Expert Systems With Applications 211 (2023) 118633

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Alzheimer's disease classification based on graph kernel support vector machines constructed with 3D texture features extracted from magnetic resonance images

Lucas José Cruz de Mendonça, Ricardo José Ferrari *, for the Alzheimer's Disease Neuroimaging Initiative[1]

Department of Computing, Federal University of São Carlos, Rod. Washington Luis, Km 235, 13565-905, São Carlos, SP, Brazil

ARTICLE INFO

Keywords:
Graph-based classification
Texture extraction
Graph kernel
Alzheimer's disease
Magnetic resonance image

ABSTRACT

Alzheimer's disease (AD) is a neurodegenerative disease characterized by cognitive and behavioral impairment that significantly interferes with social and occupational functioning. Mild cognitive impairment (MCI) is a relatively broad clinical condition involving a slight memory deficit, which in many cases represents a transitional state between a cognitively normal (CN) condition and AD. Structural magnetic resonance (sMR) imaging has been widely used in studies related to AD because it provides images with excellent anatomical details and information about structural and contrast changes induced by the disease in the brain. Many published studies restrict their analysis to a few particular regions of the brain and search for structural changes caused by the disease. Recent studies start looking for new AD biomarkers using multiple brain regions and focusing on subtle texture changes in the image. Therefore, this study proposes a new technique for MR image classification in AD diagnosis using graph kernels constructed from texture features extracted from sMR images. In our method, we first segment the MR brain images into multiple regions with the FreeSurfer. Then, we extract 22 texture features using three methods and define the graph-node attributes as the probability distributions of the extracted features. Next, for each texture feature, we build a graph and define its edge weights as the distances between pairs of node attributes using three distance metrics. After that, we use a threshold-based approach for graph edges removal and create the graph-kernels matrices. Finally, we perform image classification using Support Vector Machines (SVMs) with two graph-kernels. Results of our method have shown better performances for the CN×AD (AUC = 0.92) and CN×MCI (AUC = 0.81) classifications, and worse for the MCI×AD case (AUC = 0.78). This trend is consistent with other published results and makes sense if we consider the concept of Alzheimer's disease continuum from pathophysiological, biomarker and clinical perspectives. Besides allowing the use of different texture attributes for the diagnosis of Alzheimer's, our method uses the graph-kernel approach to represent texture features from different regions of the brain image, which considerably facilitates the image classification task via SVMs. Our results were promising when compared to the state-of-the-art in graph-based AD classification.

1. Introduction

Alzheimer's disease (AD) is a progressive and irreversible brain disease that slowly degrades brain functions. AD is the most frequent form of dementia that affects over 47 million people worldwide and this number may triple by 2050 (Livingston et al., 2017). Since this disease has currently no cure, the early diagnosis is essential so that the patient can begin a treatment to slow the worsening of symptoms. Mild cognitive impairment (MCI) is a relatively broad clinical condition involving a slight memory deficit, which most times represents a transitional state between a cognitively normal (CN) condition and AD (Morris et al., 2001).

The neuropathological diagnosis of AD depends on the presence of both neurofibrillary tangles and senile plaques. The number of

* Corresponding author.
E-mail addresses: lucasjose-ap@hotmail.com (L.J.C. de Mendonça), rferrari@ufscar.br (R.J. Ferrari).
URL: http://www.bipgroup.dc.ufscar.br (R.J. Ferrari).

[1] Data used in preparation of this article were obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (adni.loni.usc.edu). As such, the investigators within the ADNI contributed to the design and implementation of ADNI and/or provided data but did not participate in analysis or writing of this report. A complete listing of ADNI investigators can be found at: http://adni.loni.usc.edu/wp-content/uploads/how_to_apply/ADNI_Acknowledgement_List.pdf.

https://doi.org/10.1016/j.eswa.2022.118633
Received 28 December 2021; Received in revised form 14 June 2022; Accepted 17 August 2022
Available online 24 August 2022

neurofibrillary tangles is tightly linked to the degree of dementia, suggesting that the formation of neurofibrillary tangles directly correlates with neuronal dysfunction (Butterfield & Halliwell, 2019; Martins et al., 2018). In Alzheimer, the regional pattern of areas affected by the formation of neurofibrillary tangles during the course of the disease is relatively known and not difficult to predict.

Recent studies related to AD biomarkers (Dubois et al., 2014; McKhann et al., 2011) recommend the use of the measurements of cerebrospinal fluid (CSF), amyloid-β, tau and neuronal injury biomarkers to aid the diagnosis of AD. Some of these biomarkers can be obtained from neuroimaging techniques such as Magnetic Resonance Imaging (MRI) and Positron Emission Tomography (PET). In addition to imaging biomarkers, in-home sensing of daily living patterns acquired from older adults, used as input to machine learning models, has also shown great success to detect MCI cases (Teh, Rawtaer, & Tan, 2022).

MRI is usually the preferred imaging technique for structural brain analysis in AD, as it provides images with excellent soft tissue differentiation and good spatial resolution, with no known health risks. Besides the structural information, the formation of neurofibrillary tangles and the deposit of senile plaques can leave certain gray valued patterns, which may not be visible to the human eye but can be assessed from texture algorithms (Castellano, Bonilha, Li, & Cendes, 2004; Sørensen et al., 2016).

Several studies have been proposed in the literature aiming to catch the subtle gray-level intensity patterns created by soft tissue diseases, such as hepatic (Albu, Precup, & Teban, 2019) and AD progression (Bustamam, Sarwinda, & Ardenaswari, 2018; Liu, Wang, Hu, & Pan, 2017; Zhang, Yu, Jiang, Liu, & Tong, 2012). The methods commonly used for texture extraction are the Gray Level Co-Occurrence Matrix (GLCM) (Haralick, Shanmugam, & Dinstein, 1973), the Run Length Matrix (RLM) (Galloway, 1974) and the Local Binary Patterns (LBP) (Ojala, Pietikäinen, & Harwood, 1996). These methods work as structural-based texture feature extractors, since they analyze the spatial relationship between a pixel (or voxel) and its neighborhood (Cai et al., 2020). Other less common, but no less important approaches to texture analysis in AD use transform-based methods to assess the oriented spatial-frequency content of an image (Feng, Zhang, & Chen, 2020; Nanni et al., 2019).

In our previous studies we have mainly assessed structural changes of the hippocampal regions of the brain (Araújo, Poloni, & Ferrari, 2021; Cambui, Poloni, & Ferrari, 2021; Oliveira, Poloni, & Ferrari, 2020; Poloni & Ferrari, 2022a, 2022b; Silveira Souza, Poloni, & Ferrari, 2020). Although AD is well known to damage the hippocampal subfields (Hett et al., 2019; Sørensen et al., 2016), the neurodegenerative effects of the disease are not restricted to these structures. In fact, the effects of the disease are found to be widespread in the brain. Liu et al. (2017), for instance, extracted texture feature maps with the GLCM method of ninety segmented brain regions for AD classification. The average gray levels of the six extracted feature maps for each region were calculated and used to create a six-position vector to represent each brain region. To combine the information of all segmented regions, the authors built a graph using the texture feature vectors to represent its nodes and used the Pearson correlation measures of pairs of feature nodes to represent its edges. The classification was performed independently for the concatenated feature nodes and edges using multiple MKBoost (Xia & Hoi, 2012) classifiers.

Hett, Ta, Manjón, and Coupé (2018) used a graph to represent image features extracted from 134 brain regions. In their work, first, inter-subject similarities were captured by using a patch-based grading framework (PBG) (Coupé et al., 2012) applied over the entire brain images of a training dataset composed of CN and AD subjects. Second, intra-subject variability was modeled by a graph representation. The study compared the performance of intra-subject variability features (i.e., the edges of the graph) with inter-subject pattern similarity features (i.e., the vertices).

Recently, Hett et al. (2021) used graphs to represent image features extracted from 133 brain regions plus five hippocampal subfields. In their work, the graphs were constructed by using the attributes extracted with the PBG method for each region. Two graphs of each image were constructed, one with the 133 brain regions and the other with the five hippocampal subfields. A novelty in their work includes the results of five cognitive tests for each patient in the classification. The classification was performed for each graph independently (whole-brain and subfields) as an ensemble of intermediate Random Forest classifiers, where their results were unified with cognitive scores.

Unlike previous works based on the traditional graph approach, some studies in brain image classification (Cui et al., 2018; Jie, Liu, Jiang, & Zhang, 2016; Jie, Liu, Zhang, & Shen, 2018) use kernel-based graph classification. Graph kernels (Kriege, Johansson, & Morris, 2020b) is an approach that aims to learn directly from graph-structured data by using kernel functions to measure the similarity between graphs. In this approach the graphs can be plugged into a kernel machine, such as a support vector machine (SVM) (Boser, Guyon, & Vapnik, 1992; Cristianini, Shawe-Taylor, et al., 2000), and the posted problem can now be treated as a conventional classification task.

More recently, some studies have explored graph-based approaches with deep learning to identify the relationship between early-MCI and late-MCI images. Song, Elazab, and Zhang (2020), for instance, proposed a method based on a combination of high-order network and graph convolutional network (GCN). In their work, high-order network combined static, dynamic and high-level information to construct functional connectivity network (FCN) while GCN included non-image information to improve classifier's performance. Features of the combined high-order FCNs were extracted by using a recursive feature elimination method and the results were inputted into the GCN, in which MCI-graph establishes interactions between individuals and populations by using non-image information, and finally the GCN outputs the binary classification result. Liu, Tan, Lan, and Wang (2020) propose a new method for identification of early MCI (eMCI) cases using multi-modal data and GCNs. First, the authors performed image preprocessing and feature representation for both T1-weighted (T1-w) MRI and resting state functional MRI (rs-fMRI) data of each subject. Then, a multi-task feature selection method was used to obtain features that are more helpful in identifying eMCI. After that, they constructed a subject graph using imaging phenotypic measures and non-imaging phenotypic measures of each subject. Finally, a GCN model was applied to perform the eMCI identification task.

In this study, we propose a method based on graph kernels constructed from image texture features for AD diagnosis. We extract multiple texture features of 92 segmented brain regions using the GLCM, RLM and LBP algorithms. The graphs are constructed by considering the brain regions, represented by either probability distributions of texture feature maps or statistical moments and brain region volume as node attributes and the distance between the node attributes as edges. Three different distance metrics defining the graph edges were investigated. To learn the differences between the graph representations of the MR images in the classification of CN×MCI, CN×AD and MCI×AD cases, we use graph-based kernels and SVM classifiers. In our classification scheme, we selected the best combination of graph kernel, texture feature, and edge distance metric and introduced a new threshold-based approach for graph edges selection to help determine the most discriminatory region connections of the graphs.

