The dissimilarity approach: a review

Yandre M. G. Costa, Diego Bertolini, Alceu S. Britto, George D. C. Cavalcanti & Luiz E. S. Oliveira

Artificial Intelligence Review
An International Science and Engineering Journal

ISSN 0269-2821

Artif Intell Rev
DOI 10.1007/s10462-019-09746-z

Springer

Artificial Intelligence Review
https://doi.org/10.1007/s10462-019-09746-z

The dissimilarity approach: a review

Yandre M. G. Costa[1] · Diego Bertolini[2] · Alceu S. Britto Jr.[3] · George D. C. Cavalcanti[4] · Luiz E. S. Oliveira[5]

Abstract
Dissimilarity representation is a very interesting alternative for the traditional feature space representation when addressing large multi-class problems or even problems with a small number of training samples. This paper describes the existing possibilities in terms of dissimilarity representation through some comprehensive examples. The justification for using such a problem representation strategy is discussed, followed by a complete review of the state-of-art and a critical analysis in which the original purpose of the dissimilarity representation and its perspectives are discussed. Dissimilarity space derived from automatically learned features and the possibility of transiting from one space to another when performing the tasks of the classification process are good examples of promising research directions in this field.

Keywords Dissimilarity · Pattern recognition · Classification

✉ Yandre M. G. Costa
yandre@din.uem.br

Diego Bertolini
diegobertolini@utfpr.edu.br

Alceu S. Britto Jr.
alceubritto@gmail.com

George D. C. Cavalcanti
gdcc@cin.ufpe.br

Luiz E. S. Oliveira
luiz.oliveira@ufpr.br

1 State University of Maringá (UEM), Maringá, Brazil

2 Federal Technological University of Paraná (UTFPR), Curitiba, Brazil

3 Pontifical Catholic University of Paraná (PUCPR), Curitiba, Brazil

4 Federal University of Pernambuco (UFPE), Recife, Brazil

5 Federal University of Paraná (UFPR), Curitiba, Brazil

Published online: 02 August 2019

1 Introduction

In the context of pattern recognition, the use of dissimilarity approaches is based on the idea that, somehow, humans consider (dis)similarities between patterns to distinguish one from the other. According to Pękalska et al. (2002), (dis)similarities can be taken as a bridge that connects perception and higher-level knowledge, playing an important role in the way on how humans perform recognition and categorization. However, there are some intriguing issues which may lead researchers to wonder whether or not they should consider the use of dissimilarity to address a given pattern classification problem.

In this scenario, some of the first questions that come to mind are: (i) when should someone evaluate the possibility of using a dissimilarity based approach instead of using traditional feature spaces?; (ii) what are the limitations of traditional feature spaces which would justify the use of dissimilarity?; and lastly (iii) what are the most remarkable dissimilarity strategies presented in the literature? Actually, there is not a definitive answer to these questions, but some interesting insights about them can be found in the works presented by some of the most important authors regarding dissimilarity approaches: Pękalska (2005) and Cha (2001).

The answer to the first question, should be tried in a twofold way. According to Cha (2001), the dissimilarity approach proposed in his work might be a suitable choice to address large multi-class problems. Cha claims that there is a trade off between tractability and accuracy that becomes critical in this kind of problem, especially because they are hard to be sampled. The dissimilarity approach proposed by Cha transpose a multi-class problem to a binary one, as will be discussed in details in Sect. 2.2, reducing difficulties caused by the lack of samples and/or the complexity to deal with the creation of an effective large multi-class classification model.

In another vein, Pękalska (2005) advocates that the dissimilarity strategy presented in her work is appropriate for classification problems in which the patterns have an intrinsic and detectable organization (e.g. shapes, spectra, images or texts). According to her, the patterns have some latent aspects (e.g. order, time, hierarchy or functional relationships) which express the connection between morphological primitives. In addition, Pękalska also gives a direction regarding the central point of the second question raised here. She points out that, in some cases, it is difficult to obtain an efficient feature-based description to patterns for learning purposes. Sometimes, even experts are not able to describe features in an unequivocal way, obtaining a high dimensional representation or still a representation with continuous and categorical variables mixed together.

To answer the third question we performed a systematic review of the literature and find two main approaches to build dissimilarity based classifiers; (i) the dissimilarity space and (ii) the dissimilarity vectors. In the former, Pękalska et al. (2001) have presented a proposal based on a matrix of distances calculated between a set of training patterns T and a set of reference patterns R, composed of patterns selected from all the classes involved in the problem. In that case, a dissimilarity matrix is built considering one row for each sample of T and one column for each sample of R. Each cell of the matrix is filled with the distance between the sample assigned to that row and the sample assigned to that column. By this way, one can expect along the rows of the matrix, values close to zero for that columns assigned to samples which belong to the same class of the sample of that row. In the opposite way, higher values are expected to columns assigned to other classes. Taking into account this approach, one can use the rows of the dissimilarity matrix as feature vectors to build a classifier model. Although the terms distance and dissimilarity are used interchangeably thorough the literature, the dissimilarity is a broader term than distance. As pointed out by

Pękalska (2005), the dissimilarity value captures the notion of closeness between two objects, which can be interpreted as a distance in a suitable space, or which can be used to build other spaces. In fact, the dissimilarity embedding can be produced by a broad range of functions that cannot be described as distance functions (non-metric, non-symmetric, even functions returning both positive and negative values, etc.), comparing object representations that can be of many forms (graphs, vectors, combinations, etc). More details about dissimilarity matrix and how to use it will be described in Sects. 2.1 and 2.3.

In the latter case, i.e. dissimilarity vectors, Cha and Srihari (2000b) transform a seemingly insurmountable pattern recognition problem, in which the number of classes is very large or unspecified, into a two-class problem. This strategy is also known as dissimilarity in a dichotomous way. In that sense, dissimilarity vectors are made from feature vectors taken from two samples, by computing the difference between them. If those samples are assigned to the same class, the obtained dissimilarity vector is labeled as positive, otherwise the dissimilarity vector is labeled as negative. The classifier model obtained with dissimilarity vectors is supposed to distinguish whether a dissimilarity vector was made from feature vectors of samples from the same class or not. More details about dissimilarity vectors and how they can be used are found in Sects. 2.2 and 2.3. Considering several works published in the last years, we can point out that this approach has been presenting outstanding performance especially for classification problems with a large number of classes, notably when the number of samples per class is scarce.

In this work, we present a review on the two aforementioned dissimilarity based classification approaches. We describe a set of works that use those approaches, and discuss their advantages and drawbacks. The terms commonly used to refer to these approaches are not strongly standardized. Aiming to establish a uniform nomenclature to be used in this work, we will refer to the approach introduced by Pekalska and Duin as "Dissimilarity space", and the approach proposed by Cha and Srihari will be referred to as "Dissimilarity vectors".

The remainder of this paper is organized as follows: in Sect. 2 we present more details about the dissimilarity space and dissimilarity vectors approaches; in Sect. 3 we briefly describe the most remarkable works presented in literature considering the dissimilarity based approaches addressed here; in Sect. 4 we describe a critical review and perspectives on the research area for both dissimilarity approaches discussed here. Lastly, the final considerations are presented.

2 Basic concepts about dissimilarity

Humans perform pattern classification all the time, and they use to do it in different ways. However, is it possible to define what is the best way to assign class labels to the patterns we want to categorize? What is the first step to perform classification? What kind of attributes are the most discriminant? Or even, how accurate is our initial perception about similarities and differences between the patterns to be classified?

By analyzing similarities and differences between patterns, Pękalska et al. (2001) introduced the concept of dissimilarity. The rationale behind this concept is based on the way on how humans perform classification. Although the differences and similarities observation seems to be quite simplistic, some discriminative characteristics can be found in this way. In this vein, several works have been described taking into account the differences between samples of different classes in distinct scenarios (Cha and Srihari 2000b; Pavelec et al. 2008; Santini and Jain 1999).

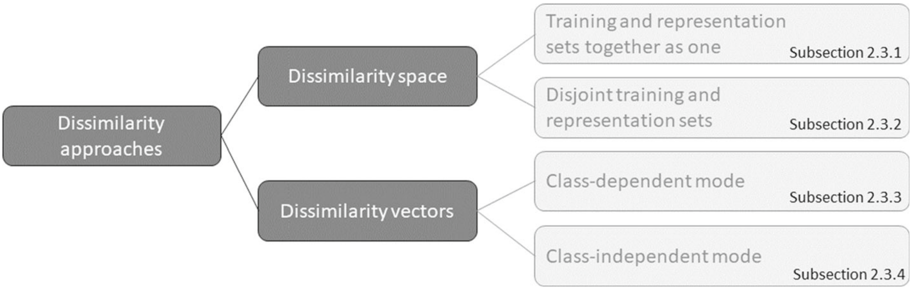


Fig. 1 Taxonomy of dissimilarity approaches described here

The concept of semblance between two objects can be expressed both by similarity and dissimilarity, but from different perspectives. Issues such as the type of data and the problem itself are important aspects that must be taken into account to define which one is more suitable to be used. In many cases, the proximity is a function of the observed variables or simply the measurements collected (Pękalska 2005).

The term "dissimilarity" has been used in different contexts and defined in different ways. However, as mentioned in Sect. 1, from our point of view it is possible to point out two main approaches regarding the use of dissimilarity for classification purposes. The first of these approaches is based on the construction of a dissimilarity matrix, presented by Pękalska and Duin. Details about this approach are presented in Sect. 2.1. In another direction, Cha and Srihari (2000a) described the use of dissimilarity vectors calculated from the difference between two feature vectors originally used to describe the patterns of the problem. Section 2.2 describes some details about this approach.

