Alma Mater Studiorum · University of Bologna

School of Science
Department of Physics and Astronomy
Master Degree in Physics

SYNTHETIC GENERATION OF AMYLOID PET IMAGES BY NON-LINEAR DIMENSIONALITY REDUCTION INVERSION

Supervisor:	Submitted by:
Prof. Daniel Remondini	Lorenzo Di Meco

Co-supervisor:
Prof. Andrea Chincarini

Academic Year 2021/2022

Abstract

Privacy and ethical restrictions and data scarcity in positron emission tomography field call for efficient methods for expanding datasets through synthetic generation of new data that cannot be traced back to real patients and that are at the same time realistic. In this thesis, machine learning techniques were applied to 1001 amyloid-beta PET images, which had undergone a diagnosis of Alzheimer's disease: the evaluations were 540 positive, 457 negative and 4 unknown. The Isomap algorithm was adopted as a manifold learning technique in order to reduce the dimensionality of the PET dataset; once a low-dimensional representation of the data was obtained, a numerical scale-free interpolation method was applied in order to explicitly define an inverse of the dimensionality reduction mapping. The interpolant was tested on the original PET images via a leave-one-out approach, where the removed images were compared with the reconstructed ones through a mean structural similarity index measure (MSSIM = 0.76 ± 0.06). The effectiveness of this measure for the scope of this thesis is questioned, since it indicated slightly higher performance for a method of comparison exploiting principal component analysis (MSSIM = 0.79 ± 0.06), which gave clearly poor quality reconstructed images with respect to those recovered by the numerical inverse mapping, as visually assessed by comparison with the original images. Ten new synthetic PET images were finally generated and, after having been mixed with ten originals, were sent to a team of clinicians, experts in amyloid PET, for the visual assessment of their realism; no significant agreements were found either between clinicians and the true image labels or among the clinicians, meaning that original and synthetic images were indistinguishable. The future perspective of this thesis points to the improvement of the research framework in the amyloid-beta PET field by considerably increasing available data, overcoming the constraints of data acquisition and privacy issues. Potential improvements in obtained results can be achieved through refinements of the manifold learning and the inverse mapping stages during the PET image analysis, by exploring different combinations in the choice of algorithm parameters and by applying other non-linear dimensionality reduction algorithms. An additional prospect suggested by this work is the search for new methods to assess image reconstruction quality.

Contents

Introduction

Nowadays, extremely high-dimensional data are processed by a large number of scientific fields, such as climatology, astrophysics, neuroscience, biology and applied mathematics. For instance, a simple 50-by-50 grayscale image belongs to a space with 2500 dimensions; on one hand, this amount of features provides an accurate characterisation of the object of study; on the other hand, as more dimensions are added, the processing power required to analyse the data and the amount of training data needed to make meaningful models grow exponentially. This problem is called "curse of dimensionality": when moving to higher dimensions, the volume containing the data quickly grows, hence becoming more and more sparse. In order to keep the same density of the feature space, one would have to increase exponentially the number of observations.

A widely explored approach to tackle the "curse of dimensionality" is to reduce the dimensions of a dataset consisting of a large number of features, which are assumed to be highly interrelated (non-independent). "The human brain confronts the same problem in everyday perception, extracting from its high-dimensional sensory inputs a manageably small number of perceptually relevant features" [1]. For instance, a 2500-dimensional grayscale image dataset could be described by far fewer variables, based on the patterns hidden in it, due to the presence of correlations between neighboring pixels. This is the underlying idea of dimensionality reduction techniques, which seek to define new variables by "combining" the ones of the high-dimensional space in a suitable way in order to find a low-dimensional representation, in which data model can be trained more efficiently and data samples can be visualised more easily.

Several dimensionality reduction algorithms have been proposed in the literature. The easiest to implement are those that search for linear dependencies between variables and seek to preserve the global structure of the data space; two instances are Principal Component Analysis [2, 3] and Multidimensional Scaling [4, 5]. These methods define linear combinations of the original variables, so they can characterise only linear structures in data spaces; however, many high-dimensional data have underlying non-linear degrees of freedom that cannot be captured by linear methods. Other techniques were then developed in order to investigate these non-linear structures; some of the most commonly used are Isomap [1], Locally Linear Embedding (LLE) [6], Laplacian eigenmaps [7, 8], t-distributed Stochastic Neighbor Embedding (t-SNE) [9], LargeVis [10] and Uniform Manifold Approximation and Projection (UMAP) [11].

These latter methods fall under the category of manifold learning techniques, since they assume the data to lie on or near a low-dimensional non-linear manifold embedded in a high-dimensional space and seek to find the intrinsic geometric structure of it; the aim is to preserve the local relationships between the data samples, that is to keep similar data points close and dissimilar data points far apart, in the low-dimensional space [10]. The dimensions of the manifold would then represent the meaningful degrees of freedom of the data.

In order to understand the importance and usefulness of manifold learning, an example of semi-supervised classification is helpful. Figure 1 [12, Figure 1] shows the problem of classifying an unlabelled point "?" giving some labelled and many unlabelled points of two classes "+" and "o" on a plane curve, that is a 1-dimensional manifold embedded in a 2-dimensional space. The distances between two arbitrary points in the original feature space do not necessarily reflect their intrinsic similarity, so one would like another representation of the points that preserves the intrinsic geometry of the data manifold and that better describes the relationship between the points. Panel 5 shows the positions of labelled points on the curve in the new representation, obtained by means of the Laplacian eigenmap [7, 8]. The point "?" now falls in the middle of "+" points and can easily be classified as belonging to the class "+".

The problem of manifold learning also brings with it another issue: how to invert a dimensionality reduction mapping to return to the original feature space. PCA

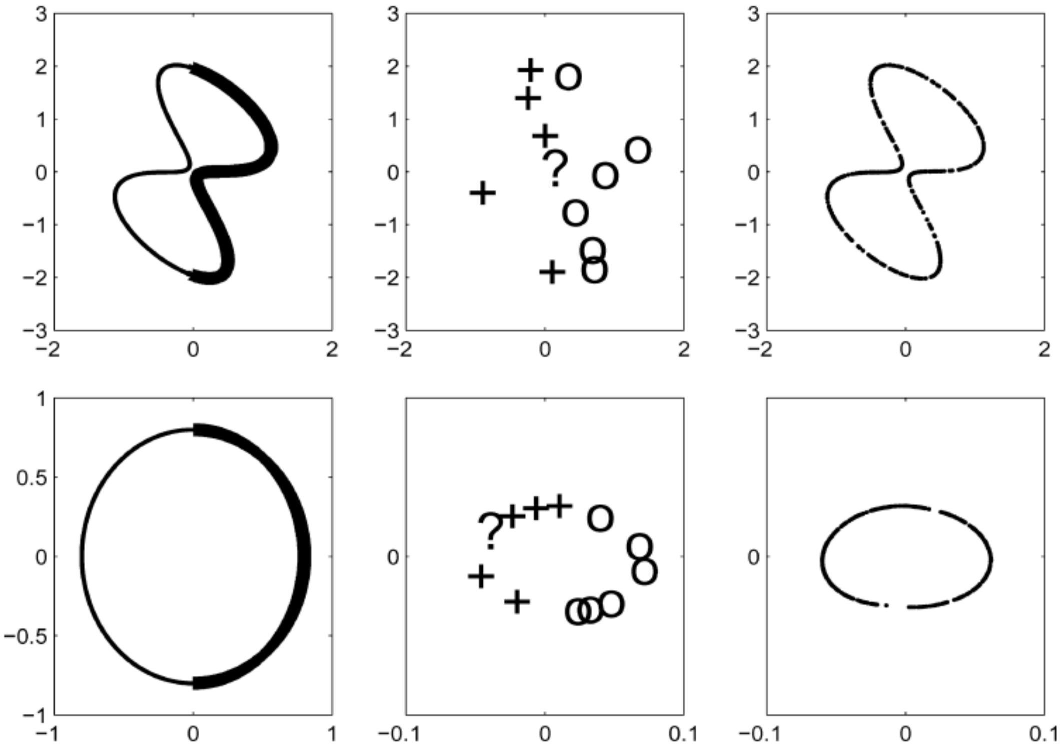


Figure 1: Top row: Panel 1. Two classes "+" and "o" on a plane curve. Panel 2. Labelled points of the classes and unlabelled point "?" to be classified. Panel 3. Unlabelled points. Bottom row: Panel 4. Alternative representation of the curve. Panel 5. Positions of labelled points and "?" on the new representation. Panel 6. All points in the new representation. [12, Figure 1]

provides an easy answer, since it defines a rotation matrix which then can be inverted effortlessly; however, PCA is linear and does not have the flexibility to learn non-linear manifolds embedded in high-dimensional spaces. The non-linear methods mentioned above, instead, are only defined on a discrete set of data points, so an explicit mapping that could then be inverted is not provided. Therefore, one usually needs numerical methods in order to define the inverse of non-linear dimensionality reduction mappings. Although non-linear mapping are more difficult to invert, the importance of understanding the intrinsic geometric structure of the manifold data makes their use worthwhile.

This thesis work was aimed at generating synthetic amyloid-beta PET images in order to overcome the privacy and ethical restrictions regarding management of sensitive data and to provide a method to increase the availability of data in PET field. The approach adopted was then the study of the manifold underlying the high-dimensional space of PET images; a low-dimensional representation of the data was achieved by means of the Isomap algorithm [1]. Once the meaningful degrees of freedom of the manifold had been obtained, new synthetic PET images could be generated by interpolating new data points on the low-dimensional space and by recovering their corresponding high-dimensional PET images through the inverse of the Isomap mapping. A numerical scale-free interpolation method based on cubic radial basis functions was adopted.

The structure of this thesis is organised as follow: Chapter 1 presents the materials used for this thesis work. Chapter 2 explains the algorithms adopted in order to reduce the dimensionality of the PET image space and to invert the non-linear mapping. Chapter 3 describes the preprocessing and analysis of PET images, as well as the generation of the synthetic images. Chapter 4 gives the results of this thesis work. Chapter 5 discusses the results and offer the future perspectives of this thesis.

Chapter 1

Dataset description

For this thesis, Positron Emission Tomography (PET) image data were collected from the European Alzheimer's Disease Consortium (EADC) database. The EADC is a consortium of 47 Alzheimer's centres in 13 European countries and is funded by the European Union [13]. Its primary goal is to develop and maintain an organisational structure capable of rapidly carrying outrivals of interventions designed to prevent, slow or ameliorate the primary and secondary symptoms of Alzheimer's disease [14].

The dataset is composed of 4 groups of patients who had been submitted to ^{18}F-florbetaben (286 subjects), ^{18}F-florbetapir (377 subjects), ^{18}F-flutemetamol (191 subjects) and ^{11}C-Pittsburgh Compound B (147 subjects) PET respectively, in 21 research centres across Europe.

The four tracers used in the PET scans are radioactive compounds that detect amyloid-beta deposition in the brain [15], associated with Alzheimer's disease. From a clinical point of view, these tracers provide roughly the same cortical information; nevertheless, they are different molecules, characterised by their own specific and non-specific binding mechanisms. They should then be regarded separately, by dividing the dataset in subsets, according to the tracers. However, this practise is usually not observed and, typically, mixed datasets are analysed; therefore, for this thesis work, the whole set of PET images was used, ignoring the technical differences between the tracers.

The 1001 PET scans are classified as negative (457 "NEG" subjects), positive

Table 1.1: Demographics and Clinical Diagnosis of the PET image dataset

	PET images
Sample size	1001
Age [y]	69.4 ± 8.3 [41; 90]
Sex	485 females 415 males 101 other/unknown
Diagnosis (NEG:POS:UNK)	457:540:4

(540 "POS" subjects) or unknown (4 "UNK" subjects).

The demographics (including age and sex) and clinical diagnosis of the subjects are summarized in Table 1.1; the ages of 111 patients are not known.

Each PET image is stored in a 193-by-229-by-193 matrix, saved in a *NIfTI* file. An example of PET labelled "negative" (rescaled into an 8bit image) is showed in Figures 1.1, 1.2, 1.3, which illustrate the brain axial, coronal and sagittal planes, respectively.

Another example of a PET image is given by the PET labelled "positive", shown in Figures 1.4, 1.5, 1.6.

Finally, an example of a PET image labelled "unknown" is shown in Figures 1.7, 1.8, 1.9.

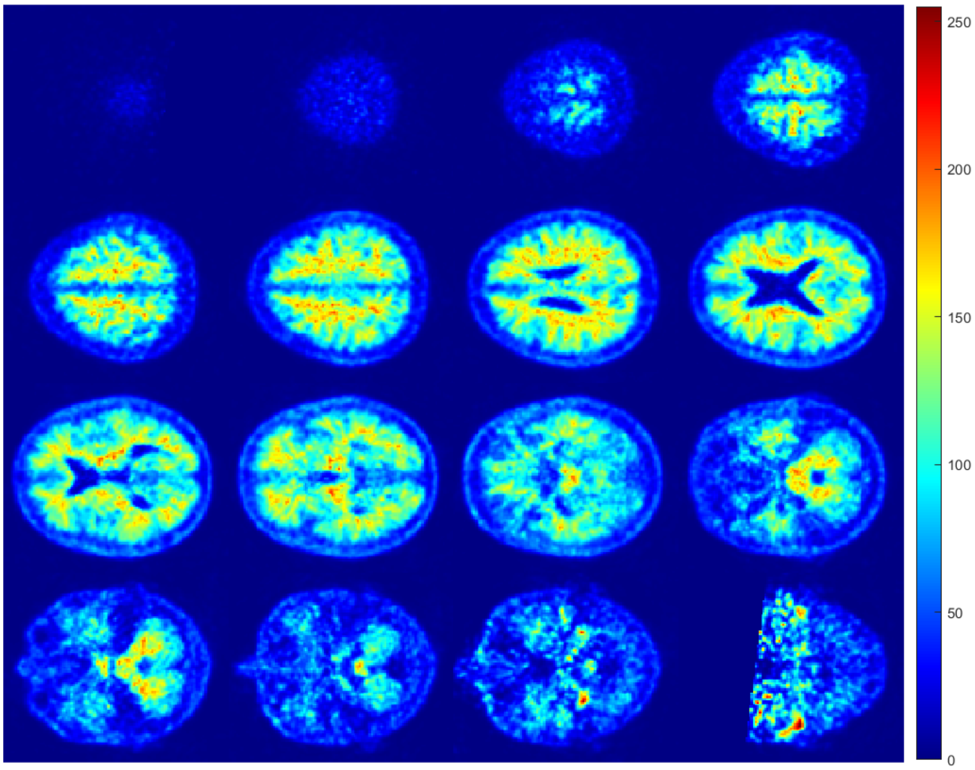


Figure 1.1: Visualisation of a PET image labelled negative on the brain axial planes.

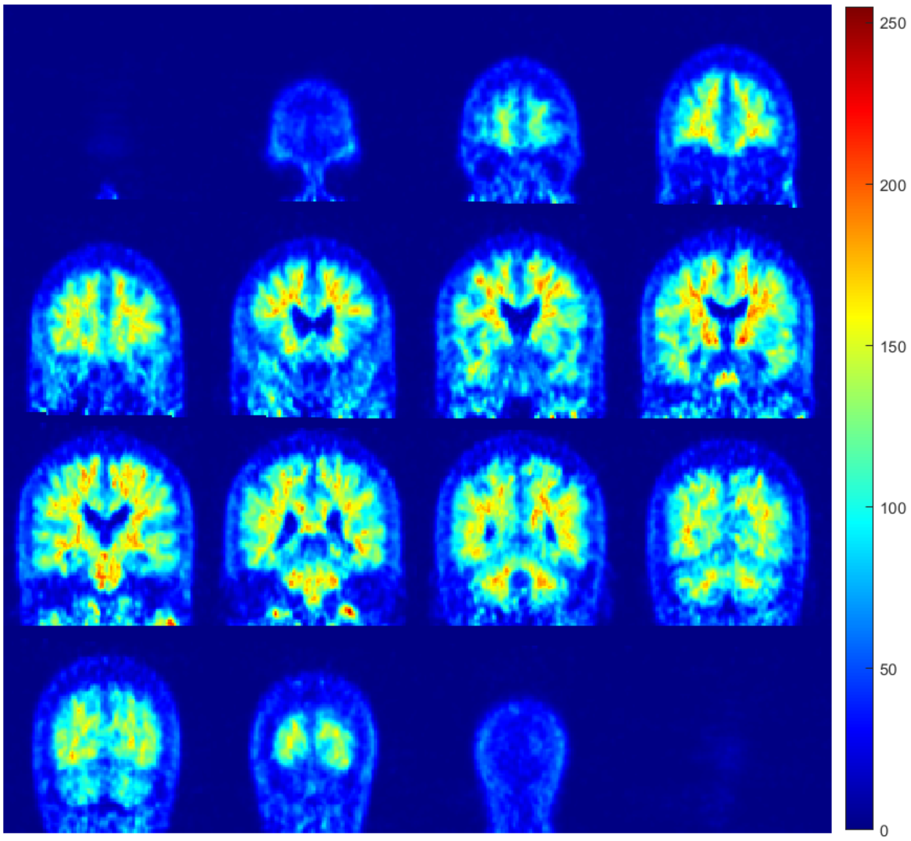


Figure 1.2: Visualisation of a PET image labelled negative on the brain coronal planes.