In summary, our contributions to 3D MR image classification in Alzheimer's diagnostic include: (i) development of a new method based on graph kernels that allows using various 3D image texture attributes, (ii) evaluation of three texture filters and two feature extraction methods and demonstration of their capabilities for classification, (iii) development of a threshold-based method to remove edges from graphs to help select the most discriminating model, (iv) assessment of three different distances to compare probability distributions, and (v) exhaustive analysis of the proposed method using a significant number of MR images.

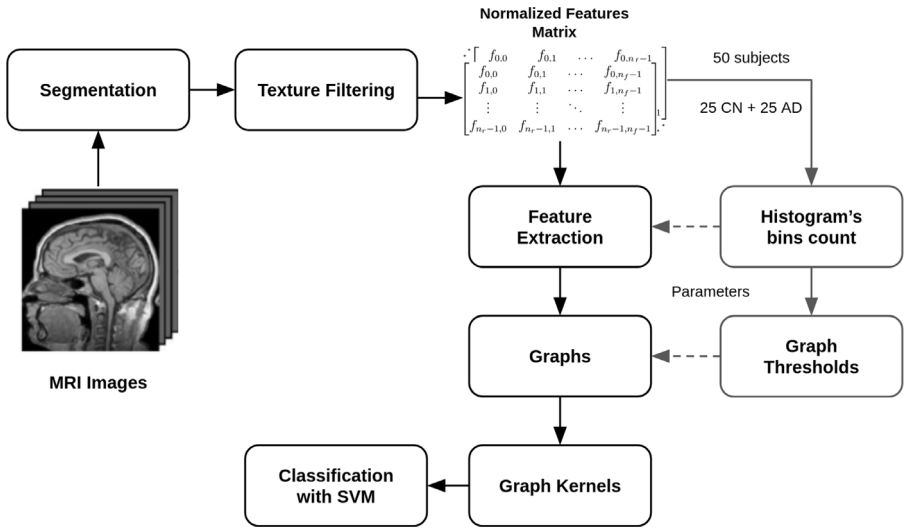

Fig. 1. Overview of our method.

Table 1
Demographic information of the 474 subjects extracted from the ADNI database. The MMSE stands for Mini-Mental State Examination.

	CN	MCI	AD
Number of subjects	200	153	121
Age (mean ± deviation)	73.7 ± 4.2	76.6 ± 4.2	76.9 ± 4.2
MMSE (mean ± deviation)	29.5 ± 0.4	26.9 ± 0.8	21.9 ± 2.6
Gender (F : M)	101 : 99	57 : 96	55 : 66

2. Material and methods

In this section, we describe in detail each step of the processing pipeline as shown in Fig. 1 and provide information about the MR image dataset used in our experiments and how we split the data into two groups to estimate the parameters of our methods. In addition, we describe our methods for extracting texture features, constructing brain graphs, removing graph edges with a threshold-based approach, and finally classifying the MR images.

2.1. Dataset

The MR images used in this study are from the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset (Jack et al., 2008), which is composed of MR and positron emission tomography (PET) images along with clinical test scores. In this study, we used a total of 474 structural MR T1-w images (200 CN, 153 MCI and 121 AD) from a wide variety of 1.5T and 3T scanners, all acquired using the Magnetization Prepared Rapid Gradient Echo (MPRAGE) protocol. Table 1 presents the demographic information of individuals. Furthermore, fifty images (25 CN and 25 AD) were randomly selected to form a data subset to estimate the parameters of the developed methods.

2.2. Segmentation of brain regions

In this study we segmented 105 brain regions using an atlas-based approach provided by the FreeSurfer[2] software suite to build the graphs. The main pipeline of this suite applies the required preprocessing steps and resample the study images to $256 \times 256 \times 256$ to match the image space used by the FreeSurfer segmentation approach. After that, the brain regions are extracted using the FreeSurfer's subcortical segmentation (ASEG) and Desikan–Killiany–Tourville (DKT) atlases (Fischl et al., 2004; Klein & Tourville, 2012). In this case, the DKT atlas acts as a complementary ASEG-atlas by providing brain regions known to be related with the AD progression, such as the entorhinal cortex. In addition, it has been validated for its capability to estimate brain region volumes when compared to other atlases (Yaakub et al., 2020). As a complementary procedure, we evaluated all segmented regions of our database and removed those whose voxel numbers were outside of a defined range, $[N_{min}, N_{max}]$, where N_{min} is the average number of voxels of the entorhinal cortex and N_{max} was defined as 25,000 voxels. This procedure that reduced the number of regions from 105 to 92 was used to avoid considering tiny and huge brain structures which could affect our analysis.

2.3. Texture filters

After the segmentation step, we extracted 22 texture features, $F_1, \ldots, F_{22}$, from each segmented brain region defined by a binary brain mask generated by the FreeSurfer. For the texture feature extraction we used the GLCM, RLM and LBP techniques. Since the features are computed directly from the image intensity values, we decided to use the least processed image generated by FreeSurfer pipeline while maintaining its alignment with segmented region of interest (ROI) masks.

2.3.1. Gray level co-occurrence matrix

The GLCM technique (Haralick et al., 1973) extracts image texture features from the second order statistics of the spatial relationship between image intensity values, which can be seen as the co-occurrence of pairs of pixels/voxels that are at a particular distance and orientation from each other. In this study we used the GLCM method from the Insight Toolkit (ITK) library (McCormick, Liu, Ibanez, Jomier, & Marion, 2014; Vimort, McCormick, Budin, & Paniagua, 2017) that implements eight texture features, $F_1, F_2, \ldots, F_8$. More details on the implementation of these features are available in the Appendix A. We used the library's default parameters, which for 3D images are set to thirteen orientations and voxel distance equal to one.

2.3.2. Run length matrix

The RLM technique (Galloway, 1974) generates a matrix $P_\theta(i, l)$ containing the number of runs of pixels/voxels that exists in an image for a gray value i and length l in a direction θ. The statistical parameters derived from this matrix can describe texture features in the sense of

[2] https://surfer.nmr.mgh.harvard.edu.

their coarseness, for long-runs, and finest, for short-runs. In this study we used ten texture features implemented in the ITK library that are derived from thirteen directions and run length of one. The definition of all eighteen texture features obtained using the GLCM and RLM methods implemented in the ITK library can be found in (Vimort et al., 2017). Taking into account the eight GLCM texture features sequence, we define the RLM texture features, $F_9, F_{10}, \ldots, F_{18}$, as described in the Appendix B.

2.3.3. Local binary patterns

The LBP method was first introduced by Ojala et al. (1996) as an efficient method for texture description in 2D images. In this study we used a 3D version of the LBP proposed by Banerjee, Moelker, Niessen, and Van Walsum (2012) which applies spherical harmonic functions in a sampling scheme to provide rotational invariant features. The 3D LBP method requires to adjust three parameters, the number of spherical harmonic levels (n_{shl}), the radius (r_n) in which the neighbors should be sampled, and the number of subdivisions (n_i) to apply in the icosphere. By changing these parameters, we can obtain different texture images. Based on experimental analysis using a data subset of fifty MR images (25 CN and 25 AD) as mentioned in Section 2.1, we found that the parameter values $n_{shl} = 3$, $r_n = 2$, and $n_i = 1$ provide the best compromise between high accuracy results and low computational cost. In addition to the texture images resulting using the parameters described above, the method provides a kurtosis image that is also incorporated in our analysis. Therefore, the LBP methods provides a total of four texture features, defined herein as F_{19} for the first level of harmonic spheres, F_{20} for the second level, F_{21} for the third, and F_{22} for the kurtosis.

2.4. Feature extraction

Since each processed MR image produces 22 filtered counterparts, we extracted and represented the texture features using two main approaches. The first approach represents each segmented region by a set of probability distributions (one distribution for each of the 22 texture features) obtained from the normalized texture feature histograms with a fixed number of bins. The distributions are used as node attributes and also serve to compute the edge weights between pairs of nodes. The second approach uses a vector composed of the first four statistical moments extracted from the texture maps of each segmented brain region and the corresponding brain region volume to represent the node attributes. For better understanding, we represent the feature matrix for each subject s as follows:

$$M^s = \begin{bmatrix} f_{0,0} & f_{0,1} & \cdots & f_{0,n_f-1} \\ f_{1,0} & f_{1,1} & \cdots & f_{1,n_f-1} \\ \vdots & \vdots & \ddots & \vdots \\ f_{n_r-1,0} & f_{n_r-1,1} & \cdots & f_{n_r-1,n_f-1} \end{bmatrix}, \tag{1}$$

where $n_r = 92$ is the number of segmented ROIs (please refer to Appendix C), $n_f = 22$ is the number of texture features, i.e., the matrix rows represent the segmented regions and its columns each texture feature. In this case, each element corresponds to a particular texture feature of a given segmented brain region. It is worthy mentioning that before the texture feature extraction, we perform a min–max normalization per feature of all analyzed brain regions on all subjects (matrix columns normalization).

This matrix indicates how each subject is represented from this step onward, as shown in Fig. 1, and this is the starting point for the next two approaches for the graph attributes extraction. A visual representation of the required steps to obtain the feature matrix of Eq. (1) is shown of Fig. 2. As can be noticed, in the first step a raw image is preprocessed by FreeSurfer and resampled to a new spatial space. Next, binary masks of several segmented brain regions from two atlases are used to define the analyzed regions on the texture image maps obtained using the GLCM, RLM and LBP techniques. The information from the texture image maps are used to build a feature matrix that represents for each subject the texture features by brain regions.

2.4.1. Texture feature distributions

Defining the number and width of bins to construct the texture feature distributions of the segmented brain regions to allow computing the distance between distributions is an essential task, considering the distinct range of values of the texture maps. To facilitate the calculation of the statistical distances between the feature distributions of pairs of brain regions so that to define the graph edge weights, we applied a mathematical strategy to determine automatically a fixed number of bins for the representation.