Figure 1 illustrates the taxonomy of the dissimilarity approaches described here. Note that both approaches may be used in different ways, depending on the properties of the available data. Details about how to use these approaches in all different scenarios conjectured here are described in Sect. 2.3.

2.1 Dissimilarity space

As far as we know, dissimilarity space, presented in Pękalska et al. (2001, 2002), is one of the best-known and widely used dissimilarity strategies. In general terms, the presented strategy consists of using a training dataset $T = \{x_1, x_2, \ldots, x_n\}$, composed of n patterns, and a representation dataset $R = \{p_1, p_2, \ldots, p_m\}$, composed of m patterns. R is a set of prototypes with samples taken from all the classes involved in the problem. Although T and R might be disjunct sets ($T \cap R = \emptyset$), the most common is to take R as a subset of T, i.e. $R \subseteq T$. From this, the learning step is conducted using the $n \times m$ distance matrix $D(T, R)$. In the $D(T, R)$ matrix, one row is assigned to each sample of T, and one column is assigned to each sample of R. The information in each cell of the matrix is given by the distance between the training sample assigned to its row and the sample from R assigned to its column, as described in Eq. 1.

$$D(T, R) = \begin{bmatrix} d(x_1, p_1) & d(x_1, p_2) & \ldots & d(x_1, p_m) \\ d(x_2, p_1) & d(x_2, p_2) & \ldots & d(x_2, p_m) \\ \ldots & \ldots & \ldots & \ldots \\ d(x_n, p_1) & d(x_n, p_2) & \ldots & d(x_n, p_m) \end{bmatrix} \quad (1)$$

The Dissimilarity Representation of a single pattern x_i is $D(x_i, R) = [d(x_i, p_1), d(x_i, p_2), \ldots, d(x_i, p_m)]^T$, where $d(\cdot, \cdot)$ is the dissimilarity between two patterns, commonly given by the Euclidean distance, described in Eq. 2.

$$d(x_a, x_b) = \sqrt{\sum_{h=1}^{V} \left(w_h^{x_a} - w_h^{x_b}\right)^2} \tag{2}$$

where $w_h^{x_a}$ and $w_h^{x_a}$ are the h-th features of patterns x_a and x_b respectively. In this case $x_a \in T$ and $x_b \in R$. On this point, the classifier model can be obtained by taking the rows of $D(T, R)$ matrix as feature vectors for training.

In order to better illustrate the concept and the creation of a dissimilarity matrix, we describe a visual example, using geometric shapes to represent five different classes. This example is particularly useful to illustrate classification problems in which the patterns have an intrinsic and detectable organization and it is difficult to obtain an efficient feature-based description to patterns for learning purposes. Figure 2 illustrates a square dissimilarity matrix created starting from five objects belonging to different classes of geometric shapes, presented in Fig. 2a. In Fig. 2b, we can see the superimposition of pairs of objects considering all the possible combinations, in the form of a dissimilarity matrix. Figure 2c shows the dissimilarity between each pair of objects highlighted, and in Fig. 2d we can see the similarities.

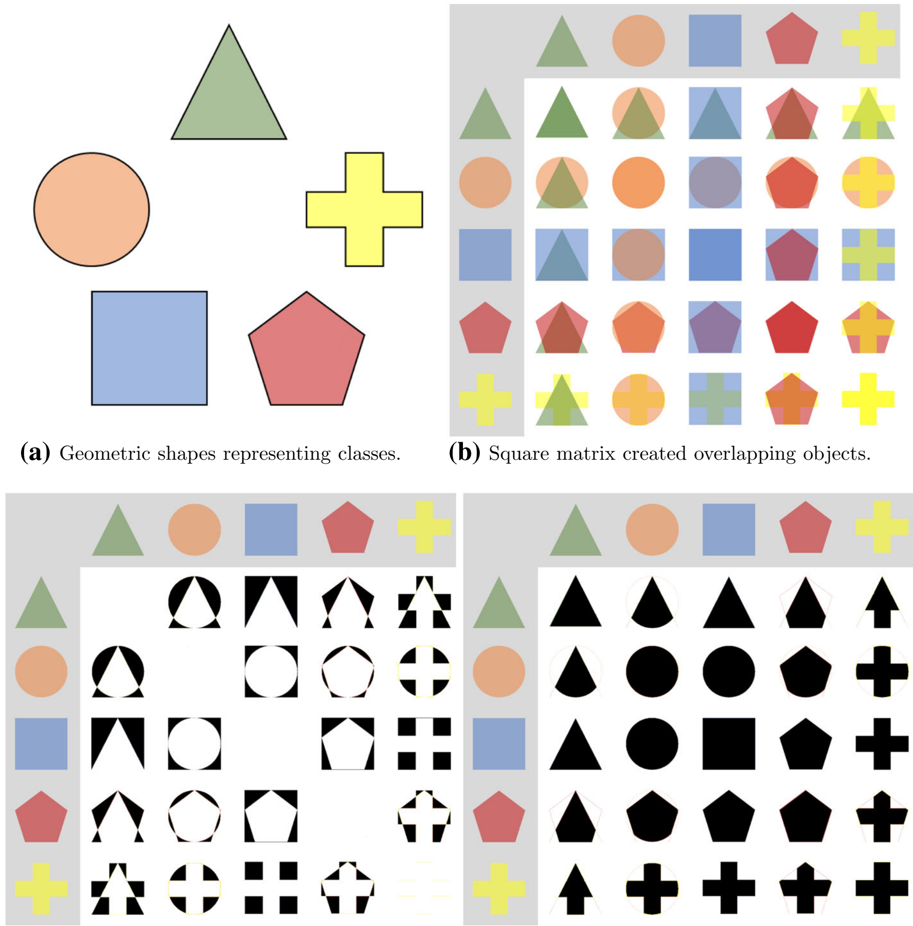
In most cases, dissimilarities are much more useful than similarities in this kind of situation. One can see that, in some situations, the similarities between two distinct pairs of objects, involving different classes, are exactly the same, and there are also other cases in which they are quite similar. Just in case, we decided to show also the similarities highlighted to clearly illustrate how more discriminative the dissimilarity usually is.

Last, but not least, we also depict in Fig. 3, originally presented in Pękalska et al. (2001), a 2D dissimilarity space. In this illustration, the first and third plots show the theoretical, artificial data, with a quadratic classifier, still in the original feature space. The representation set consists of two objects, i.e. $R = [p1, p2]$. The second and the fourth plots present the dissimilarity spaces $D(\cdot, R)$, where the representative objects are marked by circles on the first and third plots. One can note that if R is well chosen, a linear classifier on a dissimilarity kernel $D(\cdot, R)$ separates the data very well.

2.2 Dissimilarity vectors

One of the most remarkable contributions of Cha and Srihari (2000b) was the conversion of a K-class problem into a binary problem through a dichotomy transformation, by using what here we call dissimilarity vectors. Although naturally appropriate to deal with binary problems, the dissimilarity vectors approach somehow has been presenting good results also on large multi-class problems, as writer manuscript identification (Bertolini et al. 2013), signature verification (Bertolini et al. 2010), bird species identification (Zottesso et al. 2018), forest species recognition (Martins et al. 2015), and author identification (Pavelec et al. 2008), in which each writer or specie must be considered a particular class.

Cha and Srihari presented the dissimilarity vector as a vector calculated from the difference between two feature vectors. Dissimilarity vectors are so labeled as within class (or positive class—$\oplus$) or between class (or negative class—$\ominus$) whether the features vectors used to make the dissimilarity vector belong to the same class or not, respectively. By this way, the classifier created is expected to be able to distinguish if two samples belong to same class or not.



(a) Geometric shapes representing classes. **(b)** Square matrix created overlapping objects.

(c) Dissimilarities between objects. **(d)** Similarities between objects.

Fig. 2 Illustration of (dis)similarity matrices created using objects from five different classes

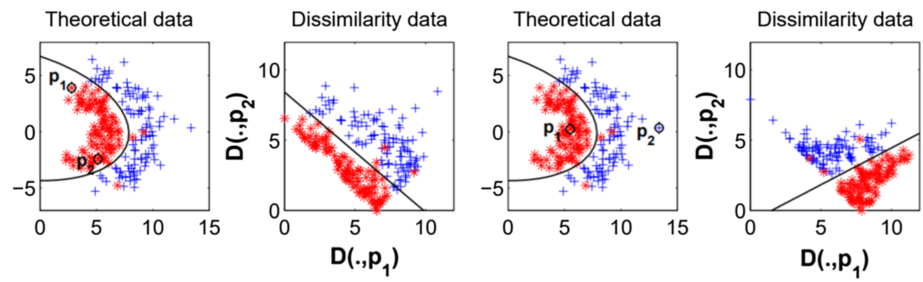
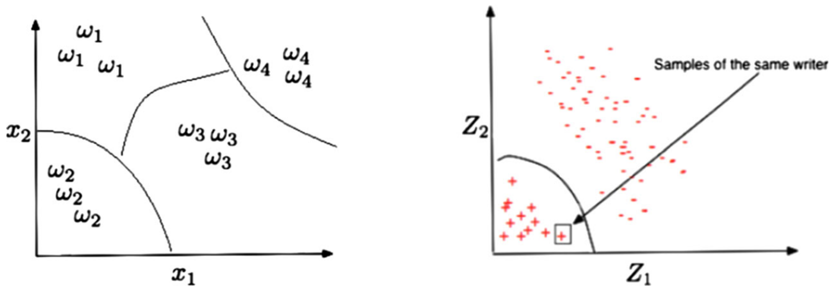


Fig. 3 Illustration of a 2D dissimilarity space (Pękalska et al. 2001)



(a) Samples in the original feature space of a four classes problem.

(b) Samples in the dissimilarity space after a dichotomy transformation.