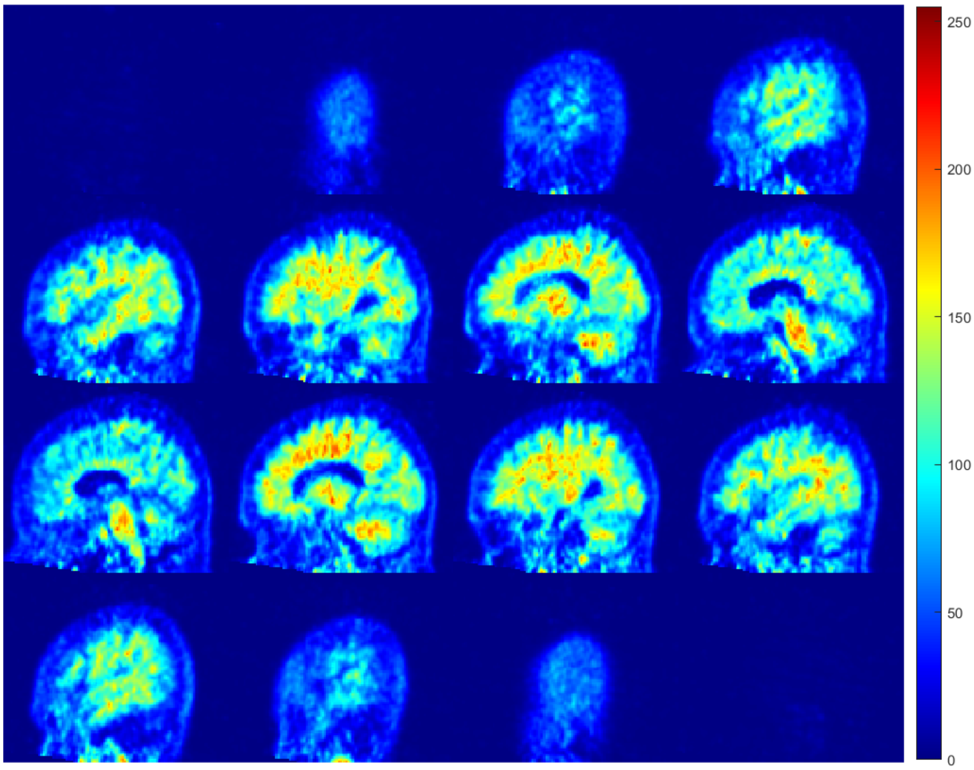


Figure 1.3: Visualisation of a PET image labelled negative on the brain sagittal planes.

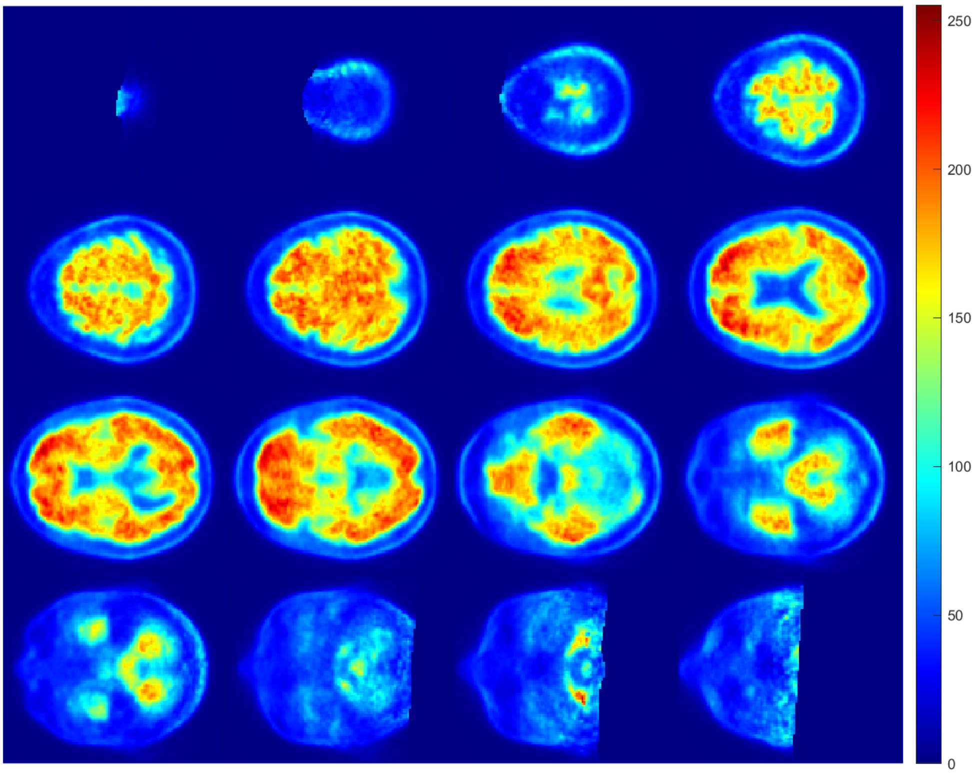


Figure 1.4: Visualisation of a PET image labelled positive on the brain axial planes.

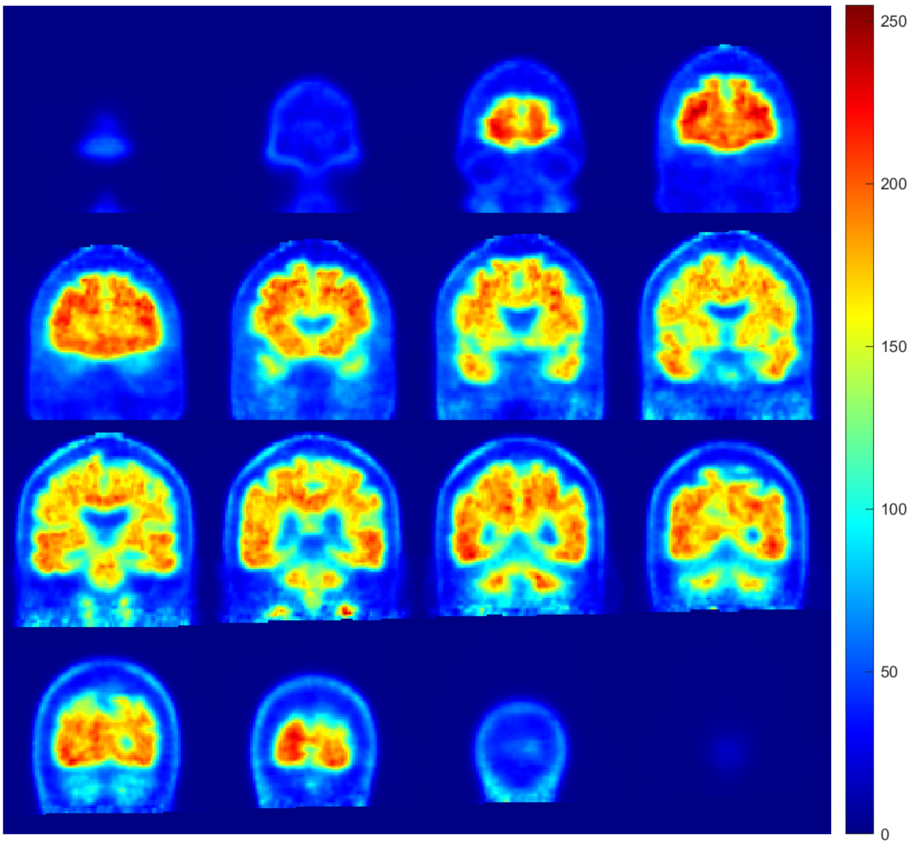


Figure 1.5: Visualisation of a PET image labelled positive on the brain coronal planes.

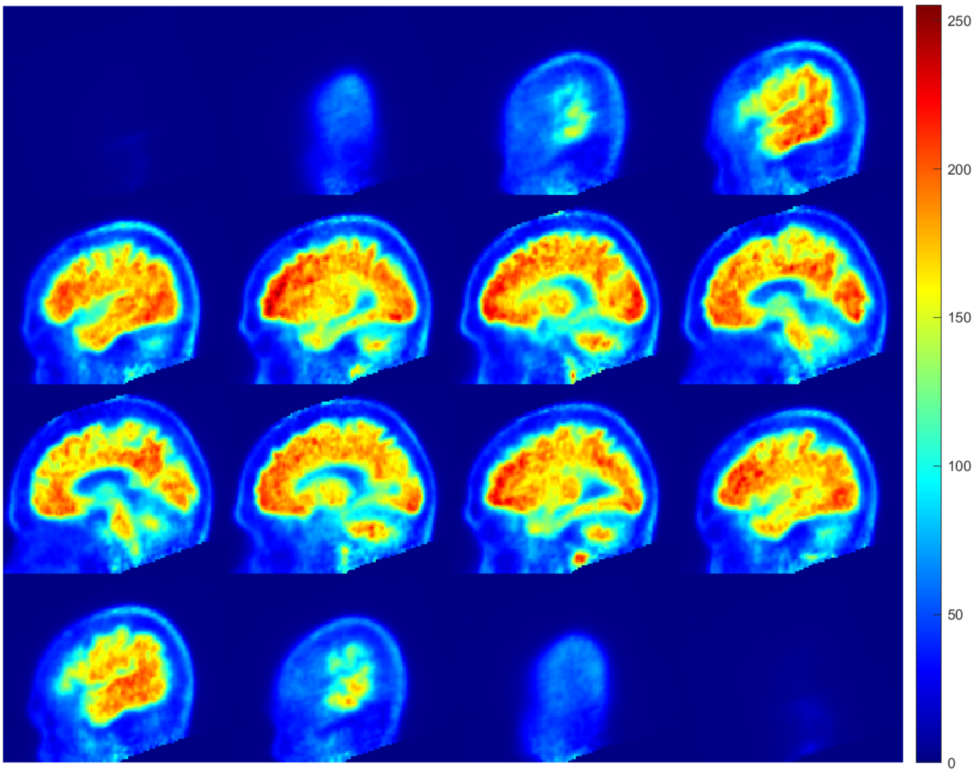


Figure 1.6: Visualisation of a PET image labelled positive on the brain sagittal planes.

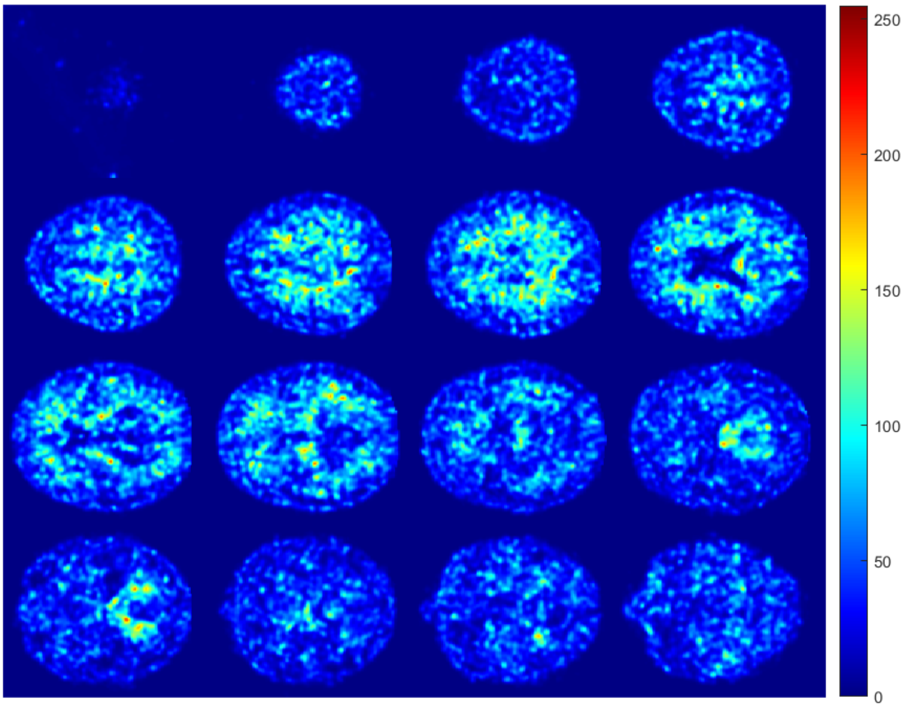


Figure 1.7: Visualisation of a PET image labelled unknown on the brain axial planes.

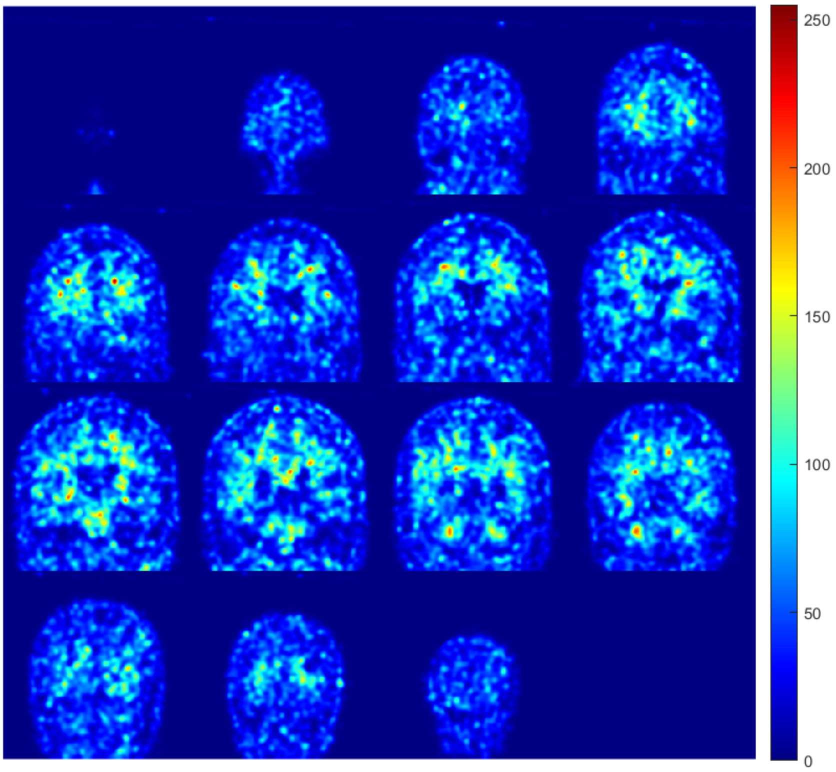


Figure 1.8: Visualisation of a PET image labelled unknown on the brain coronal planes.