The Freedman–Diaconis (FD) rule (Freedman & Diaconis, 1981), which was recently used for the discretization of texture features (Dieckmeyer et al., 2021; Noortman et al., 2020), is a robust estimator (i.e., resilient to outliers) that considers data variability and size. The rule to determine the optional number and width of bins for a distribution are defined as

$$\text{bin width} = \frac{2 \cdot \text{IQR}}{\sqrt[3]{N}}, \tag{2}$$

and

$$\#\text{bins} = \left\lfloor \frac{\max(I_{\text{ROI}}) - \min(I_{\text{ROI}})}{\text{bin width}} \right\rfloor, \tag{3}$$

where IQR is the mean interquartile range and N is the number of voxels in the assessed image ROI (I_{ROI}).

In our study, we first calculated the number of bins of the texture feature distributions using Eq. (3) for all segmented regions of our data subset of 50 images (25 AD and 25 CN). Next, for each texture attribute we average the number of bins from all regions and use the resulting value to build the histograms and distributions for the further graph analysis. Therefore, the elements of matrix M^s are defined as

$$\mathbf{f}_{i,j} = \left[probability\text{-}distribution\text{-}vector_{i,j} \right]. \tag{4}$$

In addition, it is performed the min–max normalization on the distributions per region over all features for each subject (matrix rows normalization).

2.4.2. Region volume and statistical moments

As another feature extraction strategy, we replaced the elements of matrix M^s with vectors composed of volume and the first four statistical moments extracted from the texture maps of each segmented brain region. Therefore, the elements of the new matrix are defined as

$$\mathbf{f}_{i,j} = \left[volume_i, mean_{i,j}, variance_{i,j}, skewness_{i,j}, kurtosis_{i,j} \right], \tag{5}$$

where i and j correspond to the indices of a segmented brain region and a texture feature map, respectively.

2.5. Brain graphs methodology

In order to capture the relationships between the texture patterns extracted from different brain regions of the MR images for the detection of Alzheimer's disease, in this study we used undirected attributed graphs $G = (V, E, a)$ without loops. In this case, $V = \{v_1, \ldots, v_N\}$ is the set of vertices for the N segmented brain regions, E is the set of weighted edges, with each edge defined as $(u, v) \in E$ for $u, v \in V$ and $a : V \to \mathbb{R}^D$ is a node attribute function that maps V to attribute vectors of dimension D.

For each image we extracted 22 texture features, as defined in Section 2.3, from all 92 brain regions (Section 2.2). For each texture feature we build two types of graphs. In the first, each node attribute, representing a given brain region, was defined by the distribution of the texture map of that region, while in the second the node attribute was defined by the vector composed of the first four gray level statistical moments and the region volume. Details of the graph construction are provided in the subsequent sections.

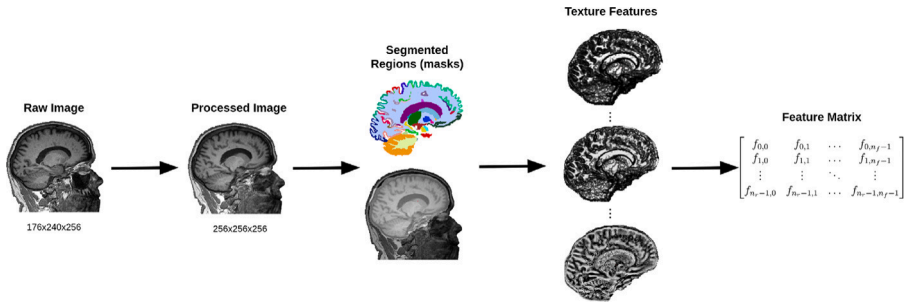

Fig. 2. Visual representation of the first steps shown in Fig. 1 for a single subject.

2.5.1. Node attributes

As mentioned previously, we have two approaches to build graph node attributes according to the feature extraction methods. For each texture feature, the node attributes are defined as:

1. The distribution of texture map values (Eq. (4)).
2. The feature vector composed of statistical moments and the region volume (Eq. (5)).

2.5.2. Edges and distances

The edge weights of our two types of graphs were defined by the distance between the two corresponding discrete nodes' attributes u and v of dimension d.

As we are using undirected graphs, we evaluate three distances that meet two crucial requirements, which are: symmetry and adequacy for comparing probability distributions. The two distances are described as follows.

Wasserstein distance (D_{ws}) (Rubner, Tomasi, & Guibas, 2000), also known as the earth mover distance, is a measure of distance between two probability distributions over a region D. In this study, we used the equivalent formulation proposed by Ramdas, Trillos, and Cuturi (2017), where the distance is defined as

$$D_{ws}(\mathbf{u},\mathbf{v}) = \sum |cdf(\mathbf{u}) - cdf(\mathbf{v})|, \tag{6}$$

where $cdf(\cdot)$ is the cumulative distribution function.

Kullback–Leibler distance (D_{kl}) (Kullback & Leibler, 1951), also known as relative entropy, is commonly used to measure the amount of information lost from one probability distribution to another. Although this distance is not symmetric, in order to obtain the symmetry property in this study we used the average distances between $D_{kl}(\mathbf{u},\mathbf{v})$ and $D_{kl}(\mathbf{v},\mathbf{u})$ as

$$D_{sym-kl}(\mathbf{u},\mathbf{v}) = \frac{1}{2}\left[D_{kl}(\mathbf{u} \parallel \mathbf{v}) + D_{kl}(\mathbf{v} \parallel \mathbf{u})\right], \tag{7}$$

where

$$D_{kl}(\mathbf{p} \parallel \mathbf{q}) = \sum_{x \in \mathcal{X}} \mathbf{p}(x) \ln \frac{\mathbf{p}(x)}{\mathbf{q}(x)}, \tag{8}$$

where x corresponds to an instance of the texture probability space $\mathcal{X}$.

Hellinger distance (D_{hg}), initially proposed by Hellinger (1909), is similar to the total variation distance, except that it resembles to the L2-norm. In its discrete form the distance is computed as

$$D_{hg}(\mathbf{u},\mathbf{v}) = \frac{1}{\sqrt{2}}\left\|\sqrt{\mathbf{u}} - \sqrt{\mathbf{v}}\right\|_2. \tag{9}$$

2.5.3. Threshold-based edge removal

When working with graphs constructed from images, one of the main challenges is how to construct discriminative graphs to be used in a classification task. In the literature, a handful of works use the Pearson correlation coefficient (Jie et al., 2016; Liu et al., 2016, 2017) to set a minimum threshold-value based on the p-value, while other works use the minimum spanning tree frequency between nodes to find the most frequent sub-network (Cui et al., 2018). One approach that is common in several works in the literature when working with brain graphs is to initiate with a complete graph, which is the case where all nodes are connected to each other, and then apply an edge removal method to obtain the most discriminative graph representation.

In this study we develop a threshold-based approach to graph edge removal that generates a threshold value for each texture feature.

Let G_{CN} and G_{AD} be two sets of complete graphs built from CN and AD MR image subjects, respectively, where the graphs have the same number of nodes and $G_{CN}^{(i)} \in G_{CN}$ and $G_{AD}^{(i)} \in G_{AD}$, for $i = 1, \ldots, n_s$, represent the ith graphs of each set. In this study $n_s = 25$ corresponds to the number of images selected for parameter estimation, as mentioned in Section 2.1.

Now, considering $\mathbf{d}_f \in \mathbb{R}^{n_s}$ a vector of the average weight distances between all graphs in the G_{AD} and G_{CN} sets (pairwise comparison) for a given texture feature f, with $f \in [0, \ldots, 21]$, then each vector element is calculated as

$$d_{f,i} = \frac{\sum_{e \in E}\left|w\left(G_{CN}^{(i)}, e\right) - w\left(G_{AD}^{(i)}, e\right)\right|}{|E|}, \tag{10}$$

where $e = (u, v) \in E$ is an edge formed by the nodes $u, v \in V$, $|E|$ is the size of the edge set, and $w : G, e \to \mathbb{R}$ is a weight function that receives a graph G and an edge e and outputs an edge weight, computed herein as the distance between the texture distributions of the nodes. In this case, any of the distances defined in Section 2.5.2 can be used.

To determine the most discriminative graph representation for our image classification task, we defined a threshold T_f for each texture feature f which is the mean value of distances between the diagnostic group pairwise comparison stored in the vector $\mathbf{d}_f$ as

$$T_f(n) = \mu_f + n \cdot \sigma_f, \tag{11}$$

where n is a multiplicative factor, and μ_f and σ_f are the mean and standard deviation, respectively, for an arbitrary texture feature f are obtained as

$$\mu_f = \frac{\sum_{i=1}^{n_s} d_{f,i}}{n_s}, \tag{12}$$

and

$$\sigma_f = \sqrt{\frac{\sum_{i=1}^{n_s}\left|d_{f,i} - \mu_f\right|^2}{n_s}}. \tag{13}$$

The optimal n values are determined experimentally and their values are provided with the results in Section 3. In our study, graph edges are removed when their distances are less than the average of all graph edges of the estimation dataset, which are computed independently to the classification subjects.

2.6. Classification model

In this section we describe our image classification model that is based on graph-kernels and SVM classifiers. First, in Section 2.6.1, we briefly present the main theoretical ideas of the SVM classifier.

Then, in Section 2.6.2 we provide details on the graph kernels used in this study. Finally, in Section 2.6.3 we discuss how the graph-kernels are integrated into the SVM classifier to create our graph-kernel SVM models and how the regularization parameter C is tuned in the cross-validation strategy.

2.6.1. SVM classifier

In this study we consider the standard binary classification problem, applied to each of the cases CN×AD, CN×MCI and MCI×AD, where training vectors $\mathbf{x}_i \in \mathbb{R}^n$, $i = 1, \dots, l$ from two classes and a label vector $\mathbf{y} \in \mathbb{R}^l$, such that $y_i \in \{1, -1\}$, are provided and we must design a linear classifier, specified by the function $h^{\boldsymbol{\omega},b}(\mathbf{x}) = \text{sign}(\boldsymbol{\omega}^T \phi(\mathbf{x}) + b)$. For our regularized SVM classifier, the parameters $(\boldsymbol{\omega}, b)$ are obtained by solving the following convex optimization problem (Boser et al., 1992; Cristianini et al., 2000):

$$\begin{aligned} \min_{\boldsymbol{\omega},b,\xi} \quad & \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} + C\sum_{i=1}^{l}\xi_i, \\ \text{subject to} \quad & y_i(\boldsymbol{\omega}^T\phi(\mathbf{x}_i) + b) \geq 1 - \xi_i, \\ & \xi_i \geq 0, \quad i = 1, \dots, l, \end{aligned} \tag{14}$$

where $\phi(\mathbf{x}_i)$ maps $\mathbf{x}_i$ into a higher-dimensional space and $C > 0$ is the regularization parameter that tells the optimizer what it should minimize more between the $\frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega}$ and $\sum_{i=1}^{l}\xi_i$ terms. For instance, if $C = 0$, then only the term $\frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega}$ will be minimize whereas if C is a large value, then $\sum_{i=1}^{l}\xi_i$ will be minimized. Due to the possible high dimensionality of the vector variable $\boldsymbol{\omega}$, it is common to solve the following dual problem via the method of Lagrange multipliers (Boser et al., 1992; Cristianini et al., 2000):

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & \frac{1}{2}\boldsymbol{\alpha}^T Q\boldsymbol{\alpha} - \mathbf{r}^T\boldsymbol{\alpha}, \\ \text{subject to} \quad & \mathbf{y}^T\boldsymbol{\alpha} = 0, \\ & 0 \leq \alpha_i \leq C, \quad i = 1, \dots, l, \end{aligned} \tag{15}$$

where $\boldsymbol{\alpha} = [\alpha_i, \dots, \alpha_l]$ are Lagrange multipliers, $\mathbf{r} = [1, \dots, 1]^T$ is a vector of all ones, Q is an l by l positive semidefinite matrix, $Q_{ij} \equiv y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$, and $K(\mathbf{x}_i, \mathbf{x}_j) \equiv \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$ is the kernel function.