Fig. 4 Transformation from a four classes problem to a binary problem in the dissimilarity space . Adapted from Bertolini et al. (2013)

Figure 4 illustrates the transformation of a four classes problem to a binary problem in the dissimilarity space. Suppose we have a writer identification problem, in which we intend to correctly identify the person who wrote a given manuscript. Considering that we have four different writers involved in the problem, we would have a four classes problem. In this case, writers ω_1, ω_2, ω_3, and ω_4, and each one of them provides three samples. The feature extraction process extracts a vector $(X1, X2)$ from each sample, and the representation of these samples in the original feature space in which the patterns are represented is shown in Fig. 4a. In Fig. 4b we can see the representation of dissimilarity vectors obtained from the same instances after a dichotomization. The dichotomy transformation takes place and computes the dissimilarity between the features of each pair of samples to form vectors $(Z1, Z2)$. These vectors, which we call dissimilarity feature vectors, are shown in Fig. 4b. In this case, the dissimilarity vectors made from feature vectors of the same class are represented by $x_{\oplus}$, while negative dissimilarity vectors are represented by $x_{\ominus}$.

Let d_{ij} denote the jth sample of the class ith, where n is the number of classes and m is the number of samples available per class. Let us assume that $|.|$ is the absolute value, the dissimilarity feature vector ($x_{\oplus}$ and $x_{\ominus}$) has the same dimensionality of d_{ij}. In this case, we could define $x_{\oplus}$ by Eq. 3, and $x_{\ominus}$ by Eq. 4, as follows.

$$x_{\oplus} = |d_{ij} - d_{ik}| \ \textit{in which} \ i = 1 \ \textit{to} \ n, \quad j, k = 1 \ \textit{to} \ m \ \textit{and} \ j \neq k \tag{3}$$

$$x_{\ominus} = |d_{ij} - d_{kl}| \ \textit{in which} \ i, k = 1 \ \textit{to} \ n, \ i \neq k \ \textit{and} \ j, l = 1 \ \textit{to} \ m \tag{4}$$

By this way, one can transform the problem from its original representation in the feature space to a dichotomic representation, in a dissimilarity space. In this new representation, the classes refer to within class distances and between classes distances.

Once we have performed the aforementioned dichotomic transformation, it is desirable that both classes ($x_{\oplus}$ and $x_{\ominus}$) describe their actual classes in the feature space, i.e. $x_{\oplus}$ samples must be close to the origin of the coordinate axis, while $x_{\ominus}$ samples must be as far as possible. However, one cannot ensure that, because several factors may affect the classes representation in the new space, as a high standard deviation between samples from the same class, or even a high similarity between samples from different classes. Such a behaviour may cause confusion during the domains transposition. One can observe the occurrence of this situation by carefully looking at Fig. 4b. In the next subsection, we describe some toy problems considering the most common ways of use for both dissimilarity approaches described in this section.

Table 1 Dissimilarity matrix for dissimilarity space Example 1

Sample	Tr1	Tr2	Tr3	Tr4	Tr5	Tr6	Tr7	Tr8	Tr9	Class
Tr1	0.00	0.93	1.22	5.89	4.85	5.22	11.30	11.15	9.44	a
Tr2	0.93	0.00	1.13	5.79	4.64	5.14	11.27	11.00	9.32	a
Tr3	1.22	1.13	0.00	6.86	5.45	6.19	12.26	12.03	10.35	a
Tr4	5.89	5.79	6.86	0.00	3.60	1.23	5.74	5.66	3.81	b
Tr5	4.85	4.64	5.45	3.60	0.00	3.13	7.55	7.19	6.10	b
Tr6	5.22	5.14	6.19	1.23	3.13	0.00	6.56	6.06	4.82	b
Tr7	11.30	11.27	12.26	5.74	7.55	6.56	0.00	3.18	2.57	c
Tr8	11.15	11.00	12.03	5.66	7.19	6.06	3.18	0.00	3.94	c
Tr9	9.44	9.32	10.35	3.81	6.10	4.82	2.57	3.94	0.00	c

2.3 Toy problems

To provide a better insight of practical issues of the dissimilarity approaches discussed in this paper, we present toy problems for both approaches. First we describe two examples for the so called dissimilarity space approach, described in Sect. 2.1. Then, we present two other examples for the dissimilarity vectors approach, described in Sect. 2.2. Thus, we cover the four cases depicted in Fig. 1 through a set of comprehensive examples.

2.3.1 Dissimilarity space—Example 1

In this first example, we intend to show how the dissimilarity space approach can be used considering the use of only two datasets: training and test. In this case, the resulting dissimilarity matrix will be a square matrix $n \times n$ sized, where n is the cardinality of the training set.

Let us consider a training set T and a test set T_s composed of patterns belonging to classes a, b and c. Consider also the following composition for these sets: $T = \{(Tr1, a), (Tr2, a), (Tr3, a), (Tr4, b), (Tr5, b), (Tr6, b), (Tr7, c), (Tr8, c), (Tr9, c)\}$, where each pair (Tri, x) refer to a training sample assigned to a label $x \in \{a, b, c\}$; and $T_s = \{(Te1, a), (Te2, a), (Te3, b), (Te4, b), (Te5, c), (Te6, c)\}$, where each pair (Tei, x) refer to a test sample assigned to a label $x \in \{a, b, c\}$.

As aforementioned, in this example we have a square dissimilarity matrix, in which we have each row and each column assigned to one pattern of the training set. The cells of the matrix are filled with a distance measure between the patterns assigned to its row and to its column. Table 1 shows a hypothetical dissimilarity matrix obtained from the training patterns of this example. As one can see, the obtained matrix is a symmetric matrix and all the cells in the main diagonal have a value of zero. As described in Sect. 2.1, the rows of this matrix are used as feature vectors to train a statistical classifier.

Once we got the classification model, we use the feature vectors described in Table 2, obtained computing distances between testing patterns and training patterns, as entrance to a classifier.

Table 2 Feature vectors for test patterns for dissimilarity space Example 1

Sample	Tr1	Tr2	Tr3	T4	Tr5	Tr6	Tr7	Tr8	Tr9	Class
Te1	0.85	1.08	0.73	6.57	5.33	5.83	12.02	11.73	10.16	a
Te2	0.92	0.62	0.85	6.22	5.03	5.50	11.71	11.40	9.78	a
Te3	3.80	3.46	4.56	2.58	2.65	2.02	8.08	7.66	6.07	b
Te4	6.15	5.76	6.88	2.06	4.06	2.52	6.67	6.26	4.40	b
Te5	8.26	8.10	9.11	2.91	4.40	3.55	3.54	3.62	2.05	c
Te6	8.11	7.91	8.94	2.79	4.26	3.40	3.88	3.70	2.26	c

Table 3 Dissimilarity matrix for dissimilarity space Example 2

Sample	Re1	Re2	Re3	Class
Tr1	1.22	5.22	9.44	a
Tr2	1.13	5.14	9.32	a
Tr3	6.86	1.23	3.81	b
Tr4	5.45	3.13	6.10	b
Tr5	12.26	6.56	2.57	c
Tr6	12.03	6.06	3.94	c

2.3.2 Dissimilarity space—Example 2: non-overlapping training and representation sets

In this example, we want to demonstrate that the Dissimilarity Space approach can be used taking into account a representation set R independent from the training set T, or disjunct sets, i.e. $T \cap R = \emptyset$.

Let us consider a training set T, a representation set R, and a test set T_s composed of patterns belonging to classes a, b and c. Consider also the following composition for these sets: $T = \{(Tr1, a), (Tr2, a), (Tr3, b), (Tr4, b), (Tr5, c), (Tr6, c)\}$, where each pair (Tri, x) refer to a training sample assigned to a label $x \in \{a, b, c\}$; $R = \{(Re1, a), (Re2, b), (Re3, c)\}$, where each pair (Rei, x) refer to a representation sample assigned to a label $x \in \{a, b, c\}$; and $T_s = \{(Te1, a), (Te2, a), (Te3, b), (Te4, b), (Te5, c), (Te6, c)\}$, where each pair (Tei, x) refer to a test sample assigned to a label $x \in \{a, b, c\}$.

Next, we compute the dissimilarity matrix by calculating the distances between samples of training and representation sets. Observe that, in this case, we have six rows and three columns in the matrix, since these are the number of samples found respectively in the training and representation sets.

The rows of Table 3 are used as feature vectors to train a classifier. After the classification model is obtained, we classify the test feature vectors described in Table 4, obtained by computing distances between samples from test set and representation set.

2.3.3 Dissimilarity vectors—Example 1

As aforementioned, the dissimilarity approach proposed by Cha and Srihari (2000a), also called dissimilarity vector in this work, can be used both in class-dependent and in class-independent modes. By this way, we decided to exemplify both situations here. The Example 1 refers to the class-dependent mode, while Example 2 is a class-independent case.

