

New Graph Embedding Approach for 3D Protein Shape Classification

Kamel Madi

Umanis, Research & Innovation
Levallois-Perret, 92300, France
Email: kmadi@umanis.com

Eric Paquet

National Research Council Canada
Ottawa, Canada
Email: Eric.Paquet@nrc-cnrc.gc.ca

Abstract—We address the problem of 3D protein deformable shape classification. Proteins are macromolecules characterized by deformable and complex shapes which are related to their function making their classification an important task. Their molecular surface is represented by graphs such as triangular tessellations or meshes. In this paper, we propose a new graph embedding based approach for the classification of these 3D deformable objects. Our technique is based on graphs decomposition into a set of substructures, using triangle-stars, which are subsequently matched with the Hungarian algorithm. The proposed approach is based on an approximation of the Graph Edit Distance which is characterized by its robustness against both noise and distortion. Our algorithm defines a metric space using graph embedding techniques, where each object is represented by a set of selected 3D prototypes. We propose new approaches for prototypes selection and features reduction. The classification is performed with supervised machine learning techniques. The proposed method is evaluated against 3D protein benchmark repositories and state-of-the-art algorithms. Our experimental results consistently demonstrate the effectiveness of our approach.

Contributions—We propose a new graph embedding approach to classify 3D deformable protein shapes and new techniques for prototypes selection and dimensionality reduction then we performed the classification using a Naive Bayes (NB) classifier and we achieve better results than the state-of-the-art.

Keywords—3D Protein Shape Classification, 3D object recognition, 3D Deformable object recognition, Pattern recognition, Graph Embedding, Graph matching, Graph edit distance, Graph decomposition, Graph metric, Metric learning.

I. INTRODUCTION

Shape recognition is one of the most important task in computer vision as demonstrated by the large body of work dealing with 3D shape analysis. Recent advances in 3D data acquisition as well as the availability of large 3D repositories have been instrumental in the design of new and more efficient algorithms for shape recognition and classification.

Many shapes may be represented by graphs and consequently, graph matching techniques are suitable for their recognition and classification. In graph matching, the similarity in between a pair of graphs is obtained by matching, respectively, their corresponding vertices and edges while making the process subject to a certain number of constraints which ensure that substructures, in between graphs, are properly mapped

to each other. Various approaches have been proposed in the literature in order to solve the graph matching problem [1], [2], [3]. Graph Edit Distance (GED) is one of the most preeminent method for determining the distance in between two graphs [4], [5]. The latter is defined as the sequence of operations that maps a graph into another while ensuring that the cost associated with the sequence of operations is kept to a minimum. This method is both noise and distortion oblivious but, regrettably, is characterized by a high computational exponential complexity [6]. In this paper, we address the problem of 3D protein deformable shape classification. Proteins are complex macromolecules with highly variable conformations which generate local and global deformations. Their structures is related to their functions which makes their classification an important task: namely for drug discovery and diseases characterization. In this study, proteins are assimilated to 3D deformable objects represented by graphs such as triangular tessellations. We propose a new graph embedding based approach to classify these 3D deformable objects. In our method, graphs are decomposed into a set of substructures called triangle-stars (ts) [7], which are subsequently matched with the Hungarian algorithm. An approximation of the Graph Edit Distance is evaluated. The latter is fault-tolerant to noise and distortion making our method particularly relevant for deformable 3D shapes comparison. Our approach defines a metric space using graph embedding techniques, where each object is represented by a set of selected 3D prototypes. We propose new approaches for prototypes selection and dimensionality reduction. The classification is performed with supervised machine learning techniques.

The remainder of the paper is organized as follows. In Section II, we briefly review some related works. Our approach is described in Sections III and IV. In Section V, we present and discuss our experimental results while comparing them with state-of-the-art benchmark shape-matching algorithms. Finally, Section VI concludes the paper.

II. RELATED WORKS

We present in this section a brief review of 3D objects recognition methods, with a particular emphasis on graph-based techniques. We can generally divide existing methods into three main categories [8]: feature-based methods, graph-based methods and others.

Graphs are particularly well suited for representing shape properties as well as for capturing the interrelations in between substructures. The former are associated with the nodes or vertices while the latter are associated with the edges. The specific nature of the representation is determined by the underlying application. Indeed, nodes may be associated to points, geometrical regions or any substructure resulting from a segmentation process while, on the other hand, the edges establish a topological relationship in between the nodes such as proximity and adjacency. Various graph comparison techniques have been proposed in the literature either focusing on the specific nature of the graphs or addressing specific applications related issues [8]. For example, 3D shapes may be converted to skeletons by means of a thinning procedure [9]. It is also possible to use a mapping function, directly on a manifold, in order to generate a Reeb graph [10]. A shape may also be divided into substructures using segmentation techniques. The interrelations in between the various elements resulting from the segmentation process are then represented by a graph [11].

Graph embedding techniques unify and combine complementary properties associated with statistical and structural methods [12], [13]. Graph embedding techniques map graphs into a vector space, therefore representing graphs with a set of vectors. Graph embedding techniques can be divided into two different classes [14]. The first class contains methods mapping the graph vertices (or substructures) into a set of points in a vector space, associating a vector representation to each node, where similar vertices (or substructures) are mapped to near by points in the vector space [15]. The second class encompasses techniques that maps whole graphs into points in a vector space, where similar graphs correspond, in the vector space, to neighboring points. Among the most salient works are: [16] for isometric embedding; [17], [18] for spectral embedding; [19], [20] for prototype-based embedding. Several strategies for selecting the graph prototypes have been proposed, such as: [21].