After solving Eq. (15), using the primal–dual relationship, the optimal $\boldsymbol{\omega}$ satisfies

$$\boldsymbol{\omega} = \sum_{i=1}^{l} y_i\alpha_i\phi(\mathbf{x}_i), \tag{16}$$

and the decision function, $h^{\boldsymbol{\omega},b}(x)$, that maximizes the margin of separation between the classes is given by

$$h^{\boldsymbol{\omega},b}(\mathbf{x}) = \text{sign}(\boldsymbol{\omega}^T\phi(\mathbf{x}) + b) = \text{sign}\left(\sum_{i=1}^{l} y_i\alpha_i K(\mathbf{x}_i, \mathbf{x}) + b\right). \tag{17}$$

In this study the libsvm library (Chang & Lin, 2011) was used for the implementation of the SVM classifier.

2.6.2. Graph kernels

Since graph is a complex structure with no direct representation that can be useful for classification, graph kernel is one of the main approaches to solve the challenge that is to compute the similarity between two or more complex graphs (Kriege, Johansson, & Morris, 2020a; Richiardi, Achard, Bunke, & Van De Ville, 2013). Graph kernel methods are commonly used in conjunction with machine learning algorithms, especially with SVM.

In this study we used two graph kernels, the Propagation Kernel (P2K) (Neumann, Garnett, Bauckhage, & Kersting, 2016) and the Hash Graph Kernel (HGK) (Morris, Kriege, Kersting, & Mutzel, 2016). The HGK framework has two implicit graph kernels, the Weisfeiler–Lehman (WL) subtree kernel (Shervashidze, Schweitzer, Van Leeuwen, Mehlhorn, & Borgwardt, 2011) and the shortest-path kernel (SP) (Borgwardt & Kriegel, 2005).

Considering χ as a non-empty set and letting $K : \chi \times \chi \to \mathbb{R}$ be a function, then K is a kernel on χ if there is a real Hilbert space $\mathcal{H}_K$ and a mapping $\phi : \chi \to \mathcal{H}_K$ such that $K(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x})^T\phi(\mathbf{y})$ for $\mathbf{x}, \mathbf{y} \in \chi$. In this case, we call $\phi(\cdot)$ a feature map, and $\mathcal{H}_K$ a feature space. Therefore, to define a graph kernel we let $\mathcal{G}$ to be a non-empty set of attributed graphs, then a graph kernel is $K : \mathcal{G} \times \mathcal{G} \to \mathbb{R}$.

The Propagation Kernel is a graph kernel that can handle graphs with continuous-valued attributes. The main idea is to compare two graphs by their attributes distribution with a diffusion scheme, where it updates a transition matrix of random walks. The similarity is calculated by mapping distribution vectors from all random walks iterations into discrete bins with *locality-sensitive hashing* and then computing their similarity.

The Hash Graph Kernel iteratively hashes continuous attributes to discrete labels and allows the use of different base kernels to handle those hashed attributes, like the WL and SP kernels. In this study, the number of iterations was set to 20, which is the number suggested by the authors of the method. The HGK between two graphs G_1 and G_2 is defined as

$$HGK\left(G_1, G_2\right) = \frac{1}{\lceil\mathfrak{H}\rceil}\sum_{i=1}^{\lceil\mathfrak{H}\rceil} K\left(\mathfrak{h}_i\left(G_1\right), \mathfrak{h}_i\left(G_2\right)\right), \tag{18}$$

where K is a base graph kernel, such as the WL or SP, and $\mathfrak{H} = \left\{\mathfrak{h}_1, \mathfrak{h}_2, \dots \lceil\mathfrak{H}\rceil\right\}$ is a finite family of hash functions produced with *locality-sensitive hashing* with each element $\mathfrak{h}_i \in \mathfrak{H}$ being a function $\mathfrak{h}_i : \mathbb{R}^D \to \mathbb{N}$. In addition, we also tested a modified version of the shortest-path (SP) graph kernel, where the SP is computed with edge weights. In this study we named this modification as HGK+wSP (weighted SP).

The P2K and HGK kernels used in this study meet the requirements of handle node attributed graphs with undirected edges. When using a graph kernel in a set χ with n graphs, it will produce a Gram matrix GM, which is a positive and semidefinite square matrix of $n \times n$ elements, where each element is the result of the graph kernel function between each possible pair of graphs in χ, i.e., $GM(i, j) = K\left(G_i, G_j\right)$ for $G_i, G_j \in \chi$. In other words, the GM stores the similarity computed by the graph kernel between all graphs in the dataset.

2.6.3. Graph-kernel SVM model

As a last step of our method, a Gram matrix is generated for each possible combination of graph kernels, texture feature, node attributes, distance metric for graph weights and different values of n in Eq. (11), as represented in Fig. 3. All possible combinations, as illustrated in Fig. 3, were evaluated for the image classification tasks, CN×AD, CN×MCI and MCI×AD.

The SVM receives a Gram matrix to perform its classification with a precomputed kernel matrix. In this study, we train and evaluate the performance of our Kernel-Graph SVM classifier using 426 MR images (175 CN, 153 MCI and 96 AD) on a grid-search with 10-fold cross-validation and the accuracy as the performance metric. For each fold, we search for the best hyper-parameter C using the following range $[10^{-6}; 10^{3}]$ with steps defined by a base of 10 with the exponent incremented by 1.0, i.e., $10^{-6}, 10^{-5}, \dots, 10^{3}$.

Due to the small random behavior of the graph kernels used in this study, for instance the random walk in the WL, P2K and HGK, each cross-validation was repeated 10 times to obtain more accurate results, which means that the final result is an arithmetic mean of the 10 repetitions. To a better evaluation of the classification results of ours experiments, we also computed accuracy, sensitivity, specificity and AUC.

Fig. 3 shows the classification scheme proposed in this study. For each graph kernel, we construct graphs using two node attributes approaches and, for each node attributes we define its edge weights using the three distances as described in Section 2.5.2. Once we select a distance, we can use the precomputed threshold values for the given

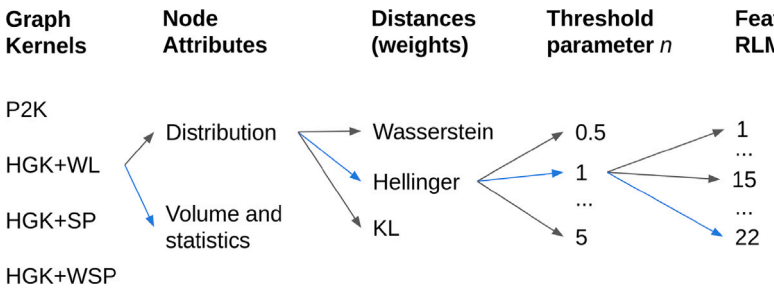

Fig. 3. Classification scheme.

distance and texture feature extracted from GLCM, RLM or LBP. In addition, each possible path generates a Gram Matrix that is used for the SVM for classification. As an example, we have draw a path highlighted in blue that corresponds to the path for the graph kernel HGK+WL, node attributes Vol+Stats, Hellinger distance weights, parameter $n = 1$, and F_{22}.

The following section presents the results and discussion of our experiments, which includes feature extraction and classification.

3. Results and discussions

In this section, we provide results of the feature extraction step and of each classification analysis, CN×AD, CN×MCI and MCI×AD, for the dataset described in Section 2.1. To evaluate the performance of different combinations of texture attributes, graph properties and types of graph kernels, we present three Tables with the experiment results. In Table 2 we show the best accuracy results of each combination for the top ranked texture attributes for CN×AD. The same arrangements are shown in Tables 3 and 4 for the CN×MCI and MCI×AD cases, respectively.

3.1. Feature extraction

The image texture features and statistical values extracted from the brain regions play a major role in this study since their distributions compose the graph node attributes. By applying the normalization step to the texture distributions, the attributes that are separable between classes should remain separable, and the distance between the distributions, obtained with a fixed number of bins using the Freedman–Diaconis rule, should also remain relatively the same.

Fig. 4 shows the texture distributions of three random subjects from the AD, MCI and CN classes. Each distribution represents the values of the feature F_{14} (High Gray-Level Run Emphasis), extracted with the RLM. Theses distributions are from each subject's left Amygdala, as this region is one of the primary regions affected by the development of Alzheimer, along with the Hippocampus. The hypothesis in this study is that the histopathological changes in AD, caused by the accumulation of amyloid β (Aβ) plaques and neurofibrillary tangles, will change the texture pattern of the brain regions related to cognition and this change will be different between subjects of the CN, MCI and AD classes.

In Fig. 4(a) each non-normalized distribution represents the raw values computed by the RLM filter. A clear separation can be noticed between the distributions of the diagnostic classes, showing a distance range from 20 to over 100, approximately. Fig. 4(b) presents the results of applying the two normalization steps as described in Section 2.4. The first step uses the min–max normalization per texture feature for all analyzed brain regions on all subjects, and the second applies the min–max over all texture distributions by region. In this case, after the first step, the number of bins of each region distribution was fixed per texture feature.

