Graph Similarity through Entropic Manifold Alignment[*]

Francisco Escolano[†], Edwin R. Hancock[‡], and Miguel A. Lozano[†]

Abstract. In this paper we decouple the problem of measuring graph similarity into two sequential steps. The first step is the linearization of the quadratic assignment problem (QAP) in a low-dimensional space, given by the *embedding trick*. The second step is the evaluation of an information-theoretic distributional measure, which relies on deformable manifold alignment. The proposed measure is a normalized conditional entropy, which induces a positive definite kernel when symmetrized. We use bypass entropy estimation methods to compute an approximation of the normalized conditional entropy. Our approach, which is purely topological (i.e., it does not rely on node or edge attributes although it can potentially accommodate them as additional sources of information) is competitive with state-of-the-art graph matching algorithms as sources of correspondence-based graph similarity, but its complexity is linear instead of cubic (although the complexity of the similarity measure is quadratic). We also determine that the best embedding strategy for graph similarity is provided by commute time embedding, and we conjecture that this is related to its inversibility property, since the inverse of the embeddings obtained using our method can be used as a generative sampler of graph structure.

Key words. graph similarity, graph matching, graph embedding, graph kernels, nonparametric entropy estimation

AMS subject classifications. 68T45, 05C85

DOI. 10.1137/15M1032454

1. Introduction.

1.1. Motivation and previous work.
The accurate and effective measurement of graph similarity has proved to be a challenging problem in structural pattern recognition. Since state-of-the-art methods for object matching aim at incorporating structural information, advances in measuring graph similarity are pivotal to the development of successful object retrieval techniques. The problem of quantifying graph similarity has challenged researchers for over three decades. Early approaches included the work of Fischler and Elschlager [21], who exploited an elastic spring analogy, and Barrow and Poppleston's work [3] based on cliques of the association graph. In the late 1980s [19], with the emergence of structural pattern recognition as a distinct field of study, several attempts were made to extend the concept of edit distance from strings to graphs and trees. Earlier, Sanfeliu and Fu [45] showed how to associate

[†]Department of Computer Science and Artificial Intelligence, University of Alicante, Alicante 03690, Spain (sco@dccia.ua.es, malozano@ua.es).

[‡]Department of Computer Science, University of York, YO10 5GH York, UK (erh@cs.york.ac.uk).

edit costs with the insertion, deletion, and relabeling of nodes and edges, and developed greedy algorithms to find optimal matches. In 1981 Shapiro and Haralick [46] developed an elegant framework based on consistent clique counting, and in 1997 Bunke [7] used the maximum common subgraph to define the edit distance between graphs. However, these approaches are based on goal directed considerations motivated by graph theory and are not information theoretic. One of the earliest attempts to draw on information-theoretic concepts to measure graph similarity was presented by Boyer and Kak [6], who exploited the concept of mutual information. Christmas, Kittler, and Petrou [11] and Wilson and Hancock [54] later showed how relaxation labeling could be applied to the graph matching problem by modeling the probability distribution for matching errors using simple error models. Drawing on ideas from the connectionist literature, Gold and Rangarajan [22] developed a relaxation scheme based on soft-assign, and Finch, Wilson, and Hancock [20] took this work one step further by using ideas from statistical mechanics to develop a nonlinear version of Gold and Rangarajan's method. A Bayesian model has been designed for learning generative models using minimum description length and has exploited the model to compute information-theoretic edit distances [50].

Recently there has been renewed interest in the graph matching problem, stimulated in part by developments in object retrieval. Here a number of authors have attempted to extend the matching process to incorporate higher order relations. Zass and Shashua [56] were among the first to investigate this problem by introducing a probabilistic hypergraph matching framework, in which higher order relationships are marginalized to unary order. Chertok and Keller [9] improved this work by marginalizing the higher order relationships to be pairwise and then adopting pairwise graph matching methods. However, these two methods only approximate the hypergraph representation by using a clique graph. It has already been pointed out in [1] that this graph approximation is just a low pass representation of the original hypergraph and causes information loss and inaccuracy. On the other hand, Duchenne et al. [14] have developed the spectral technique for graph matching [29] into a higher order matching framework using the so-called tensor power iteration. Although they adopt an L_1 norm constraint in computation, the original objective function is subject to an L_2 norm and does not satisfy the basic probabilistic properties. The method described by Umeyama [51] can be interpreted as an implicit embedding method, while the more recent methods of Caelli and Kosinov [8] and Robles-Kelly and Hancock [44] make use of explicit embeddings. However, one of the weaknesses of these methods is that they are again not information theoretic in their development.

More recently, the factorized/deformable graph matching (FGM/DGM) approach proposed in [57] and [58] has uncovered the interplay between the topological information derived from node attributes and the attributes themselves. In this way, a unified approach to graph matching has been proposed, using a convex-concave relaxation of the quadratic assignment problem (QAP), similar to that used in the well-known path-following algorithm [55].