III. DISTANCE BETWEEN 3D DEFORMABLE OBJECTS

We propose a new graph embedding based approach for proteins classification. Our approach is based on the decomposition of graphs into a set of substructures called triangle-stars [7], which are subsequently matched with the Hungarian algorithm [22]. Our approach defines a metric space using graph embedding techniques, where each object is represented by a set of selected 3D objects prototypes. We propose new approaches for prototypes selection and features reduction. Firstly, we introduce the triangle-stars decomposition as well the distance between triangle-stars and triangular tessellations. Then, in Section IV, we describe the proposed approaches based on graph embedding.

A. Triangle-stars decomposition

The triangle-stars decomposition [7] is a decomposition method of a triangular tessellation (of a 3D shape) into a set of connected components called triangle-stars for a given

neighborhood of order N_k . From a triangle-star representation, a description is defined which is invariant or at least oblivious under most common deformations. Prior to the decomposition, a strict total order on the triangles is established, in order to reduce the number of triangle-stars and to guarantee the uniqueness of the decomposition.

a) **N_k -triangle-star and N_k -neighborhood:** A N_k -triangle-star (N_k -ts) is a subgraph constituted by a triangle and the set of its N_k -neighbors. Two triangles t_0 and t_k are N_k -neighbors, if there is between them a chain of at most $(k-1)$ distinct triangles, which are pairwise consecutive neighbors ($\exists t_{i=1..k-1}$ where: $\forall i \in 1..(k-1)$, t_i and t_{i+1} are neighbors). Two triangles sharing at least one common node, are neighbors ($k=1$). [7].

b) **Triangle-star vector representation:** A vector representation is associated to each triangle-star ts_i based on the following set of descriptors: the perimeter $P(t_{i,l})$ and the area $A(t_{i,l})$ of each triangle t_l belonging to ts_i ($t_l \in ts_i$), the global perimeter $PG(ts_i) = \sum_{j=1}^{j=\|T(ts_i)\|} P(t_{i,j})$ and the global area $AG(ts_i) = \sum_{j=1}^{j=\|T(ts_i)\|} A(t_{i,j})$ of ts_i , the degrees $deg_{i,l,k}$ associated with their vertices v_k as well as the weights $W_{i,l,k}$ (Euclidean distance) associated with their edges e_k belonging to the triangles $t_l \in ts_i$. This vector representation is given by: $\{AG(ts_i), PG(ts_i), \{A(t_{i,l}), P(t_{i,l}), W_{i,l,k=1..3}, deg_{i,l,k=1..3}\}_{l=1}^{l=\|T(ts_i)\|}\}$ [7].

c) **Triangle-stars decomposition process:** According to a descending strict total order applied on the set of triangles, based on the number of neighbors $\|neighbors\|$ and the nodes coordinates; the first N_k -triangle-star is constructed using the first triangle and its N_k -neighbors (triangles not belonging to any other N_k -triangle-stars). Then, the triangles and the resulting N_k -triangle-stars sets are updated. The process is repeated, as long as there is at least one triangle not belonging to any N_k -triangle-star [7].

B. Distance between triangular tessellations

The *Triangle-Star Measure* TSM [7] is considered to compute the dissimilarity between two triangular tessellations represented by triangle-stars TS_1 and TS_2 . The *Triangle-Star Measure* $TSM(TS_1, TS_2)$ is defined as:

$$TSM(TS_1, TS_2) = \frac{\min_{m \in M} \sum_{ts_i \in TS_1, m(ts_i) \in TS_2} d(ts_i, m(ts_i))}{\max(\|TS_1\|, \|TS_2\|)} \quad (1)$$

where the similarity measure d between two triangle-stars ts_i and ts_j is defined as:

$$d(ts_i, ts_j) = 1 - \frac{\sum_{k=1}^{k=6} sim_k(ts_i, ts_j)}{\sum_{k=1}^{k=6} \alpha_k} \quad (2)$$

The similarity measure d requires six auxiliary functions sim_k :

$$dsim_k(ts_i, ts_j) = \alpha_k *$$

$$(3) \quad \left\{ \begin{array}{l} \frac{|AG(ts_i) - AG(ts_j)|}{AG_{MAX}} \\ \frac{|PG(ts_i) - PG(ts_j)|}{PG_{MAX}} \\ \frac{\sum_{l=1}^{\Gamma} |A(T(ts_i)_l) - A(T(ts_j)_l)|}{A_{MAX} \Gamma} \\ \frac{\sum_{l=1}^{\Gamma} |P(t_{i,l}) - P(t_{j,l})|}{P_{MAX} \Gamma} \\ \frac{\sum_{l=1}^{\Gamma} \sum_{k=1}^3 |W_{i,l,k} - W_{j,l,k}|}{W_{MAX} \Gamma} \\ \frac{\sum_{l=1}^{\Gamma} \sum_{k=1}^3 |Deg_{i,l,k} - Deg_{j,l,k}|}{Deg_{MAX} \Gamma} \end{array} \right. \quad \begin{array}{l} k=1 \\ k=2 \\ k=3 \\ k=4 \\ k=5 \\ k=6 \end{array}$$

Where Γ is the max number of triangles in the triangle-stars and $\alpha_{k=1..6}$ are parameters associated to the descriptors ($\alpha_k \in \mathbb{N}$ and $\sum_{k=1}^6 \alpha_k > 0$). See paragraph (b) of the subsection III-A for the descriptors definition.