The increase of the granularity in the distributions from Fig. 4(a) to (b) can be attributed to the different sizes (voxel counts) of the brain regions. As a result of the Freedman–Diaconis rule, the distributions

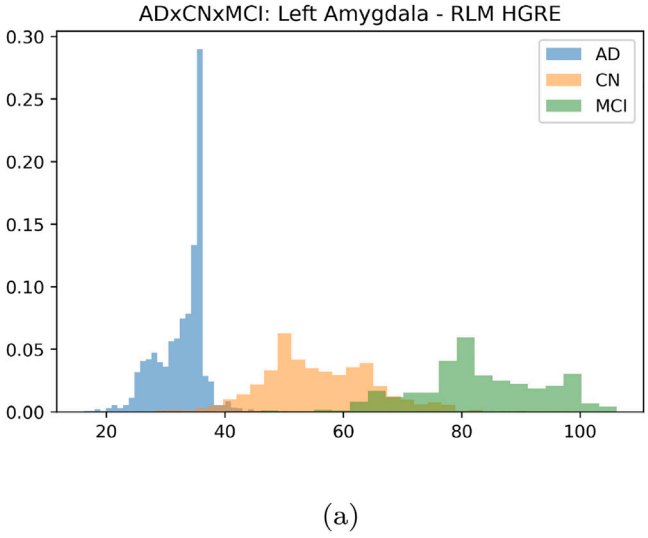

Fig. 4. (a) Non-normalized and (b) normalized texture distributions for the left Amygdala regions of a patient for each diagnostic class. The pairwise Wasserstein distances of the normalized distributions are $D_{ws} = 0.370$, $D_{ws} = 0.221$, and $D_{ws} = 0.148$ for the MCI×AD, CN×MCI and CN×AD cases, respectively.

built from small regions, such as the amygdala, will show an increasing number of bins, while for large regions, like the putamen, this number will decrease.

We can also observe that the distances between the distributions in Figs. 4(a) and (b) remained similar, with a slightly reduction in

Table 2
Classification results of the CN×AD experiments. The values of accuracy, sensitivity, and specificity are given in percentage.

Graph kernel	Node attributes	Weight metric	Texture feature	Threshold parameter	Accuracy	Sensitivity	Specificity	AUC
P2K	Distribution	Hellinger	F_{19} (LBP)	$n = 3$	75.6 ± 2.1	46.0 ± 5.2	91.6 ± 1.6	0.72 ± 0.02
	Vol+Stats	Hellinger	F_{21} (LBP)	$n = 3$	70.2 ± 1.8	24.0 ± 4.8	95.5 ± 2.2	0.65 ± 0.02
HGK+WL	Distribution	Hellinger	F_{21} (LBP)	$n = 2$	77.3 ± 0.6	42.0 ± 2.4	96.8 ± 0.6	0.84 ± 0.01
	Vol+Stats	Wasserstein	F_{21} (LBP)	$n = 3$	81.0 ± 1.3	52.2 ± 3.0	97.3 ± 0.6	0.87 ± 0.01
HGK+SP	Distribution	Wasserstein	F_2 (GLCM)	$n = 2$	75.9 ± 1.7	48.3 ± 7.4	91.0 ± 2.2	0.78 ± 0.02
	Vol+Stats	KL	F_{19} (LBP)	$n = 4$	83.8 ± 1.5	68.5 ± 2.7	92.7 ± 1.7	0.92 ± 0.01
HGK+wSP	Distribution	Wasserstein	F_{19} (LBP)	$n = 5$	73.6 ± 0.8	25.4 ± 2.8	100 ± 0.0	0.66 ± 0.01
	Vol+Stats	Hellinger	F_{19} (LBP)	$n = 2$	77.3 ± 1.6	49.5 ± 6.3	93.1 ± 1.7	0.83 ± 0.01

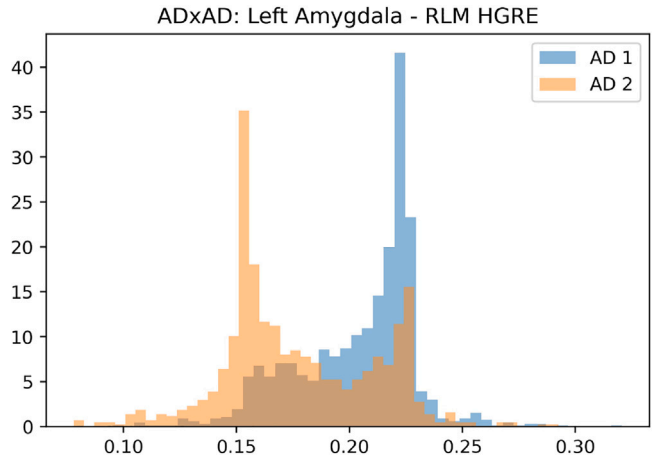

Fig. 5. RLM-HGRE texture distributions for two AD subjects (AD 1 and AD 2) showing the overlap in a dark color.

the overlap between the CN×MCI distribution after the normalization steps. Among the distributions shown in Fig. 4(b), the MCI×AD is the farthest apart ($D_{ws} = 0.370$) following by the CN×MCI ($D_{ws} = 0.221$) and CN×AD ($D_{ws} = 0.148$) distributions.

Fig. 5 shows the RLM-HGRE texture distributions of the left amygdala of two AD subjects. Since the distributions are from the same diagnostic group, we can observe a significant overlap between them. The Wasserstein distance in this case is equal to 0.027, which is considerably smaller than the inter-class distances reported for the plots in Fig. 4.

3.2. Classification results

To evaluate the performance of our method, we organized our results according to classification scheme shown in Fig. 3. We have assessed all combinations of node attributes, weight metrics, and texture attribute by using accuracy, sensitivity, specificity and AUC metrics, and their values were calculated by running the classification experiments ten times for each combination, as described in Section 2.6.3.

The threshold parameter (n), which multiplies the standard deviation in Eq. (11), was tested for the following values $n = 0.5, 1, 2, 3, 4, 5$. Since we constructed complete graphs, the larger the n is, the more edges are deleted.

Table 2 shows the results for the CN×AD classification, where we highlighted the best results of each metric. As it can be seen, the best result (83.8% of accuracy, 68.5% of sensitivity, 92.7% of specificity and 0.92 of AUC) was achieved by using the HGK+SP graph kernel for the LBP texture feature (F_{19}), with volume+statistics as node attributes and edge weights calculated with the KL distance. For the other results we can observed that the sensitivities are considerably lower when compared with the specificity values.

Except for the HGK+SP+Distribution+Wasserstein combination, the prevalence of the LBP F_{19} and F_{21} features is in the top ranked for CN×AD, showing that in our study the LBP features perform better than GLCM and RLM features to discriminate AD from CN. These results also show that for the CN×AD experiment the graphs with node attributes composed by the statistical moments and volume of the assessed brain regions performed better than node attributes defined by the texture probability distributions. This might be explained by the considerable differences in brain atrophy of the assessed regions in the CN and AD images.

Table 3 presents the results of the CN×MCI classification using the same organization as the CN×AD case. The best result (74.1% of accuracy, 71.6% of sensitivity, 76.2% of specificity and 0.81 of AUC) was obtained using the HGK+WL graph kernel for the RLM texture feature F_{17} with volume+statistics as node attributes and edge weights calculated with the Hellinger distance. Similarly to the CN×AD, the accuracy results using volume+statistics as node attributes were higher compared with the case of using probability distributions. However, in contrast to the CN×AD, the differences between the specificity and sensitivity values were lower. Analyzing the texture feature column, it can be noticed that six out of eight results were obtained using the RLM technique. Furthermore, it can be seen that the HGK+SP with volume+statistics as node attributes resulted in the same AUC value (0.81) as the best result, but with slightly lower accuracy (74.0%).

As for the MCI×AD classification results in Table 4, the best result (75.4% of accuracy, 50% of sensitivity, 91% of specificity and 0.74 of AUC) was achieved by the HGK+SP graph kernel for the RLM texture F_{13} with volume+statistics as node attributes and, edge weights calculated with the KL distance, which is the same combination used to get the best CN×AD result.

The best classification results in this experiment were obtained for higher values of the threshold parameter n when compared with the ones for the CN×AD and CN×MCI cases. This indicates that for the MCI×AD classification it was necessary to remove more edges to obtain discriminant graphs between the diagnostic classes. Similarly to the CN×AD case, the sensitivity values were very low.

Another point to note is that none of the eight GLCM texture features appeared in the top results of this experiment. In addition, comparing the three classification cases, CN×AD, CN×MCI and MCI×AD, we observed that the texture features from the RLM and LBP methods showed to be more discriminatory for the diagnosis of AD, i.e., for the CN×AD and MCI×AD cases.

3.2.1. Comparison of results with other methods

For comparison purposes, Table 5 shows the accuracy and AUC values of our method and four other methods published in the literature that use graph-based approaches and the ADNI datasets. Please notice that this comparison must be assessed with reservations since the images used in these studies possibly differ and, therefore, the parameters adjustment of the methods and reported results were not obtained using the same images.

A brief description of how our method compares to these other studies is presented bellow:

Table 3
Classification results of the CN×MCI experiments. The values of accuracy, sensitivity, and specificity are given in percentage.

Graph kernel	Node attributes	Weight metric	Texture feature	Threshold parameter	Accuracy	Sensitivity	Specificity	AUC
P2K	Distribution	KL	F_{13} (RLM)	$n = 0.5$	68.4 ± 2.1	59.6 ± 7.8	76.6 ± 4.2	0.75 ± 0.01
	Vol+Stats	Hellinger	F_7 (GLCM)	$n = 1$	64.9 ± 1.8	50.2 ± 7.3	77.8 ± 3.7	0.70 ± 0.01
HGK+WL	Distribution	Hellinger	F_{17} (RLM)	$n = 4$	68.8 ± 1.4	57.3 ± 6.2	79.0 ± 3.5	0.78 ± 0.01
	Vol+Stats	Hellinger	F_{17} (RLM)	$n = 3$	74.1 ± 1.3	71.6 ± 5.0	76.2 ± 2.4	0.81 ± 0.01
HGK+SP	Distribution	Hellinger	F_{17} (RLM)	$n = 3$	68.5 ± 2.9	63.0 ± 7.1	74.2 ± 2.3	0.77 ± 0.01
	Vol+Stats	KL	F_{18} (RLM)	$n = 1$	74.0 ± 1.0	79.8 ± 2.3	68.9 ± 2.6	0.81 ± 0.01
HGK+wSP	Distribution	KL	F_{17} (RLM)	$n = 0.5$	64.8 ± 1.8	51.0 ± 6.7	78.1 ± 3.0	0.69 ± 0.01
	Vol+Stats	Wasserstein	F_5 (GLCM)	$n = 5$	70.4 ± 2.2	66.1 ± 6.9	75.7 ± 2.6	0.79 ± 0.01

Table 4
Classification results of the MCI×AD experiments. The values of accuracy, sensitivity, and specificity are given in percentage.