1.2. Contributions. In this paper, we decouple the problem of computing graph similarity into that of solving an approximate graph matching problem, followed by the estimation of an information-theoretic similarity measure computed from the available matching. This decoupling is motivated by the need to reduce the cubic complexity of state-of-the-art graph matching algorithms. We avoid measuring graph similarity in terms of the number of correct

correspondences, which has driven the continuous improvement of polynomial solutions to the QAP. Instead, we turn our attention to solving a linearized version of the QAP in an embedding space (we exploit the *embedding trick*) and then use this solution to estimate a highly discriminative graph similarity measure. In this paper, we define the conditions that must be satisfied for a graph embedding to be a good linearizer of the QAP, namely (a) the dimensionality is bounded by the intrinsic dimension, (b) it approximates the geodesic with an L_2 norm, and (c) the manifold embedding is reversible. We focus our analysis on (c) (reversibility) and contribute a formal development.

We show that despite being a fairly rough approximation of the QAP solution, the linearized solution obtained has sufficient inliers to support a low-energy global transformation between the manifolds induced by the embeddings. Such a transformation imitates the topological regularizing role of the QAP cost function (via the rectangle rule) but in a geometric space. Given this transformation, the computation of graph similarity is posed in terms of a normalized conditional entropy between the aligned manifolds. In this way we account for the high order statistical dependencies between the sampled manifolds. We prove that the similarity measure obtained induces a positive definite kernel.

The remainder of this paper is organized as follows. In section 2 we define the linearized version of QAP, referred to as the structural embedding graph matching (SEgm). SEgm is *purely topological*, i.e., it relies exclusively on the adjacency matrices of the graphs being matched, although it can also additionally accommodate attributes coming from edges or node characteristics, depending on the application domain. This purely topological approach allows us to understand the power of the embedding trick without relying on node or edge attributes. Significantly, it paves the way to a *distributional* graph similarity measure, the so-called normalized squared conditional entropy (NSCE). We detail the NSCE in section 3 and also prove that its symmetrized version is a positive definite kernel. Section 4 is devoted to approximating the kernel with a bypass entropy estimator. This requires that we perform some simplifications for the sake of efficiency. Then in section 5 we validate our approach through (a) testing the proposed strategy of *linearization + similarity*, referred to as *entropic alignment* or EA (see Figure 1) on a standard database (houses images dataset), (b) evaluating alternative information-theoretic measures and embeddings for a more challenging database (Gator), and (c) comparing with alternative algorithms, specifically FGM/DGM and path following, in terms of graph retrieval performance for both databases. From the experiments, we conclude that our strategy is competitive with FGM/DGM and path following. Moreover, the best performance is provided when the commute time embedding is used in the linearized step. In section 6 we formulate the problem of inverse embedding and prove that the commute time embedding is reversible. We conjecture that the success of such an embedding may be motivated by this property. Finally, in section 7 we present our conclusions and suggest directions for future work.

2. Structural embedding graph matching. Let $X = (V_X, E_X)$ and $Y = (V_Y, E_Y)$ be two undirected and unweighted graphs with respective node-sets V_X and V_Y, edge-sets E_X and E_Y, and numbers of nodes $n = |V_X|$ and $m = |V_Y|$. Also let $f_X : V_X \to \mathbb{R}^d$ and $f_Y : V_Y \to \mathbb{R}^d$ with $d \ll \max\{n, m\}$ be two *embedding functions* satisfying the following:

(a) They induce two low-dimensional subspaces (manifolds) $\mathcal{M}_X$ and $\mathcal{M}_Y$ of $\mathbb{R}^d$, where d is bounded by the intrinsic dimensions of the manifolds.
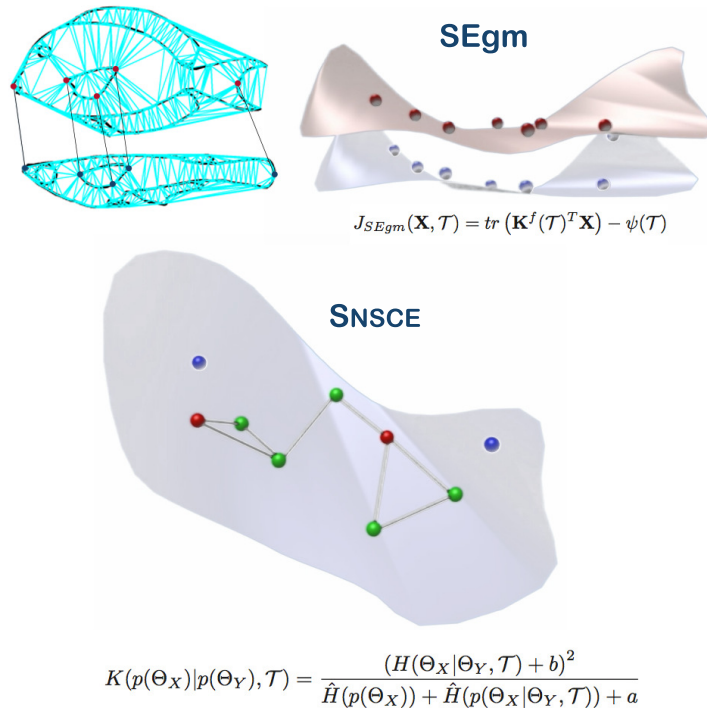
Figure 1. *Entropic alignment. Top-left: Inliers provided by SEgm. Top-right: SEgm formulation and manifold alignment. Bottom: After the optimal alignment we proceed to measure $\mathcal{S}_{NSCE}$, with inlier correspondences in green and outliers in red and blue; both are used for the kNN estimation of the entropies and then the positive definite kernel.*