Table 4 Feature vectors for test patterns for dissimilarity space Example 2

Sample	Re1	Re2	Re3	Class
Te1	0.73	5.83	10.16	a
Te2	0.85	5.50	9.78	a
Te3	4.56	2.02	6.07	b
Te4	6.88	2.52	4.40	b
Te5	9.11	3.55	2.05	c
Te6	8.94	3.40	2.26	c

Table 5 Feature vectors of training, reference, and test sets for dissimilarity vectors Example 1

Sample	Features					Class
	F1	F2	F3	F4	F5	
Train						
Tr1	1.60	1.56	1.57	1.27	0.52	a
Tr2	1.29	1.65	1.20	1.41	1.29	a
Tr3	0.88	1.12	0.69	1.36	0.54	a
Tr4	4.06	4.48	4.50	2.72	3.57	b
Tr5	2.41	2.50	3.49	4.88	2.81	b
Tr6	3.83	3.30	4.51	2.52	3.53	b
Tr7	4.76	5.22	5.51	4.88	4.85	c
Tr8	5.12	5.32	5.29	5.08	4.81	c
Tr9	5.05	5.14	4.82	4.76	5.11	c
Reference						
Re1	1.19	0.92	1.25	1.07	0.59	a
Re2	1.38	1.12	1.05	1.21	1.10	a
Re3	2.71	3.06	2.86	2.44	3.33	b
Re4	3.25	4.88	3.28	2.54	4.94	b
Re5	5.34	5.24	4.62	4.82	4.81	c
Re6	5.52	4.89	5.36	5.29	4.58	c
Test						
Te1	0.62	0.94	0.94	0.79	1.67	a
Te2	3.17	3.66	2.85	3.54	3.70	b
Te3	5.16	5.14	5.03	4.90	4.72	c

Table 5 presents the feature vectors of training set, reference set, and test set of the dissimilarity vectors class-dependent example.

Next, we describe in Table 6 the positive and negative samples created using the training vectors of the Table 5, calculated with Eqs. 3 and 4 respectively. These positive and negative samples will be used in the dissimilarity training.

Once we have obtained the positive and negative dissimilarity vectors, a classifier model is created. After that, the dissimilarity vectors for test, described in Table 7 are created by using the reference samples, for which we already know the actual labels. By this way, dissimilarity vectors are created between a given test pattern and reference samples from all the classes involved in the problem. Finally, the label of the reference sample used to make

Table 6 Positive and negative dissimilarity vectors for Examples 1 and 2

Sample	Features				
	F1	F2	F3	F4	F5
Positive samples					
(Tr1, Tr2)	0.31	0.09	0.37	0.15	0.77
(Tr1, Tr3)	0.72	0.45	0.88	0.09	0.02
(Tr2, Tr3)	0.41	0.53	0.51	0.06	0.75
(Tr4, Tr5)	1.66	1.99	1.00	2.16	0.76
(Tr4, Tr6)	0.23	1.19	0.01	0.20	0.04
(Tr5, Tr6)	1.43	0.80	1.01	2.36	0.72
(Tr7, Tr8)	0.36	0.10	0.22	0.20	0.04
(Tr7, Tr9)	0.29	0.08	0.69	0.12	0.26
(Tr8, Tr9)	0.07	0.19	0.47	0.32	0.30
Negative samples					
(Tr1, Tr4)	2.47	2.92	2.93	1.45	3.05
(Tr2, Tr5)	1.12	0.85	2.30	3.47	1.52
(Tr3, Tr6)	2.96	2.18	3.82	1.16	2.99
(Tr1, Tr7)	3.17	3.65	3.94	3.62	4.33
(Tr2, Tr8)	3.84	3.67	4.09	3.67	3.52
(Tr3, Tr9)	4.17	4.02	4.13	3.41	4.57
(Tr4, Tr7)	0.70	0.73	1.01	2.16	1.28
(Tr5, Tr8)	2.72	2.82	1.80	0.20	2.00
(Tr6, Tr9)	1.22	1.84	0.31	2.24	1.58

the dissimilarity vector which maximizes the value for positive class will be taken as the final decision for a given test sample.

2.3.4 Dissimilarity vectors—Example 2: class independent mode

From now, we start to describe a class independent example using dissimilarity vectors. Observe that, in this case, the training step will be developed using samples from classes a, b, and c, and the test will be made by using reference and test set composed of samples belonging to other two classes: d and e. Table 8 presents the feature vectors of the training, reference and test sets.

Considering that the training set of the Example 2 is the same already used in the Example 1, the positive and negative dissimilarity vectors for Example 2 are the same of the Example 1, described in Table 6. Following, we present in Table 9 the dissimilarity vectors of the test set for Example 2.

3 State of the art

This section presents, predominantly in chronological order, some works that somehow have used one of the dissimilarity approaches described in Sect. 2. Our goal is to describe some different application domains in which those approaches has been successfully employed. This section was divided in two subsections, each related to one approach.

Table 7 Dissimilarity vectors of test set for Example 1

Combined samples	Features				
	F1	F2	F3	F4	F5
Positive vectors					
(Te1, Re1)	0.57	0.02	0.31	0.29	1.08
(Te1, Re2)	0.76	0.19	0.11	0.42	0.57
(Te2, Re3)	0.46	0.60	0.01	1.10	0.37
(Te2, Re4)	0.08	1.21	0.43	1.00	1.24
(Te3, Re5)	0.18	0.09	0.42	0.07	0.09
(Te3, Re6)	0.36	0.26	0.33	0.39	0.15
Negative vectors					
(Te1, Re3)	2.09	2.13	1.92	1.66	1.65
(Te1, Re4)	2.63	3.94	2.33	1.76	3.27
(Te1, Re5)	4.72	4.30	3.67	4.04	3.14
(Te1, Re6)	4.90	3.95	4.42	4.50	2.90
(Te2, Re1)	1.98	2.75	1.60	2.47	3.11
(Te2, Re2)	1.79	2.54	1.79	2.34	2.60
(Te2, Re5)	2.17	1.57	1.77	1.28	1.11
(Te2, Re6)	2.35	1.22	2.51	1.74	0.87
(Te3, Re1)	3.97	4.23	3.78	3.82	4.13
(Te3, Re2)	3.78	4.02	3.98	3.69	3.62
(Te3, Re3)	2.71	3.06	2.86	2.44	3.33
(Te3, Re4)	3.25	4.88	3.28	2.54	4.94

3.1 Dissimilarity space

Since the early 2000s, a lot of efforts have been devoted towards investigation of the use of dissimilarity in pattern classification. In Pękalska et al. (2001) examine two different ways to build classifiers using dissimilarity. In the first approach, the dissimilarity was embedded in a pseudo-Euclidean space. In the second approach, the classifiers were built on distance kernels. Experiments were accomplished on a 2-class digit recognition problem and on two 6-class shape recognition problems. The investigations showed that in cases in which the 1-NN rule presents a poor performance, more complex approaches based on dissimilarity kernels perform better. Regarding pseudo-Euclidean spaces, the influence of noise can be reduced by employing dimensionality reduction.

Following the investigations, Pękalska et al. (2002) used a dissimilarity representation to address classification on two versions of the NIST digit sets, with 10 classes each, and on the chromosome banding profile task, with 24 classes. In that work, the authors introduce the use of a "dissimilarity matrix" constructed based on the distances between the samples of the train set and the samples of a representation set, as described in Sect. 2.1. The authors evaluate the use of linear and quadratic classifiers built on dissimilarity representations and compare their performance to k-NN classifiers. The experiments carried out showed that linear classifiers created from dissimilarities most of the time perform better than k-NN rule based on the same representation set. Regarding quadratic classifiers, they can perform better yet if there is a reliable way to estimate the class covariance matrices.

Table 8 Feature vectors of training, reference, and test sets for dissimilarity vectors Example 2

Sample	Features					Class
	F1	F2	F3	F4	F5	
Train						
Tr1	1.60	1.56	1.57	1.27	0.52	a
Tr2	1.29	1.65	1.20	1.41	1.29	a
Tr3	0.88	1.12	0.69	1.36	0.54	a
Tr4	4.06	4.48	4.50	2.72	3.57	b
Tr5	2.41	2.50	3.49	4.88	2.81	b
Tr6	3.83	3.30	4.51	2.52	3.53	b
Tr7	4.76	5.22	5.51	4.88	4.85	c
Tr8	5.12	5.32	5.29	5.08	4.81	c
Tr9	5.05	5.14	4.82	4.76	5.11	c
Reference						
Re1	2.10	1.96	1.96	2.28	2.06	d
Re2	2.04	2.32	1.84	2.14	2.17	d
Re3	2.23	2.07	1.94	2.00	1.95	d
Re4	5.14	5.52	4.87	5.04	4.98	e
Re5	4.61	4.91	4.64	5.17	4.82	e
Re6	5.28	4.79	4.91	4.95	5.22	e
Test						
Te1	1.95	2.04	1.77	1.77	2.02	d
Te2	5.02	4.89	5.06	4.88	5.10	e

Table 9 Dissimilarity vectors of the test set for Example 2

Combined samples	Features				
	F1	F2	F3	F4	F5
Positive samples					
(Te1, Re1)	0.15	0.08	0.19	0.51	0.04
(Te1, Re2)	0.09	0.27	0.07	0.37	0.15
(Te1, Re3)	0.27	0.03	0.17	0.23	0.07
(Te2, Re4)	0.12	0.63	0.19	0.16	0.11
(Te2, Re5)	0.41	0.02	0.42	0.29	0.28
(Te2, Re6)	0.26	0.10	0.15	0.07	0.12
Negative samples					
(Te1, Re4)	3.19	3.47	3.10	3.27	2.96
(Te1, Re5)	2.66	2.87	2.87	3.40	2.80
(Te1, Re6)	3.32	2.75	3.14	3.18	3.20
(Te2, Re1)	2.92	2.93	3.10	2.60	3.04
(Te2, Re2)	2.98	2.57	3.22	2.74	2.93
(Te2, Re3)	2.79	2.82	3.12	2.88	3.15

Pękalska and Duin (2006) have also investigated the circumstances in which dissimilarity-based techniques are suitable for deriving classifiers in feature vectors spaces. In order to create the dissimilarity space, the authors assume a training set T composed of N patterns and a representation set $R = \{p_1, p_2, \ldots, p_n\}$ composed of n prototypes. The patterns $x \in T$ are described by a vector of distances between x and the prototypes from R. Once we have a vector representation, several traditional classifiers established in vector spaces can be used. The authors described a series of experiments on ten different datasets taken from the UCE repository,[1] in which the features are categorical, continuous and mixed. For categorical or mixed features, city block distance was used, and for continuous features, the Euclidean distance was chosen. The authors assessed three different classifiers on dissimilarity spaces: a normal density based linear classifier (NLC); a normal density based quadratic classifier (NQC); and a logistic linear classifier (LOGC). Finally, the authors present a two-fold conclusion: (i) NLC and NQC dissimilarity-based classifiers perform better than or equal to the best NN rule applied on the full original training set; (ii) NLC and NQC dissimilarity-based classifiers are indicated when the data have categorical or mixed variables, or when there is a high class overlap probability.