In order to calculate $TSM(TS_1, TS_2)$, a $n \times n$ matrix D is defined (where n is given by $n = \max(\|TS_1\|, \|TS_2\|)$), transforming the issue to solving the assignment problem, which is one of the fundamental combinatorial optimization problems that consists of finding, the minimum or maximum weight matching in a weighted bipartite graph. Each matrix element $D_{i,j}$ is the dissimilarity measure $d(ts_i, ts_j)$ Eq. (2) between a triangle-star $ts_i \in TS_1$ and a corresponding triangle-star $ts_j \in TS_2$. The smallest set of triangle-stars is completed by $(\max(\|TS_1\|, \|TS_2\|) - \min(\|TS_1\|, \|TS_2\|))$ empty triangle-stars ε , if $\|TS_1\| \neq \|TS_2\|$. Then, the Hungarian algorithm [22] is applied on the matrix D to find the best assignment in $\mathcal{O}(n^3)$ time.

IV. GRAPH EMBEDDING BASED APPROACH

We propose a new graph embedding based approach to classify 3D protein deformable shapes. Our approach utilizes the distance TSM (Section III) and defines a metric space using graph embedding techniques, where each object is represented by a set of selected 3D objects prototypes. In this section, we present the proposed approach based on graph embedding. We firstly introduce the graph embedding paradigm, then the mesh reduction algorithm, the proposed approaches for prototypes selection and features reduction, and finally the classification.

A. Graph embedding formal definition

Graph embedding techniques map graphs into a vector space, therefore representing graphs with a set of vectors. Graph embedding techniques can be divided into two different classes [14]. The first class contains methods mapping the graph nodes into a set of points in a vector space, associating a vector representation to each node, where similar nodes are mapped to near by points in the vector space [15]. The second class covers techniques that map whole graphs into points in a vector space, where similar graphs correspond, in the vector space, to neighboring points, as illustrated in Figure 1.

Formally, let consider a set of n graphs $G = \{g_1, g_2, \dots, g_n\}$ and a set of p graph prototypes $GP = \{gp_1, gp_2, \dots, gp_p\} \subseteq G$. The graph embedding is defined as a mapping $\Omega_p^{GP} : G \rightarrow \mathbb{R}^p$ from a graph domain G to a vector space \mathbb{R}^p . Each graph is represented with p dimensional vector as follow: $\Omega_p^{GP}(g) = (d(g, gp_1), d(g, gp_2), \dots, d(g, gp_p))$, where

$(d(g, gp_i))$ is a graph distance measuring the dissimilarity or the similarity between the two graphs g and gp_i .

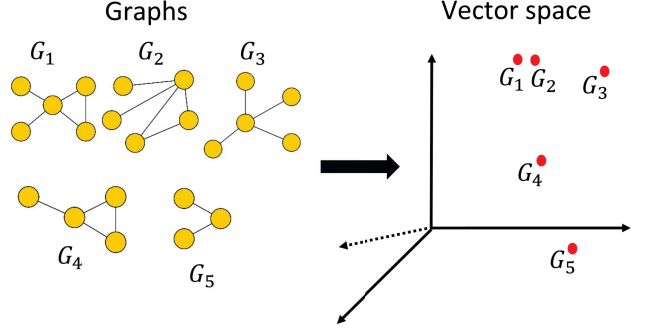


Fig. 1. Graph embedding example.

B. Mesh reduction

It is important to consider mesh reduction as well as re-meshing as both alter the structure of the graph which may impact the accuracy. The negative impact is rather limited mainly because of the robustness of GED against noise and distortion [7]. In addition, the shape descriptors associated with GED are characterized by their obliviousness, if not invariance, against most common geometrical deformations [7]. These features ensure both the robustness against re-meshing and mesh decimation. Mesh reduction was achieved with a quadratic edge collapse decimation technique [23]. The latter iteratively contracts pairs of vertices, while maintaining the geometrical faithfulness of the decimated shape, with quadric matrices. The meshes associated with the macromolecular surfaces were reduced to 15,000 triangles. Figure 2 shows an example of mesh reduction of a protein from the SHREC18 database [24].

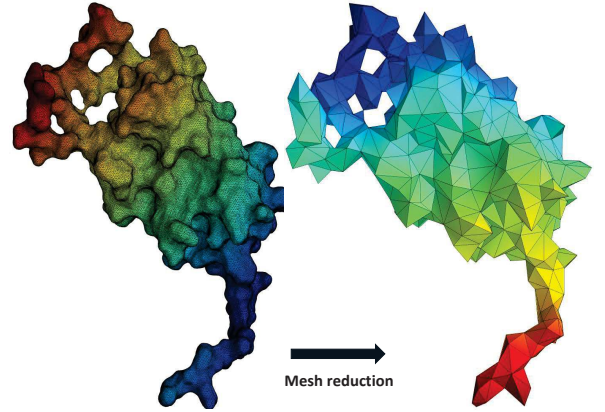


Fig. 2. Mesh reduction example.

C. Multi-criteria algorithm for prototypes selection

Our approach defines a metric space using graph embedding techniques, where each object is represented by a set of selected 3D objects prototypes. We propose a new technique based on the multi-criteria algorithm TOPSIS for prototypes

selection. In this section, we introduce the multi-criteria algorithm TOPSIS. Then we present our approach for prototypes selection.