(b) For each pair $i, j \in V_X$ we have that $g_X(i,j) \approx ||f_X(i) - f_X(j)||^2$, where $g_X(i,j)$ is the length of the geodesic between i and j, and similarly for $u, v \in V_Y$ and $g_Y(u,v) \approx ||f_Y(u) - f_Y(v)||^2$.

(c) E_X ans E_Y, respectively, can be inferred from $\mathcal{D}_X = \{||f_X(i) - f_X(j)||^2 \ \forall \ i, j \in V_X\}$ and $\mathcal{D}_Y = \{||f_Y(u) - f_Y(v)||^2 \ \forall \ u, v \in V_Y\}$ with bounded errors ϵ_X and ϵ_Y.

Embeddings f_X and f_Y rely on *topological* properties computed from adjacency matrices $\mathbf{A}_X$ and $\mathbf{A}_Y$, such as node degree, path-length distributions, and diffusive processes leading to random walks. When *geometric* properties of the nodes (position, relative angle, local image features, etc.) are available, then the graphs will be weighted (i.e., characterized by weight matrices $\mathbf{W}_X$ and $\mathbf{W}_Y$), and f_X, f_Y will be computed from, respectively, $\mathbf{A}_X + \alpha_x \mathbf{W}_X$ and $\mathbf{A}_Y + \alpha_y \mathbf{W}_Y$, $\alpha_x > 0$, $\alpha_Y > 0$.

Given two *extended graphs* $\mathcal{G}_X = \{V_X, E_X, \mathbf{A}_X, f_X\}$ and $\mathcal{G}_Y = \{V_Y, E_Y, \mathbf{A}_Y, f_Y\}$, with $\mathbf{A}_X \in \{0,1\}^{n \times n}$ and $\mathbf{A}_Y \in \{0,1\}^{m \times m}$, and embeddings f_X, f_Y with dimensionality $d \ll \max\{n, m\}$, the aim of SEgm is to find the one-to-one mapping (matching or correspondence) encoded by a partial permutation matrix $\mathbf{X} \in \Pi$, $\Pi = \{\mathbf{X} | \mathbf{X} \in \{0,1\}^{n \times m}, \mathbf{X}\mathbf{1}_m \leq \mathbf{1}_n, \mathbf{X}^T\mathbf{1}_n \leq \mathbf{1}_m\}$ maximizing

$$(1) \qquad\qquad J_{SEgm}(\mathbf{X}, \mathcal{T}) = \text{tr}\left(\mathbf{K}^f(\mathcal{T})^T\mathbf{X}\right) - \psi(\mathcal{T}) \ ,$$

where

(a) $\mathcal{T}(.,\mathbf{W})$ is a global nonrigid transformation parameterized by $\mathbf{W}$, and $\psi(.)$ is a regularization function typically given by $\psi(\mathcal{T}) = \lambda\mathrm{tr}(\mathbf{W}^T\mathbf{G}\mathbf{W})$ where $\mathbf{G}$ is a Green's function;

(b) $\mathbf{K}^f \in \mathbb{R}^{n \times m}$ is a *structural deformation matrix* given by

$$\mathbf{K}^f_{iu} = ||f_X(i) - \mathcal{T}(f_Y(u);\mathbf{W})||^2 \;,$$

i.e., by the deformation costs associated with the alignment of $\mathcal{M}_X$ and $\mathcal{M}_Y$.

Therefore, SEgm can be seen as a *linearization* of the *purely structural* version of QAP, whose objective is to maximize

$$(2) \qquad\qquad J_{QAP}(\mathbf{X}) = \mathrm{tr}\left(\mathbf{K}_q^T\left(\mathbf{G}_X^T\mathbf{X}\mathbf{G}_Y \circ \mathbf{H}_X^T\mathbf{X}\mathbf{H}_Y\right)\right) \;,$$

where $\mathbf{K}_q^T \in \mathbb{R}^{|E_X| \times |E_Y|}$ is the edge attributes matrix (only applying when $\mathbf{W}_X$ and $\mathbf{W}_Y$ are defined), $\circ$ is the Hadamard product, and $\mathbf{G}_X \in \{0,1\}^{n \times |E_X|}$, $\mathbf{G}_Y \in \{0,1\}^{m \times |E_Y|}$ are the *binary* node-edge incidence matrices ($\mathbf{G}_X^{ic} = \mathbf{H}_Y^{jc} = 1$ if the cth edge starts from i and ends at j, and similarly for $\mathbf{G}_Y$ and $\mathbf{H}_Y$). Here we follow the factorized graph matching formulation [57].

