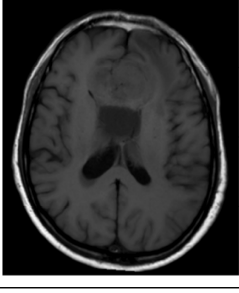
<i>training_001</i>	
	
Dimensions: 512 x 512 x 29 Thickness: 4.0 mm	Dimensions: 256 x 256 x 26 Thickness: 4.0 mm
<i>patient_109</i>	
	
Dimensions: 512 x 512 x 40 Thickness: 3.0 mm	Dimensions: 256 x 256 x 52 Thickness: 3.0 mm
<i>Real patient</i>	
	
Dimensions: 512 x 512 x 28 Thickness: 5.0 mm	Dimensions: 288 x 288 x 20 Thickness: 6.0 mm

Table 6.1: Patients used in the test

6.2 Quality of the Fusion

6.2.1 Evaluation Criteria

Determining the quality of a fusion strategy is a difficult task. The knowledge of an expert is required to determine in each situation the information of the input models that have to be represented in the fusion. These information can vary according to the image modality and the pathology. To overcome these limitations we contacted with a set of experts of different units of an image diagnosis service. The first one is an expert in anatomy from the Hospital Clinic from Barcelona, the second one is a neuroradiologist from the Hospital Josep Trueta from Girona and the last one is a radiology technician from the same hospital.

First of all, we asked to the anatomist to evaluate each one of the image modalities (CT and MR) and determine for each one which parts he considers that have to be represented in the fused model. His selection was clear and was the following. From the CT he wants to see the bone structure and the paranasal sinuses. These parts can be considered crucial in a CT-MR fusion since they define the boundaries of the brain parenchyma (see Figure 6.1). From the MR model he is interested in the brain parenchyma which contains the white and grey matter, the subcutaneous fat, the ventricles and the extension of the injury if it exists.



Figure 6.1: Basic parts of the brain.

Taking into account these considerations we define the **form** of Table 6.2 where each column represents the type of material and also the data model from which we want to obtain it to be represented in the fused model (i.e. parenchyma from MR, ventricles from MR, soft tissues from MR, injury from MR, bone from CT and sinuses from CT). Each row corresponds to one of the fusion strategies. Using our framework, for each one of the testing data sets we apply the different fusion strategies and the expert evaluates

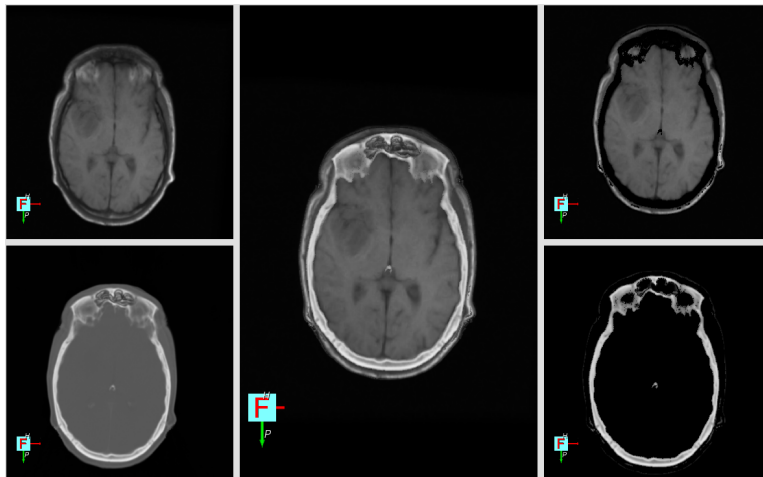
the results ranking as perfect match “++”, good match “+”, and no match “-”.

As we have seen in previous chapters, we proposed automatic and semi-automatic fusion methods. Using the semi-automatic methods the medical experts can modify the fusion result in order to improve it or to vary the contribution of each model in the final result. Basically, the semi-automatic methods are the asymmetric fusion methods (see Section 3.4). However, in some symmetric fusion methods some modifications can be done. To guarantee that all the evaluators analyze the same images, for each method we define the *best configuration* and we do not allow to modify it during the tests. These configurations were based on the considerations that the anatomist previously defined.