Graph kernel	Node attributes	Weight metric	Texture feature	Threshold parameter	Accuracy	Sensitivity	Specificity	AUC
P2K	Distribution	Hellinger	F_{14} (RLM)	$n = 2$	70.7 ± 1.7	33.0 ± 3.9	94.1 ± 1.9	0.67 ± 0.01
	Vol+Stats	Wasserstein	F_{22} (LBP)	$n = 3$	68.0 ± 1.1	19.5 ± 3.9	98.8 ± 0.9	0.63 ± 0.01
HGK+WL	Distribution	Wasserstein	F_{12} (RLM)	$n = 2$	74.1 ± 1.3	38.7 ± 3.2	96.4 ± 0.5	0.68 ± 0.01
	Vol+Stats	KL	F_{15} (RLM)	$n = 3$	74.0 ± 2.5	41.1 ± 3.2	94.3 ± 1.3	0.72 ± 0.02
HGK+SP	Distribution	Wasserstein	F_{15} (RLM)	$n = 5$	74.8 ± 1.3	41.3 ± 2.0	95.4 ± 2.4	0.72 ± 0.02
	Vol+Stats	KL	F_{13} (RLM)	$n = 3$	75.4 ± 1.3	50.0 ± 3.2	91.0 ± 2.4	0.74 ± 0.01
HGK+wSP	Distribution	Hellinger	F_{22} (LBP)	$n = 5$	71.9 ± 1.9	34.5 ± 5.3	96.3 ± 0.8	0.73 ± 0.01
	Vol+Stats	Hellinger	F_{20} (LBP)	$n = 3$	70.9 ± 0.6	24.9 ± 2.3	99.9 ± 0.2	0.71 ± 0.01

Table 5
Comparison of results with other methods. The accuracy values are given in percentage.

Methods	Subjects			CN×AD		CN×MCI		MCI×AD	
	CN	MCI	AD	ACC	AUC	ACC	AUC	ACC	AUC
Liu et al. (2017)	230	280	200	95.3	0.96	86.6	0.91	86.5	0.85
Jie et al. (2018)	50	99	34	–	–	82.6	0.78	–	–
Cui et al. (2018)	21	25	22	91.3	0.99	98.3	0.99	77.3	0.97
Wee et al. (2019)	242	415	355	81.0	–	67.7	–	65.4	–
Hett et al. (2021)	213	–	130	91.6	–	–	–	–	–
Proposed method	200	153	121	83.8	0.92	74.1	0.81	75.4	0.74

- Liu et al. (2017) presented a graph-based approach built from multiple regions that uses GLCM texture features and performs two stages of feature selection. Although in general their average results are superior to those obtained by our method, for the CN×AD case the AUC value of our method is only slightly lower. However, it is important to note that in their work, Liu et al. did not provide the variance values of the metrics used to evaluate the results.
- Jie et al. (2018) proposed a graph kernel approach built from multiple brain regions and which uses features extracted from fMRI data of the ADNI dataset. In comparison to this method, our approach achieved superior AUC value for CN×MCI, i.e., 0.81 of our method against 0.78 of their approach. Their work also used fewer images when compared to our study.
- Cui et al. (2018) proposed a method based on graph kernel that uses attributes from fMRI readings of multiple brain regions. Despite the superior results reported in their study, they used less than 30 images for each diagnostic class.
- Wee et al. (2019) proposed a method based on a spectral graph convolutional neural network that incorporates cortical thickness and its underlying geometry information to identify MCI and AD using T1-w MRI data from the ADNI datasets. Although their focus was on the prediction of conversion of MCI to AD, they also used their method for image classification. In comparison to their work, our method resulted in superior accuracy in all classification cases. However, it is noteworthy that the number of images used in their study was higher for the MCI and AD classes.
- As previously described in the introduction section of this paper, Hett et al. (2021) used multiple regions and extracted their attributes using the PBG method (Coupé et al., 2012). The focus of their study was primarily to distinguish stable MCI from progressive MCI, but they also used their method for CN×AD where they obtained results slightly above ours with similar number of subjects.

In contrast to our method, which uses only a threshold-based method for removing edges from graphs, the related methods described above use sophisticated edge and node attribute selection strategies to build class discriminating graphs. This processing step may have contributed positively to the classification results.

4. Conclusions

We presented a new technique to perform MR image classification for AD diagnosis using kernel-based graph methods that are constructed from image texture features extracted from structural MR images. For each MR image in the dataset, we extracted 22 texture features and statistical moments and volume of multiple segmented brain regions and used this information within two strategies to build our graph representations. Furthermore, we proposed an edge removal technique to help build discriminative graphs for the AD classification task. In addition, we evaluated three distances as the edge metric and performed the MR image classification using SVMs with four different graph kernels.

Our results showed a better performance for the CN×AD (AUC = 0.92) and CN×MCI (AUC = 0.81) classifications, and worse for the MCI×AD case (AUC = 0.78). These results are consistent with other results published in the literature and make sense if we consider the concept of Alzheimer's disease continuum from pathophysiological, biomarker and clinical perspectives.

Comparing to the state-of-the-art in graph-based Alzheimer's classification our results superior in some cases and slightly inferior in others. We believe that by using a more elaborated technique for edge and attribute selection for the image classification and incorporating other attributes such as MMSE score and ranking (Hett et al., 2021), our results can improve significantly.

This project was developed using python and C++ programming languages. All source codes is available at GitHub[3].

CRediT authorship contribution statement

Lucas José Cruz de Mendonça: Conceptualization, Methodology, Software, Writing – original draft. **Ricardo José Ferrari:** Conceptualization, Supervision, Writing – review & editing, Resources.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

Data collection and sharing for this project was funded by the Alzheimer's Disease Neuroimaging Initiative (ADNI) (National Institutes of Health Grant U01 AG024904) and DOD ADNI (Department of Defense award number W81XWH-12-2-0012). ADNI is funded by the National Institute on Aging, the National Institute of Biomedical Imaging and Bioengineering, and through generous contributions from the following: AbbVie, Alzheimer's Association; Alzheimer's Drug Discovery Foundation; Araclon Biotech; BioClinica, Inc.; Biogen; Bristol-Myers Squibb Company; CereSpir, Inc.; Cogstate; Eisai Inc.; Elan Pharmaceuticals, Inc.; Eli Lilly and Company; EuroImmun; F. Hoffmann-La Roche Ltd and its affiliated company Genentech, Inc.; Fujirebio; GE Healthcare; IXICO Ltd.; Janssen Alzheimer Immunotherapy Research & Development, LLC.; Johnson & Johnson Pharmaceutical Research & Development LLC.; Lumosity; Lundbeck; Merck & Co., Inc.; Meso Scale Diagnostics, LLC.; NeuroRx Research; Neurotrack Technologies; Novartis Pharmaceuticals Corporation; Pfizer Inc.; Piramal Imaging; Servier; Takeda Pharmaceutical Company; and Transition Therapeutics. The Canadian Institutes of Health Research is providing funds to support ADNI clinical sites in Canada. Private sector contributions are facilitated by the Foundation for the National Institutes of Health (www.fnih.org). The grantee organization is the Northern California Institute for Research and Education, and the study is coordinated by the Alzheimer's Therapeutic Research Institute at the University of Southern California. ADNI data are disseminated by the Laboratory for Neuro Imaging at the University of Southern California.

Funding statement

The authors would like to thank the São Paulo Research Foundation (FAPESP) (grant number 2018/08826-9) and the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001 for the financial support of this research.

Appendix

All feature definitions in this appendix were obtained from the Ref. Vimort et al. (2017).

[3] https://github.com/lucasjome/alzheimer.

Appendix A. GLCM texture features

Considering a GLCM matrix P and $p(i,j)$ as an element in the cell (i,j) of P and $\mu = \sum_{i,j} i \cdot p(i,j) = \sum_{i,j} j \cdot p(i,j)$ and $\sigma = \sum_{i,j}(i-\mu)^2 p(i,j) = \sum_{i,j}(j-\mu)^2 p(i,j)$ to be the weighted average and variance, respectively. Assuming that μ_t and σ_t are the mean and standard deviation, respectively, of the row or column sums, then the GLCM texture features are defined as:

Energy is a feature that measures the local uniformity of a texture. The higher the energy value is, the bigger the uniformity and organization of the texture.

$$F_1 = \sum_{i,j} p(i,j)^2.$$

Entropy is a feature that expresses the level of organization of a texture. A completely random distribution of gray-level intensities in a image volume would have very high entropy, while an image with the same gray-level across all pixels would have a very low value of entropy.

$$F_2 = \begin{cases} \sum_{i,j} p(i,j)\log_2\left(p(i,j)\right), & \text{if } p(i,j) \neq 0 \\ 0, & \text{if } p(i,j) = 0. \end{cases}$$

Correlation is a feature that measures the linear dependency of gray level values in the co-occurrence matrix.

$$F_3 = \sum_{i,j} \frac{(i-\mu)(j-\mu)p(i,j)}{\sigma^2}.$$

Inverse Difference Moment is a feature that measures the homogeneity of an image. This feature will be low for in-homogeneous images, and a high for homogeneous images.

$$F_4 = \sum_{i,j} \frac{p(i,j)}{1+(i-j)^2}.$$

Inertia or **contrast** is a feature that measures local gray-level variation in the GLCM matrix. If the neighboring pixels in a texture are very similar in their gray-level values then the image contrast is very low. Contrast is zero for a constant image.

$$F_5 = \sum_{i,j} (i-j)^2 p(i,j).$$

Cluster Shade is a feature of the skewness of the matrix and it is believed to be linked to perception of uniformity in an image. When this feature is high the image is asymmetric.

$$F_6 = \sum_{i,j} ((i-\mu)+(j-\mu))^3 p(i,j).$$

Cluster Prominence is a feature also related to the perceptual symmetry of an image. When the cluster prominence value is high, the image is less symmetric.

$$F_7 = \sum_{i,j} ((i-\mu)+(j-\mu))^4 p(i,j).$$

Haralick's Correlation is the original correlation measure designed by Haralick in 1973, and it measures the linear dependence between pixels relative to each other.