Pękalska et al. (2006) investigate the use of prototype selection for the development of dissimilarity-based classifiers. In that work, the dissimilarity representation was considered as a vector space. Experiments were made on twelve dissimilarity representations obtained from seven different datasets, most of them regarding binary classification problems, and few of them regarding multi-class problems. The authors claim that a better classification performance can be obtained, both in terms of accuracy and speed, if just a few suitable prototypes are selected instead of using all the training set. They still conclude that: (i) when the classification is performed using a quadratic function (BayesNQ), very few prototypes (3–12% of the training set) are enough to get a performance comparable to that obtained by using k-NN on the entire training set; (ii) for two-class problems, BayesNQ performs significantly better than k-NN when the representation set is large (20% for instance); (iii) in general, a well designed strategy for prototype selection works better than a random selection.

Nguyen et al. (2006) described a new strategy to address content based image retrieval (CBIR) by learning dissimilarity for interactive search. The dissimilarity was adjusted in a prototype-based dissimilarity space, as proposed in Pękalska et al. (2002), by using relevance feedback. The dissimilarity was computed from contexture feature vectors (Van Gemert et al. 2006), which combines color and texture. Experiments were developed on two image collections. The first one, Corel collection, which contains 10,000 images on 100 different non-overlapping categories. The second, taken from TrecVid 2005 benchmark,[2] containing 43,907 images extracted from news videos. It was defined 29 categories to perform the classification. The authors concluded that, in general, learning on the dissimilarity space performs better than learning on the feature space, since a proper number of initial prototypes is provided.

In the context of the ICPR 2010 Classifier Domains of Competence contest, a research competition aimed at finding out the relation between data complexity and the performance of learners, Duin et al. (2010) compare dissimilarity-based classifiers with traditional feature-based classifiers. Experiments were conducted on 301 datasets of the contest and in a large subset of its datasets, using a set of classifiers based on dissimilarity representation. The authors concluded that the feature-based dissimilarity space classification performs better than or comparable to the linear and nonlinear SVMs.

[1] http://www.ics.uci.edu/.

[2] https://www-nlpir.nist.gov/projects/trecvid/trecvid.data.html#tv05.

In Duin and Pękalska (2012), the authors consider the dissimilarity representation as a strategy to bridging the gap between statistical and structural approaches for pattern recognition tasks. In that sense, the authors advocate that objects can be compared by using structural descriptions. By this way, vectors can be derived from pairwise dissimilarities aiming to use a statistical learning method. In this context, some important aspects concerning to object representation are discussed, as feature representation and feature space (including structural representation, dissimilarity representation, kernel representation, pseudo-Euclidean space, kernel space, dissimilarity space, and class models). In the conclusion, the authors point out that there are a lot of room for improvement regarding dissimilarity measures proposal, because there is not a definitive way on how to make it.

In Theodorakopoulos et al. (2013), the authors investigate for the first time the performance of the collaborative representation-based classification schemes' in the dissimilarity space. For this purpose, they used the datasets and experimental setup from previous works of Pekalska.

Theodorakopoulos et al. (2014b) present an alternative classifier based on dissimilarity space to boost the classification of real time pose estimation. For that purpose, the authors proposed a method for action recognition by using a set of robust and invariant pose representations. The coordinate system was created considering skeletal data and Euler angles from some particular skeletal primitives, such a way that human motion is taken as a progression of poses, in a multidimensional feature space. After that, the dissimilarity space was created considering a set of prototype actions, as proposed by Pčkalska and Duin (2005). The recognition was accomplished in the dissimilarity space. Experiments were conducted on UPCV action dataset, MSR Action3D dataset, and UTKinect-Action dataset. The proposed method presented performance equal or better when compared to other methods on all the evaluated datasets. Finally, the authors claims that the method is suitable to overcome some limitations of low-cost commercial RGB-D sensors.

In another work, Theodorakopoulos et al. (2014a) have presented a distinguished way to address a quite unique problem, the HEp-2 cells classification starting from fluorescence images. The automatic classification of HEp-2 patterns on fluorescence images is useful in the context of autoimmune diseases detection. In the proposed framework, the authors combined descriptors to capture both gradient and textural characteristics of the patterns, i.e. SIFT and GoC-LBP, using sparse representation into dissimilarity space. The use of sparse representations in the dissimilarity space was one of the central aspects of that work, and the dissimilarity was applied as proposed in Pękalska et al. (2006). Lastly, the authors claim that the experimental results showed that the proposed method is very efficient, outperforming all the methods submitted to a recent contest on HE-2 cells classification, obtaining 75.1% accuracy in cell-level classification and 85.7% accuracy in image level classification.

In Bunke and Riesen (2008) and Riesen and Bunke (2009), the authors address graph embedding using dissimilarity representations. In that vein, an embedding procedure which allows to map graphs to a vector space is used. Thus, statistical pattern recognition methods can be adopted, as well as the dissimilarity approach described in Pčkalska and Duin (2005). More recently, in Livi et al. (2014), the authors discuss the optimization of the dissimilarity space representation embedded for label graphs.

Another work that employed dissimilarity approach was published by Pinheiro et al. (2017). The focus of that work is dissimilarity representation and multiple classifier systems in text categorization systems. The dissimilarity approach was employed like proposed by Pękalska et al. (2002). The authors used bag-of-words to represent a document and the dissimilarity space because of its universality. The universality is the capability of using the dissimilarity representation with any measure and with any classifier that requires a feature vector as input. In that case, two measures (cosine and Euclidean) were used. Multiple classifiers were trained on data from different dissimilarity spaces such a way that these different spaces were combined. The classification was performed using SVM with linear kernel and the final decision was obtained by using majority vote. The authors used 47 text categorization databases and they claim that the proposed approach obtained results better than those presented in the literature.

Table 10 summarizes some information about related works described in this subsection. Although the dissimilarity space approach has not been intensely used in recent years to directly address classification tasks, as one can see in Table 10, we must consider that it has still been a great source of inspiration for many works recently published.

3.2 Dissimilarity vector

Oliveira et al. (2007) employed a writer-independent approach to perform off-line signatures verification (SV). The proposed approach is based on the concept of dichotomization introduced by Cha and Srihari (2000a). The authors observed some coincidences between the way how forensic experts use to perform signature verification and the way in which dissimilarity works. The authors also evaluated the impact of combining classifiers in ROC (Receiver Operating Characteristic), and the combination based on the maximum fusion rule performed better then the combination based on majority vote rule. Experiments were accomplished on a database composed of samples made by 100 writers using SVM classifier. In conclusion, it was possible to observe a performance improvement over false rejection and false acceptance.

Pavelec et al. (2008) employed the strategy presented by Cha and Srihari (2000a) on the author identification task. The authors described a comparison between the Writer-independent (WI) approach proposed in that work and the pairwise approach, which results on a Writer-dependent (WD) strategy. Experiments were conducted on a database composed of 30 short manuscript texts written by 20 different people. The manuscripts content were taken from the online edition of two Brazilian newspapers. A feature vector composed of 171 features, obtained from conjunctions and adverbs, was used. In the WD approach, the authors used 10 documents from each writer for training and other 10 documents for test. In the WI approach, the same 10 samples were used for test and five samples per author were used on training set. In this case the WD approach presented better results, 83.2% versus 75.1% of the WI approach. However, it is worth of mention that even with better performance, the WD approach presents a higher computational cost. On the other hand, the WI approach with lower performance rates has a very accessible computational cost. Anyway, if the number of classes is small, the WD approach is an interesting approach, while the WI approach should be considered if there are a large number of classes.

Hanusiak et al. (2012) evaluated the dissimilarity approach on the writer verification task using handwritten texts. The proposed approach is based on the dissimilarity framework proposed by Cha and Srihari (2000a). In this case, starting from a handwritten text the authors want to verify whether or not the text was written by a determined writer, i.e. verification task, and not to identify who wrote the text. The work presents the idea of combining classifiers through ROC curves using textural content taken from the manuscripts to represent handwriting. A database composed of samples written by 315 people was used. Three samples (manuscript letters) were authored by each writer, always containing the same content (text-dependent). The authors evaluated only two dissimilarity parameters. However they used the concept of writer-independent, in which classes used in the training set are not

Table 10 Summary of the works that uses dissimilarity space approach

References	Application domain/dataset	Number of classes
Pękalska et al. (2001)	Digit recognition	2
	Shape recognition	6
Pękalska et al. (2002)	NIST digit set	10
	Chromosome banding profile	24
Pękalska and Duin (2006)	UCE repository datasets:	
	Australian	2
	Biomed	2
	Breast (Wisconsin)	2
	Diabetes	2
	Heart	2
	Ecoli	3
	Glass	4
	Ionosphere	2
	Liver	2
	Musk	2
	Sonar	2
	Wine	3
Pękalska et al. (2006)	Polydisth	2
	Polydistm	2
	NIST-38	2
	Zongker-12	2
	RoadSign	2
	GeoSam	2
	GeoShape	2
	Wine	3
	Ecoli-p1	3
	Ecoli-p08	3
	ProDom	4
	Zongker-all	10
Nguyen et al. (2006)	Corel collection	100
	TrecVid 2005 benchmark	29
Duin et al. (2010)	301 datasets provided by ICPR 2010 Classifier Domains of Competence contest	–
Duin and Pękalska (2012)	Pen digits dataset	10
Theodorakopoulos et al. (2014b)	UPCV action dataset	10
	MSR Action 3D dataset	20
	UT-KinectAction dataset	10
Theodorakopoulos et al. (2014a)	HEp-2 cells images dataset	5
Bunke and Riesen (2008)	Letter	15
	NIST-4	4
	AIDS	2
	Webgraph	20