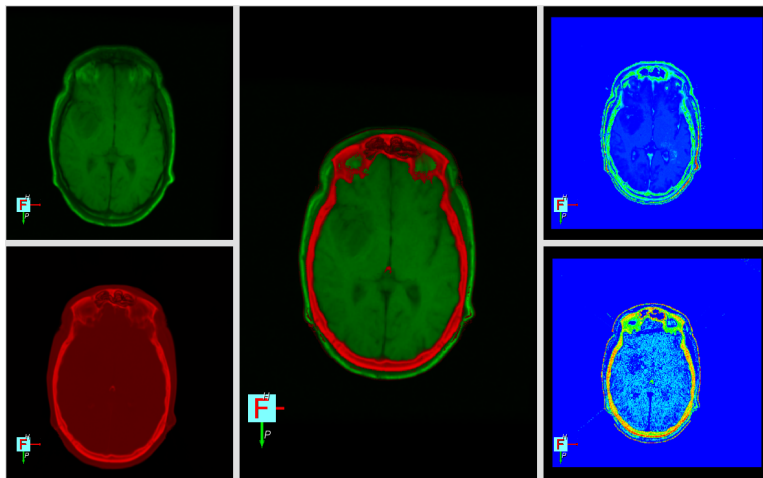
Fusion Strategy	parenchyma (MR)	ventricles (MR)	soft tissues (MR)	injury (MR)	bone (CT)	sinuses (CT)
MCE						
NMCE						
NMCE collap.						
MMI						
NMMI						
NMMI collap.						
CELTT (CT)						
MIMTT (CT)						
CELTT (MR)						
MILTT (MR)						
MER						
NMER						
NMER collap.						
MEE						
NMEE						
NMEE collap.						
EEMTT (CT)						
ERMTT (CT)						

Table 6.2: Test table used in order to evaluate the fusion approaches developed. In the table, *collap.* means that fusion result had been manipulated and *(CT)* or *(MR)* means the CT or MR is the reference model.

To reduce the complexity of the evaluation process, we show to the evaluator the grey scale images and also the colour maps generated using the green and red scales just to highlight the contribution of each model (see Figure 6.2). The use of the green to red scale visualization allows us to use the helper viewer to visualize other information. However, the evaluators prefer the gray scale mode.



(a)



(b)

Figure 6.2: There are two ways to evaluate the fusion. In (a) we can see the contribution of each model in the helper viewers and in (b) we apply a green and red scales for each model in order to know exactly which is the origin of the information of each pixel.

6.2.2 Results

In this section, we present the obtained results. Each evaluator analyzed 3 models and 18 fusion techniques which leads to 3 forms. For the sake of simplicity, we grouped the results of the evaluators in models. In the following forms we report the obtained results.

Fusion Strategy	parenchyma (MR)	ventricles (MR)	soft tissues (MR)	injury (MR)	bone (CT)	sinuses (CT)
MCE	+	-	-	+	+	+
NMCE	+	-	++	++	+	+
NMCE collap.	++	+	+	++	-	+
MMI	++	++	+	++	-	-
NMMI	-	-	+	-	-	-
NMMI collap.	++	++	+	++	++	+
CELTT (CT)	++	++	-	++	+	++
MIMTT (CT)	++	++	+	++	++	+
CELTT (MR)	+	-	-	+	-	+
MILTT (MR)	+	-	-	+	+	+
MER	+	+	-	+	-	+
NMER	+	-	-	+	-	+
NMER collap.	+	+	-	+	+	+
MEE	+	+	+	+	-	++
NMEE collap.	++	+	-	+	+	++
ERMTT (CT)	+	+	-	+	++	++
EEMTT (CT)	+	+	-	+	+	++