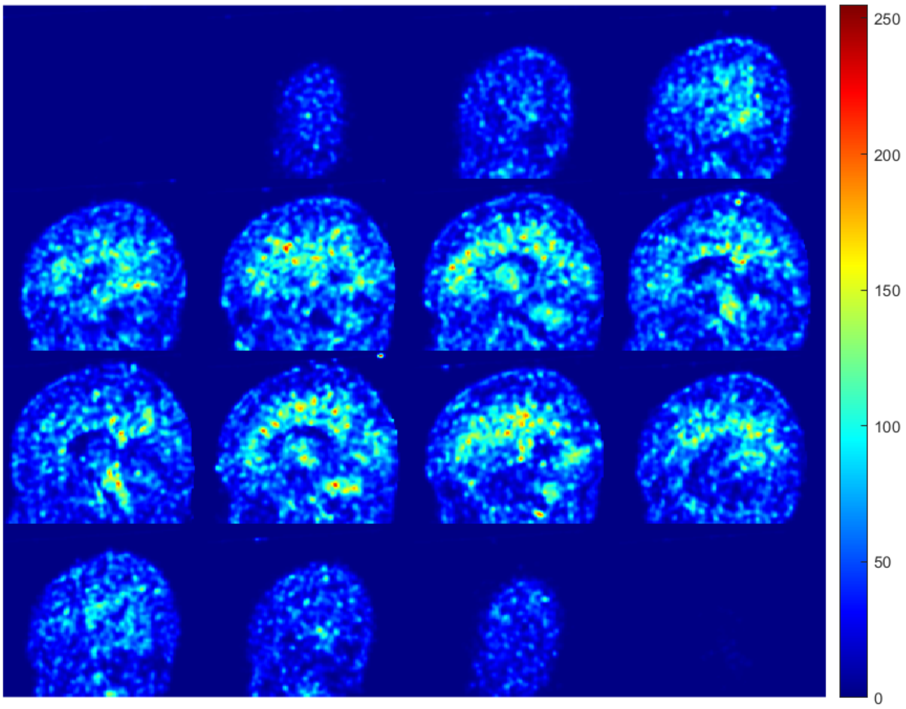


Figure 1.9: Visualisation of a PET image labelled unknown on the brain sagittal planes.

Chapter 2

Methods

This chapter presents an explanation of the algorithms used in this thesis work.

In the first section, the Isometric Feature Mapping, or Isomap, algorithm [1] is introduced as the method of choice for the non-linear dimensionality reduction of the dataset.

The second section offers the theoretical underpinning and the description of a scale-free radial basis interpolation [16] used to numerically invert the dimensionality reduction mapping.

2.1 The Isomap algorithm

The generic problem of dimensionality reduction is the following [8]. Given a set of n data points $\{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(n)}\}$ in $\mathbb{R}^D$, one searches for a set $\{\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(n)}\}$ of points in $\mathbb{R}^d$, where $d \ll D$, such that $\boldsymbol{z}^{(i)}$ is a "good" representation of $\boldsymbol{x}^{(i)}$, in some sense. The assumption that $\{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(n)}\} \subset \mathcal{M}$ will be made, where $\mathcal{M}$ is a non-linear d-dimensional manifold embedded in $\mathbb{R}^D$. Intuitively $\mathcal{M}$ can be thought of as a d-dimensional "surface" in $\mathbb{R}^D$ [12].

The goal is therefore to find a Euclidean embedding $\boldsymbol{\Phi}_n$ of the data points from the high-dimensional space $\mathbb{R}^D$, which is called input or feature space, to the low-dimensional space $\mathbb{R}^d$:

$$\boldsymbol{\Phi}_n : \mathbb{R}^D \to \mathbb{R}^d,\, \boldsymbol{x}^{(i)} \mapsto \boldsymbol{\Phi}_n(\boldsymbol{x}^{(i)}) = \boldsymbol{z}^{(i)} \ , \ \text{ for } i = 1, \dots, n \, . \tag{2.1.1}$$

For two arbitrary points on the underlying manifold, their distance, measured by the standard Euclidean metric or some other domain-specific metric, may appear deceptively small and may not accurately reflect their intrinsic similarity, which is instead correctly measured by the geodesic distance along the manifold.

For neighboring points, input-space distance provides a good approximation to geodesic distance. For faraway points, geodesic distance can be approximated by adding up a sequence of "short hops" between neighboring points. These approximations are computed by finding shortest paths in a graph with edges connecting neighboring data points.

These are the ideas behind the Isometric Feature Mapping, or Isomap, algorithm [1], and its steps are the following:

1. The first step determines which points are neighbours in the manifold $\mathcal{M}$. The neighbourhood of each point $\boldsymbol{x}^{(i)}$ depends on the Euclidean distances $d_{\mathbb{R}^D}(i,j)$ between $\boldsymbol{x}^{(i)}$ and the other points $\boldsymbol{x}^{(j)}$ measured in their input space. The two main methods to find the neighbours are to connect each point to all points within some fixed radius ϵ or to all of its K nearest neighbours. A weighted graph G is then created over the data points with edges of weights $d_{\mathbb{R}^D}(i,j)$.

2. In the second step, Isomap estimates the geodesic distances $d_{\mathcal{M}}(i,j)$ between all pairs of points on the manifold $\mathcal{M}$ by computing their shortest path distances $d_G(i,j)$ in the graph G. The graph distances $d_G(i,j)$ provide increasingly better approximations to the intrinsic geodesic distances $d_{\mathcal{M}}(i,j)$ as the number of points is increased [1]. The shortest paths are computed via the Floyd's algorithm [17]: we initialise $d_G(i,j) = d_{\mathbb{R}^D}(i,j)$ if $\boldsymbol{x}^{(i)}$ and $\boldsymbol{x}^{(j)}$ are linked by an edge, $d_G(i,j) = \infty$ otherwise; then, for each value of $k = 1, \dots, n$ in turn, all entries $d_G(i,j)$ are replaced by $\min\{d_G(i,j), d_G(i,k) + d_G(k,j)\}$. The matrix of final values $\mathcal{D}_G = \{d_G(i,j)\}$ contains the shortest path distances between all pairs of points in G.

3. The final step applies classical Multidimensional Scaling (MDS) [18] to the matrix of graph distances $\mathcal{D}_G$, constructing an embedding of the data in $\mathbb{R}^d$ that best preserves the manifold's estimated intrinsic geometry. The coordinate vectors $\boldsymbol{z}^{(i)}$ for the points in $\mathbb{R}^d$ are chosen to minimize the cost function

$E = \|\tau(\mathcal{D}_G) - \tau(\mathcal{D}_Z)\|_{L^2}$: $\mathcal{D}_Z$ indicates the matrix of Euclidean distances $d_Z(i,j) = \|\boldsymbol{z}^{(i)} - \boldsymbol{z}^{(j)}\|$ and $\|A\|_{L^2}$ the matrix norm $\sqrt{\sum_{i,j} A_{ij}^2}$. The τ operator converts distances to inner products in order to guarantee an efficient optimization and is defined as $\tau(\mathcal{D}) = -HSH/2$, where $S = \{\mathcal{D}_{ij}^2\}$ is the matrix of squared distances and $H = \{\delta_{ij} - 1/N\}$ is the "centering matrix" [1, 18]. The global minimum of the cost function E is achieved by setting the coordinate vectors $\boldsymbol{z}^{(i)}$ to the top d eigenvectors of the matrix $\tau(\mathcal{D}_G)$: if λ_p is the p-th eigenvalue (in decreasing order) of the matrix $\tau(\mathcal{D}_G)$ and $v^i{}_p$ the i-th component of the p-th eigenvector, the p-th component of the d-dimensional coordinate vector $\boldsymbol{z}^{(i)}$ is set equal to $\sqrt{\lambda_p}\, v^i{}_p$.

The Euclidean embdedding $\boldsymbol{\Phi}_n : \mathbb{R}^D \to \mathbb{R}^d$ sought is thus defined as follows:

$$\boldsymbol{\Phi}_n(\boldsymbol{x}^{(i)}) = \left[\sqrt{\lambda_1}\, v^i{}_1\,, \ldots\,, \sqrt{\lambda_d}\, v^i{}_d\right]^T . \tag{2.1.2}$$

For finite datasets, $d_G(i,j)$ may fail to approximate $d_{\mathcal{M}}(i,j)$ for a small fraction of points that are disconnected from the giant component of the neighborhood graph G [1]. These outliers are easily detected as having infinite graph distances from the majority of other points and arise from being outside the neighbourhood of size ϵ of the points in the giant component (in case the radius ϵ was chosen as the Isomap parameter in the first step of the algorithm). Therefore, the graph turns out to be split in multiple connected components and one has to choose between increasing the neighbourhood size to include all the points in a single connected component, or deleting the outliers from further analysis; the latter was actually the choice for the study of this thesis, as will be seen in Section 3.2.1.

The intrinsic dimensionality of the data, that is the dimension d of the manifold $\mathcal{M}$, can be estimated from the decrease of the residual variance of Isomap as the dimensionality of the low-dimensional space $\mathbb{R}^d$ is increased: we may look for the "elbow" at which the curve of the residual variance ceases to decrease significantly with added dimensions.

A classic example of application of the Isomap algorithm is the dimensionality reduction of the "Swiss roll" dataset: the data lie on a 2-dimensional manifold which

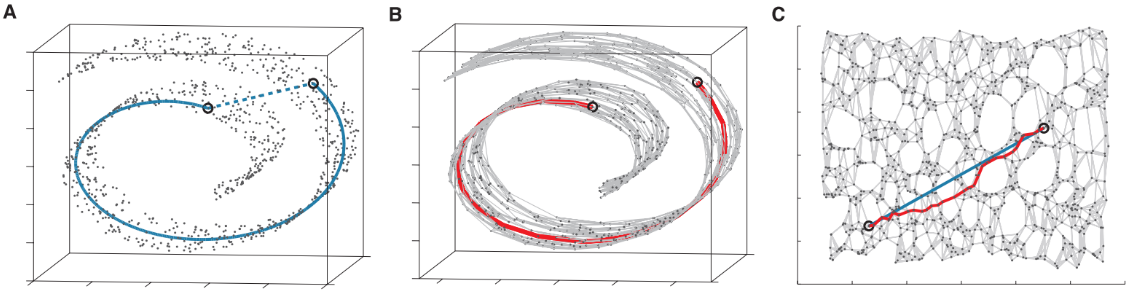


Figure 2.1: Application of Isomap on the “Swiss roll” dataset. Panel A. Comparison between Euclidean (dashed line) and geodesic (solid line) distances. Panel B. Neighborhood graph G providing an approximation of the geodesic distance via shortest-path length (red segments). Panel C. Comparison between geodesic distance and shortest-path length on the 2-dimensional representation. [1, Fig. 3]

is embedded in a 3-dimensional space. A visualisation of the main Isomap ideas applied on this problem is given in Figure 2.1 [12, Fig. 3]. In Panel A, the two circled points appear deceptively close in the original space, as measured by their Euclidean distance (dashed line); instead the geodesic distance (solid line) reflects their intrinsic similarity. Panel B shows the neighborhood graph G constructed from the Euclidean distances in the first step of Isomap; the red segments provide an approximation of the geodesic distance by means of the shortest path distance, computed in the second step of the algorithm. Panel C shows the embedding of the data points in a 2-dimensional space, representing the manifold; the shortest-path and geodesic distances are compared, and it can be seen that the former gives a good approximation of the latter. The 2-dimensional Isomap embedding therefore well approximates the geometric structure of the manifold, finding a configuration that preserves the relationships between the data points.

2.2 Inverse mapping

Let there be n data points $\{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(n)}\} \subset \mathbb{R}^D$ lying on a bounded d-dimensional smooth manifold $\mathcal{M} \subset \mathbb{R}^D$. These points are embedded into $\mathbb{R}^d$ by a non-linear mapping:

$$\boldsymbol{\Phi}_n : \mathbb{R}^D \to \mathbb{R}^d,\ \boldsymbol{x}^{(i)} \mapsto \boldsymbol{\Phi}_n(\boldsymbol{x}^{(i)})\ , \quad \text{for } i = 1, \dots, n\,, \tag{2.2.1}$$

which is approximated by a dimensionality reduction mapping, for instance the one given by the Isomap algorithm and expressed in Equation (2.1.2). Assume the existence of a continuous operator $\mathbf{\Phi} : \mathcal{M} \to \mathbf{\Phi}(\mathcal{M})$, given by the extension of the map $\mathbf{\Phi}_n$ on the whole manifold.

Given $\mathbf{\Phi}^{-1}$ the inverse of the mapping $\mathbf{\Phi}$, the goal is to find an approximate inverse:

$$\mathbf{\Phi}^{\dagger} : \mathbf{\Phi}(\mathcal{M}) \to \mathbb{R}^D,\, \boldsymbol{z} \mapsto \mathbf{\Phi}^{\dagger}(\boldsymbol{z}) \tag{2.2.2}$$

such that $\lim\limits_{n\to\infty} \mathbf{\Phi}^{\dagger}(\boldsymbol{z}) = \mathbf{\Phi}^{-1}(\boldsymbol{z})$ for all $z \in \mathbf{\Phi}(\mathcal{M})$.

2.2.1 Radial basis function interpolant

Let there be a collection of radial basis functions (RBF) defining the kernel $g : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ and the set of points $\{\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(n)}\} \subset \mathbb{R}^d$, with corresponding data points $\boldsymbol{x}^{(i)} = \mathbf{\Phi}^{-1}(\boldsymbol{z}^{(i)})$ for $i = 1, \dots, n$; the RBF interpolant $\boldsymbol{s} : \mathbb{R}^d \to \mathbb{R}^D$ is then defined:

$$s_k(\boldsymbol{z}) = \sum_{j=1}^{n} \alpha_k^{(j)}\, g(\boldsymbol{z}, \boldsymbol{z}^{(j)})\ ,\ \ k = 1, \dots, D\,. \tag{2.2.3}$$

The weights $\alpha_k^{(j)}$ are found by interpolating the data points $\boldsymbol{x}^{(i)}$, so that $s_k(\boldsymbol{z}^{(i)}) = \Phi_k^{-1}(\boldsymbol{z}^{(i)}) = x_k^{(i)}$ holds. The conditions that fix the weights are described by the following system of equations:

$$\begin{bmatrix} g(\boldsymbol{z}^{(1)}, \boldsymbol{z}^{(1)}) & \cdots & g(\boldsymbol{z}^{(1)}, \boldsymbol{z}^{(n)}) \\ \vdots & & \vdots \\ g(\boldsymbol{z}^{(n)}, \boldsymbol{z}^{(1)}) & \cdots & g(\boldsymbol{z}^{(n)}, \boldsymbol{z}^{(n)}) \end{bmatrix} \cdot \begin{bmatrix} \alpha_1^{(1)} & \cdots & \alpha_D^{(1)} \\ \vdots & & \vdots \\ \alpha_1^{(n)} & \cdots & \alpha_D^{(n)} \end{bmatrix} =:$$

$$=: G\,A = \begin{bmatrix} x_1^{(1)} & \cdots & x_D^{(1)} \\ \vdots & & \vdots \\ x_1^{(n)} & \cdots & x_D^{(n)} \end{bmatrix} = \begin{bmatrix} (\boldsymbol{x}^{(1)})^T \\ \vdots \\ (\boldsymbol{x}^{(n)})^T \end{bmatrix} =: X\ , \tag{2.2.4}$$

where G is the matrix of the RBF kernel such that $G_{ij} = g(\boldsymbol{z}^{(i)}, \boldsymbol{z}^{(j)})$, the i-th row of $X \in \mathbb{R}^{n\times D}$ identifies the coordinates of $\boldsymbol{x}^{(i)} \in \mathbb{R}^D$ and $A \in \mathbb{R}^{n\times D}$ is the matrix of the interpolation weights: $A_{jk} = \alpha_k^{(j)}$.

Solving the system provides the weight matrix $A = G^{-1}X$ and thus the D-dimensional interpolant $\boldsymbol{\Phi}^{\dagger} : \mathbb{R}^d \to \mathbb{R}^D$ that gives an approximation of the inverse mapping $\boldsymbol{\Phi}^{-1}$; therefore, a new data point $\boldsymbol{x} = \boldsymbol{\Phi}^{-1}(\boldsymbol{z}) \in \mathcal{M}$ can be approximated as follows:

$$\boldsymbol{\Phi}^{\dagger}(\boldsymbol{z}) = \boldsymbol{s}(\boldsymbol{z}) = \left[\boldsymbol{g}(\boldsymbol{z},\cdot)^T A\right]^T = \left[G^{-1}X\right]^T \boldsymbol{g}(\boldsymbol{z},\cdot)\,, \tag{2.2.5}$$

where $\boldsymbol{g}(\boldsymbol{z},\cdot) = \left[g(\boldsymbol{z},\boldsymbol{z}^{(1)}),\ldots,g(\boldsymbol{z},\boldsymbol{z}^{(n)})\right]^T$.