SEgm relies on the assumption that $\mathbf{X}_{SEgm}$, the global optimizer of $J_{SEgm}(\mathbf{X},\mathcal{T})$, is a reasonable approximation of $\mathbf{X}_{QAP}$, the global optimizer of $J_{QAP}(\mathbf{X})$. The error of the approximation depends on the following two factors:

1. The quality of the *embedding trick*. Graph embedding methods are designed to capture high order similarities between nodes. If the approximation $g_X(i,j) \approx ||f_X(i)-f_X(j)||^2$ is sufficiently good, we capture the long-range interactions between nodes which are by far more informative than the existence of edges and the node degrees. In this regard, the *inversibility property* (to what extent the original edges can be recovered from all pairs of Euclidean distances $||f_X(i) - f_X(j)||^2$) plays a critical role in the effectiveness of an embedding for the purposes of graph matching.

2. The *regularizing power* of $\mathcal{T}$. The apparent simplicity of $\mathbf{K}^f_{iu} = ||f_X(i)-\mathcal{T}(f_Y(u);\mathbf{W})||^2$ is misleading. The role of the nonrigid transformation $\mathcal{T}$ in SEgm is purely structural (i.e., it does not rely on node attributes), and it simulates the role of the topological regularization imposed by $\mathbf{G}_X^T\mathbf{X}\mathbf{G}_Y \circ \mathbf{H}_X^T\mathbf{X}\mathbf{H}_Y$ in QAP. It is well known that the *rectangle rule* represented by $\mathbf{X}_{iu}\mathbf{A}_X^{ij}\mathbf{A}_Y^{uv}\mathbf{X}_{jv}$ imposes the constraint that the adjacent nodes $i \in V_X$, $j \in V_X$ should match adjacent nodes $u \in V_Y$, $v \in V_Y$. This is the role of the quadratic cost of QAP and the origin of the NP-hard complexity of graph matching. Similarly, $\mathcal{T}$ enforces that the d-dimensional neighbors of both $f_X(j) \in \mathcal{M}_X$ and $f_Y(v) \in \mathcal{M}_Y$ match if $||f_X(i) - \mathcal{T}(f_Y(u);\mathbf{W})||^2$ is sufficiently small. Therefore, the combinatorial requirements are replaced by the regularizing power of a geometric transformation.

We should not expect the SEgm approximation to be sufficiently good for low-error correspondence recovery, even after a careful choice of the embedding and the regularizer. The reason for this is that the above linearization resembles that used in DGM [58], and which is designed for graph recovery. If our final objective is graph recovery or classification, then SEgm provides a set of inliers supporting the global alignment of $\mathcal{M}_X$ and $\mathcal{M}_Y$. Given such an

alignment, it leads to a similarity measure that is sufficiently discriminative to work effectively since such similarity is *distributional*.

3. The distributional graph similarity. Let Θ_X and Θ_Y be two random variables whose realizations are points in $\mathbb{R}^d$ belonging to $\mathcal{M}_X$ and $\mathcal{M}_Y$, respectively. Then, the conditional probability of observing Θ_X given Θ_Y after the alignment $\mathcal{T}(.; \mathbf{W})$ can be modeled by the factorization

$$(3) \qquad p(\Theta_X|\Theta_Y, \mathcal{T}) = \prod_{u=1}^{m} p_u(f_X(c(u))|f_Y(u), \mathcal{T}) ,$$

where

$$(4) \qquad p_u(f_X(c(u))|f_Y(u), \mathcal{T}) \propto \exp -\frac{1}{2}\left\{\left\|\frac{f_X(c(u)) - \mathcal{T}(f_Y(u); \mathbf{W})}{\sigma}\right\|^2\right\} ,$$

where $i \in V_X$, $u \in V_Y$ are graph nodes, $c : V_Y \to V_X$ is a *correspondence function* given by the optimal solution of the SEgm, and σ is a bandwidth parameter determined during the alignment. The bandwidth parameter σ is proportional to the global error, i.e., $\sigma \propto \sum_{i=1}^{n}\sum_{u=1}^{m}||f_X(i) - \mathcal{T}(f_Y(u)\mathbf{W})||^2 = \sum_i \sum_j \mathbf{K}_{iu}^{f}$, where $\mathbf{K}_{iu}^{f}$ is the structural deformation matrix.

Given the conditional density $p(\Theta_X|\Theta_Y, \mathcal{T})$, our similarity function relies on the conditional entropy $H(p(\Theta_X|\Theta_Y, \mathcal{T}))$, defined as

$$(5) \qquad H(p(\Theta_X|\Theta_Y, \mathcal{T})) \approx \hat{H}\left(p(\Theta_X)\right) - \hat{H}\left(p(\Theta_X|\Theta_Y, \mathcal{T})\right),$$

where $H(.)$ denotes the Shannon entropy and $\hat{H}(.)$ is an estimator of the Rényi entropy [28], whose limit is the Shannon entropy. As we will see later in section 4, it is the choice of estimators of Rényi type that validates the approximation in (5).