Table 6.3: Evaluation results of patient *training_001*

Fusion Strategy	parenchyma (MR)	ventricles (MR)	soft tissues (MR)	injury (MR)	bone (CT)	sinuses (CT)
MCE	++	++	++	++	-	-
NMCE	++	-	++	++	-	+
NMCE collap.	+	+	+	++	-	+
MMI	+	+	-	+	+	+
NMMI	-	+	-	+	++	+
NMMI collap.	++	++	-	++	++	+
CELTT (CT)	++	++	++	++	++	++
MIMTT (CT)	++	++	-	++	++	+
CELTT (MR)	+	-	+	++	-	+
MILTT (MR)	++	-	-	-	-	-
MER	-	+	+	+	-	-
NMER	-	+	+	+	-	-
NMER collap.	++	++	++	++	++	++
MEE	+	+	+	+	-	-
NMEE	-	+	-	+	-	-
NMEE collap.	+	+	-	+	+	+
ERMTT (CT)	+	+	-	+	+	-
EEMTT (CT)	+	+	-	+	+	-

Table 6.4: Evaluation results of patient *patient_109*

Fusion Strategy	parenchyma (MR)	ventricles (MR)	soft tissues (MR)	injury (MR)	bone (CT)	sinuses (CT)
MCE	-	-	-	-	-	-
NMCE	++	-	+	++	-	-
NMCE collap.	++	++	+	++	-	-
MMI	-	++	+	-	-	+
NMMI	-	++	+	-	+	-
NMMI collap.	++	++	+	++	+	-
CELTT (CT)	++	++	++	++	++	++
MIMTT (CT)	++	++	+	++	++	+
CELTT (MR)	-	-	-	-	-	-
MILTT (MR)	++	-	+	++	-	-
MER	+	-	-	+	-	-
NMER	+	-	-	+	-	-
NMER collap.	+	+	-	+	+	+
MEE	-	-	-	-	-	-
NMEE	+	-	-	-	-	-
NMEE collap.	+	-	+	-	+	+
ERMTT (CT)	+	-	-	+	++	++
EEMTT (CT)	+	-	-	+	++	++

Table 6.5: Evaluation results of the real patient

From the above evaluation forms we can observe that the fusion methods generate similar results when we change the input models even though in some cases the results are quite different, such as the results obtained using *MinConditionalEntropy fusion*. On the other hand, analyzing in depth the tables we can observe that:

- *MinConditionalEntropy (MCE)* is a very irregular method. It generates a bad fusion in the real patient, it detects the interest MR parts in *training_001* and CT parts in *patient_109*. It is not able to detect all the MR and CT parts in any case.
- *NormalizedMinConditionalEntropy (NMCE)* uses the same measure used in the *MCE fusion* but it has been normalized. Working with the normalized CE measure of CT and MR models we can obtain more consistent results. It detects very well the parenchyma, the soft tissues and the injury extension but it does not identifies well the ventricles and the bone in any patient. However, it is a semi-automatic strategy which allow us to manipulate the results. In that case, we are able to achieve a good or very good detection of the ventricles as it can be seen in *NMCE collap.* rows.

- *MaxMutualInformation (MMI)* is not a regular strategy even though it is more regular than the MCE fusion. It detects well the ventricles, the soft tissues and the sinuses. The other materials not always are detected, it depends on the patient.
- *NormalizedMaxMutualInformation (NMMI)* is a semi-automatic method which uses the normalized MI measure. It generates interesting results when some adjustments are applied. It detects very well the parenchyma, the ventricles and the injury extension. The bone structure, the soft tissues and the sinuses are not always well identified, it depends on the patient. Therefore, it can be qualified as a good method.
- *ConditionalEntropyLessThanThreshold (CELTT)* is an asymmetric fusion which needs a reference model. Analyzing the results we can conclude that it is not useful when MR model is considered the reference model. On the other hand, when CT is the reference model very interesting results are obtained. Note that all the materials, except soft tissue, are well detected. This method can be considered as one of the best methods.
- *MutualInformationMoreThanThreshold (MIMTT)* fusion, as in *CELTT fusion*, only generates useful results when the reference model is the CT study. It detects very well all the materials.
- *MaxEntropyRate (MER)* and *NormalizedMaxEntropyRate (NMER)* use the ER measure and both generate similar bad results. However, manipulating the fusion, more consistent results can be achieved, as can be seen in *NMER collap.* rows. It detects the parenchyma, ventricles, bone structure, injury extension and the sinuses but never identifies the soft tissues.
- *MaxErasureEntropy (MEE)* is a non consistent fusion. It detects a lot of materials in patient *training_001* but only the parenchyma in the real patient. The drawback of this method is the non detection of the bone structure which is an important part.
- *NormalizedMaxErasureEntropy (NMEE)* produces more consistent results than *MEE* approach, since it is able to detect the bone structure.
- Finally, we can observe that *EntropyRateMoreThanThreshold (ERMTT)* and *ErasureEntropyMoreThanThreshold (EEMTT)* fusions produce very similar results. Both approaches detect the parenchyma, injury extension and bone structure which are considered the more relevant parts. Note that the ventricles and the sinuses are not always identified while the soft tissues are never detected.