Table 10 continued

References	Application domain/dataset	Number of classes
Riesen and Bunke (2009)	Letter database	15
	GREC database	32
	Image database	4
	COIL-100 database	100
	Web database	20
	Fingerprint database	4
	AIDS database	2
	Multagenicity database	2
Livi et al. (2014)	Synthetic data	2
	Letter LOW	15
	Letter MED	15
	Letter HIGH	15
	AIDS	2
	Proteins	6
	GREC	22
	Multagenicity	2
Pinheiro et al. (2017)	47 text categorization databases	2–120

necessarily part of test set. The presented approach is quite interesting because new classes can be added without the need for retrain a new model. In the experiments performed, 200 writers were used on the training set and other 115 on the test. Five texture blocks 800 × 600 sized were created on the manuscript images. The authors used the well-known Gray Level Co-occurrence Matrix (GLCM) texture descriptor, proposed by Haralick et al. (1973), evaluating different statistical measures extracted from the matrix. The impact of increasing the number of writers in the training set, combining the different classifiers through ROC curves was evaluated, among others. The best obtained rate was 97.1% of global hit, similar to the state of the art.

Swanepoel and Coetzer (2012) addressed off-line SV by using dissimilarity, on a WI way. In that case, the dissimilarity approach differs from what had been described so far, leading to a method that the authors called Dissimilarity Normalization. The WI concept explored in that work is similar to that proposed by Cha and Shirari where the purpose is to create an universal binary classifier. The authors used discrete Radon Transform and Dynamic Time Warping in the dissimilarity space, and the experiments were conducted using the Dolfing's dataset containing 1530 genuine signatures and 3000 skilled forgeries. Briefly summarizing, the authors claim that the use of dissimilarity in conjunction with the proposed feature extraction method is a successful model to address that task.

Eskander et al. (2013) proposed a hybrid system using WI, on the basis of dissimilarity approach, and WD to carry out the SV task. The authors address the lack of samples for training on writer identification task, a problem commonly encountered in several application domains. According to the authors, the dissimilarity approach is a good alternative to tackle problems when only few samples are available. In this vein, the authors propose to use one approach or another depending on the number of samples available for training. A universal classifier is trained and this is used when no user sample is available to train a WD classifier.

The system may change between both approaches depending on the availability of user samples being sufficient or not for training. Using two databases, Eskander et al. (2013) concluded that the proposed approach presents results comparable to the literature, with the benefit of a low complexity in the classification step.

Rivard et al. (2013) employed the approach proposed by Cha and Srihari (2000a) on off-line SV. According to the authors, the dissimilarity proposal contributes to this task due to two main aspects: firstly, the number of classes, and secondly, the low number of samples per class. Because of this, dissimilarity may be a robust approach in tasks with such characteristics. SV systems generally have millions of classes, the dissimilarity approach as presented by Cha and Srihari transforms a multi-class problem into a binary problem, as described in Sect. 2.2, this can contribute to improve the system performance. Another characteristic in SV is that writers rarely give in a large number of signature samples. With this motivation, Rivard et al. (2013) employed dissimilarity in their work. However, the main contribution of that work is the use of boosting feature selection to design low cost classifiers which automatically select features during the training step. The database employed is composed of samples from 168 authors in which each author yields 40 signatures. From this total, 108 authors were used for development and 60 authors for exploitation. Thirty genuine signatures were used for the learning set and 10 for validation. Extended Shadow Code (ESC) and Directional Probability Density Function (DPDF) features were extracted from the signature database. The authors present a series of experiments focused on boosting feature selection, cardinality of reference set and ESC and DPDF features. Finally, they present the overall error rates between 7.24% and 5.19%, in the best cases.

Okawa and Yoshida (2013) addressed online writer verification by using dissimilarity vectors made with features of pen pressure information obtained from infrared images, Gray Level Co-occurrence Matrix (GLCM) texture features, and Local Directional pattern (LDP) features. Dissimilarity vectors were submitted to SVM classifier, and the authors performed experiments on a dataset composed of samples made by 54 volunteers. Obtained results show that the use of pen pressure information combined with LDP allows for a reduction of the error rate from 10 to 4.6%.

The dissimilarity approach was employed by Martins et al. (2015) in the forest species recognition task. The strategy was based on the Cha and Shirari proposal, by transposing a n-class problem into a binary one. The authors used dissimilarity in conjunction with texture descriptors and SVM classifier. That work was focused on the evaluation of the impact of the use of dynamic classifier selection strategies in the investigated task. Experiments were accomplished assessing dynamic classifier selection strategies on a database composed of 2240 microscopic images from 112 forest species using an ensemble of classifiers created with 10 different texture descriptors. A considerable improvement on the performance was achieved in different scenarios. In addition, the authors claim that the use of the dissimilarity approach in this task was unprecedented and the idea of using disjoint training and test sets is quite suitable to address this problem, such a way that there is no need to retrain the model when new classes are inserted.

Souza et al. (2018) employed features obtained using a representation learning strategy (i.e. CNN) in conjunction with the dissimilarity approach for off-line SV. The authors claim that CNN features used in the WI mode is well-successful to address that task. The class-independence was obtained through the dichotomy transformation, proposed by Cha and Srihari (2000a), such a way that positive and negative samples were created by using intra-class and inter-class feature vectors respectively. The authors speculate about advantages and disadvantages of using the dichotomy transformation. According to them, one possible disadvantage of the dissimilarity approach is that writers perfectly grouped in the original feature domain may be not perfectly dichotomized in the distance domain. In other words, the greater the dispersion between sample distributions among writers, the less the dichotomizer is able to detect differences among signatures. The experimental protocol was developed on the PUC-PR and GPDS datasets, containing samples from 168 and 881 writers respectively. The WI approach presented better results than the WD approach on the PUC-PR dataset. Moreover, authors conclude that the proposed method performs better than other methods described in the literature that use complex feature extraction strategies and also classifiers selection.

Zottesso et al. (2018) performed bird species identification by using the dissimilarity approach proposed by Cha and Srihari. The authors extracted features from the time-frequency (i.e. spectrogram) representation of the audio signal, and then they used the dissimilarity framework to accomplish bird species identification on eight quite challenging subsets taken from the LifeClef 2015 bird task contest database. In the hardest scenario, there were 915 species in the classification task. The obtained results confirm once again the evidence that dissimilarity should be considered as a suitable and effective alternative to deal with problems with a large number of classes.

Bouibed et al. (2018) investigated the use of dissimilarity approach, as proposed by Cha and Srihari, to perform writer retrieval task. Experiments were accomplished by using CVL and ICDAR 2011 databases, containing 309 and 26 classes respectively. The authors do not mention the use of any preprocessing step widely used in this kind of application, such as zoning the documents into blocks. Textural features were obtained by using Histogram of Oriented Gradients (HOG) and the Gradient Local Binary Patterns (GLBP). A classifier model was created using SVM and the results obtained are similar to those described in the literature. In a nutshell, the dissimilarity in this work was performed through the differences between intra-class and inter-class samples, generating respectively positive and negative examples for the SVM classifier.

In Bertolini et al. (2010, 2013, 2016), the authors describe a series of investigations using the dissimilarity approach proposed by Cha and Srihari. In the first work (Bertolini et al. 2010), the authors perform off-line SV task claiming about the suitability of dissimilarity vectors due to its robustness in terms of class-independence and its appropriateness to be used with universal classifiers. The authors also point out the adequacy of that framework taking into account that it is not required to retrain the classification model as new classes are inserted. The dissimilarity approach in conjunction with an ensemble of classifiers was evaluated on a database composed of samples taken from 100 writers, and a significant gain in performance was reported. In addition, a comprehensive set of experiments assessing two different scenarios using simple, random and simulated forgeries were evaluated. Through schemes of combination of classifiers it was possible to notice that some weak features can contribute to increase the performance and to reduce the false acceptance considerably in SV systems.

In the second and third works of the series, Bertolini et al. (2013, 2016) used dissimilarity for the writer identification task. The authors argue to use dissimilarity due to two main points, the first is the idea of class independence, since new classes are inserted and there is no need to retrain the model. Another positive point is the transposition of a n-class problem into a binary one. The authors employed a texture generation scheme which converts the original document into a denser texture that is supposed to preserve characteristics of the writer. Using the dissimilarity approach in conjunction with SVM classifier and different schemes of combination rules the authors obtained performance similar to those described in the literature. In Bertolini et al. (2013), the authors explored different texture descriptors and especially the impact of using more classes in the training set, as well as the impact of ranging the number of reference samples, according to its availability. It was possible to note that as higher the number of classes in the training set and the number of reference samples, as better the performance. BFL and IAM databases were used in that work, respectively with 115 and 240 writers in the test.

In the third work of the series (Bertolini et al. 2016), the idea of dissimilarity was used in the task of multi-script writer identification where the same writer gives texts in different scripts, like English and Arabic. Thus, using the same parameters used in Bertolini et al. (2013) the authors evaluate the impact of using different scripts for training and testing. Therefore, scripts in English were used for training and scripts in Arabic were used in the test set, and vice-versa. The use of dissimilarity was crucial to the success of that work because, in that case, the classes present in the training set are not part of the test set. Thus, the dissimilarity approach was decisive for the optimal performance described. A subset of the QUWI database containing 475 writers was used in that work. In conclusion, the authors argue about the successfulness of the proposed strategy using both scripts for training and test, obtaining in the multi-script scenario results superior to those already presented in the literature.