Then, the *normalized conditional entropy* between two random variables Θ_Y and Θ_X after the alignment is given by

$$(6) \qquad \bar{H}(\Theta_X|\Theta_Y, \mathcal{T}) = \frac{\hat{H}(p(\Theta_X)) - \hat{H}(p(\Theta_X|\Theta_Y, \mathcal{T}))}{\hat{H}(p(\Theta_X)) + \hat{H}(p(\Theta_X|\Theta_Y, \mathcal{T}))} .$$

Then, $\bar{H}(\Theta_X|\Theta_Y, \mathcal{T})$ has the following properties:
(a) The numerator is the conditional entropy $H(p(\Theta_X|\Theta_Y, \mathcal{T}))$ between the (sampled) manifolds given the transformation $\mathcal{T}$; i.e., it is the reduction in entropy of $p(\Theta_X)$ after the alignment. If the alignment provides two identical manifolds, then the conditional entropy is zero.
(b) Normalization by $\hat{H}(p(\Theta_X)) + \hat{H}(p(\Theta_X|\Theta_Y, \mathcal{T}))$ is key when we compare manifolds induced by graphs with a significantly different number of nodes. We prefer this form of the numerator to the alternative $\hat{H}(p(\Theta_X)) + \hat{H}(p(\Theta_Y))$, since it enforces the role of the conditional probability and the transformation.
(c) $\bar{H}(\Theta_X|\Theta_Y, \mathcal{T})$ is *directional*, i.e., $\mathcal{T} : \Theta_Y \to \Theta_X$, so that $p_u(f_X(c(u))|f_Y(u), \mathcal{T}) > 0$ whenever it is possible given the smoothness constraint imposed by the minimization of $\psi(\mathcal{T})$.

The above properties lead us to *define a kernel between the probability functions for the manifolds* and, thus, implicitly between the graphs. Such kernels are of pivotal importance for principled comparisons of the probability distributions associated with the manifolds [33]. In this regard, we have the following:

- Since the Shannon/Rényi entropy is negative definite (nd), and negative definiteness is closed under the sum, we have that $\hat{H}(p(\Theta_X)) + \hat{H}(p(\Theta_X|\Theta_Y, \mathcal{T}))$ is nd. Then

$$\frac{1}{\hat{H}(p(\Theta_X)) + \hat{H}(p(\Theta_X|\Theta_Y, \mathcal{T})) + a}$$

  is positive definite (pd) for any $a > 0$ (see Proposition 20 in [33]).
- Since $H(p(\Theta_X|\Theta_Y, \mathcal{T}))$ is nd, to ensure positive definiteness we add a nonnegative constant and square the alignment-based conditional entropy; i.e.,

$$(H(\Theta_X|\Theta_Y, \mathcal{T}) + b)^2 = \left(\hat{H}(p(\Theta_X)) - \hat{H}(p(\Theta_X|\Theta_Y, \mathcal{T})) + b \right)^2$$

  is pd for $b > 0$.

In addition to the above considerations is the fact that the product of two pd measures is pd, i.e., we have that

$$(7) \qquad K(p(\Theta_X)|p(\Theta_Y), \mathcal{T}) = \frac{(H(\Theta_X|\Theta_Y, \mathcal{T}) + b)^2}{\hat{H}(p(\Theta_X)) + \hat{H}(p(\Theta_X|\Theta_Y, \mathcal{T})) + a}$$

is a pd measure, for $a, b > 0$, between the density functions of Θ_X and Θ_Y. We refer to this measure as the NSCE between the two densities and, consequently, between the extended graphs $\mathcal{G}_X$ and $\mathcal{G}_Y$ given the alignment $\mathcal{T}$.

However, the NSCE is still not a kernel, since it is not symmetric due to the directionality of the transformation $\mathcal{T} : \Theta_Y \to \Theta_X$. Then, let $\mathcal{T}' : \Theta_X \to \Theta_Y$ be the nonrigid transformation given by $\mathcal{T}' = (.; \mathbf{W}')$. Such a transformation optimizes

$$(8) \qquad J_{SEgm}(\mathbf{X}', \mathcal{T}') = \mathrm{tr}\left(\mathbf{K}'^f (\mathcal{T}')^T \mathbf{X}' \right) - \psi(\mathcal{T}') \,,$$

where $\mathbf{X}' \in \{0,1\}^{m \times n}$ and $\mathbf{K}'^f \in \mathbb{R}^{m \times n}$ is the structural deformation matrix which has ui entries given by