Considering all the experiments, the three evaluators agree that the best approaches are the *CELT* and *MIMTT* fusion since they are able to detect from each model the relevant materials. They also consider that the *NMMI* strategy also performs well although it does not detect the soft tissues correctly. The *NMER* method also has to be considered an approach to be taken into account. Note that all these methods are semi-automatic approaches. On the other hand, all evaluators agree that the worse methods are the asymmetric strategies when the MR is considered the reference model.

6.3 Performance Requirements

The next test was designed to evaluate the requirements of the fusion method in terms of memory and computation time. Obtaining a good results is the most important goal. However, we are working with big models and a computer with limited memory to allocate them, and we also have to take into account that we are developing a software which has to be used in a real environment where the time is crucial.

In Sections 4.4 and 5.4 we describe the structures needed for each of the proposed methods. Now, we compare them considering, first, the memory requirements and, then, the computation time. In both tests we use the CT and MR-T1 studies of *training_001* patient described in Section 6.1.

6.3.1 Memory Requirements

To evaluate the memory requirements we measured the megabytes required to compute the measures of each fusion strategy. We consider the input models represented quantized with 16, 32, 64, 128, 256, 512 and 1024 intensity values. The obtained results are represented in Table 6.6 .

	16	32	64	128	256	512	1024
CE	0,0013	0,0046	0,0171	0,0654	0,2559	1,0117	4,0234
MI	0,0013	0,0046	0,0171	0,0654	0,2559	1,0117	4,0234
ER	116,07	116,52	120,06	148,25	373	2168	16516
EE	116,07	116,55	120,19	148,75	375	2176	16548

Table 6.6: The memory in megabytes required using two models and varying their intensity range values.

If we analyze the table, we can see that the memory required to compute the conditional entropy and mutual information measures is negligible compared to the memory required to store the models. On the other hand, the higher-order entropy measures need more than the memory required to store the models. The difference between these two groups of measures is that the

higher-order entropy measures need 3D arrays and too much space is needed if we have to work with models which have more than 256 intensity values using current computers. However, in the previous chapters we showed how the approaches work when we vary the intensity rang and we concluded that the results with quantized images are similar or better compared to the original models. Therefore, this is an interesting property because we can work with these models decreasing the intensity range values.

Moreover, we can also appreciate the importance to avoid the 3D tables if we compare the higher-order entropy measures. The erasure entropy needs two 2F tables more than entropy rate but it is not a drawback since the excess of memory required is negligible compared to the memory required to store the 3D table.

The plot of Figure 6.3 illustrates the difference in terms of memory required by the measures.