2.2.2 Cubic interpolant

As suggested in [16], we choose a cubic kernel to define the RBF interpolant; thus we have:

$$g(\boldsymbol{z},\boldsymbol{w}) = \|\boldsymbol{z}-\boldsymbol{w}\|^3\ , \tag{2.2.6}$$

which is an instance from the set of RBF kernels $g(\boldsymbol{z},\boldsymbol{w}) = \|\boldsymbol{z}-\boldsymbol{w}\|^{\beta}$ (with $\beta = 1, 3, 5, \ldots$), known as *radial powers*, which in turn belong to the family of the RBF kernels named *polyharmonic splines*.

Thus, the cubic RBF interpolant takes the following form:

$$s_k(\boldsymbol{z}) = \sum_{j=1}^{n} \alpha_k^{(j)} \|\boldsymbol{z}-\boldsymbol{z}^{(j)}\|^3\ ,\ \ k = 1,\ldots,D\,, \tag{2.2.7}$$

where again the weights $\alpha_k^{(j)}$ characterizing the matrix A are to be found by interpolating the data points $\boldsymbol{x}^{(i)}$, as in Equation (2.2.4).

In order to uniquely define our interpolant, the cubic RBF basis is augmented with constant and linear polynomials.

As will be seen later, and as reported in [19], the addition of constant and linear polynomial terms in the cubic RBF basis helps to guarantee the non-singularity of the interpolation matrix; moreover, these terms improve the behaviour of the interpolant near the boundaries of its domain, defined by the points $\{\boldsymbol{z}^{(1)},\ldots,\boldsymbol{z}^{(n)}\} \subset \mathbb{R}^d$ [16].

Lastly, we will have an interpolant that exactly reproduces constant and linear functions, which is a quality often sought in interpolation methods [19].

Let us then assume the new form of the interpolant:

$$s_k(\boldsymbol{z}) = \sum_{j=1}^{n} \alpha_k^{(j)} \, \|\boldsymbol{z} - \boldsymbol{z}^{(j)}\|^3 + \beta_{0k} + \sum_{l=1}^{d} \beta_{lk} \, z_l \ , \ \ k = 1, \ldots, D \, , \tag{2.2.8}$$

where z_l is the l-th coordinate of $\boldsymbol{z} \in \mathbb{R}^d$.

In order to find the interpolation weights we now have, for each dimension $k = 1, \ldots, D$, a system of n equations (given by the conditions $s_k(\boldsymbol{z}^{(i)}) = x_k^{(i)}$ for $i = 1, \ldots, n$) in $n + d + 1$ unknowns $\alpha_k^{(j)}$, β_{lk} and β_{0k}; $d + 1$ conditions must then be added to guarantee unique solvability of the system of interpolation equations.

For the sake of clarity, we shall now focus on a single dimension k of the D-dimensional RBF interpolant $\boldsymbol{s} : \mathbb{R}^d \to \mathbb{R}^D$ and we shall drop the k index of the weights and the interpolant; let us hence consider, without loss of generality, the one-dimensional interpolant $s : \mathbb{R}^d \to \mathbb{R}$, $\boldsymbol{z} \mapsto s(\boldsymbol{z}) = x$.

In order to justify the choice of using constant and linear pylonomials, we will make use of the following definition [16, 19]. Let us consider a continuous even function $f : \mathbb{R}^d \to \mathbb{R}$; it is said to be *conditionally positive definite of order m on $\mathbb{R}^d$* if

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha^{(i)} \alpha^{(j)} \, f(\boldsymbol{z}^{(i)} - \boldsymbol{z}^{(j)}) \geq 0 \tag{2.2.9}$$

for n distinct points $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(n)}\} \subset \mathbb{R}^d$ and $\boldsymbol{\alpha} = (\alpha^{(1)}, \ldots, \alpha^{(n)})^T \in \mathbb{R}^n$ so that, for any real-valued polynomial $p(\boldsymbol{z})$ of degree at most $m - 1$, the condition

$$\sum_{j=1}^{n} \alpha^{(j)} \, p(\boldsymbol{z}^{(j)}) = 0 \tag{2.2.10}$$

holds. If the quadratic form (2.2.9) is equal to zero if and only if $\boldsymbol{\alpha} = 0$, then f is called *strictly conditionally positive definite.*

The function $f(\boldsymbol{z} - \boldsymbol{w}) = \|\boldsymbol{z} - \boldsymbol{w}\|^3$ associated to the cubic RBF kernel $g : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is strictly conditionally positive definite of order 2 on $\mathbb{R}^d$ [19]; this means that, for the set of points $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(n)}\} \subset \mathbb{R}^d$, we have

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha^{(i)} \alpha^{(j)}\, g(\boldsymbol{z}^{(i)}, \boldsymbol{z}^{(j)}) \geq 0\,, \tag{2.2.11}$$

with the coefficients $\alpha^{(j)}$ satisfying:

$$\sum_{j=1}^{n} \alpha^{(j)}\, p_l(\boldsymbol{z}^{(j)}) = 0\,,\ \ l = 1, \ldots, d+1\ , \tag{2.2.12}$$

where the polynomials $p_l(\boldsymbol{z})$ form a basis for the linear space of all polynomials in $\boldsymbol{z}$ up to degree 1: $p_1(\boldsymbol{z}) = 1$, $p_l(\boldsymbol{z}) = z_{l-1}$, for $l = 2, \ldots d+1$.

The condition (2.2.12) provides us with $d+1$ equations to be added to uniquely solve the system of the interpolation equations given by $s(\boldsymbol{z}^{(i)}) = x^{(i)}$, for $i = 1, \ldots, n$.

In order to guarantee unique solvability, however, we need another condition on the points $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(n)}\}$: let us recall the definition of *m-unisolvency.*

A set of points $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(n)}\} \subset \mathbb{R}^d$ is said to be *m-unisolvent* if the unique polynomial of total degree at most m interpolating zero data on $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(n)}\}$ is the zero polynomial.

We can now exploit [19, Theorem 7.2], which states: if the real-valued even function $f : \mathbb{R}^d \to \mathbb{R}$ is strictly conditionally positive definite of order m on $\mathbb{R}^d$ and the points $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(n)}\}$ form an $(m-1)$-unisolvent set, then the following system of linear equations is uniquely solvable:

$$\begin{bmatrix} G & P \\ P^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{bmatrix} = \begin{bmatrix} \boldsymbol{x} \\ 0 \end{bmatrix}\,, \tag{2.2.13}$$

where $G_{ij} = f(\|\boldsymbol{z}^{(i)} - \boldsymbol{z}^{(j)}\|)$ for $i, j = 1, \ldots, n$; $\boldsymbol{x} = (x^{(1)}, \ldots, x^{(n)})^T$; $P_{ml} = p_l(\boldsymbol{z}^{(m)})$ for $m = 1, \ldots, n$ and $l = 1, \ldots, M$, and the polynomials $p_l(\boldsymbol{z})$ for $l = 1, \ldots, M$ form a basis for the linear space of all polynomials in $\boldsymbol{z}$ up to degree $(m-1)$.

Since the cubic RBF kernel is strictly conditionally positive definite of order $m = 2$, we ask the points $\{\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(n)}\}$ to form a 1-unisolvent set in $\mathbb{R}^d$; the condition is equivalent to the following [16]:

$$\operatorname{span}\{(\boldsymbol{z}^{(i)} - \boldsymbol{z}^{(j)}) \ \text{ for } \ i, j = 1, \dots, n\} = \mathbb{R}^d \,. \tag{2.2.14}$$

Let us introduce back the index k referring to the k-th dimension of the interpolant $\boldsymbol{s} : \mathbb{R}^d \to \mathbb{R}^D$ and to the k-th coordinate of the data point $\boldsymbol{x}^{(i)} \in \mathbb{R}^D$, for $i = 1, \dots, n$; then, for each dimension $k = 1, \dots, D$, the system of interpolation equations given by the conditions $s_k(\boldsymbol{z}^{(i)}) = x_k^{(i)}$ is uniquely solvable. The cubic RBF interpolant (2.2.8) is thus guaranteed to be unique and, according to Equation (2.2.5), the approximate inverse mapping $\boldsymbol{\Phi}^\dagger$ sought is defined as follows:

$$\Phi_k^\dagger(\boldsymbol{z}) = s_k(\boldsymbol{z}) = \sum_{j=1}^{n} \alpha_k^{(j)} \, \|\boldsymbol{z} - \boldsymbol{z}^{(j)}\|^3 + \beta_{0k} + \sum_{l=1}^{d} \beta_{lk} \, z_l \,, \tag{2.2.15}$$

where $k = 1, \dots, D$.

Chapter 3

Application on PET data

3.1 Image preprocessing

PET images were extracted from the NIfTI files by using the MATLAB function *$niftiread$*, which saved the images in double-valued 193-by-229-by-193 matrices.

Due to limits on computational resources available, the PET matrices were resized with the MATLAB function *$imresize3$* via a nearest-neighbor interpolation, by choosing a scale equal to 0.5 in order to reduce the size of the three matrix dimensions. New 97-by-115-by-97 matrices were thus obtained.

The PET matrices were then reshaped into $97 \times 115 \times 97$-dimensional vectors, hence providing a dataset of 1001 points in $\mathbb{R}^{1082035}$.

The range of values in the PET vector components differ widely from one PET to another across the dataset on account of the different data acquisition methods of the research centres; as a consequence, vectors have quite different Euclidean norms, as reported in Figure 3.1. To overcome such heterogeneity in the data, each vector was normalised to have norm equal to 1; this method of standardisation was also adopted in order to avoid too much pieces of data being lost during the dimensionality reduction stage, as will be seen in Section 3.2.1 .

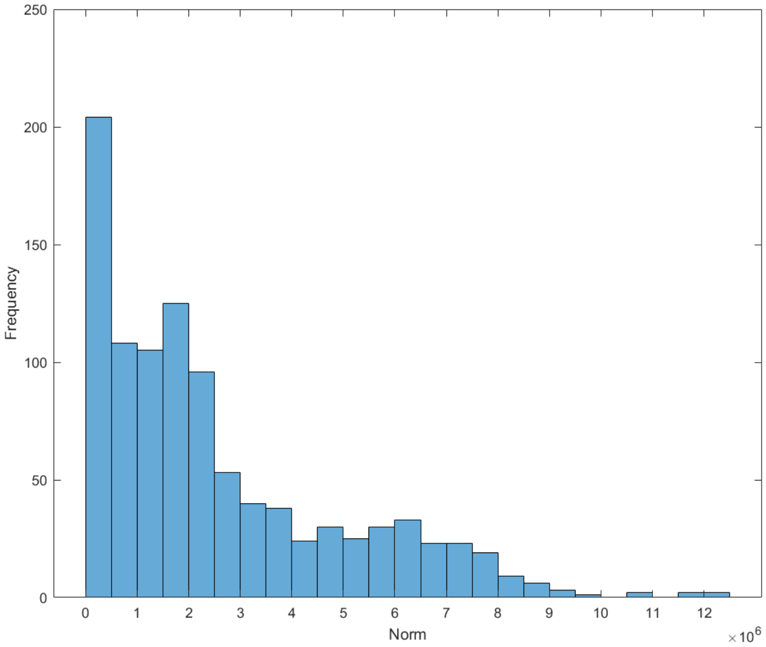


Figure 3.1: Histogram of norm of PET images treated as vectors in $\mathbb{R}^{1082035}$.

3.2 Image analysis

3.2.1 Manifold learning

The dataset dimensionality reduction was approached with the application of the Isomap algorithm on the MATLAB environment; the implementation code provided by [20] was used. Principal component analysis (PCA) [2, 3] was also adopted as a method of comparison.

The PET matrices had been reshaped into $97 \times 115 \times 97$-dimensional vectors for the purpose of computing distance between images. The space of PET physical degrees of freedom was assumed to have the natural structure of a low-dimensional manifold $\mathcal{M}$ embedded in a Euclidean vector space with dimension $D = 1082035$; referring back to Section 2.1, the $n = 1001$ PET vectors from the dataset were the input-space data points $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(n)}\} \subset \mathbb{R}^D$ whose low-dimensional representation was to be found, by computing the Isomap embedding in $\mathbb{R}^d$.

In order to learn the manifold to which the data are supposed to belong, the algorithm takes as input the Euclidean distances $d_{\mathbb{R}^D}(i,j)$ between all the data points $\boldsymbol{x}^{(i)}$ and $\boldsymbol{x}^{(j)}$, for $i,j = 1,\ldots,n$. Those were thus used to connect each point to all points within the chosen fixed radius $\epsilon = 0.45$, creating a graph G to best recover the geometric structure of the manifold $\mathcal{M}$ and to approximate the geodesic distances $d_{\mathcal{M}}$ from the shortest path distances $d_G(i,j)$ in the graph.

Although choosing to use the number of nearest neighbours K of the points $\boldsymbol{x}^{(i)}$ as the Isomap parameter tends to lead to connected graphs, the neighbourhood size ϵ is more geometrically motivated [8], since it fixes the radius within which the Euclidean distances $d_{\mathbb{R}^D}(i,j)$ can be considered as good approximations of the geodesic distances $d_{\mathcal{M}}(i,j)$. Moreover, the K parameter may yield misleading results when the local dimensionality varies across the dataset [1]. That is why the use of the radius ϵ was preferred.

The value of ϵ was chosen after several trials: what emerged is that, in order to have a connected graph G, the Isoamp parameter would have needed to be $\epsilon = 0.64$, for which most of the geodesic distances would have been equal to the Euclidean ones. At that point, the algorithm would have just recovered the classical MDS, which for Euclidean metric is equivalent to PCA. Having a single connected component was then traded-off with the non-linear nature of the Isomap algorithm and 0.64 was taken as an upper bound for ϵ. The lower bound was arbitrary set at 0.3 so as not to have too many data points disconnected from the giant component of the graph. Eventually, the $\epsilon = 0.45$ was chosen by comparing the overall results of this thesis work, computing the performances for different values of ϵ on a fraction of the dataset, as will be explained in Section 3.2.2.

The construction of the neighbourhood graph generated distinct connected components: the first was the giant component, consisting of 990 data points; 11 points were instead disconnected from the first ones. These outliers were thus deleted from further analysis. The reduced dataset was then composed of 990 PET images.

Notice that, without the normalisation procedure of the PET vectors adopted in the preprocessing stage of Section 3.1, the deleted images would have been many more due to the heterogeneity of the dataset.

A 10-dimensional Euclidean embedding $\boldsymbol{\Phi} : \mathbb{R}^{1082035} \to \mathbb{R}^{10}$ of the data was finally constructed.

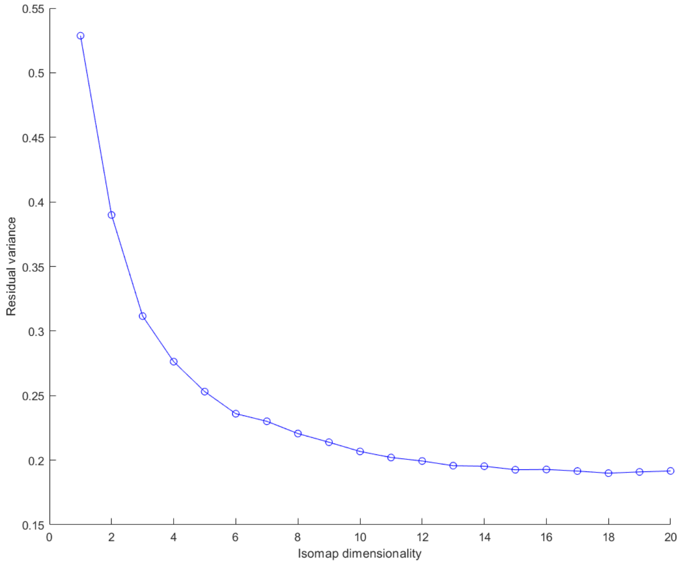


Figure 3.2: Residual variance at different dimensions of the Isomap embedding.

The process of choosing the dimensionality of the low-dimensional Isomap representation started from a qualitative study of the residual variance, showed in Figure 3.2 as a function of the Isomap dimensionality. The "elbow" of the curve may suggest that the true meaningful dimensionality of the PET image dataset is around $d = 6$. However, as with the selection of ϵ and as explained in Section 3.2.2, a comparison of the results of this thesis work indicated that a more appropriate value of choice was around $d = 10$; this is one of the last values of dimensionality where the residual variance can still be seen decreasing sufficiently, before reaching its "plateau".