1) *TOPSIS algorithm*: Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) is a multi-criteria algorithm for decision making (alternatives ranking). The original version was proposed in [25]. The main idea behind TOPSIS is that selected alternatives must have the shortest geometric distance from the Positive Ideal Solution (PIS) and the longest geometric distance from the Negative Ideal Solution (NIS). The TOPSIS algorithm consists of the following steps: 1) the decision matrix construction and normalization, 2) the weighted normalized matrix calculation, 3) the positive and the negative ideals determination, 4) the separation measures S_i^+ and S_i^- calculation and finally 5) evaluation of the relative proximity to the best solution and ranking the alternatives. TOPSIS algorithm is summarized in Algorithm 1.

Algorithm 1 TOPSIS algorithm [25] [26].

```

1: Inputs:
2: A matrix  $M_{m \times n}$  of  $m$  alternatives and  $n$  criteria.
3: Values  $X_{i,j}$  of alternatives according to the  $n$  criteria.
4: Criteria : weights ( $w$ ) and types (positive or negative).
5: Outputs: alternatives ranked.
6: Begin
7:   Normalize the matrix  $M$ :  $r_{i,j} = \frac{X_{i,j}}{\sqrt{\sum_{i=1}^m X_{i,j}^2}}$ .  $\forall j = 1 \dots n$ .
8:   Calculate the weighted normalized matrix:  $v_{i,j} = w_j * r_{i,j}$ 
      $\forall i = 1 \dots m$  and  $j = 1 \dots n$ .
9:   Compute the positive ( $A^+$ ) and the negative ( $A^-$ ) ideals solutions:
10:   $A^+ = \{(\max_j v_{i,j} | i \in C), (\min_j v_{i,j} | i \in C') ; \forall j = 1 \dots n\}$ .
11:   $A^+ = \{v_1^+, v_2^+, \dots\}$ .
12:   $A^- = \{(\min_j v_{i,j} | i \in C), (\max_j v_{i,j} | i \in C') ; \forall j = 1 \dots n\}$ .
13:   $A^- = \{v_1^-, v_2^-, \dots\}$ .
14:  Where  $C$  and  $C'$  are respectively positive and negative criteria.
15:  Calculate separation measures  $S_i^+$  and  $S_i^-$ :
16:   $S_i^+ = \sqrt{\sum_{j=1}^n (v_{i,j} - v_j^+)^2}$ .  $\forall j = 1 \dots n$ 
17:   $S_i^- = \sqrt{\sum_{j=1}^n (v_{i,j} - v_j^-)^2}$ .  $\forall j = 1 \dots n$ 
18:
19:  Calculate the Relative Proximity to the best solution:
20:   $RP_j^+ = \frac{S_j^-}{S_j^+ + S_j^-}$ .  $\forall j = 1 \dots n$ .
21:  Establish an order of alternatives according to  $RP_j^+$ .
22: End

```

2) *TOPSIS-based approach*: In this section we introduce our approach for prototypes selection based on the TOPSIS algorithm. The key ideas consist in ranking (ordering) 3D objects (represented by Triangle-Stars) in each class and then in selecting a predefined maximum number of objects per class ($NBmax$). The first step is associated with the construction of the decision matrix, which is based on the confusion matrix where the distances are calculated from Eq. (1). The second step consists on applying Algorithm 1 on the decision matrix, class per class, in order to rank the objects, in term of similarity, for each class. Given a class C_i we consider the objects belonging to this class as positive criteria and the remaining objects as negative criteria. We associate a weight $W_{CI} = 0.5/NB_{obj}(C_i)$ to each object belonging to the class C_i and a weight $W_{EC} = 0.5/(NB_{All} - NB_{obj}(C_i))$ to the

objects not belonging to this class, where $NB_{obj}(C_i)$ is the number of objects belonging to class C_i and NB_{All} is the total number of objects in the dataset. We apply Algorithm 1 on class C_i which results in the ranking of all the objects belonging to class C_i . The process is repeated for each class.

The final step consists in selecting a predetermined number $NBmax$ of objects per class C_i . When $NB_{obj}(C_i) < NBmax$, we select $NB_{obj}(C_i)$ objects, formally, we select $\min(NB_{obj}(C_i), NBmax)$. The pseudo-code of our method appears in Algorithm 2.

Algorithm 2 TOPSIS based approach for prototypes selection.

```

1: Inputs:
2: A set of objects and  $NBmax$  objects to select per class.
3: Outputs: objects ordered and a set of prototypes selected.
4: Begin
5:   Construct a confusion matrix  $CM$  using  $TSM$  Eq. (1).
6:   For each class  $C_i$ 
7:     Positive criteria = objects  $\in C_i$ .
8:     Negative criteria = objects  $\notin C_i$ .
9:      $W_{CI} = 0.5/NB_{obj}(C_i)$  for positive criteria.
10:     $W_{EC} = 0.5/(NB_{All} - NB_{obj}(C_i))$  for negative criteria.
11:    Apply Algorithm 1 and order objects  $\in C_i$ .
12:    Select  $\min(NB_{obj}(C_i), NBmax)$ .
13:  End For each.
14:  Return a set of selected prototypes.
15: End