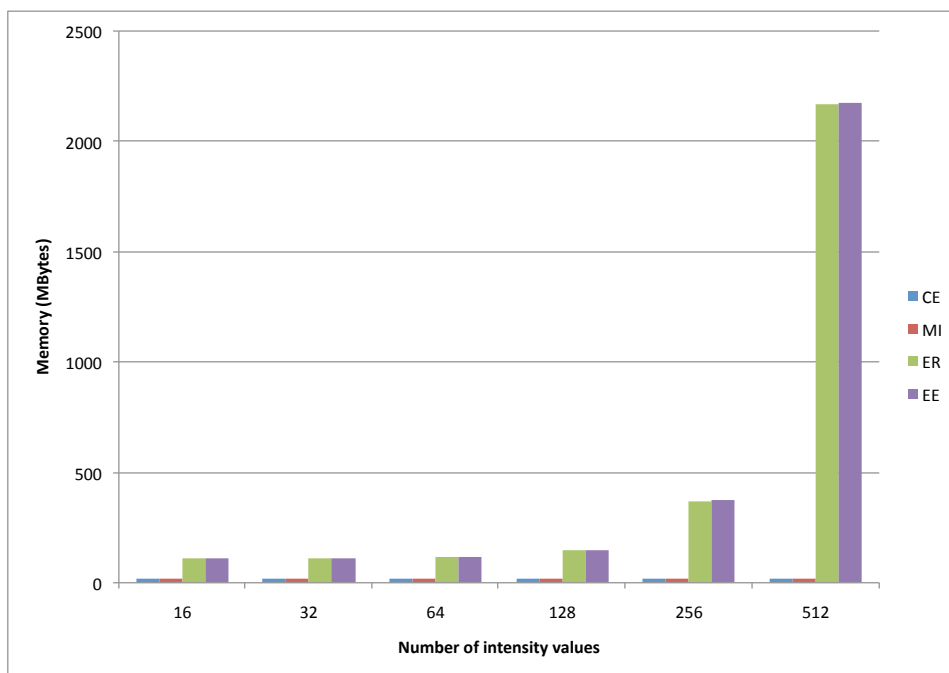


Figure 6.3: Memory requirements for the computation of the different measures considering different quantizations of the original input models.

6.3.2 Computational Time Requirements

In this section, we evaluate the computational time required for the computation of the different measures used in our fusions methods. All the tests have been done using a MacBook Pro which has an Intel Core 2 Duo 2.2 GHz processor and 4GB of RAM memory.

In Table 6.7 we describe the time in milliseconds required to compute the measures for the CT and MR models varying the intensity range of values.

Although the Conditional Entropy and the Mutual Information share information and hence one can benefit from the computation of the other, we have considered that each measure has been computed independently. The obtained results are illustrated in Table 6.7. If we carry out the computation of one measure considering the computations of the other one the time required to obtain the second measures is less than 10 milliseconds.

	16	32	64	128	256	512
CE	240	237	235	237	244	271
MI	240	236	236	237	245	273
ER	12758	12707	12840	13229	14026	17433
EE	11610	11658	11811	12288	12862	15541

Table 6.7: The time in milliseconds required for the computation of the measures applied in the fusion methods considering different quantizations of the input models.

In Figure 6.4 we plot the time required to compute the measures when considering different quantizations of the input models. Note that there is a big difference between the conditional entropy and mutual information measures and the higher-order entropy measures. The CE and MI group of measures is 60 times faster than the HE. This is due to large amount of memory required for the computation of the higher-order entropy measures. They need a lot of time in order to fill the histogram tables and to compute the measure and store it in a 3D table.

Focusing our interest on higher-order entropy measure we can see that the erasure entropy is faster even though two more 2D tables are required to compute the measure. Although the entropy rate has two 2D tables less, it has to compute more values in order to fill the tables.