3.2.2 Inverse map training and testing

The non-linear dimensionality reduction mapping $\boldsymbol{\Phi} : \mathbb{R}^{1082035} \to \mathbb{R}^{10}$ given by the Isomap algorithm was taken into account, with the aim of explicitly define an inverse of it. The low-dimensional representation of the PET images, described by the vectors $\{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(990)}\} \subset \mathbb{R}^{1082035}$, was then given by the points $\{\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(990)}\} \subset \mathbb{R}^{10}$, provided by the Isomap eigenvectors, as pointed in Equation (2.1.2).

The numerical interpolation method described in Section 2.2 was then adopted:

in order to find an approximate inverse $\boldsymbol{\Phi}^{\dagger} : \mathbb{R}^{10} \to \mathbb{R}^{1082035}$ of the Isomap dimensionality reduction mapping, the 1082035-dimensional cubic RBF interpolant (2.2.8) was thus computed by interpolating on the original PET vectors.

The inverse mapping algorithm was cross-validated via a leave-one-out approach. One image at a time was discarded from the dataset in order to train the interpolant over the remaining 989 images. The interpolant was then tested on the removed image $\boldsymbol{x}^{(i)}$ by reconstructing it from its low-dimensional representation $\boldsymbol{z}^{(i)}$; the reconstructed vector $\boldsymbol{\Phi}^{\dagger}(\boldsymbol{z}^{(i)})$ was compared with the original by means of the Euclidean distance-based error $E = \|\boldsymbol{x}^{(i)} - \boldsymbol{\Phi}^{\dagger}(\boldsymbol{z}^{(i)})\|$ and through a mean structural similarity index measure, or MSSIM, between the two vectors, after having been reshaped into matrices and rescaled into 8bit images. The structural similarity index measure between the local windows $\boldsymbol{a}$ and $\boldsymbol{b}$ of two images $\boldsymbol{A}$ and $\boldsymbol{B}$ is defined as follows [21]:

$$\text{SSIM}(\boldsymbol{a}, \boldsymbol{b}) = \frac{(2\mu_a\mu_b + C_1)\,(2\sigma_{ab} + C_2)}{(\mu_a^2 + \mu_b^2 + C_1)\,(\sigma_a^2 + \sigma_b^2 + C_2)}\,, \tag{3.2.1}$$

where μ_a, μ_b, σ_a, σ_b, and σ_{ab} are the voxel sample means, standard deviations, and cross-covariance for the local windows, respectively; C_1 and C_2 are two constants included to avoid instability when either $(\mu_a^2 + \mu_b^2)$ or $(\sigma_a^2 + \sigma_b^2)$ is very close to zero [21]. In order to evaluate the overall similarities between the entire images, a mean SSIM (MSSIM) index was used, by computing the average of the SSIM indeces over the local windows; a value closer to 1 indicates a greater similarity between images.

The code implementation adopted for the SSIM index computation was provided by the MATLAB function *ssim*.

The overall performance of the algorithm was estimated on the 990 test parts taken together and was compared with the performance of PCA reconstruction from the first 10 principal components of the PET vectors.

As mention in Section 3.2.1, before fixing the Isomap parameter ϵ and the dimension d of the Euclidean embedding, the cross-validation stage of the interpolant was repeated several times on a smaller fraction of the dataset. The parameters were chosen by evaluating the performance of the cubic RBF interpolant, computed for different values of ϵ and d. Due to limits on computational resources available, only a few combinations of these parameters could be tested, and the leave-one-out cross-validation was performed on 10% of the whole dataset.

These limitations call for a more thorough future investigation of possible combinations of parameters to be used, possibly over the whole dataset.

3.3 Synthetic generation

The last stage of this thesis work was the synthetic generation of new PET images.

The aim was to create data that could be considered realistic according to a visual inspection of clinicians; the key point of the work was that generated data could not be traced back to real patients, thus solving both data scarcity in amyloid-beta PET field and privacy issues regarding management of sensitive data.

The method for generating synthetic data was inspired by the Synthetic Minority Over-sampling Technique, or SMOTE [22], and by its incorporation with Isomap algorithm, given by [23]. The idea behind the method adopted is the following: once the 10-dimensional representation of the data is obtained, the physical degrees of freedom of the PET images are encoded in the linear Euclidean space $\mathbb{R}^{10}$. The data were previously supposed to lie on a non-linear manifold $\mathcal{M}$, embedded in $\mathbb{R}^{1082035}$. As a consequence, the local space between any two samples is not necessarily physical, meaning that a random point drawn from the line segment between the two samples would not necessarily represent a realistic PET image. However, the space $\mathbb{R}^{10}$ providing the Isomap embedding of the data is linear; therefore, chances are that a random point $\boldsymbol{z}$ selected along the line segment between two neighbours on this space would be the low-dimensional representation of a realistic PET image, which could then be recovered by inverting the Isomap mapping through the cubic RBF interpolant (2.2.15); the new synthetic data point would then be $\boldsymbol{x} = \boldsymbol{\Phi}^{\dagger}(\boldsymbol{z})$.

The method for generating a synthetic PET image was thus the following:

- a point $\boldsymbol{z}^{(i)}$ was selected randomly from the low-dimensional representation of the dataset;
- 20 nearest neighbors of $\boldsymbol{z}^{(i)}$ were computed;

- a random number between 1 and 20 was generated in order to choose one of the 20 nearest neighbours of $\boldsymbol{z}^{(i)}$;
- the difference between $\boldsymbol{z}^{(i)}$ and its nearest neighbour was computed; it was then multiplied by a random number between 0 and 1 and subtracted from $\boldsymbol{z}^{(i)}$. This caused the selection of a random point $\boldsymbol{z}$ along the line segment between $\boldsymbol{z}^{(i)}$ and its neighbour;
- the inverse mapping method was finally applied to $\boldsymbol{z}$ in order to obtain a PET vector $\boldsymbol{x} = \boldsymbol{\Phi}^{\dagger}(\boldsymbol{z})$.

This procedure was repeated ten times in order to generate five synthetic PET images from within the area of "negative" samples and five more from the "positive" area.

The new ten PET images were finally mixed with ten originals and were sent to a team of four clinicians, experts in amyloid PET, for the visual assessment of their realism.

Chapter 4

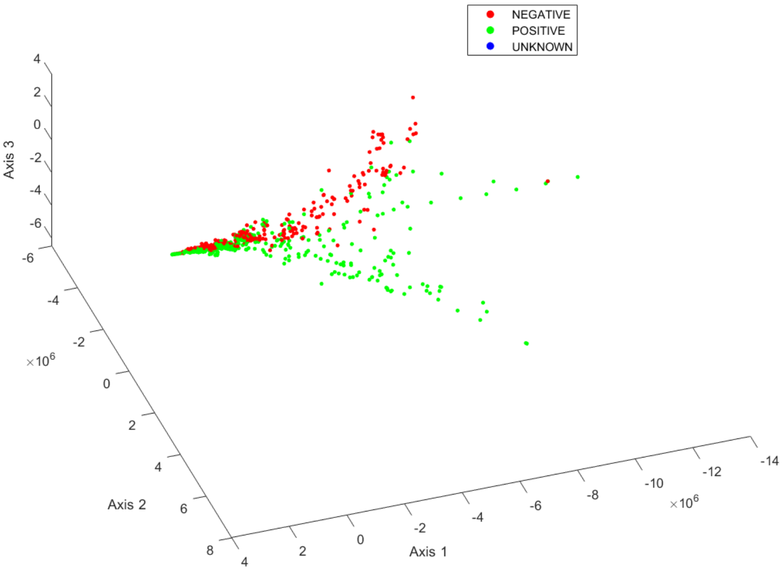
Results

In the manifold learning stage of this thesis work, described in Section 3.2.1, a 10-dimensional representation of the PET image data was obtained, both via the Isomap algorithm and with PCA, that was the method of comparison.

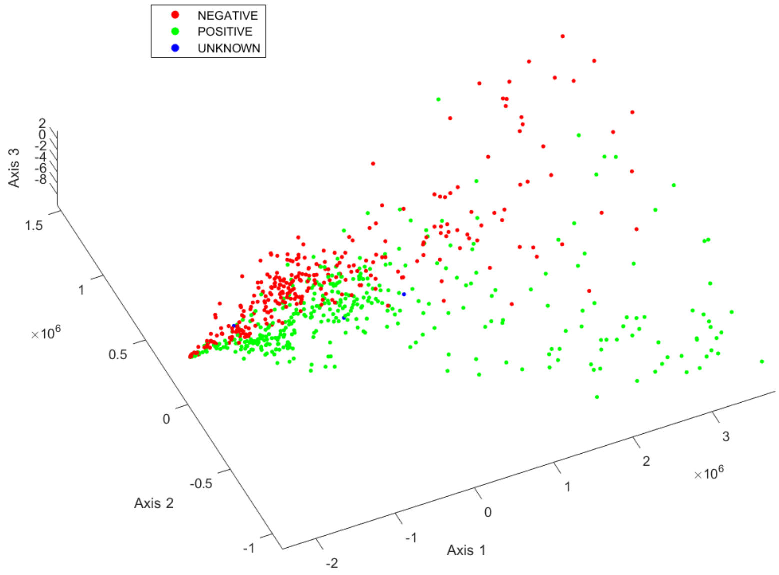
In Figure 4.1, 3-dimensional representations provided by Isomap and PCA before the preprocessing normalisation procedure are shown, with labels of clinical diagnoses highlighted. It is clear that Isomap mapping fails to separate the two main classes of positive and negative patients for the a large number of data points; the classes overlap also in PCA representation. Another issue concerning these representations is the presence of two regions with highly different data point density. Both representation have one branch with sparse data and a very dense branch, where the dissimilarity among the data points is hard to grasp, even though for Isomap the situation is worse. This problematic pattern of data point density was tackled with a normalisation of the PET vectors, as described in Section 3.1.

In Figure 4.2, the 3-dimensional projections of the Isomap and PCA representations after the normalisation are shown; the labels corresponding to the clinical diagnoses are highlighted, showing a good visual separation between the two main classes of negative and positive patients in both the representations.

In Figure 4.3, the 3-dimensional projections of both the algorithms are shown for the PET images from nine of the most numerous research centre classes. A separation between some of the classes is visible and may be due to the different data acquisition methods among the centres.

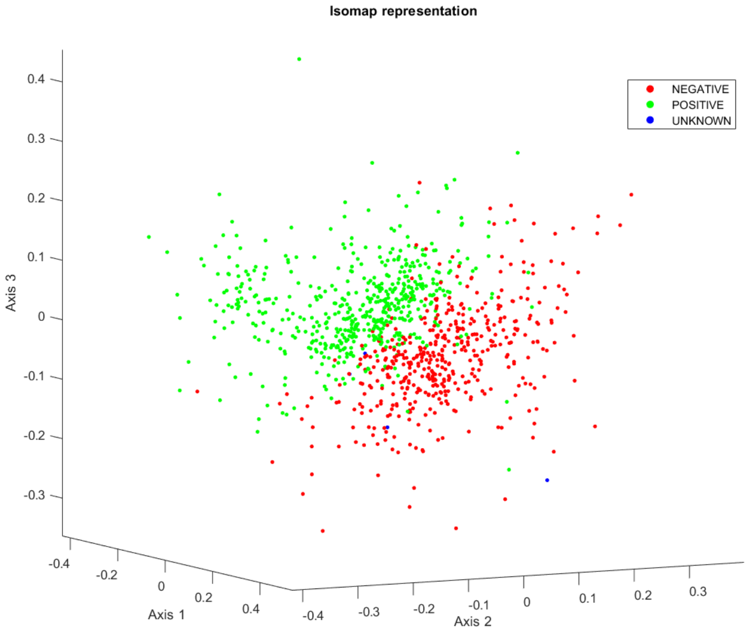


Isomap representation.

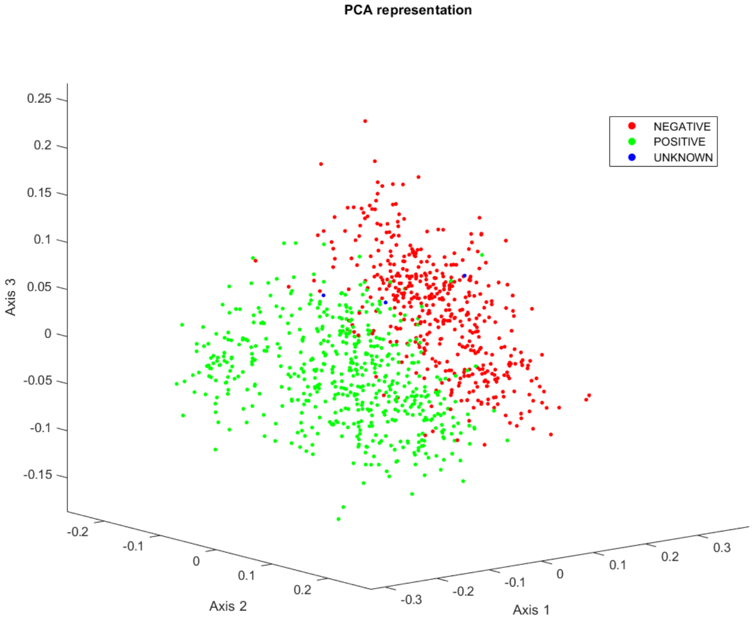


PCA representation.

Figure 4.1: Isomap and PCA 3-dimensional representations of PET images with highlighted labels of clinical diagnoses, before the normalisation.



Isomap representation.



PCA representation.

Figure 4.2: 3-dimensional projections of the Isomap and PCA representations of PET images with highlighted labels of clinical diagnoses.

In Section 3.2.2, the cross-validation of the cubic RBF inverse mapping algorithm was described; the overall performance of the interpolant was determined by the Euclidean distance-based error E and the mean structural similarity index measure MSSIM. The statistics of the performance measures are depicted in the boxplots in Figure 4.4; cubic RBF interpolant and PCA results are compared.

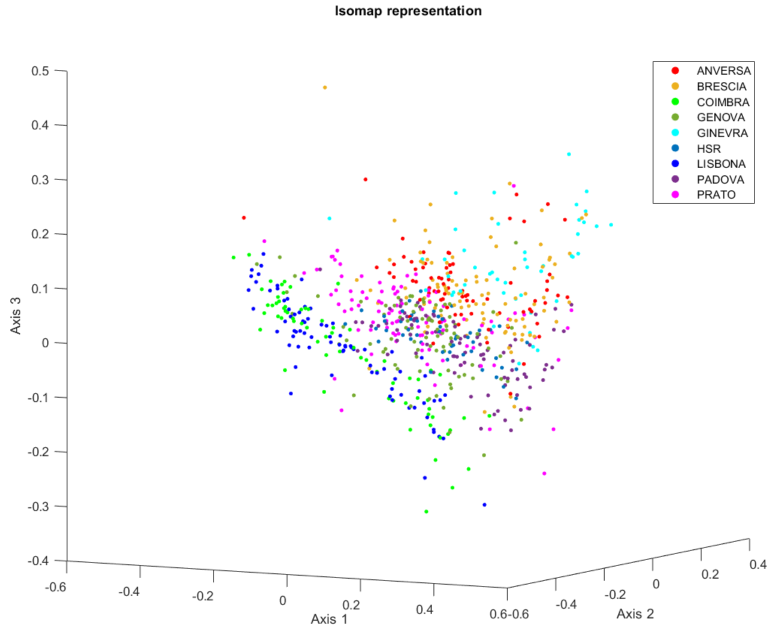
The overall performances were considered to be the average values of the performances on the single reconstructions. Therefore, for cubic interpolant reconstruction, $E_{\text{cubic}} = 0.24 \pm 0.03$ and $\text{MSSIM}_{\text{cubic}} = 0.76 \pm 0.06$; for PCA, $E_{\text{PCA}} = 0.18 \pm 0.02$ and $\text{MSSIM}_{\text{PCA}} = 0.79 \pm 0.06$. According to the reconstruction error measure E, PCA technique appears to have a fairly higher performance than the cubic RBF interpolant; also MSSIM indicates slightly higher performance for PCA.