```

D. Dimensionality reduction and classification

As demonstrated in [7], the TSM distance is a pseudo-metric. In order to classify the various proteins belonging to the SHREC18 proteins dataset, the metric is learned with machine learning techniques. This is in contrast with the standard approach in which an invariant descriptor is associated to each protein and classification is performed with machine learning techniques through supervised learning. In our approach, the TSM distance defines a metric space which characterizes the proteins. Therefore, in order to perform classification, the metric must be learned with an appropriate supervised learning method. The SHREC18 protein dataset contains 2267 macromolecular structures consisting of 107 classes. The classes are strongly unbalanced: the number of macromolecular conformations (conformers) per class varies considerably (see Section V-A for SHREC18 dataset description). In order to facilitate the evaluation, the dataset has been further divided into small (less than 20 conformers per class), medium (20 to 40 conformers per class) and large (more than 40 conformers per class). This means that the dimensions of the symmetric square matrix associated with the metric are 2267 X 2267. Therefore, each protein is characterized by 2267 features. In order to improve the classification performance, the number of features must be reduced. Indeed, some features may be either redundant or highly correlated. Furthermore, for very large datasets, such as the Protein Data Bank (154 939 macromolecular structures) [27], scalability may become an issue. For these reasons, a dimensionality reduction method was applied to the metric. The reduction was achieved with the low-rank matrix factorization algorithm (LRMF) [28].

This approach minimizes the discrepancy, in the sense of the Frobenius norm, in between the original metric and the approximated metric subject to the constraint that the approximated metric has reduced rank. As stated by the Eckart-Young-Mirsky theorem [28], a closed-form solution may be obtained from the singular value decomposition. The singular value decomposition may become computationally prohibitive for large metric spaces. For this reason, an algorithm based on the hidden matrix factorized augmented Lagrangian method was employed in order to evaluate the LRMF as proposed recently by [28]. This algorithm is more efficient and robust from a computational point of view. The number of features was reduced by a factor of hundred when applied to the whole dataset while the reduction was limited to a factor of ten when the classification was restricted to subsets of the dataset e.g. medium. This means that 227 features were considered when the whole dataset was classified. Because the metric matrix is both square and symmetric, the approximated metric generated by the LRMF algorithm is orthogonal which means that the features are independent from one another. The independence in between features is the corner stone of Naive Bayes (NB) classification [29]. Such an assumption is rarely satisfied in practice but is nonetheless perfectly satisfied in our case. For this reason, Naive Bayes (NB) was selected as our classification algorithm as it is perfectly adapted to such context [29].

V. EXPERIMENTS

The proposed method was evaluated, using the SHREC18 [24] shape repository, against state-of-the-art approaches.

A. Description of the SHREC18 database

The SHREC18 database [24] consists of 2 267 three-dimensional protein shapes originating from the Protein Data Bank (PDB) [30]. Each protein shape is represented by a triangular tessellation (mesh). The database is divided into 107 classes. Each class consists of various conformations (deformations) of the same protein. The SHREC18 database is unbalanced: there are one to hundred conformations per class. Among them, eighteen consist of only one conformation. The average number of conformations per class is 21 while the average number of vertices per mesh is 84019 which corresponds to 168041 triangles per conformation on average. Table I shows a small sample of 3D protein shapes belonging to the SHREC18 database.

B. Target methods

In order to evaluate the efficacy of our approach *GEmb3D*, our algorithm was compared against all methods employed in SHREC18 Protein Shape Retrieval Contest; these methods are described in [24]. The first one is a 3D convolutional framework (3D-FusionNet) which is based on diffusion distances, volumetric grid, a convolutional neural network for features extraction and a multi-layer perceptron for classification. The second method is a global spectral graph wavelet framework (GSGW) based on the Laplace-Beltrami operator associated

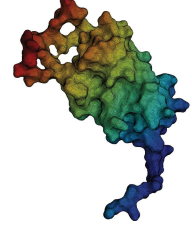
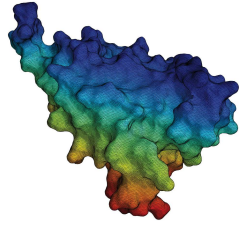
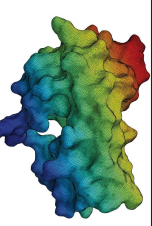
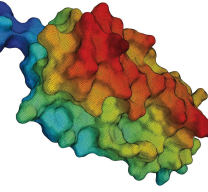
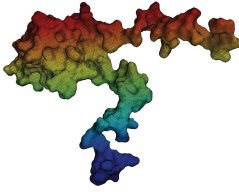
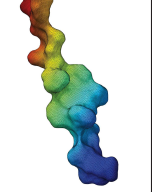
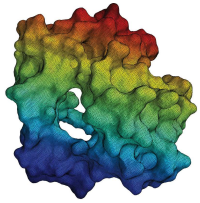
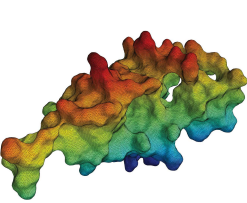
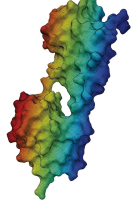
C1 p1	C2 p19	C3 p25
		
C4 p55	C5 p75	C6 p95
		
C88 p1788	C15 p290	C107 p2247
		

TABLE I
SMALL SAMPLE FROM THE SHREC18 DATABASE; WITH $C_i p_j$
REFERRING TO PROTEIN j BELONGING TO CLASS i .

with the macromolecular surface mesh. The third algorithm employs the histograms of area projection transforms (HAPT) which measures the likelihood, for the 3D points inside the protein, of being the center of a spherical symmetry. Four variants of the algorithms are evaluated namely HAPT1–4. The forth method represents proteins as digital elevation models (DEM) which are also obtained from the Laplace-Beltrami operator. The fifth approach relies on the scale-invariant wave kernel signature (SI-WKS) which involves the Laplace-Beltrami operator and the Schrödinger equation. Finally, the last technique is based on the wave kernel signature (WKS) uses in conjunction with the bag of features technique.