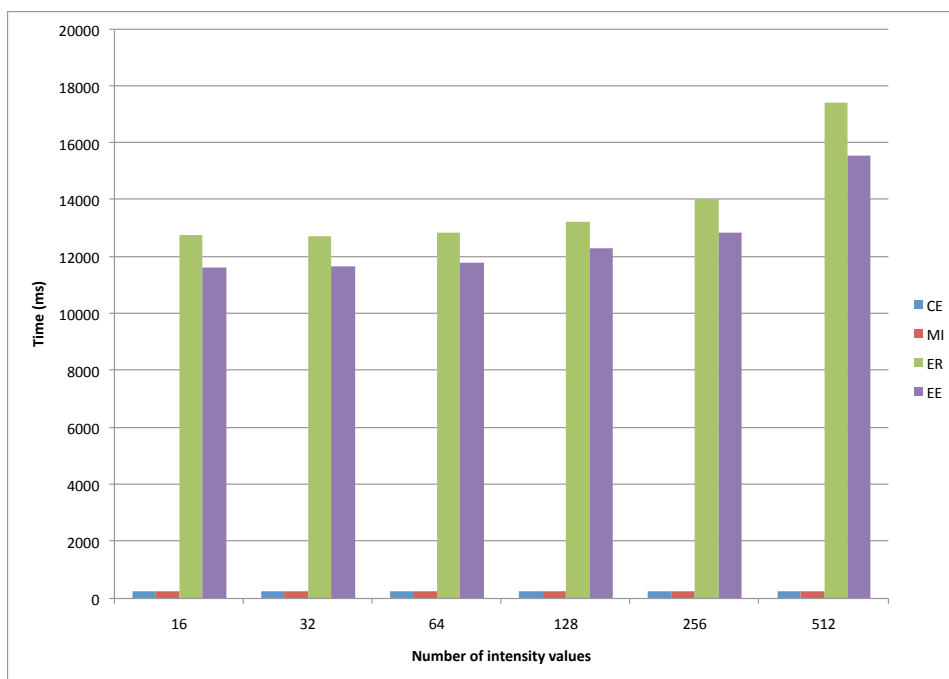


Figure 6.4: The time in milliseconds needed to compute the measures using two models and varying their intensity range values.

Chapter 7

Conclusions

Multimodal visualization is a powerful tool to be considered in medical applications. The development of new techniques that reduce the complexity of current fusion strategies allowing to assist and enhance visual image interpretation in a timely and accurate manner is fundamental in real clinical environments. The aim of this master thesis has been to contribute to the research of these techniques.

Next, the main contributions of this thesis are described:

- Several information theoretic tools have been presented in order to define new fusion strategies. A key point of our work has been the reduction of user interaction when defining the fusion process. We have achieved this objective proposing approaches which use the registration channel or higher-order entropy measures.
- An important part of our work has been the validation of the proposed techniques in a real medical environment. We have developed a multi-volume visualization framework that integrates different techniques to register, segment, and fuse medical volume data sets. We have implemented also different visualization strategies to enhance the validation process. Our techniques have been tested on synthetic and real data sets and we concluded that the best approaches are the *ConditionalEntropyLessThanThreshold* and *MutualInformationMoreThanThreshold* fusion since they are able to detect from each model the most relevant materials.

In future work, we will investigate how to automatically define a transfer function to achieve a high quality multi-volume visualization. We will also concentrate our effort on how the segmentation process can be exploited during fusion.

Bibliography

- [1] Pascual Abellán and Dani Tost. Multimodal volume rendering with 3d textures. *Comput. Graph.*, 32(4):412–419, 2008.
- [2] Ma K.L. Akiba H. A tri-space visualization interface for analyzing time-varying multivariate volume data. In *EUROVIS '07: Proceedings of Eurographics/IEEE VGTC Symposium on Visualization*, pages 115–122, Aire-la-Ville, Switzerland, Switzerland, 2007. Eurographics Association.
- [3] A.D. Brink. Minimum spatial entropy threshold selection. *Vision, Image and Signal Processing, IEE Proceedings -*, 142(3):128–132, Jun 1995.
- [4] Brian Cabral, Nancy Cam, and Jim Foran. Accelerated volume rendering and tomographic reconstruction using texture mapping hardware. In *VVS '94: Proceedings of the 1994 symposium on Volume visualization*, pages 91–98, New York, NY, USA, 1994. ACM.
- [5] Wenli Cai and Georgios Sakas. Data intermixing and multi-volume rendering. *Comput. Graph. Forum*, 18(3):359–368, 1999.
- [6] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley Series in Telecommunications, 1991.
- [7] Retrospective Image Registration Evaluation Project database. <http://www.insight-journal.org/rire/>.
- [8] David P. Feldman and James P. Crutchfield. Discovering noncritical organization: Statistical mechanical, information theoretic and computational views of patterns in one-dimensional spin systems. Working Paper 98-04-026, Santa Fe Institute, Santa Fe (NM), USA, April 1998.
- [9] Maria Ferre, Anna Puig, and Dani Tost. Decision trees for accelerating unimodal, hybrid and multimodal rendering models. *Vis. Comput.*, 22(3):158–167, 2006.