$$\mathbf{K}'^f_{ui} = ||f_Y(u) - \mathcal{T}'(f_X(i); \mathbf{W}')||^2 \,.$$

Then, the definition of $p(\Theta_Y|\Theta_X, \mathcal{T}')$ in terms of $p_{ui}(\Theta_Y|\Theta_X, \mathcal{T}')$ gives

$$(9) \qquad \bar{H}(\Theta_Y|\Theta_X, \mathcal{T}') = \frac{\hat{H}(p(\Theta_Y)) - \hat{H}(p(\Theta_Y|\Theta_X, \mathcal{T}'))}{\hat{H}(p(\Theta_Y)) + \hat{H}(p(\Theta_Y|\Theta_X, \mathcal{T}'))} \,,$$

which in turn leads to

$$(10) \qquad K(p(\Theta_Y)|p(\Theta_X), \mathcal{T}') = \frac{(H(\Theta_Y|\Theta_X, \mathcal{T}') + b)^2}{\hat{H}(p(\Theta_Y)) + \hat{H}(p(\Theta_Y|\Theta_X, \mathcal{T}')) + a} \,.$$

Finally, the *symmetrized normalized squared conditional entropy* between two extended graphs $\mathcal{G}_X$ and $\mathcal{G}_Y$ is the pd kernel given by

$$(11) \qquad S_{NSCE}(\mathcal{G}_X, \mathcal{G}_Y) = K(p(\Theta_X)|p(\Theta_Y), \mathcal{T}) + K(p(\Theta_Y)|p(\Theta_X), \mathcal{T}') \ .$$

We can then exploit the kernel trick to classify graphs, and thus we recover or recognize objects by their structure, as in [15].

In the accompanying video (M103245_01.mp4 [local/web 7.18MB]) we illustrate the concept of commute time (CT) and show the processes of embedding and manifold alignment, and SNESV computation.

4. Leonenko–Pronzato–Savani entropy estimator. The $S_{NSCE}(\mathcal{G}_X, \mathcal{G}_Y)$ similarity measure is *distributional*. Here, the term distributional emphasizes the continuous nature of manifolds $\mathcal{M}_X$ and $\mathcal{M}_Y$ sampled at $f_X(i), i \in V_X$, and $f_Y(u), u \in V_Y$. Actually, the computation of $S_{NSCE}(.,.)$ requires, in principle, the estimation of densities $p(\Theta_X)$, $p(\Theta_Y)$, $p(\Theta_X|\Theta_Y, \mathcal{T})$, and $p(\Theta_Y|\Theta_X, \mathcal{T}')$. When d is very low (for example, 2D/3D) data, we can exploit nonparametric kernel density estimators such as the Parzen windows [37]. However, Parzen windows do not scale well with d and tend to overestimate entropy for medium/high dimensions, which is the case of graph embedding.

Therefore, instead of using a *plug-in* entropy estimator (inferring the probability density function before computing the Shannon entropy), here we use a *bypass* entropy estimator. Bypass estimators account for the neighborhood structure of the samples. In Appendix A, we analyze the Kozackenko–Leonenko Rényi-type entropy estimator [28] and its implications in estimating mutual information (MI) [27], because MI satisfies

$$(12) \qquad I(\Theta_X, \Theta_Y) = H(\Theta_X) - H(\Theta_X|\Theta_Y) = H(\Theta_Y) - H(\Theta_Y|\Theta_X) \ ,$$

that is, it is closely related to conditional probabilities: MI is the amount of uncertainty reduction due to the conditioning. Actually it should be more desirable to use MI as a similarity measure instead of conditional entropy. However, the Kozackenko–Leonenko estimator is better adapted to the alternative definition of MI:

$$(13) \qquad I(\Theta_X, \Theta_Y|c) = H(\Theta_X) + H(\Theta_X) - H(\Theta_X, \Theta_Y) \ ,$$

where $I(\Theta_X, \Theta_Y)$ is the *joint entropy* and $c : V_Y \to V_X$ is the correspondence function. This function is key to constructing an estimator of $H(\Theta_X, \Theta_Y)$, so that the samples $\mathbf{z}_u = (f_X(c(u)), f_Y(u))$ of the variable $\mathcal{Z}_{XY} = (\Theta_X, \Theta_Y)$ are properly built. The correspondence function establishes a common reference system as in the case of image alignment [36], [40]. However, the optimal transformation $\mathcal{T}$ is not needed here since it is not going to be applied to $f_Y(u)$ with $u \in V_Y$.