C. Experimental results and discussion

We evaluate our approach on the SHREC18 dataset, we apply the 5-fold cross validation technique to the medium subset (*GEmb3D_medium*), the large subset (*GEmb3D_large*) as well as to both subsets (*GEmb3D_merge*). Five training and testing sets were generated for the 5-fold cross validation: each training set consisting of a random selection of 80% of the conformers while the corresponding testing set was formed from the remaining 20% structures. Four evaluation metrics, abundantly described in the literature [24], were employed in order to determine the performance of our system namely the F-measure, the area under the receiver operating characteristic (ROC) curve, accuracy and the confusion matrix [24]. The values reported correspond to an average over the

five sets associated with the cross validation process. The dimensionality of the features was reduced by a factor of 100 (see Section IV-D). With our method, as reported in Table II, we obtained high F-measures (0.877 for Merge, 0.860 for Medium and 0.895 for Large) while for the best SHREC18 algorithm, namely HAPT4, the F-measure was only 0.515; a value of one corresponding to a perfect classification while a value of zero reflects an entirely erroneous one. These results clearly illustrate the performance of our approach.

Method	Class	F-measure
GEmb3D (our method)	Medium	0.860
	Large	0.895
	Merge (Medium+Large)	0.877
3D-FusionNet	Medium	0.379
	Large	0.319
HAPT1	Medium	0.462
	Large	0.301
HAPT2	Medium	0.466
	Large	0.304
HAPT3	Medium	0.490
	Large	0.311
HAPT4	Medium	0.515
	Large	0.338
SI-WKS	Medium	0.118
	Large	0.123
DEM	Medium	0.262
	Large	0.205
WKS	Medium	0.416
	Large	0.313
GSGW	Medium	0.264
	Large	0.223

TABLE II
RESULTS SUMMARY BY METHOD FOR THE F-MEASURE FOR DIFFERENT CLASSES OF THE SHREC18 DATASET.

As mentioned earlier, we employed the 5-fold cross validation for the medium subset, the large subset as well as for both subsets. We did not use the standard 10-fold cross validation because the number of conformers per class is too small (at least for the medium subset). The small subset was leaved out as many classes have a membership of only one conformer.

Our results, in terms of F-measure and area under the ROC, are reported in Table III.

Method	Class	F-measure	Area ROC
GEmb3D	Medium	0.860 \pm 0.006	0.989 \pm 0.001
GEmb3D	Large	0.895 \pm 0.011	0.986 \pm 0.002
GEmb3D	Medium + Large	0.877 \pm 0.006	0.985 \pm 0.001

TABLE III
RESULTS SUMMARY FOR OUR APPROACH GEMB3D WITH 5-FOLD CROSS VALIDATION.

As illustrated by Table III, with the 5-fold cross validation, our approach outperforms all SHREC18 algorithms. As for SHREC18, the best results, in terms of F-measure were obtained with HAPT4 with a value of 0.515 for the medium subset and 0.338 for the large subset [24].

Finally, the confusion matrix for the large subset is shown in Figure 3. This matrix clearly illustrates the fact our algorithm is simultaneously capable of generalization and discrimination.

In the last part of our evaluation, we apply our technique for prototypes selection based on TOPSIS, with a set of selected

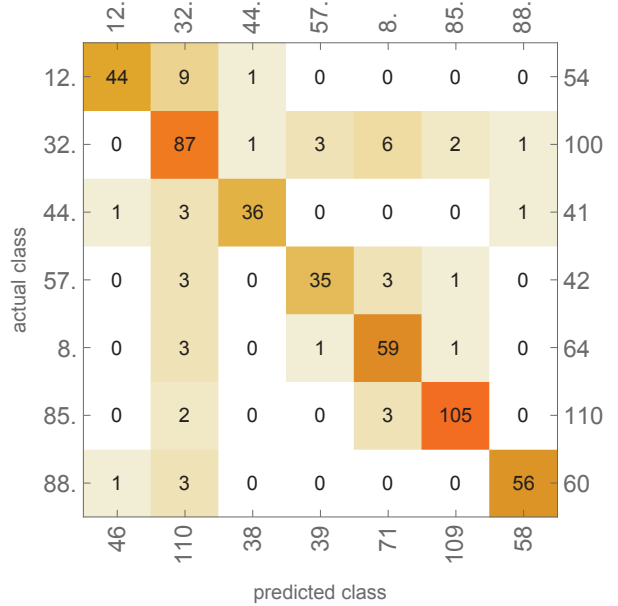


Fig. 3. Confusion matrix for the large subset as obtained with a 5 folds cross validation.

$NBmax \in \{1, 3, 5, 10, 15, 20, 25\}$. The results obtained with our GEmb3D method using our prototypes selection TOPSIS technique are reported in Table IV. These results confirm the high efficiency of our approach. Indeed, our technique outperforms all SHREC18 methods.