- [10] Sören Grimm, Stefan Bruckner, Armin Kanitsar, and Meister Eduard Gröller. Flexible direct multi-volume rendering in interactive scenes. In *Vision, Modeling, and Visualization (VMV)*, pages 386–379, October 2004.
- [11] Miquel Feixas i Feixas. An information-theory framework for the study of the complexity of visibility and radiosity in a scene, 2002.
- [12] Luis Ibáñez, Will Schroeder, Lydia Ng, and Josh Cates. Itk software guide. pages 378–392.
- [13] James T. Kajiya. The rendering equation. *SIGGRAPH Comput. Graph.*, 20(4):143–150, 1986.
- [14] James T. Kajiya and Brian P Von Herzen. Ray tracing volume densities. *SIGGRAPH Comput. Graph.*, 18(3):165–174, 1984.
- [15] Joe Kniss, Charles Hansen, Michel Grenier, and Tom Robinson. Volume rendering multivariate data to visualize meteorological simulations: a case study. In *VISSYM '02: Proceedings of the symposium on Data Visualisation 2002*, pages 189–ff, Aire-la-Ville, Switzerland, Switzerland, 2002. Eurographics Association.
- [16] Wolfgang Krueger. The application of transport theory to visualization of 3d scalar data fields. In *VIS '90: Proceedings of the 1st conference on Visualization '90*, pages 273–280, Los Alamitos, CA, USA, 1990. IEEE Computer Society Press.
- [17] Philippe Lacroute and Marc Levoy. Fast volume rendering using a shear-warp factorization of the viewing transformation. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 451–458, New York, NY, USA, 1994. ACM.
- [18] Marc Levoy. Display of surfaces from volume data. *IEEE Comput. Graph. Appl.*, 8(3):29–37, 1988.
- [19] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH Comput. Graph.*, 21(4):163–169, 1987.
- [20] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens. Multimodality image registration by maximization of mutual information. *Medical Imaging, IEEE Transactions on*, 16(2):187–198, 1997.
- [21] D. Mattes and DR Haynor. Pet-ct image registration in the chest using free-form deformations. 2003.

- [22] Nelson Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):99–108, 1995.
- [23] Klaus Mueller, Naeem Shareef, Jian Huang, and Roger Crawfis. High-quality splatting on rectilinear grids with efficient culling of occluded voxels. *IEEE Transactions on Visualization and Computer Graphics*, 5(2):116–134, 1999.
- [24] Gemma Piella. A general framework for multiresolution image fusion: from pixels to regions. *Information Fusion*, 4(4):259–280, 2003.
- [25] L.R. Schad, R. Boesecke, W. Schlegel, G.H. Hartmann, V. Sturm, L.G. Strauss, and W.J. Lorenz. Three dimensional image correlation of ct, mr, and pet studies in radiotherapy treatment planning of brain tumors. *Journal of computer assisted tomography*, 11(6):948–954, 1987.
- [26] Claude E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 623–656, July, October 1948.
- [27] R. Stokking, K. Zuiderveld, H.E. Hulshoff Pol, and M.A. Viergever. Spect/mri visualization for frontal-lobe-damaged regions. *Visualization in Biomedical Computing*, 2359(1):282–290, 1994.
- [28] ITK Insight Toolkit. www.itk.org.
- [29] QT Toolkit. www.qtsoftware.org.
- [30] VTK Visualization Toolkit. www.vtk.org.
- [31] S. Verdu and T. Weissman. The information lost in erasures. *Information Theory, IEEE Transactions on*, 54(11):5030–5058, Nov. 2008.
- [32] Lee Westover. Interactive volume rendering. In *VVS '89: Proceedings of the 1989 Chapel Hill workshop on Volume visualization*, pages 9–16, New York, NY, USA, 1989. ACM.
- [33] Lee Westover. Footprint evaluation for volume rendering. In *SIGGRAPH '90: Proceedings of the 17th annual conference on Computer graphics and interactive techniques*, pages 367–376, New York, NY, USA, 1990. ACM.
- [34] R. W. Yeung. *A First Course in Information Theory*. Kluwer, New York, 2002.