However, PET images reconstructed by inverting PCA were clearly of lower quality than those recovered by the cubic interpolant, as visually assessed by comparison with the original images. Figures 4.5 - 4.10 show an example of visual comparison between a PET image and its reconstruction with the two methods. The example was chosen so that the MSSIM between the original image and its cubic RBF interpolant reconstruction would be around the average value $\text{MSSIM}_{\text{cubic}}$; therefore, $\text{MSSIM} \approx 0.78$. The corresponding value of MSSIM for the PCA reconstruction is ≈ 0.83. It is clear that, although in this example the performance of reconstruction is higher for PCA than for the cubic RBF interpolant, the latter manages to construct an image that better preserves the anatomical structures of the brain and the PET scan characteristics.

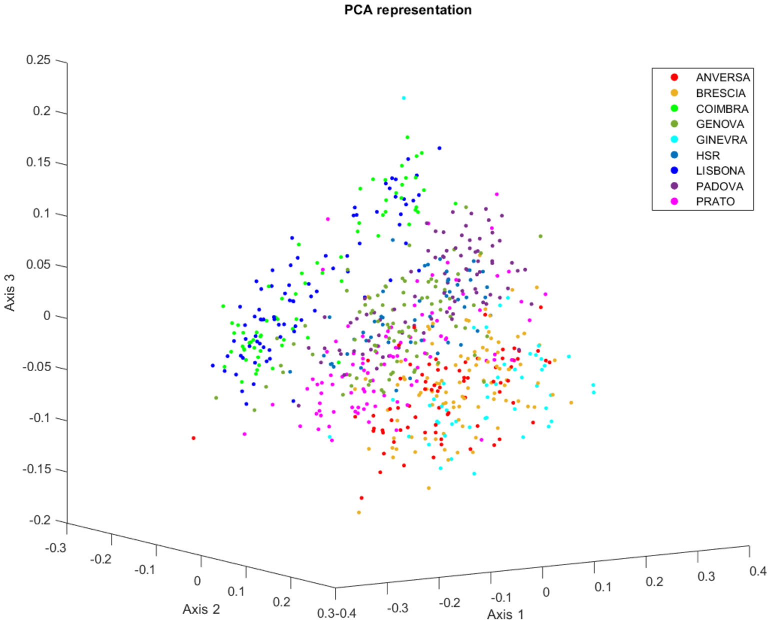
The application limits of the Euclidean distance-based error were already known; the SSIM [21] was indeed designed to have a measure that could better assess image similarities. However, the results obtained may cast doubt on the effectiveness of this measure, at least within the scope of this thesis work.

As stated in Section 3.3, ten synthetic PET images were generated and mixed with ten originals; they were then sent to a team of four clinicians, here called raters, for an independent visual assessment of their realism.

Table 4.1 shows the measures of accuracy of each rater with respect to the true labels (original or synthetic) of the PET images and the other raters; the accuracy

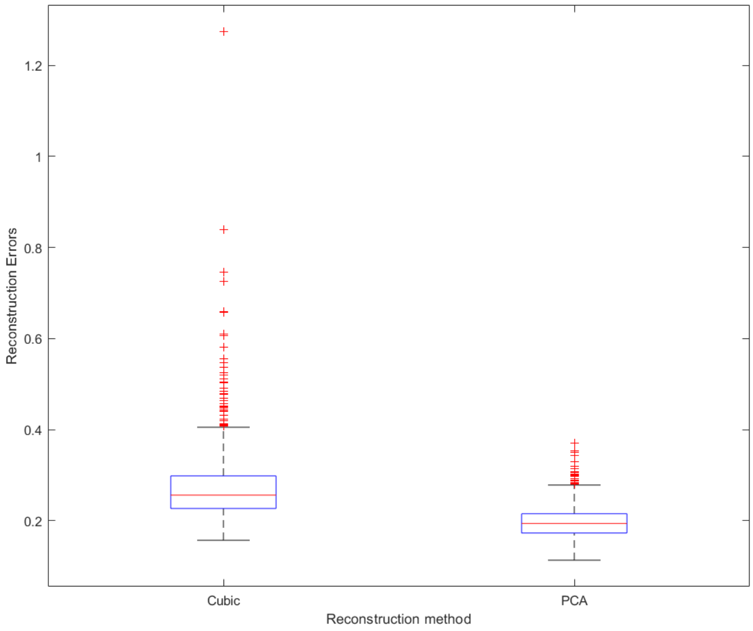


Isomap representation.



PCA representation.

Figure 4.3: 3-dimensional projections of the Isomap and PCA representations of PET images from nine research centres.



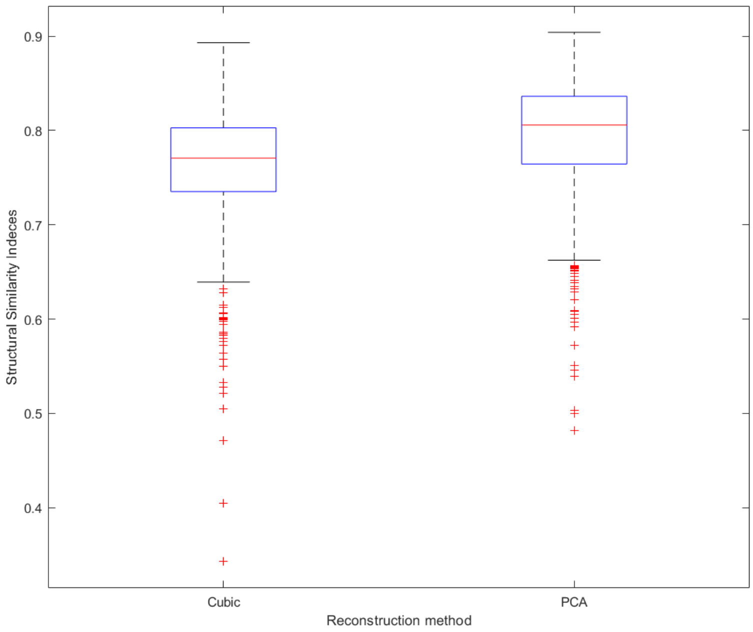


Figure 4.4: Comparisons between cubic RBF interpolant and PCA for the each of the performance measures E (above) and MSSIM (below).

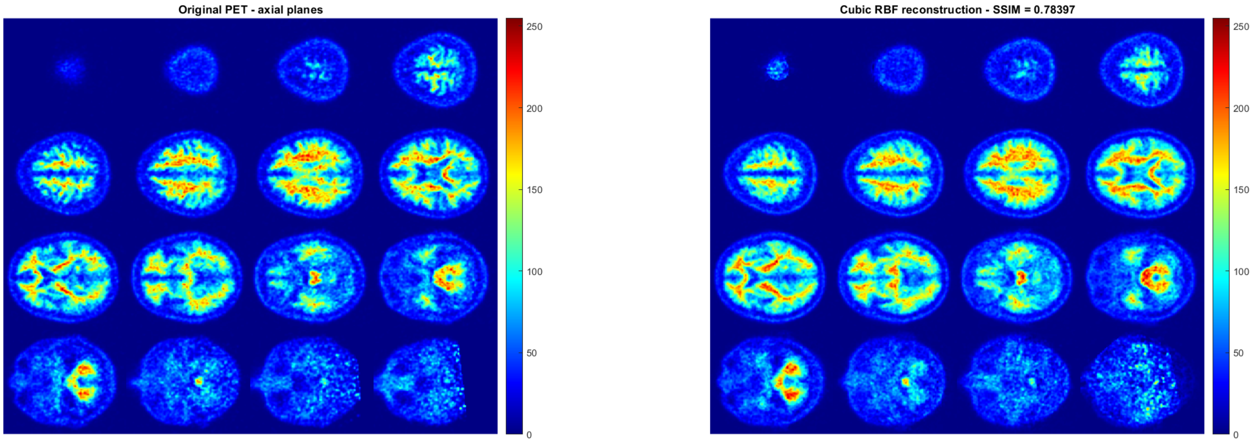

Original PET - axial planes
Cubic RBF reconstruction - SSIM = 0.78397
250
200
150
100
50
0

Figure 4.5: Example of comparison on the axial planes between a PET image and its cubic RBF reconstruction.

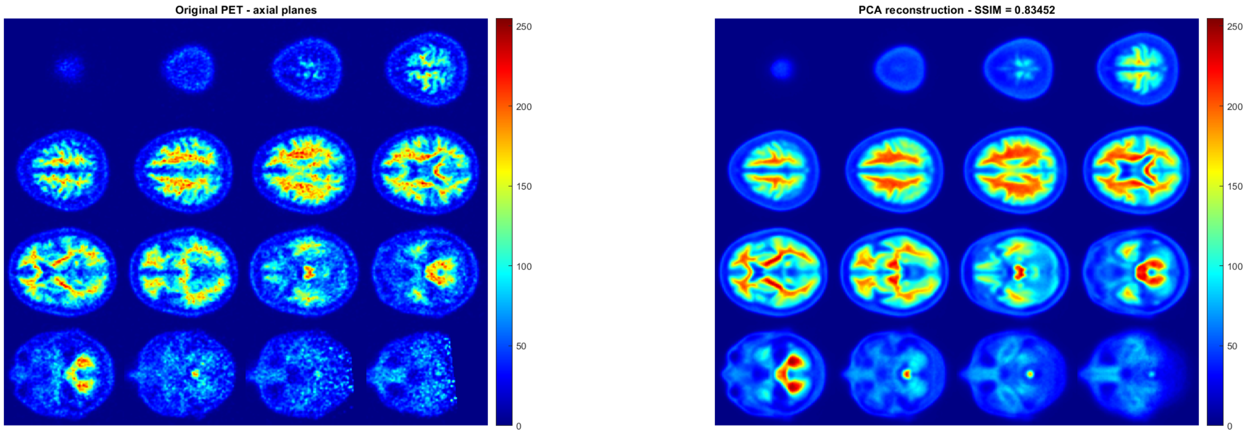

Original PET - axial planes
PCA reconstruction - SSIM = 0.83452
250
200
150
100
50
0

Figure 4.6: Example of comparison on the axial planes between a PET image and its PCA reconstruction.

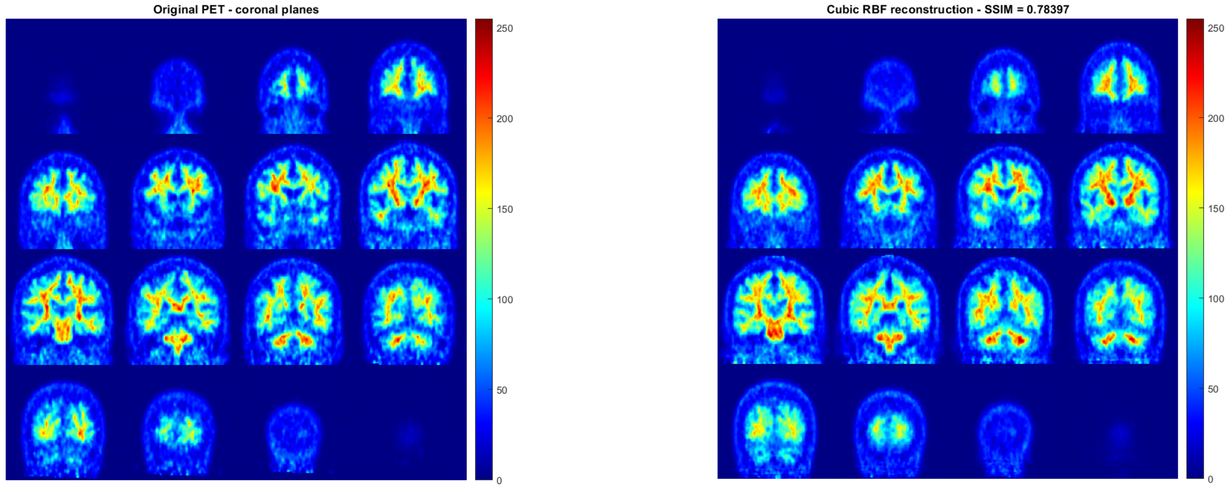

Original PET - coronal planes
250
200
150
100
50
0
Cubic RBF reconstruction - SSIM = 0.78397
250
200
150
100
50
0

Figure 4.7: Example of comparison on the coronal planes between a PET image and its cubic RBF reconstruction.

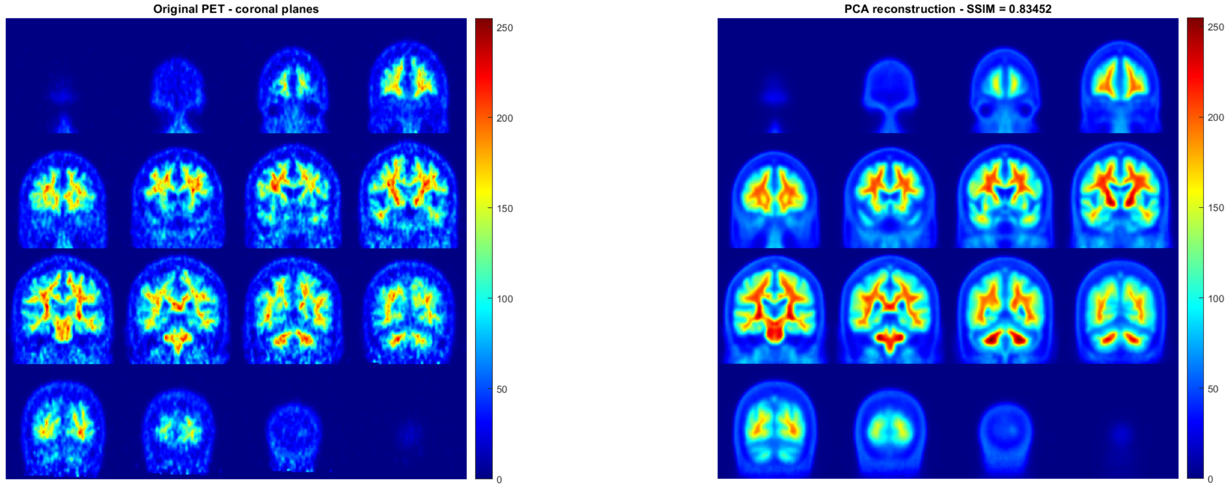

Original PET - coronal planes
250
200
150
100
50
0
PCA reconstruction - SSIM = 0.83452
250
200
150
100
50
0

Figure 4.8: Example of comparison on the coronal planes between a PET image and its PCA reconstruction.

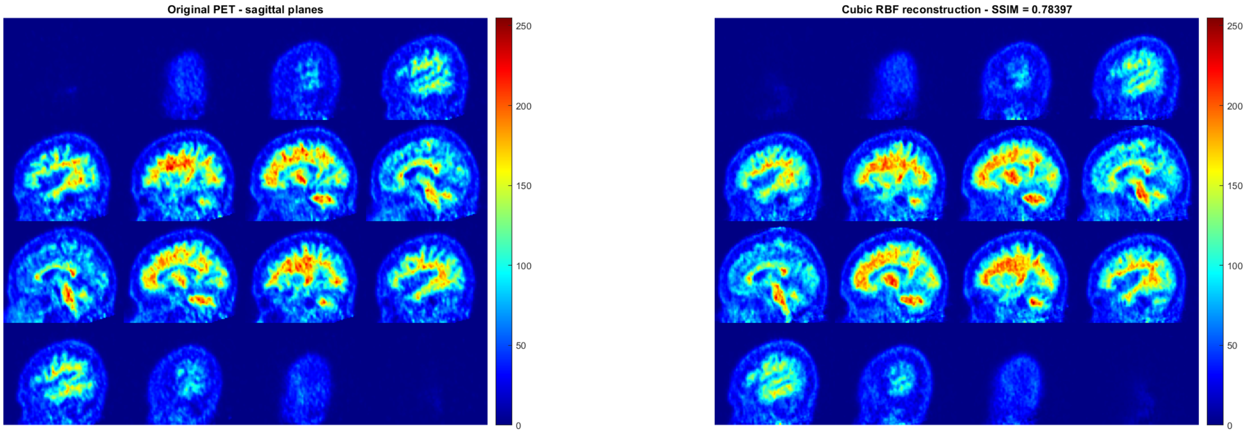

Original PET - sagittal planes
Cubic RBF reconstruction - SSIM = 0.78397
250
200
150
100
50
0

Figure 4.9: Example of comparison on the sagittal planes between a PET image and its cubic RBF reconstruction.