$NBmax$	$ Prototypes $	Accuracy
1	107	84%
3	289	85%
5	471	85%
10	926	87%
15	1330	89%
20	1792	90%
25	1829	91%

TABLE IV
ACCURACY FOR OUR GEMB3D APPROACH WHEN EMPLOYED WITH OUR TOPSIS-BASED SELECTION METHOD ACCORDING TO $NBmax$.

Our results clearly demonstrate that our approach outperforms the others. In addition, our method is capable of maintaining a high accuracy for the classification of macro-molecular conformers.

VI. CONCLUSIONS

In this paper, the problem of 3D protein deformable shape classification is addressed. The classification of proteins is an important task due to their deformable and complex shape which is related to their function. The various proteins may be assimilated to 3D deformable objects represented by graphs such as triangular tessellations. In this paper, we proposed a new graph embedding technique to classify these 3D deformable objects. Our approach is based on the decomposition of graphs into a set of substructures called triangle-stars, which are subsequently matched with the Hungarian algorithm. The

proposed algorithm is based on computing an approximation of Graph Edit Distance which is characterized by its robustness against both noise and geometrical distortion which are highly desirable features for shape comparison and make our technique particularly suited for deformable objects comparison. Our algorithm defines a metric space using graph embedding techniques, where each object is represented by a set of selected 3D prototypes. We proposed new approaches for prototypes selection and features reduction. The classification is performed with supervised machine learning techniques. The proposed method is evaluated against 3D protein benchmark databases under different evaluation criteria. Our experimental results consistently demonstrated the effectiveness and the high performances of our approach. As future work, we project to combine our approach with deep learning techniques.