$$F_8 = \frac{\sum_{i,j}(i,j)p(i,j) - \mu_t^2}{\sigma_t}.$$

Appendix B. RLM texture features

Considering $p(i,j)$ as a element of the RLM matrix, and i as the voxel intensity and j as length of the run. The RLM texture features are defined as:

Short run emphasis (SRE) measures the distribution of short runs. SRE is expected to be large for fine textures.

$$F_9 = \frac{\sum_{i,j} \frac{p(i,j)}{j^2}}{\sum_{i,j} p(i,j)}.$$

Table C.6
List of brain regions used in our analysis.

label_index	label_name	label_index	label_name
4	Left-Lateral-Ventricle	1019	ctx-lh-parsorbitalis
5	Left-Inf-Lat-Vent	1020	ctx-lh-parstriangularis
7	Left-Cerebellum-White-Matter	1021	ctx-lh-pericalcarine
10	Left-Thalamus	1022	ctx-lh-postcentral
11	Left-Caudate	1023	ctx-lh-posteriorcingulate
12	Left-Putamen	1024	ctx-lh-precentral
13	Left-Pallidum	1025	ctx-lh-precuneus
14	3rd-Ventricle	1026	ctx-lh-rostralanteriorcingulate
15	4th-Ventricle	1027	ctx-lh-rostralmiddlefrontal
16	Brain-Stem	1028	ctx-lh-superiorfrontal
17	Left-Hippocampus	1029	ctx-lh-superiorparietal
18	Left-Amygdala	1030	ctx-lh-superiortemporal
24	CSF	1031	ctx-lh-supramarginal
28	Left-VentralDC	1034	ctx-lh-transversetemporal
31	Left-choroid-plexus	1035	ctx-lh-insula
43	Right-Lateral-Ventricle	2002	ctx-rh-caudalanteriorcingulate
44	Right-Inf-Lat-Vent	2003	ctx-rh-caudalmiddlefrontal
46	Right-Cerebellum-White-Matter	2005	ctx-rh-cuneus
49	Right-Thalamus	2006	ctx-rh-entorhinal
50	Right-Caudate	2007	ctx-rh-fusiform
51	Right-Putamen	2008	ctx-rh-inferiorparietal
52	Right-Pallidum	2009	ctx-rh-inferiortemporal
53	Right-Hippocampus	2010	ctx-rh-isthmuscingulate
54	Right-Amygdala	2011	ctx-rh-lateraloccipital
60	Right-VentralDC	2012	ctx-rh-lateralorbitofrontal
63	Right-choroid-plexus	2013	ctx-rh-lingual
77	WM-hypointensities	2014	ctx-rh-medialorbitofrontal
251	CC_Posterior	2015	ctx-rh-middletemporal
252	CC_Mid_Posterior	2016	ctx-rh-parahippocampal
255	CC_Anterior	2017	ctx-rh-paracentral
1002	ctx-lh-caudalanteriorcingulate	2018	ctx-rh-parsopercularis
1003	ctx-lh-caudalmiddlefrontal	2019	ctx-rh-parsorbitalis
1005	ctx-lh-cuneus	2020	ctx-rh-parstriangularis
1006	ctx-lh-entorhinal	2021	ctx-rh-pericalcarine
1007	ctx-lh-fusiform	2022	ctx-rh-postcentral
1008	ctx-lh-inferiorparietal	2023	ctx-rh-posteriorcingulate
1009	ctx-lh-inferiortemporal	2024	ctx-rh-precentral
1010	ctx-lh-isthmuscingulate	2025	ctx-rh-precuneus
1011	ctx-lh-lateraloccipital	2026	ctx-rh-rostralanteriorcingulate
1012	ctx-lh-lateralorbitofrontal	2027	ctx-rh-rostralmiddlefrontal
1013	ctx-lh-lingual	2028	ctx-rh-superiorfrontal
1014	ctx-lh-medialorbitofrontal	2029	ctx-rh-superiorparietal
1015	ctx-lh-middletemporal	2030	ctx-rh-superiortemporal
1016	ctx-lh-parahippocampal	2031	ctx-rh-supramarginal
1017	ctx-lh-paracentral	2034	ctx-rh-transversetemporal
1018	ctx-lh-parsopercularis	2035	ctx-rh-insula

Long run emphasis (LRE) is a feature that measures distribution of long runs. LRE is expected to be large for coarse structural textures.

$$F_{10} = \frac{\sum_{i,j} p(i,j) j^2}{\sum_{i,j} p(i,j)}.$$

Gray level non-uniformity (GLN) measures the similarity of gray-level values through out the texture. The GLN is expected to be small if the gray-level values are alike throughout the whole texture.

$$F_{11} = \frac{\sum_i (\sum_j p(i,j))^2}{\sum_{i,j} p(i,j)}.$$

Run length non-uniformity (RLN) is a feature that measures the similarity of the length of runs through out the image. The RLN is expected to be small if the run lengths are alike through out the image.

$$F_{12} = \frac{\sum_j (\sum_i p(i,j))^2}{\sum_{i,j} p(i,j)}.$$

Low gray level run emphasis (LGRE) is orthogonal to SRE, and the value of the feature increases when the texture is dominated by many runs of low gray value.

$$F_{13} = \frac{\sum_{i,j} \frac{p(i,j)}{i^2}}{\sum_{i,j} p(i,j)}.$$

High gray level run emphasis (HGRE) is orthogonal to LRE, and the metric increases when the texture is dominated by many runs of high gray value.

$$F_{14} = \frac{\sum_{i,j} p(i,j) i^2}{\sum_{i,j} p(i,j)}.$$

Short run low gray level emphasis (SRLGE) is a diagonal measurement that combines SRE and LGRE. The metric increases when the texture is dominated by many short runs of low gray value.

$$F_{15} = \frac{\sum_{i,j} \frac{p(i,j)}{i^2 j^2}}{\sum_{i,j} p(i,j)}.$$

Short run high gray level emphasis (SRHGE) is orthogonal to SRLGE and LRHGE and increases when the texture is dominated by short runs with high intensity levels.

$$F_{16} = \frac{\sum_{i,j} \frac{p(i,j)}{j^2}}{\sum_{i,j} p(i,j)}.$$

Long run low gray level emphasis (LRLGE) is complementary to SRHGE, it increases when the texture is dominated by long runs that have low gray levels.

$$F_{17} = \frac{\sum_{i,j} \frac{p(i,j) j^2}{i^2}}{\sum_{i,j} p(i,j)}.$$

Long run high gray level emphasis (LRHGE) is the complementary metric to SRLGE and increases with a combination of long, high-gray value runs.

$$F_{18} = \frac{\sum_{i,j} p(i,j) i^2 j^2}{\sum_{i,j} p(i,j)}.$$

Appendix C. List of brain regions

List of brain regions used in our analysis. Regions with label index 1002 onwards (those starting with ctx) are from the DKT atlas. The other regions are from the ASEG atlas. In this list, lh stands for "Left Hemisphere" and rh stands for "Right Hemisphere" (see Table C.6).

References

Albu, A., Precup, R.-E., & Teban, T.-A. (2019). Results and challenges of artificial neural networks used for decision-making and control in medical applications. *Facta Universitatis*, *17*, 285–308. http://dx.doi.org/10.22190/FUME190327035A.

Araújo, M., Poloni, K., & Ferrari, R. (2021). Assessment of linear and non-linear feature projections for the classification of 3-D MR images on cognitively normal, mild cognitive impairment and Alzheimer's disease. In *Computational science and its applications – ICCSA 2021* (pp. 18–33). Springer International Publishing, http://dx.doi.org/10.1007/978-3-030-86960-1_2.

Banerjee, J., Moelker, A., Niessen, W. J., & Van Walsum, T. (2012). 3D LBP-based rotationally invariant region description. In *Asian conference on computer vision* (pp. 26–37). Springer.

Borgwardt, K. M., & Kriegel, H.-P. (2005). Shortest-path kernels on graphs. In *Fifth IEEE international conference on data mining (ICDM'05)* (pp. 1–8). IEEE.

Boser, B. E., Guyon, I. M., & Vapnik, V. N. 1992. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory, Pittsburgh, Pen, USA*. (pp. 144–152).

Bustamam, A., Sarwinda, D., & Ardenaswari, G. (2018). Texture and gene expression analysis of the MRI brain in detection of Alzheimer's disease. *Journal of Artificial Intelligence and Soft Computing Research*, *8*, 111–120.

Butterfield, D. A., & Halliwell, B. (2019). Oxidative stress, dysfunctional glucose metabolism and Alzheimer disease. *Nature Reviews Neuroscience*, *20*, 148–160.

Cai, J. H., He, Y., Zhong, X. L., Lei, H., Wang, F., Luo, G. H., et al. (2020). Magnetic resonance texture analysis in Alzheimer's disease. *Academic Radiology*, *27*, 1774–1783.

Cambui, V., Poloni, K., & Ferrari, R. (2021). Classification of brain MR images for the diagnosis of Alzheimer's disease features extracted from the three main brain tissues. In *Proceedings of the 6th Brazilian technology symposium (BTSym'20), Vol. 233* (pp. 212–219). Springer International Publishing, http://dx.doi.org/10.1007/978-3-030-75680-2_25.

Castellano, G., Bonilha, L., Li, L., & Cendes, F. (2004). Texture analysis of medical images. *Clinical Radiology*, *59*, 1061–1069.

Chang, C.-C., & Lin, C.-J. (2011). Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *2*, 1–27.

Coupé, P., Eskildsen, S. F., Manjón, J. V., Fonov, V. S., Pruessner, J. C., Allard, M., et al. (2012). Scoring by nonlocal image patch estimator for early detection of Alzheimer' s disease. *NeuroImage: Clinical*, *1*, 141–152.

Cristianini, N., Shawe-Taylor, J., et al. (2000). *An introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press.

Cui, X., Xiang, J., Guo, H., Yin, G., Zhang, H., Lan, F., et al. (2018). Classification of Alzheimer's disease, mild cognitive impairment, and normal controls with subnetwork selection and graph kernel principal component analysis based on minimum spanning tree brain functional network. *Frontiers in Computational Neuroscience*, *12*, 1–12.

Dieckmeyer, M., Inhuber, S., Schlaeger, S., Weidlich, D., Mookiah, M. R. K., Subburaj, K., et al. (2021). Texture features of proton density fat fraction maps from chemical shift encoding-based MRI predict paraspinal muscle strength. *Diagnostics*, *11*, 239.