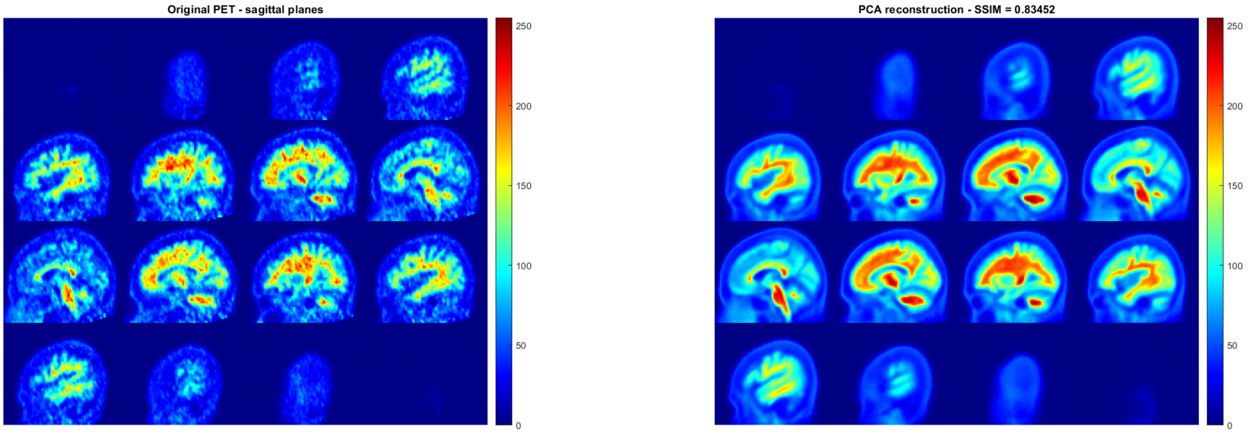

Original PET - sagittal planes
PCA reconstruction - SSIM = 0.83452
250
200
150
100
50
0

Figure 4.10: Example of comparison on the sagittal planes between a PET image and its PCA reconstruction.

Table 4.1: Measures of accuracy among the true labels and the raters

	Labels	Rater 1	Rater 2	Rater 3	Rater 4
Labels	1	0.4	0.4	0.75	0.5
Rater 1	0.4	1	0.4	0.55	0.6
Rater 2	0.4	0.4	1	0.15	0.6
Rater 3	0.75	0.55	0.15	1	0.45
Rater 4	0.5	0.6	0.6	0.45	1

Table 4.2: Measures of Cohen's kappa coefficient among the true labels and the raters

	Labels	Rater 1	Rater 2	Rater 3	Rater 4
Labels	1	-0.2	-0.2	0.5	0
Rater 1	-0.2	1	-0.2	0.06	0.2
Rater 2	-0.2	-0.2	1	-0.7	0.2
Rater 3	0.5	0.06	-0.7	1	-0.1
Rater 4	0	0.2	0.2	-0.1	1

indicates the number of identical assessments of the raters over the total number of images.

Another measure of the agreement between the raters and the true labels is provided by the Cohen's kappa coefficient, whose values for this test are shown in Table 4.2; this measure of inter-rater reliability takes into account the possibility of the agreement occurring by chance to correct the measures of accuracy. Values near to 1 indicate a high level of concordance; the results obtained, instead, show low, and also negative, values of Cohen's kappa coefficient, suggesting an agreement worse than that expected by chance [24].

Finally, given the set of assessments of each rater, Fisher's exact test can be adopted in order to test the significance of each rater assessments under the statistical hypothesis that the original and synthetic PET images are distinguishable; the null hypothesis is that the images are indistinguishable. The measures of the p-value of this null-hypothesis significance testing are shown in Table 4.3. P-values under the null-hypothesis are very high; only one rater comes close to the 0.05 threshold of

Table 4.3: Measures of Fisher's exact test (pvalue) for each rater

	p-value
Rater 1	0.62848
Rater 2	0.65628
Rater 3	0.069779
Rater 4	1

significance; therefore, the null hypothesis cannot be rejected for any rater.

The outcome of the validation test on the PET images is therefore that there are no significant agreements between the raters and the true labels and among the raters, meaning that original and synthetic images are indistinguishable.

An example of a synthetic PET image, drawn from within the negative sample region of Isomap representation, is shown in Figures 4.11 - 4.16 with the two neighbouring negative samples that were chosen at random, from which the synthetic image was interpolated.

A synthetic PET image generated instead from within the positive sample region is shown in Figures 4.17 - 4.22 with the two interpolation positive neighbours.

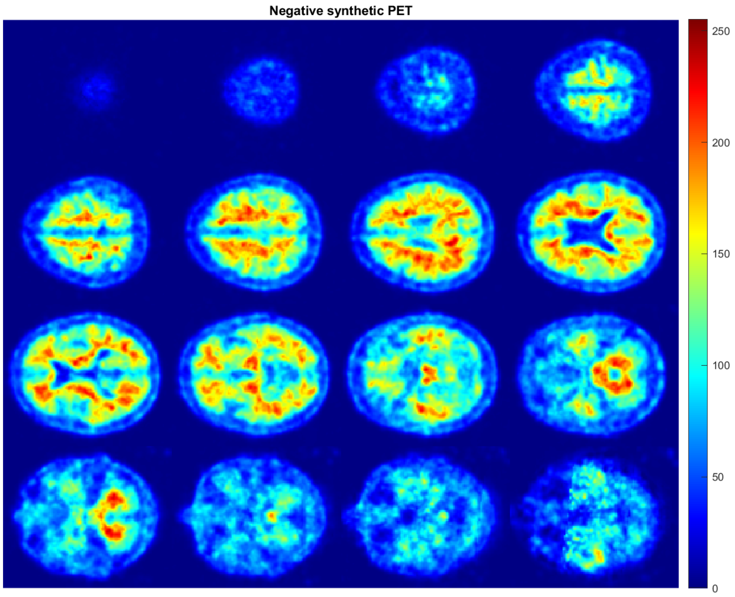


Figure 4.11: Synthetic negative PET image on the brain axial planes.

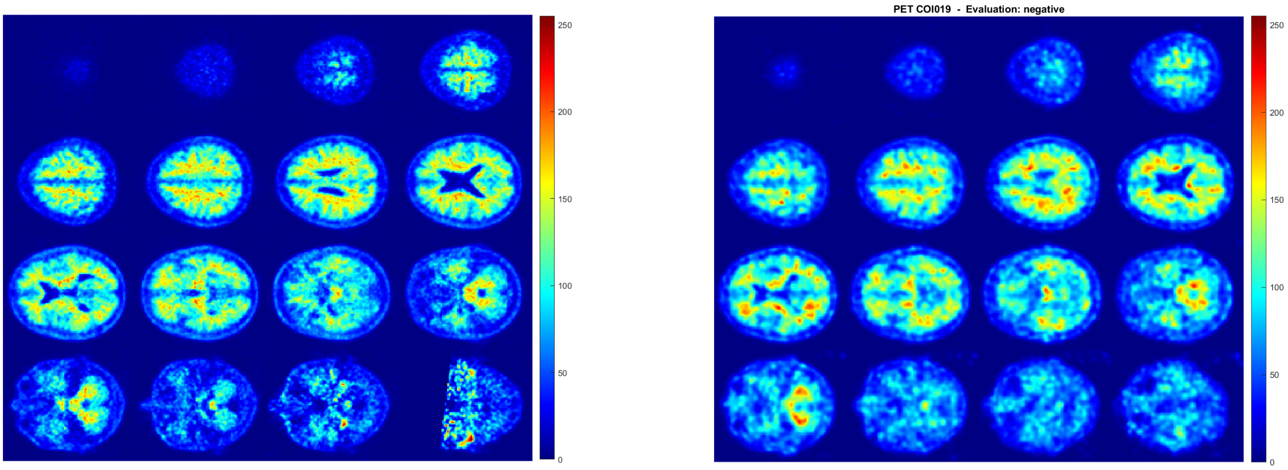


Figure 4.12: The two neighbouring negative samples of interpolation on the brain axial planes.

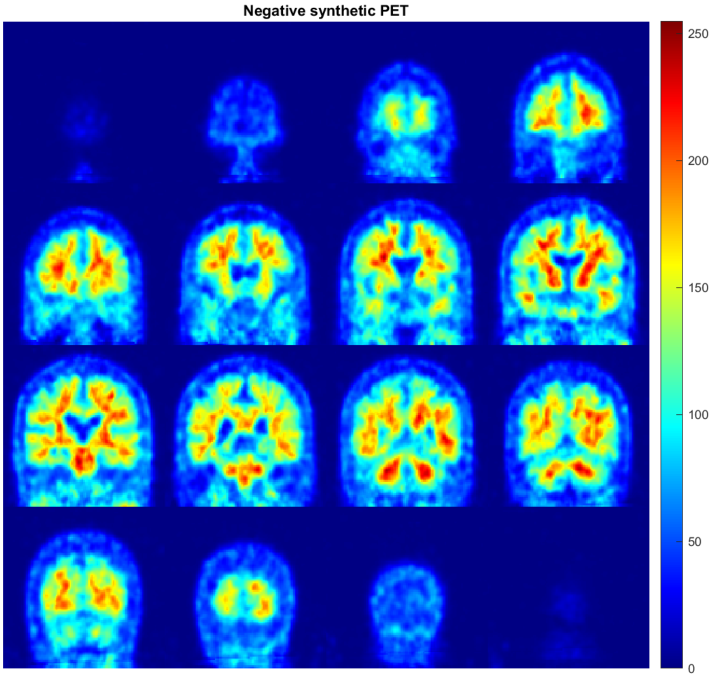


Figure 4.13: Synthetic negative PET image on the brain coronal planes.

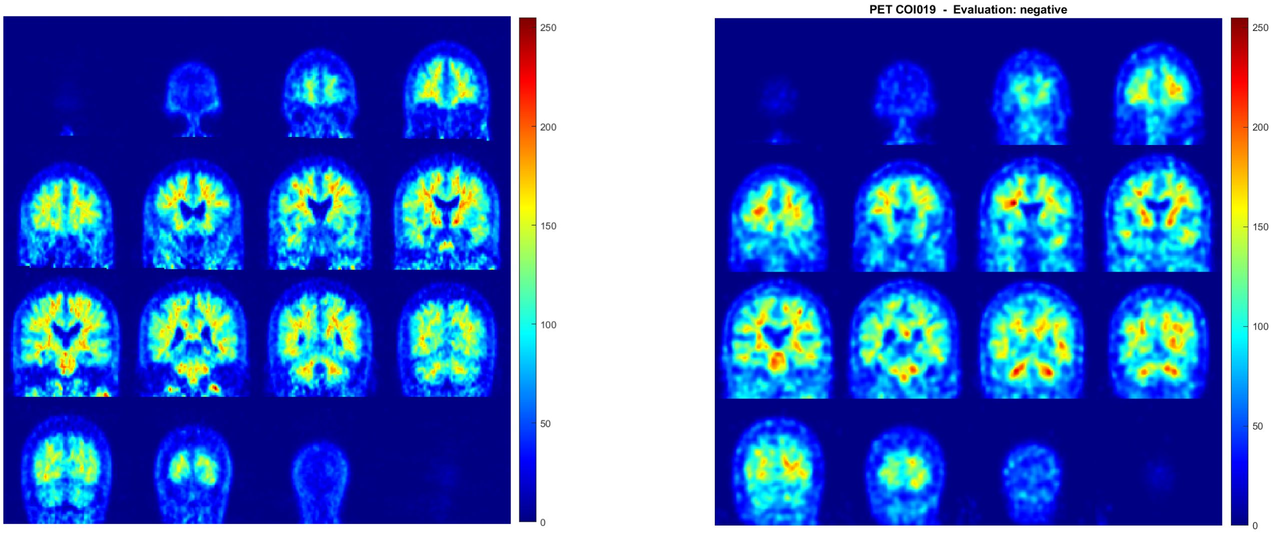


Figure 4.14: The two neighbouring negative samples of interpolation on the brain coronal planes.

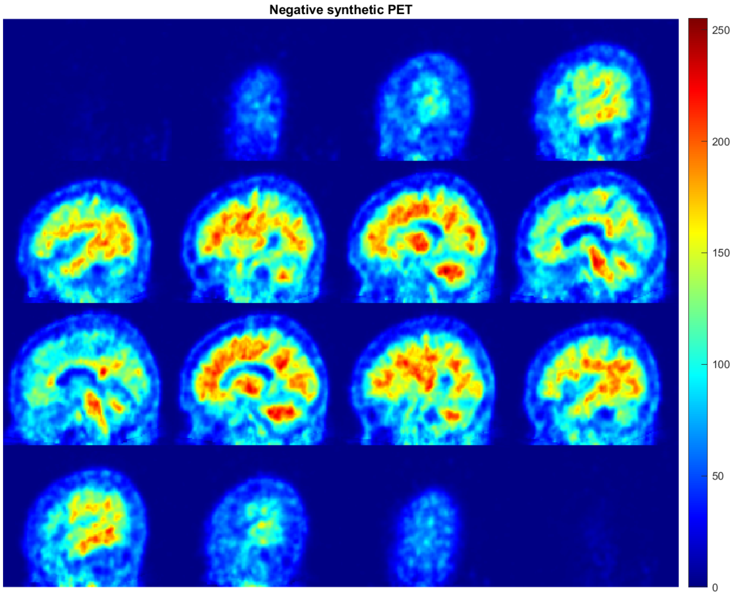


Figure 4.15: Synthetic negative PET image on the brain sagittal planes.

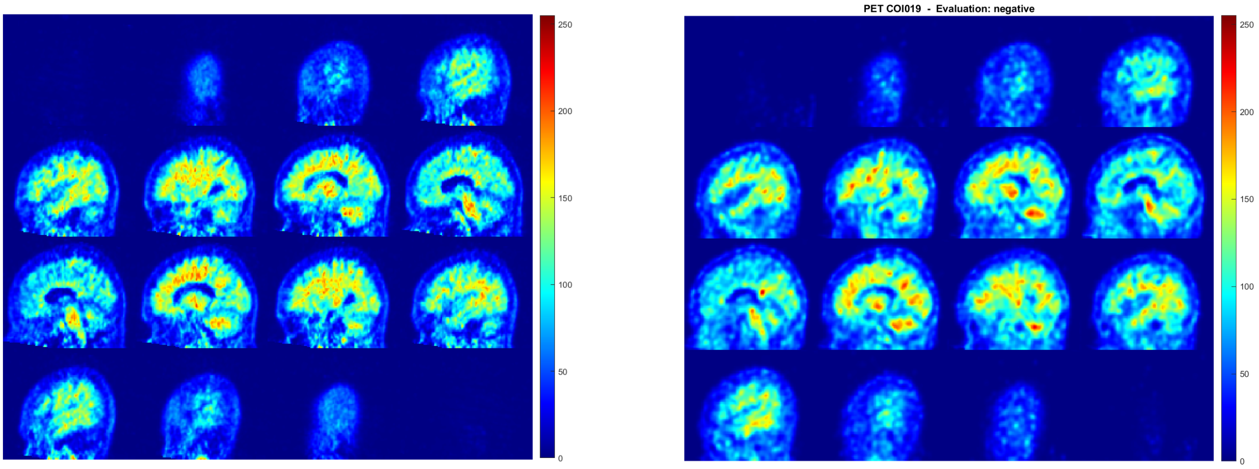


Figure 4.16: The two neighbouring negative samples of interpolation on the brain sagittal planes.

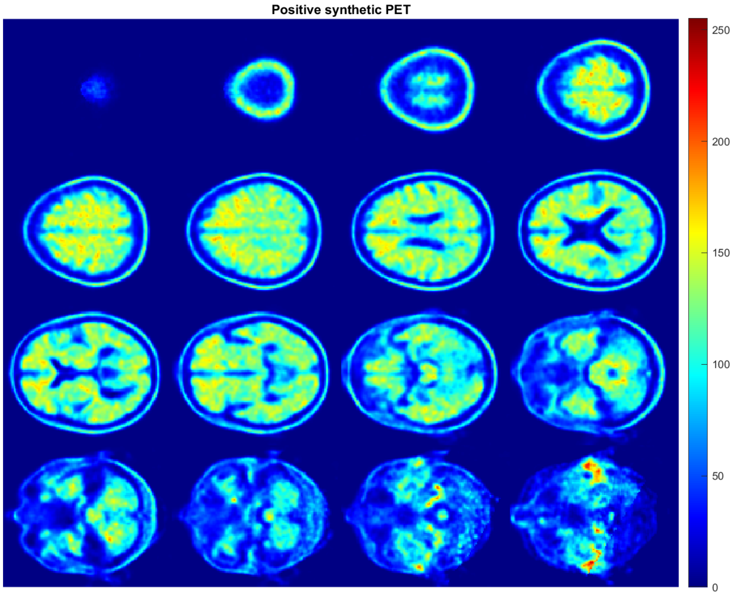


Figure 4.17: Synthetic positive PET image on the brain axial planes.