REFERENCES

- [1] H. Bunke, A. Munger, and X. Jiang, "Combinatorial search versus genetic algorithms: A case study based on the generalized median graph problem," *Pattern Recognit. Lett.*, vol. 20, no. 11-13, pp. 1271-1277, 1999. [Online]. Available: [https://doi.org/10.1016/S0167-8655\(99\)00094-X](https://doi.org/10.1016/S0167-8655(99)00094-X)
- [2] M. Gori, M. Maggini, and L. Sarti, "Exact and approximate graph matching using random walks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 7, pp. 1100-1111, 2005. [Online]. Available: <https://doi.org/10.1109/TPAMI.2005.138>
- [3] A. Zanfir and C. Sminchisescu, "Deep learning of graph matching," pp. 2684-2693, 2018. [Online]. Available: http://openaccess.thecvf.com/content_cvpr_2018/html/Zanfir_Deep_Learning_of_CVPR_2018_paper.html
- [4] H. Bunke and K. Shearer, "A graph distance metric based on the maximal common subgraph," *Pattern Recognit. Lett.*, vol. 19, no. 3-4, pp. 255-259, 1998. [Online]. Available: [https://doi.org/10.1016/S0167-8655\(97\)00179-7](https://doi.org/10.1016/S0167-8655(97)00179-7)
- [5] S. Sorlin, C. Solnon, and J. Jolion, "A generic graph distance measure based on multivalent matchings," pp. 151-181, 2007. [Online]. Available: https://doi.org/10.1007/978-3-540-68020-8_6
- [6] D. Conte, P. Foggia, C. Sansone, and M. Vento, "Thirty years of graph matching in pattern recognition," *IJPRAI*, vol. 18, no. 3, pp. 265-298, 2004. [Online]. Available: <https://doi.org/10.1142/S0218001404003228>
- [7] K. Madi, E. Paquet, and H. Kheddouci, "New graph distance for deformable 3d objects recognition based on triangle-stars decomposition," *Pattern Recognit.*, vol. 90, pp. 297-307, 2019. [Online]. Available: <https://doi.org/10.1016/j.patcog.2019.01.040>
- [8] J. W. H. Tangelder and R. C. Veltkamp, "A survey of content based 3d shape retrieval methods," *Multimedia Tools Appl.*, vol. 39, no. 3, pp. 441-471, 2008. [Online]. Available: <https://doi.org/10.1007/s11042-007-0181-0>
- [9] J. Czajkowska, C. Feinen, M. Grzegorzec, M. Raspe, and R. Wickenhofer, "Skeleton graph matching vs. maximum weight cliques aorta registration techniques," *Comp. Med. Imag. and Graph.*, vol. 46, pp. 142-152, 2015. [Online]. Available: <https://doi.org/10.1016/j.compmedimag.2015.05.001>
- [10] V. Barra and S. Biasotti, "3d shape retrieval using kernels on extended reeb graphs," *Pattern Recognit.*, vol. 46, no. 11, pp. 2985-2999, 2013. [Online]. Available: <https://doi.org/10.1016/j.patcog.2013.03.019>
- [11] Y. Kleiman, O. van Kaick, O. Sorkine-Hornung, and D. Cohen-Or, "SHED: shape edit distance for fine-grained shape similarity," *ACM Trans. Graph.*, vol. 34, no. 6, p. 235, 2015. [Online]. Available: <http://doi.acm.org/10.1145/2816795.2818116>
- [12] H. Bunke and K. Riesen, "Recent advances in graph-based pattern recognition with applications in document analysis," *Pattern Recognition*, vol. 44, no. 5, pp. 1057-1067, 2011. [Online]. Available: <https://doi.org/10.1016/j.patcog.2010.11.015>
- [13] B. Horst and R. Kaspar, "Towards the unification of structural and statistical pattern recognition," *Pattern Recognition Letters*, vol. 33, no. 7, pp. 811-825, 2012. [Online]. Available: <https://doi.org/10.1016/j.patrec.2011.04.017>
- [14] P. Foggia, G. Percannella, and M. Vento, "Graph matching and learning in pattern recognition in the last 10 years," *IJPRAI*, vol. 28, no. 1, 2014. [Online]. Available: <https://doi.org/10.1142/S0218001414500013>
- [15] F. Escolano, B. Bonev, and M. A. Lozano, "Information-geometric graph indexing from bags of partial node coverages," pp. 52-61, 2011. [Online]. Available: https://doi.org/10.1007/978-3-642-20844-7_6
- [16] S. Jouili and S. Tabbone, "Graph embedding using constant shift embedding," pp. 83-92, 2010. [Online]. Available: https://doi.org/10.1007/978-3-642-17711-8_9
- [17] X. Bai, E. R. Hancock, and R. C. Wilson, "Graph characteristics from the heat kernel trace," *Pattern Recognit.*, vol. 42, no. 11, pp. 2589-2606, 2009. [Online]. Available: <https://doi.org/10.1016/j.patcog.2008.12.029>
- [18] X. Bai, Y. Song, and P. M. Hall, "Learning invariant structure for object identification by using graph methods," *Comput. Vis. Image Underst.*, vol. 115, no. 7, pp. 1023-1031, 2011. [Online]. Available: <https://doi.org/10.1016/j.cviu.2010.12.016>
- [19] W. Lee and R. P. W. Duin, "A labelled graph based multiple classifier system," pp. 201-210, 2009. [Online]. Available: https://doi.org/10.1007/978-3-642-02326-2_21
- [20] W. Lee, R. P. W. Duin, and H. Bunke, "Selecting structural base classifiers for graph-based multiple classifier systems," pp. 155-164, 2010. [Online]. Available: https://doi.org/10.1007/978-3-642-12127-2_16
- [21] E. Z. Borzeshi, M. Piccardi, K. Riesen, and H. Bunke, "Discriminative prototype selection methods for graph embedding," *Pattern Recognit.*, vol. 46, no. 6, pp. 1648-1657, 2013. [Online]. Available: <https://doi.org/10.1016/j.patcog.2012.11.020>
- [22] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83-97, 1955.
- [23] M. Garland and P. S. Heckbert, "Surface simplification using quadric error metrics," pp. 209-216, 1997. [Online]. Available: <https://doi.org/10.1145/258734.258849>
- [24] F. Langenfeld, A. Axenopoulos, A. Chatzitofis, D. Craciun, P. Daras, B. Du, A. Giachetti, Y. Lai, H. Li, Y. Li, M. Masoumi, Y. Peng, P. L. Rosin, J. Sirugue, L. Sun, S. Thermos, M. Toews, Y. Wei, Y. Wu, Y. Zhai, T. Zhao, Y. Zheng, and M. Montes, "Protein shape retrieval," pp. 53-61, 2018. [Online]. Available: <https://doi.org/10.2312/3dor.20181053>
- [25] C. Hwang and K. Yoon, *Multiple Attribute Decision Making: Methods and Applications - A State-of-the-Art Survey*, ser. Lecture Notes in Economics and Mathematical Systems. Springer, 1981, vol. 186. [Online]. Available: <https://doi.org/10.1007/978-3-642-48318-9>
- [26] V. Yadav, S. Karmakar, P. P. Kalbar, and A. Dikshit, "Pytops: A python based tool for topsis," *SoftwareX*, vol. 9, pp. 217 - 222, 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S2352711018302279>
- [27] S. K. Burley, H. M. Berman, C. Bhikadiya, C. Bi, L. Chen, L. DiCostanzo, C. Christie, K. Dalenberg, J. M. Duarte, S. Dutta, Z. Feng, S. Ghosh, D. S. Goodsell, R. K. Green, V. Guranovi, D. Guzenko, B. P. Hudson, T. Kalro, Y. Liang, R. Lowe, H. Namkoong, E. Peisach, I. Periskova, A. Prli, C. Randle, A. Rose, P. Rose, R. Sala, M. Sekharan, C. Shao, L. Tan, Y.-P. Tao, Y. Valasatava, M. Voigt, J. Westbrook, J. Woo, H. Yang, J. Young, M. Zhuravleva, and C. Zardecki, "RCSB Protein Data Bank: biological macromolecular structures enabling research and education in fundamental biology, biomedicine, biotechnology and energy," *Nucleic Acids Research*, vol. 47, no. D1, pp. D464-D474, 10 2018. [Online]. Available: <http://www.rcsb.org>
- [28] B. Chen, Z. Yang, and Z. Yang, "An algorithm for low-rank matrix factorization and its applications," *Neurocomputing*, vol. 275, pp. 1012-1020, 2018. [Online]. Available: <https://doi.org/10.1016/j.neucom.2017.09.052>
- [29] K. P. Murphy, *Machine learning - a probabilistic perspective*, ser. Adaptive computation and machine learning series. MIT Press, 2012.
- [30] H. M. Berman, J. D. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne, "The protein data bank," *Nucleic Acids Research*, vol. 28, no. 1, pp. 235-242, 2000. [Online]. Available: <https://doi.org/10.1093/nar/28.1.235>