However, when applying the Kozackenko–Leonenko/Kraskow–Stögbauer–Grassberger approach, we obtain the entropy estimator

$$(14) \qquad \hat{H}_{N,k,1} = -\Psi(k) + \Psi(N) + \log V_d + \frac{d}{N} \sum_{i=1}^{N} \log \epsilon(i)$$

and its associated estimator of $I(\Theta_X, \Theta_Y; c)$,

$$(15) \qquad \hat{I}_{N,k,1} = \Psi(k) - \frac{1}{k} - \Psi(N) - \frac{1}{N} \sum_{u=1}^{N} (\Psi(n_x(c(u))) + \Psi(n_y(u))) \,,$$

where $\epsilon(i)$ is twice the Euclidean distance of the ith point of the manifold to its kth neighbor, and N is the number of independent and identically distributed (i.i.d.) samples of $\mathcal{X}' = \{f_X(c(u)), c(u) \in V_X\}$ and $\mathcal{Y} = \{f_Y(u), u \in V_Y\}$, i.e., $|\mathcal{X}'| = |\mathcal{Y}|$. Let $\epsilon_x(c(u))$ be the distance between $f_X(c(u))$ and its kth nearest neighbor in $\mathcal{X}'$, and let $\epsilon_y(u)$ be the distance between $f_Y(u)$ and its kth nearest neighbor in Y (here the max norm is used); then $n_x(c(u))$ and $n_y(u)$ are, respectively, the number of points $\mathbf{x}_j \in \mathcal{X}'$ with $||f_X(c(u)) - \mathbf{x}_j|| < \epsilon_x(c(u))/2$ and the number of points $\mathbf{y}_j \in \mathcal{Y}$ with $||f_Y(u) - \mathbf{y}_j|| < \epsilon_y(u)/2$. In addition, $\Psi(k) = \Gamma'(k)/\Gamma(k) = -\gamma + A_{k-1}$ is the digamma function with $\gamma \approx 0.5772$ (Euler constant), and $A_0 = 0$, $A_j = \sum_{i=1}^{j} 1/i$.

The $\hat{I}_{N,k,1}$ estimator does not include explicitly the distances $\epsilon_x(c(u))$ and $\epsilon_y(u)$. It embodies these distances in rank data: accounting for the expected number of points surrounding a given point in a ball of radius $\epsilon_x(c(u))$ or $\epsilon_y((u))$ gives an idea of the amount of joint entropy. However, $n_x(.)$ and $n_y(.)$ are the result of a marginalization of $\epsilon(u)$ (the distance between $\mathbf{z}_u = (f_X(c(u), f_Y(u))$ and its kth nearest neighbor). The marginalization is imposed by the fact that designing a $2d$ ball imposes a neighboring structure quite different from that used for estimating the marginal entropies, and this leads to larger systematic errors when d grows, because $\epsilon(u)$ tends to be much larger than the marginals. As a result, our experiments included in section 5 show that the $\hat{I}_{N,k,1}$ estimator leads to a poor discrimination.

As an alternative, the *symmetrized normalized squared conditional entropy* (11) relies on the conditional entropy $H(\Theta_X | \Theta_Y, \mathcal{T})$ (see (5)). Despite the conditional entropy being less effective than the MI for pattern discrimination, it is better adapted than MI when the Kozackenko–Leonenko/Kraskow–Stögbauer–Grassberger estimator is used. This is due to the following properties:

(a) We avoid the marginalization of $\epsilon(u)$ while preserving the consistency of the estimator. This is ensured by the compatibility of the ranges associated with both the samples of $\mathcal{X} = \Theta_X$ and those of $\mathcal{Z}_{X|Y} = (\Theta_X | \Theta_Y, \mathcal{T})$.

(b) We choose the samples for $\mathcal{Z}_{X|Y} = (\Theta_X | \Theta_Y, \mathcal{T})$ so that they are compatible with the conditional entropy $H(\Theta_X | \Theta_Y, \mathcal{T})$ *in conjunction with* the samples of $\mathcal{X} = \Theta_X$.

To commence, let us characterize the *entropy conditioning* in $\mathbb{R}^d$ in terms of replacing each point $f_X(c(u))$ by its corresponding point $f_Y(u)' = f_Y(u) + \mathcal{T}$ after the transformation $\mathcal{T}$ and then computing an entropy. The average of the m entropies is a proper approximation of the

conditional entropy, since

$$(16) \qquad H(\Theta_X | \Theta_Y, \mathcal{T}) = \sum_{u=1}^{m} p(\Theta_X) H(\Theta_X | \Theta_Y = f_Y(u)', \mathcal{T})$$

$$\approx \frac{1}{m} \sum_{u=1}^{m} H(\Theta_X | \Theta_Y = f_Y(u)', \mathcal{T})$$

$$\approx \frac{\mathcal{C}}{m} \sum_{u=1}^{m} \hat{H}_{m,k,1}(\{\Theta_X \sim f_X(c(u))\} \cup \{f_Y(u)'\})$$

$$\doteq \frac{\mathcal{C}}{m} \sum_{u=1}^{m} \hat{H}_{m,k,1}(X|u),$$

where $\mathcal{C}$ is a constant. It can be proved that

$$(17) \qquad \frac{\mathcal{C}}{m} \sum_{u=1}^{m} \hat{H}_{m,k,1}(X|u) = H_{m,k,1}(\Theta_X) + \mathcal{C} \sum_{e} \mathbb{E}_u \left(\log \left(1 \pm \frac{\delta e}{|e|} \right) \right),$$

where e are the edges of the kth neighborhood system of the points of Θ_X (see an example in Figure 2, where we drop $\mathcal{C}$ for the sake of clarity). We denote by $|e|$ the length of the edges and by δe the difference between their original lengths $|e|$ when a new point of Θ_Y is introduced. Then $\mathbb{E}_u \left(\log \left(1 \pm \delta e / |e| \right) \right)$ is the expectation of the log-relative errors for each edge over all choices of u defining $f_Y(u)'$. We refer to this approximation as *estimation from the average.*