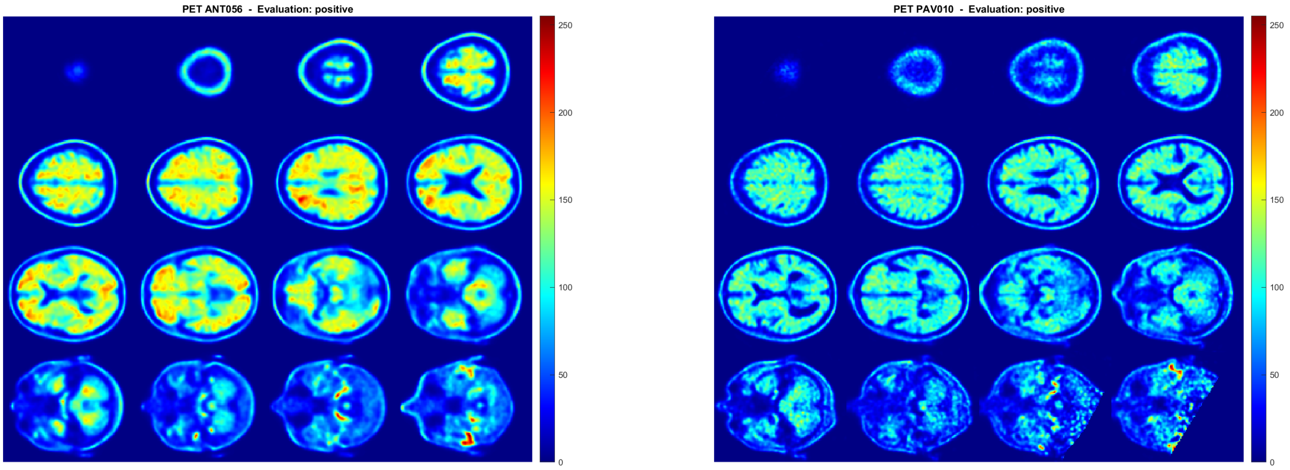

PET ANT056 - Evaluation: positive
250
200
150
100
50
0
PET PAV010 - Evaluation: positive
250
200
150
100
50
0

Figure 4.18: The two neighbouring positive samples of interpolation on the brain axial planes.

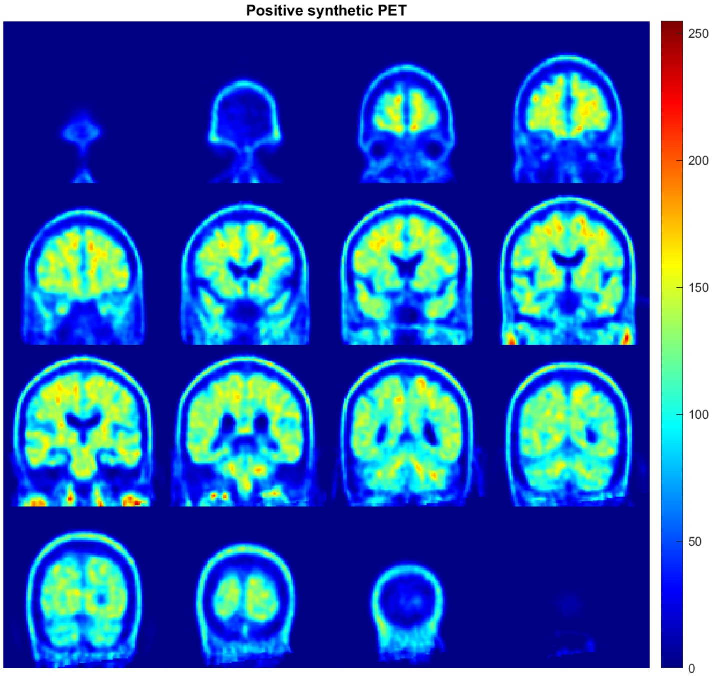


Figure 4.19: Synthetic positive PET image on the brain coronal planes.

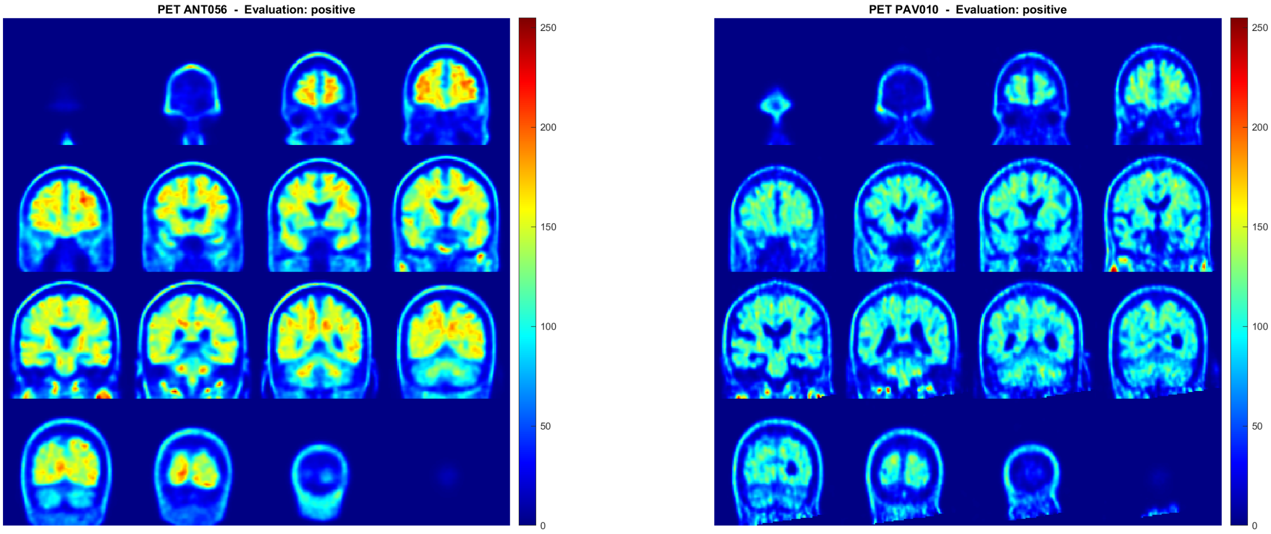

PET ANT056 - Evaluation: positive
250
200
150
100
50
0
PET PAV010 - Evaluation: positive
250
200
150
100
50
0

Figure 4.20: The two neighbouring positive samples of interpolation on the brain coronal planes.

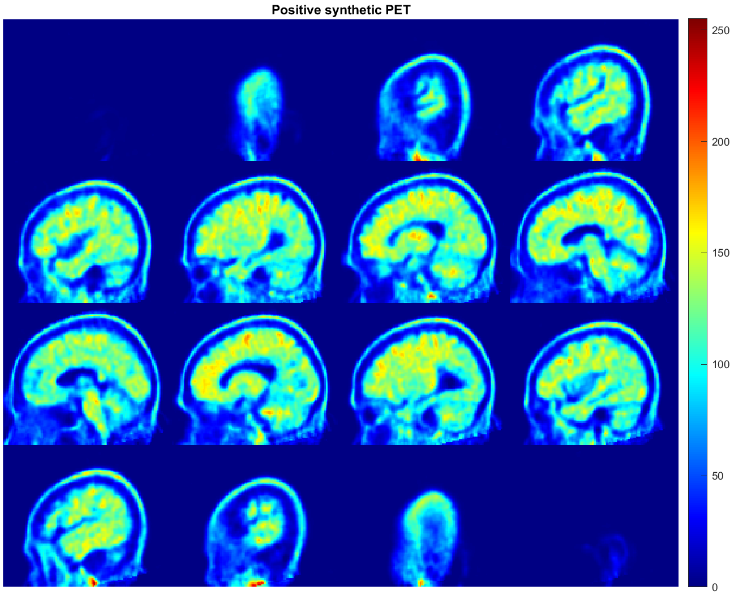


Figure 4.21: Synthetic positive PET image on the brain sagittal planes.

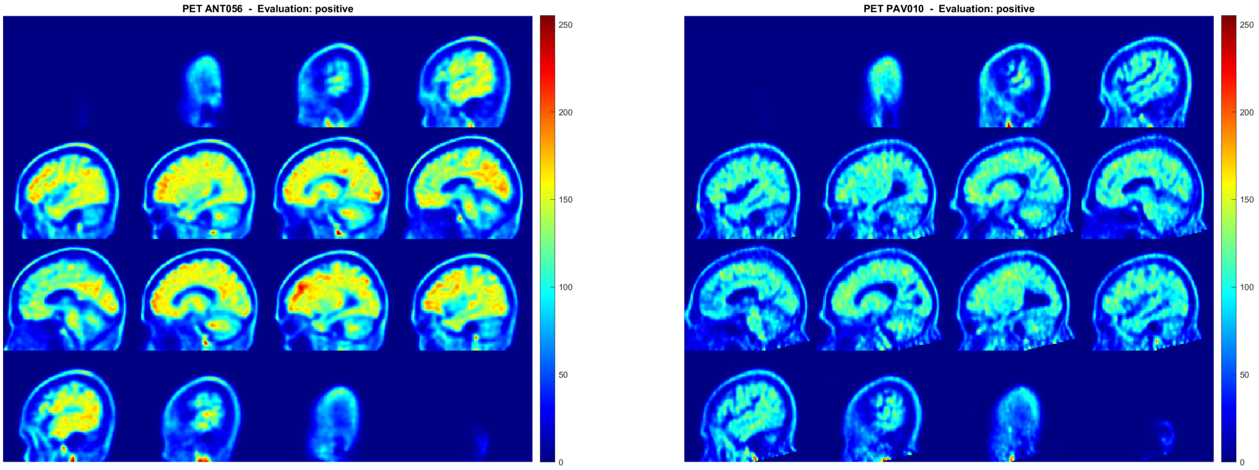

PET ANT056 - Evaluation: positive
250
200
150
100
50
0
PET PAV010 - Evaluation: positive
250
200
150
100
50
0

Figure 4.22: The two neighbouring positive samples of interpolation on the brain sagittal planes.

Chapter 5

Conclusions

The adoption of a scale-free cubic RBF interpolant was found to be effective for the inversion of the Isomap dimensionality reduction mapping. Reconstructed PET images were visually similar to the originals, at least with respect to the ones provided by the inversion of PCA. The measures adopted to define the similarity between images did not prove useful, suggesting the search for new methods to assess image reconstruction quality.

Realistic synthetic images could be generated, passing the clinicians' assessment test, which confirmed the indistinguishability between original and synthetic images.

The results of this thesis work point to the improvement of the research framework in the amyloid-beta PET field. Legal constraints arising from privacy issues about management of sensitive data could be overcome; at the same time, available data could be increased considerably without the restriction of data acquisition limitations, potentially improving other research fields where the same kind of data-wise issues are present.

Refinements in obtained results could be achieved through improvements of manifold learning and inverse mapping stages during PET image analysis, by exploring different combinations in the choice of algorithm parameters and by applying other non-linear dimensionality reduction algorithms.

Finally, a future work could be to built a classifier capable of distinguish between positive and negative PET images, to be trained on synthetic images generated from within the bulk of the two classes and from the boundary separating them; the next step would then be to search for characteristics of the PET images that associate them to the two classes, by inspecting their variability when moving from one class to the other.

Bibliography

[1] Joshua B. Tennenbaum, Vin de Silva, and John C. Langford. “A global geometric framework for nonlinear dimensionality reduction”. In: *Science* 290.5500 (2000), pp. 2319–2323.

[2] Karl Pearson. “LIII. On lines and planes of closest fit to systems of points in space”. In: *The London, Edinburgh, and Dublin philosophical magazine and journal of science* 2.11 (1901), pp. 559–572.

[3] Harold Hotelling. “Analysis of a complex of statistical variables into principal components.” In: *Journal of educational psychology* 24.6 (1933), p. 417.

[4] Warren S Torgerson. “Multidimensional scaling: I. Theory and method”. In: *Psychometrika* 17.4 (1952), pp. 401–419.

[5] Joseph B. Kruskal. “Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis”. In: *Psychometrika* 29.1 (1964), pp. 1–27.

[6] Sam T Roweis and Lawrence K Saul. “Nonlinear dimensionality reduction by locally linear embedding”. In: *science* 290.5500 (2000), pp. 2323–2326.

[7] Mikhail Belkin and Partha Niyogi. “Laplacian eigenmaps and spectral techniques for embedding and clustering”. In: *Advances in neural information processing systems* 14 (2001).

[8] Mikhail Belkin and Partha Niyogi. “Laplacian eigenmaps for dimensionality reduction and data representation”. In: *Neural computation* 15.6 (2003), pp. 1373–1396.

[9] Laurens Van der Maaten and Geoffrey Hinton. “Visualizing data using t-SNE.” In: *Journal of machine learning research* 9.11 (2008).

[10] Jian Tang et al. "Visualizing large-scale and high-dimensional data". In: *Proceedings of the 25th international conference on world wide web.* 2016, pp. 287–297.

[11] Leland McInnes, John Healy, and James Melville. "Umap: Uniform manifold approximation and projection for dimension reduction". In: *arXiv preprint arXiv:1802.03426* (2018).

[12] Mikhail Belkin and Partha Niyogi. "Semi-supervised learning on Riemannian manifolds". In: *Machine learning* 56.1 (2004), pp. 209–239.

[13] Philippe H. Robert et al. "Grouping for behavioral and psychological symptoms in dementia: clinical and biological aspects. Consensus paper of the European Alzheimer disease consortium". In: *European Psychiatry* 20.7 (2005), pp. 490–496.

[14] European Commission. *European alzheimer's disease consortium (EADC).* Accessed 17 November 2022. `https://cordis.europa.eu/project/id/QLK6-CT-2001-30003`.

[15] Keshav Anand and Marwan Sabbagh. "Amyloid imaging: poised for integration into medical practice". In: *Neurotherapeutics* 14.1 (2017), pp. 54–61.

[16] Nathan D. Monnig, Bengt Fornberg, and Francois G. Meyer. "Inverting nonlinear dimensionality reduction with scale-free radial basis interpolation". In: *arXiv preprint arXiv:1305.0258* (2013).

[17] Robert W. Floyd. "Algorithm 97: shortest path". In: *Communications of the ACM* 5.6 (1962), p. 345.

[18] K.V. Mardia, J.T. Kent, and J.M. Bibby. *Multivariate analysis, 1979.* 1979.

[19] Gregory E. Fasshauer. *Meshfree approximation methods with MATLAB.* Vol. 6. World Scientific, 2007.

[20] Utkarsh Trivedi. *Isomap(D, n_fcn, n_size, options).* MATLAB Central File Exchange. Retrieved 19 November 2022. `https://www.mathworks.com/matlabcentral/fileexchange/62449-isomap-d-n_fcn-n_size-options`. 2022.

[21] Zhou Wang et al. “Image quality assessment: from error visibility to structural similarity”. In: *IEEE transactions on image processing* 13.4 (2004), pp. 600–612.

[22] Nitesh V. Chawla et al. “SMOTE: synthetic minority over-sampling technique”. In: *Journal of artificial intelligence research* 16 (2002), pp. 321–357.

[23] Qiong Gu, Zhihua Cai, and Li Zhu. “Classification of imbalanced data sets by using the hybrid re-sampling algorithm based on isomap”. In: *International symposium on intelligence computation and applications*. Springer. 2009, pp. 287–296.

[24] Julius Sim and Chris C. Wright. “The kappa statistic in reliability studies: use, interpretation, and sample size requirements”. In: *Physical therapy* 85.3 (2005), pp. 257–268.