Pinheiro et al. (2019) present a Text Categorization system that combines binary classifiers in the dichotomy space, called Combined Dichotomy Transformations (CoDiT). CoDit combines binary classifiers that are trained with different dichotomy sets using Dichotomy Transformation (DT). By this way, much more training examples are obtained if compared with the original training set. Thus, the classifiers can be trained using different data without the reduction of the number of examples or features. So, an ensemble with diverse and strong classifiers can be obtained. The authors performed experiments using 14 databases, and the CoDiT achieved statistically better results in comparison to SVM, Bagging, Random Subspace, BoosTexter, and Random Forest.

As one can note by analyzing this subsection, many of the works already developed using dissimilarity vectors are devoted to manuscript/signature verification/identification tasks. It can be explained by the fact that these tasks typically present suitable characteristics to the use of that framework (i.e. large number of classes and few samples per class). Furthermore, it is even possible to observe that the SVM classifier is chosen also in most of the works, what can be explained by the fact that this classifier was originally proposed to deal with binary problems, like what happens after the dichotomization. In any case, it is worth of mention the fact that in recent works, other large multi-class problems, related to different application domains, has also been addressed. In Table 11 we summarize some information about related works described in this subsection.

4 Critical review

As pointed in Sect. 1, the dissimilarity space introduced by Pękalska and Duin was originally proposed considering the development of classifiers in situations where the feature extraction step is (virtually) unfeasible. Pčkalska and Duin (2005) point out that step, typically developed in the traditional classification framework, maybe a superfluous step in the description of a class, based on the intuition that a class is a set of similar objects and, in this way, one can derive the proximity straight from raw or pre-processed measurements, like images or spectra. In order to put this matter into a proper context, it is important to remind that this strategy was presented at the beginning of the 2000s. At that time, most of the common strategies used to create classifiers systems were based on the traditional feature engineering framework,

Table 11 Summary of the works that uses dissimilarity vector approach

References	Application domain/dataset	Number of classes
Oliveira et al. (2007)	Writer independent off-line SV	2[a]/100[c]
Pavelec et al. (2008)	Author identification task both in class dependent and class independent mode	20[b,c]
Bertolini et al. (2010)	Writer independent off-line SV	2[a]/100[c]
Hanusiak et al. (2012)	Writer verificaton task databases	2[a]/315[c]
Swanepoel and Coetzer (2012)	Off-line SV/Dolfing's dataset	2[a]/51[c]
Eskander et al. (2013)	SV task both in class dependent and class independent mode/Brazilian dataset and GPDS dataset	2[a]
		168[c]
		300[c]
Rivard et al. (2013)	Offline SV	2[a]/168[c]
Okawa and Yoshida (2013)	Class independent online writer verification/54 volunteers handwriting	2[a]
		54[c]
Bertolini et al. (2013)	Writer identification and verification/BFL and IAM database	2[a]/115[b]
		2[a]/240[b]
Martins et al. (2015)	Forest species recognition	112[b,c]
Bertolini et al. (2016)	Multi-script writer identification/QUWI dataset	475[b,c]
Souza et al. (2018)	Writer independent handwritten SV/Brazilian dataset and GPDS dataset	2[a]
		168[c]
		300[c]
Zottesso et al. (2018)	Bird species identification	23–915[b,c]
Bouibed et al. (2018)	Writer retrieval/CVL and ICDAR 2011 datasets	2[a]
		309[c]
		26[c]
Pinheiro et al. (2019)	14 text categorization databases	4–25[c]

[a]Verification, [b]identification, [c]number of classes on the database

such a way that handmade features should be defined to feed machine learning algorithms. However, what is important to observe is that more than replace the feature representation, the dissimilarity space can be built on the top of it (Eskander et al. 2013). In such a case, the quality of the obtained dissimilarity space depends directly on the quality of the feature representation used, beyond the properties of the utilized dissimilarity function.

Thus, the recent growth of the use of representation learning strategies based on deep models, boosted mainly by the popularization of Graphics Processing Units (GPUs), which can be perceived as a suitable and successful way to address different classification problems, opens new possibilities for the use of dissimilarity based solutions. For classification problems characterized by few samples for training and many classes, the feature extracted from pre-trained models could be used for the construction of promising dissimilarity spaces.

Regarding the dissimilarity vectors approach, proposed by Cha and Srihari, one can highlight that this strategy is still quite suitable to address problems for which the number of

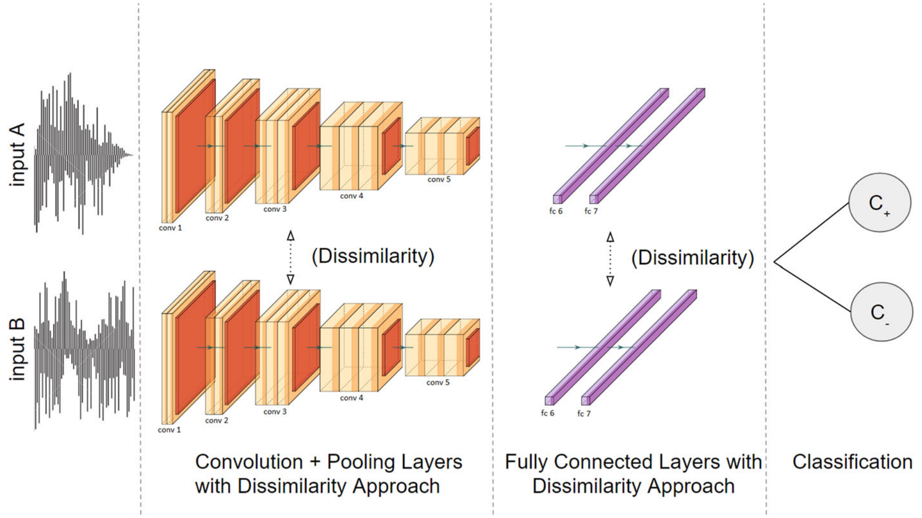


Fig. 5 Generic framework proposed to guide the creation of better dissimilarity vectors on the deep learning scenario

classes is not very well defined in advance, or it can increase over time. In addition, it is worthy of mention that this strategy has been achieving interesting results in large class classification problems, many times obtaining results better than those obtained by using traditional modeling. It can be explained by the fact that in problems with a large number of classes, it is often very difficult to sufficiently sample all the classes involved in the problem. Nonetheless, it is also important to remind that this strategy highly depends on the suitable choice of reference samples (prototypes) for all the classes involved in the problem. The classification performance for a given class depends on the selection of representative prototypes which should cover well the center of its data distribution. As corroborated by Bertolini et al. (2015), Garcia et al. (2011) and Triguero et al. (2012). Furthermore, we would conjecture that the number of reference samples taken for each class may be another crucial point for favoring the performance of this approach. As higher the number of reference samples for a given class, as better its covering in the space. With respect to the representation learning strategies, the use of dissimilarity vectors is still an interesting strategy to create class-dependent and class-independent solutions based on features automatically extracted.

Moreover, there is plenty of room for improvement regarding the investigations on how deep learning strategies could help to find better dissimilarity vectors starting from two given feature vectors A and B, and their class labels. Figure 5 illustrates a generic framework that could be used to guide the development of this kind of strategy.

In the suggested framework, given two pairs composed of feature vectors and their respective class labels (i.e. (A, l_A) and (B, l_B)), a deep neural network based algorithm could be used to find a more suitable dissimilarity vector for A and B. The dissimilarity using deep learning could be performed both on "Convolution plus Pooling" layer, or on "Fully Connected" layer. The deep neural network should be trained aiming to perform as an optimal mapping function, such a way that if $l_A = l_B$, then the resulting dissimilarity vector should have its values as close as possible to zero, otherwise, its values should be far from zero. In this sense, the feature learning process could be accomplished taking into account difference between the inputs A and B. Thus, the input B could be taken from a reference sample when we want to perform the verification (genuine or false).

Independent on the strategy used to explore dissimilarity, different aspects may be considered. As suggested in Eskander et al. (2013), perhaps the main could be the possible combination of tasks performed in different spaces. For instance, one may perform feature selection, prototype (reference) selection, and classifier design while making transitions between feature and dissimilarity representation.

Finally, we can also make some notes regarding the scenarios in which the two dissimilarity approaches discussed here have been examined in the literature, and regarding the scenarios in which these approaches are more promising.

On the one hand, the dissimilarity space approach has been experimented in a wide range of application domains (Table 10), and the performance improvement varies from one situation to another. It is important to point out that this approach can be seen as a way out to deal with problems in which it is not easy (or feasible) to find a proper vector representation to describe the objects, such a way that the differences between images, graphs, or strings that describe them could be most appropriate (Duin and Pękalska 2012).

On the other hand, the dissimilarity vector approach has been achieving impressive results in situations in which the number of classes is large, as in writer identification, bird classification, signature verification, and forest species recognition tasks, as described in Table 11. Another issue which calls attention is that in many of these applications, the dissimilarity vectors were built from feature vectors created to describe the textural content of the images of the patterns to be classified. In general, we can highlight three conditions that are favorable to the use of this approach for a given classification task: (i) few examples for each class, (ii) a large number of classes, and (iii) high frequency of class inclusion (or exclusion) demanding the classifier retraining.