Dubois, B., Feldman, H., Jacova, C., Hampel, H., Molinuevo, J., Blennow, K., et al. (2014). Advancing research diagnostic criteria for Alzheimer's disease: the IWG-2 criteria. *The Lancet Neurology*, *13*, 614–629.

Feng, J., Zhang, S. W., & Chen, L. (2020). Identification of Alzheimer's disease based on wavelet transformation energy feature of the structural MRI image and NN classifier. *Artificial Intelligence in Medicine*, 108.

Fischl, B., Van Der Kouwe, A., Destrieux, C., Halgren, E., Ségonne, F., Salat, D. H., et al. (2004). Automatically parcellating the human cerebral cortex. *Cerebral Cortex*, *14*, 11–22.

Freedman, D., & Diaconis, P. (1981). On the histogram as a density estimator: L2 theory. *Zeitschrift Für Wahrscheinlichkeitstheorie Und Verwandte Gebiete*, *57*, 453–476.

Galloway, M. M. (1974). *Texture analysis using grey level run lengths*: *NASA STI/Recon technical report N, 75*, (p. 18555).

Haralick, R. M., Shanmugam, K., & Dinstein, I. H. (1973). Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-3.

Hellinger, E. (1909). Neue begründung der theorie quadratischer formen von unendlichvielen veränderlichen. *Journal Für Die Reine Und Angewandte Mathematik*, *1909*, 210–271.

Hett, K., Ta, V.-T., Catheline, G., Tourdias, T., Manjón, J. V., & Coupé, P. (2019). Multimodal hippocampal subfield grading for alzheimer's disease classification. *Scientific Reports*, *9*, 1–16.

Hett, K., Ta, V.-T., Manjón, J., & Coupé, P. (2018). Graph of brain structures grading for early detection of Alzheimer's disease. In *Medical image computing and computer assisted intervention – MICCAI 2018* (pp. 429–436). Springer International Publishing, http://dx.doi.org/10.1007/978-3-030-00931-1_49.

Hett, K., Ta, V.-T., Oguz, I., Manjón, J. V., Coupé, P., Initiative, A. D. N., et al. (2021). Multi-scale graph-based grading for Alzheimer's disease prediction. *Medical Image Analysis*, *67*, Article 101850.

Jack, C. R., Jr., Bernstein, M. A., Fox, N. C., Thompson, P., Alexander, G., Harvey, D., et al. (2008). The Alzheimer's disease neuroimaging initiative (ADNI): MRI methods. *Journal of Magnetic Resonance Imaging*, *27*, 685–691.

Jie, B., Liu, M., Jiang, X., & Zhang, D. (2016). Sub-network based kernels for brain network classification. In *7th ACM conference on bioinformatics, computational biology, and health informatics* (pp. 622–629). Seattle, WA, USA: ACM.

Jie, B., Liu, M., Zhang, D., & Shen, D. (2018). Sub-network kernels for measuring similarity of brain connectivity networks in disease diagnosis. *IEEE Transactions on Image Processing*, *27*, 2340–2353.

Klein, A., & Tourville, J. (2012). 101 Labeled brain images and a consistent human cortical labeling protocol. *Frontiers in Neuroscience*, *6*, 1–12.

Kriege, N. M., Johansson, F. D., & Morris, C. (2020a). A survey on graph kernels. *Applied Network Science*, *5*.

Kriege, N. M., Johansson, F. D., & Morris, C. (2020b). A survey on graph kernels. *Applied Network Science*, *5*, 1–42.

Kullback, S., & Leibler, R. (1951). *On information and sufficiencyannals of mathematical statistics, Vol. 22* (pp. 79–86). MathSciNet MATH.

Liu, J., Li, M., Lan, W., Wu, F.-X., Pan, Y., & Wang, J. (2016). Classification of Alzheimer's disease using whole brain hierarchical network. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *15*, 624–632.

Liu, J., Tan, G., Lan, W., & Wang, J. (2020). Identification of early mild cognitive impairment using multi-modal data and graph convolutional networks. *BMC Bioinformatics*, *21*, 1–12.

Liu, J., Wang, J., Hu, F.-X., & Pan, Y. (2017). Alzheimer's disease classification based on individual hierarchical networks constructed with 3-D texture features. *IEEE Transactions on Nanobioscience*, *16*, 428–437.

Livingston, G., Sommerlad, A., Orgeta, V., Costafreda, S. G., Huntley, J., Ames, D., et al. (2017). Dementia prevention, intervention, and care. *The Lancet*, *390*, 2673–2734.

Martins, R. N., Villemagne, V., Sohrabi, H. R., Chatterjee, P., Shah, T. M., Verdile, G., et al. (2018). Alzheimer's disease: a journey from amyloid peptides and oxidative stress, to biomarker technologies and disease prevention strategies—gains from AIBL and DIAN cohort studies. *Journal of Alzheimer's Disease*, *62*, 965–992.

McCormick, M. M., Liu, X., Ibanez, L., Jomier, J., & Marion, C. (2014). ITK: enabling reproducible research and open science. *Frontiers in Neuroinformatics*, *8*(13).

McKhann, G. M., Knopman, D. S., Chertkow, H., Hyman, B. T., Jack, C. R., Jr., Kawas, C. H., et al. (2011). The diagnosis of dementia due to Alzheimer's disease: recommendations from the national institute on aging-Alzheimer's association workgroups on diagnostic guidelines for Alzheimer's disease. *Alzheimer's & Dementia*, *7*, 263–269.

Morris, C., Kriege, N. M., Kersting, K., & Mutzel, P. (2016). Faster kernels for graphs with continuous attributes via hashing. In *IEEE 16th international conference on data mining (ICDM)* (pp. 1095–1100). IEEE.

Morris, J. C., Storandt, M., Miller, J. P., McKeel, D. W., Price, J. L., Rubin, E. H., et al. (2001). Mild cognitive impairment represents early-stage alzheimer disease. *Archives of Neurology*, *58*, 397–405.

Nanni, L., Brahnam, S., Salvatore, C., Castiglioni, I., Initiative, A. D. N., et al. (2019). Texture descriptors and voxels for the early diagnosis of Alzheimer's disease. *Artificial Intelligence in Medicine*, *97*, 19–26.

Neumann, M., Garnett, R., Bauckhage, C., & Kersting, K. (2016). Propagation kernels: efficient graph kernels from propagated information. *Machine Learning*, *102*, 209–245.

Noortman, W. A., Vriens, D., Slump, C. H., Bussink, J., Meijer, T. W., de Geus-Oei, L.-F., et al. (2020). Adding the temporal domain to PET radiomic features. *PLoS One*, *15*, Article e0239438.

Ojala, T., Pietikäinen, M., & Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, *29*, 51–59.

Oliveira, I., Poloni, K., & Ferrari, R. (2020). Exploring hippocampal asymmetrical features from magnetic resonance images for the classification of Alzheimer's disease. In *2020 IEEE 33rd international symposium on computer-based medical systems (CBMS)* (pp. 59–64). http://dx.doi.org/10.1109/CBMS49503.2020.00019.

Poloni, K., & Ferrari, R. (2022a). A deep ensemble hippocampal CNN model for brain age estimation applied to Alzheimer's diagnosis. *Expert Systems with Applications*, *195*, Article 106581. http://dx.doi.org/10.1016/j.eswa.2022.116622.

Poloni, K., & Ferrari, R. (2022b). Automated detection, selection and classification of hippocampal landmark points for the diagnosis of Alzheimer's disease. *Computer Methods and Programs in Biomedicine*, *214*, Article 106581. http://dx.doi.org/10.1016/j.cmpb.2021.106581.

Ramdas, A., Trillos, N. G., & Cuturi, M. (2017). On wasserstein two-sample testing and related families of nonparametric tests. *Entropy*, *19*, 1–18.

Richiardi, J., Achard, S., Bunke, H., & Van De Ville, D. (2013). Machine learning with brain graphs: predictive modeling approaches for functional imaging in systems neuroscience. *IEEE Signal Processing Magazine*, *30*, 58–70.

Rubner, Y., Tomasi, C., & Guibas, L. J. (2000). The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, *40*, 99–121.

Shervashidze, N., Schweitzer, P., Van Leeuwen, E. J., Mehlhorn, K., & Borgwardt, K. M. (2011). Weisfeiler-lehman graph kernels. *Journal of Machine Learning Research*, *12*.

Silveira Souza, B., Poloni, K., & Ferrari, R. (2020). Detector of 3-D salient points based on the dual-tree complex wavelet transform for the positioning of hippocampi meshes in magnetic resonance images. *Journal of Neuroscience Methods*, *341*, Article 108789.

Song, X., Elazab, A., & Zhang, Y. (2020). Classification of mild cognitive impairment based on a combined high-order network and graph convolutional network. *IEEE Access*, *8*, 42816–42827.

Sørensen, L., Igel, C., Liv Hansen, N., Osler, M., Lauritzen, M., Rostrup, E., et al., The Australian Imaging Biomarkers, & of Ageing, L. F. S. (2016). Early detection of Alzheimer's disease using MRI hippocampal texture. *Human Brain Mapping*, *37*, 1148–1161.

Teh, S.-K., Rawtaer, I., & Tan, A.-H. (2022). Predictive self-organizing neural networks for in-home detection of mild cognitive impairment. *Expert Systems with Applications*, *205*, Article 117538. http://dx.doi.org/10.1016/j.eswa.2022.117538.

Vimort, J.-b., McCormick, M., Budin, F., & Paniagua, B. (2017). Computing textural feature maps for N-dimensional images. *The Insight Journal*, http://dx.doi.org/10.54294/qy48ty.

Wee, C.-Y., Liu, C., Lee, A., Poh, J. S., Ji, H., Qiu, A., et al. (2019). Cortical graph neural network for AD and MCI diagnosis and transfer learning across populations. *NeuroImage: Clinical*, *23*, 1–15.

Xia, H., & Hoi, S. C. (2012). Mkboost: A framework of multiple kernel boosting. *IEEE Transactions on Knowledge and Data Engineering*, *25*, 1574–1586.

Yaakub, S. N., Heckemann, R. A., Keller, S. S., Mcginnity, C. J., Weber, B., & Hammers, A. (2020). On brain atlas choice and automatic segmentation methods : a comparison of MAPER & FreeSurfer using three atlas databases. *Scientific Reports*, *10*, 1–15.

Zhang, J., Yu, C., Jiang, G., Liu, W., & Tong, L. (2012). 3D texture analysis on MRI images of Alzheimer's disease. *Brain Imaging and Behavior*, *6*, 61–69.