5 Concluding remarks

The dissimilarity approach has shown to be a very interesting alternative for the traditional feature space representation when addressing large multi-class problems or even problems with a small number of training samples. We have shown in this paper the existing possibilities in terms of dissimilarity representations, describing them by means of some comprehensive examples. In addition, we have discussed when and why one may apply such a problem representation strategy. A complete review of the state-of-art was presented, followed by a critical analysis. The original purpose of the dissimilarity representation was discussed, and some perspectives for this still promising field in the era of the representation learning based on deep models was presented.

In summary, there is still an interesting field of research that can explore different possible combinations of tasks performed in both spaces, feature and dissimilarity. In the same classification problem, one may investigate the possibility of making transitions from one space to another to perform feature and prototype (reference) selection, or even classifier training. Thus, an interesting research direction could be to produce classification methods using the best of both dissimilarity and feature representation.

Finally, we would point two interesting open issues which could be properly explored in future works: (i) one could investigate the development of a meta-learning strategy aiming to find the best dissimilarity representation starting from the vectors obtained from two given objects, deep metrics could be used to address this matter, as Siamese networks more particularly. (ii) Bertolini et al. (2015) have already shown that instance selection can be useful to improve the dissimilarity performance, however the investigation regarding feature selection in this context still remains an open question.

Acknowledgements We thank the Brazilian Research Support Agency CNPq - National Council for Scientific and Technological Development (Grants #171193/2017-2 and #156956/2018-7) for its financial support.

References

Bertolini D, Oliveira LS, Justino E, Sabourin R (2010) Reducing forgeries in writer-independent off-line signature verification through ensemble of classifiers. Pattern Recognit 43(1):387–396

Bertolini D, Oliveira LS, Justino E, Sabourin R (2013) Texture-based descriptors for writer identification and verification. Expert Syst Appl 40(6):2069–2080

Bertolini D, Oliveira LS, Sabourin R (2015) Improving writer identification through writer selection. In: Iberoamerican Congress on pattern recognition. Springer, pp 168–175

Bertolini D, Oliveira LS, Sabourin R (2016) Multi-script writer identification using dissimilarity. In: 2016 23rd International conference on pattern recognition (ICPR). IEEE, pp 3025–3030

Bouibed ML, Hassiba N, Chibani Y (2018) Evaluation of gradient descriptors and dissimilarity learning for writer retrieval. In: 2018 Eighth international conference on information science and technology (ICIST), pp 252–256. https://doi.org/10.1109/ICIST.2018.8426179

Bunke H, Riesen K (2008) Graph classification based on dissimilarity space embedding. In: Joint IAPR international workshops on statistical techniques in pattern recognition (SPR) and structural and syntactic pattern recognition (SSPR). Springer, pp 996–1007

Cha S-H (2001) Use of distance measures in handwriting analysis. PhD thesis, Buffalo, AAI3010803

Cha S-H, Srihari SN (2000a) Writer identification: statistical analysis and dichotomizer. In: Ferri FJ, Iñesta JM, Amin A, Pudil P (eds) Advances in pattern recognition. Springer, Berlin, pp 123–132. ISBN 978-3-540-44522-7

Cha S-H, Srihari SN (2000b) Writer identification: statistical analysis and dichotomizer. In: Joint IAPR international workshops on statistical techniques in pattern recognition (SPR) and structural and syntactic pattern recognition (SSPR). Springer, pp 123–132

Duin RPW, Pękalska E (2012) The dissimilarity space: bridging structural and statistical pattern recognition. Pattern Recognit Lett 33(7):826–832

Duin RPW, Loog M, Pękalska E, Tax DMJ (2010) Feature-based dissimilarity space classification. In: Recognizing patterns in signals, speech, images and videos. Springer, pp 46–55

Eskander GS, Sabourin R, Granger E (2013) Eric hybrid writer-independent–writer-dependent offline signature verification system. IET Biom 2:169–181(12). ISSN 2047-4938

Garcia S, Derrac J, Cano JR, Herrera F (2011) Prototype selection for nearest neighbor classification: taxonomy and empirical study. IEEE Trans Pattern Anal Mach Intell 3:417–435

Hanusiak RK, Oliveira LS, Justino E, Sabourin R (2012) Writer verification using texture-based features. Int J Doc Anal Recognit (IJDAR) 15(3):213–226. ISSN 1433-2825. https://doi.org/10.1007/s10032-011-0166-4

Haralick RM, Shanmugam K, Dinstein IH (1973) Textural features for image classification. IEEE Trans Syst Man Cybern 3(6):610–621

Livi L, Rizzi A, Sadeghian A (2014) Optimized dissimilarity space embedding for labeled graphs. Inf Sci 266:47–64

Martins JG, Oliveira LS, Britto AS, Sabourin R (2015) Forest species recognition based on dynamic classifier selection and dissimilarity feature vector representation. Mach Vis Appl 26(2–3):279–293

Nguyen GP, Worring M, Smeulders AWM (2006) Similarity learning via dissimilarity space in CBIR. In: Proceedings of the 8th ACM international workshop on multimedia information retrieval. ACM, pp 107–116

Okawa M, Yoshida K (2013) User generic model for writer verification using multiband image scanner. In: 2013 IEEE International conference on technologies for homeland security (HST). IEEE, pp 375–380

Oliveira LS, Justino E, Sabourin R (2007) Off-line signature verification using writer-independent approach. In: 2007 International joint conference on neural networks, pp 2539–2544. https://doi.org/10.1109/IJCNN.2007.4371358

Pavelec D, Justino E, Batista LV, Oliveira LS (2008) Author identification using writer-dependent and writer-independent strategies. In: Proceedings of the 2008 ACM symposium on applied computing, SAC '08. ACM, New York, pp 414–418. ISBN 978-1-59593-753-7. https://doi.org/10.1145/1363686.1363788

Pčkalska E, Duin RPW (2005) The dissimilarity representation for pattern recognition: foundations and applications. World Scientific, Singapore
Pękalska EM (2005) Dissimilarity representations in pattern recognition. concepts, theory and applications. Thesis. http://rduin.nl/papers/pekalska_thesis.pdf
Pękalska E, Duin RPW (2006) Dissimilarity-based classification for vectorial representations. In: 18th International conference on pattern recognition, 2006. ICPR 2006, vol 3. IEEE, pp 137–140
Pękalska E, Paclik P, Duin RPW (2001) A generalized kernel approach to dissimilarity-based classification. J Mach Learn Res 2(Dec):175–211
Pękalska E, Paclik P, Duin RPW (2002) Dissimilarity representations allow for building good classifiers. Pattern Recognit Lett 23(8):943–956. ISSN 0167-8655. https://doi.org/10.1016/S0167-8655(02)00024-7
Pękalska E, Duin RPW, Paclík P (2006) Prototype selection for dissimilarity-based classifiers. Pattern Recognit 39(2):189–208
Pinheiro RHW, Cavalcanti GDC, Tsang IR (2017) Combining dissimilarity spaces for text categorization. Inf Sci 406–407:87–101. ISSN 0020-0255. https://doi.org/10.1016/j.ins.2017.04.025
Pinheiro RHW, Cavalcanti GDC, Tsang IR (2019) Combining binary classifiers in different dichotomy spaces for text categorization. Appl Soft Comput 76:564–574. ISSN 1568-4946. https://doi.org/10.1016/j.asoc.2018.12.023
Riesen K, Bunke H (2009) Reducing the dimensionality of dissimilarity space embedding graph kernels. Eng Appl Artif Intell 22(1):48–56
Rivard D, Granger E, Sabourin R (2013) Multi-feature extraction and selection in writer-independent off-line signature verification. Int J Doc Anal Recognit (IJDAR) 16(1):83–103. ISSN 1433-2825. https://doi.org/10.1007/s10032-011-0180-6
Santini S, Jain R (1999) Similarity measures. IEEE Trans Pattern Anal Mach Intell 21(9):871–883
Souza VLF, Oliveira ALI, Sabourin R (2018) A writer-independent approach for offline signature verification using deep convolutional neural networks features. arXiv preprint arXiv:1807.10755,
Swanepoel JP, Coetzer J (2012) Writer-specific dissimilarity normalisation for improved writer-independent off-line signature verification. In: 2012 International conference on frontiers in handwriting recognition, pp 393–398. https://doi.org/10.1109/ICFHR.2012.290
Theodorakopoulos I, Economou G, Fotopoulos S (2013) Collaborative sparse representation in dissimilarity space for classification of visual information. In: Bebis G, Boyle R, Parvin B, Koracin D, Li B, Porikli F, Zordan V, Klosowski J, Coquillart S, Luo X, Chen M, Gotz D (eds) Advances in visual computing. Springer, Berlin, pp 496–506. ISBN 978-3-642-41914-0
Theodorakopoulos I, Kastaniotis D, Economou G, Fotopoulos S (2014a) Hep-2 cells classification via sparse representation of textural features fused into dissimilarity space. Pattern Recognit 47(7):2367–2378
Theodorakopoulos I, Kastaniotis D, Economou G, Fotopoulos S (2014b) Pose-based human action recognition via sparse representation in dissimilarity space. J Vis Commun Image Represent 25(1):12–23
Triguero I, Derrac J, Garcia S, Herrera F (2012) A taxonomy and experimental study on prototype generation for nearest neighbor classification. IEEE Trans Syst Man Cybern Part C (Appl Rev) 42(1):86–100
Van Gemert JC, Geusebroek J, Veenman CJ, Snoek CGM, Smeulders AWM (2006) Robust scene categorization by learning image statistics in context. In: Conference on computer vision and pattern recognition workshop, 2006. IEEE, pp 105–105
Zottesso RHD, Costa YMG, Bertolini D, Oliveira LS (2018) Bird species identification using spectrogram and dissimilarity approach. Ecol Inform 48:187–197. ISSN 1574-9541. https://doi.org/10.1016/j.ecoinf.2018.08.007 **(in press)**