However, if we fix the edges e of Θ_X and express those e' of Θ_Y in terms of e (named *estimation at a time* in Figure 2), we have that

$$(18) \qquad \hat{H}_{m,k,1}(\Theta_Y') \approx H_{m,k,1}(\Theta_X) + \mathcal{C} \sum_{e} \log \left(1 \pm \frac{\delta^2 e}{|e|} \right);$$

i.e., each expectation can be approximated by the log of a second-order error (as in the variance). This leads to the following approximation of the conditional entropy:

$$(19) \qquad H(p(\Theta_X | \Theta_Y, \mathcal{T})) \approx H_{m,k,1}(\Theta_X) - \hat{H}_{m,k,1}(\Theta_Y') = K \sum_{e} \log \left(1 \pm \frac{\delta^2 e}{|e|} \right).$$

This approximation can be interpreted in terms of a sum of log-likelihood ratios, since for $n = m$ we have that for the Leonenko–Pronzato–Savani entropy estimator [28] used in this paper (see more details in Appendix A),

$$(20) \qquad H(p(\Theta_X | \Theta_Y, \mathcal{T})) = \hat{H}_{m,k,2}(\mathcal{X}) - \hat{H}_{m,k,2}(\mathcal{Y}')$$

$$= \left(-\frac{\Psi(k)}{m} + \frac{\log(m-1)}{m} + \log V_d + \frac{d}{2m} \sum_{i=1}^{m} \log \epsilon_X(i) \right.$$

$$\left. + \frac{\Psi(k)}{m} - \frac{\log(m-1)}{m} - \log V_d - \frac{d}{2m} \sum_{u=1}^{m} \log \epsilon_{Y'}(u) \right)$$

$$= \frac{d}{2m} \sum_{i=1}^{m} \log \frac{\epsilon_X(i)}{\epsilon_{Y'}(u)},$$
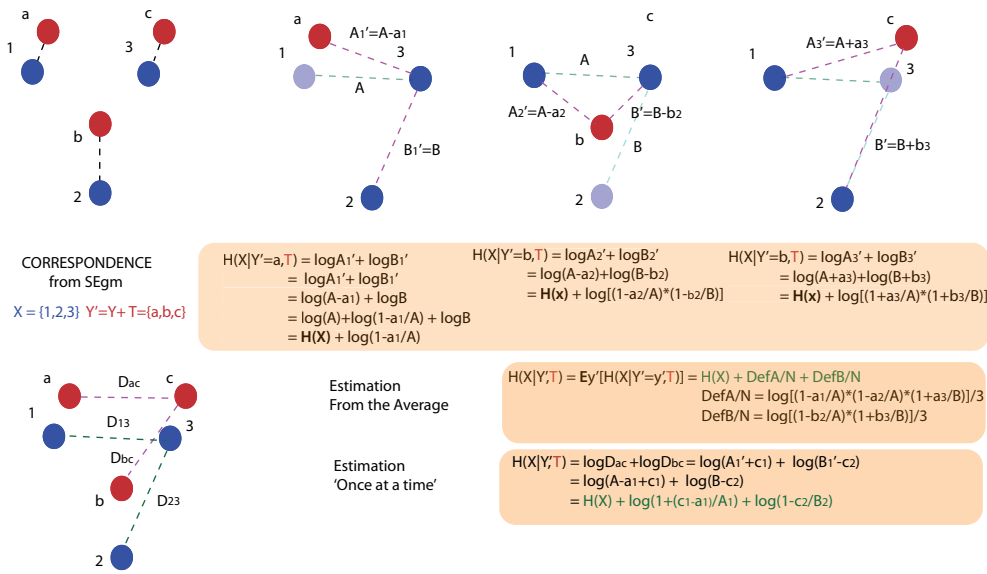
Figure 2. *Estimating conditional entropy (toy example). Blue dots are samples of X, and red ones are those of $Y' = Y + \mathcal{T}$. After the optimal alignment we proceed to compute $\hat{H}(X|Y', \mathcal{T})$. Top-right: We replace each X by a value of Y' and recompute the entropy. Rényi entropy is encoded by the neighborhood structures of X (in green) and of X after replacing X_i by the corresponding Y_i (magenta). Bottom: The average distortion is similar to that of making all the replacements at a time, and both depend on $\hat{H}(X)$. In all cases $k = 1$.*

where $i = c(u)$, and $\epsilon_X(.), \epsilon_{Y'}(.)$ are, respectively, twice the distances to the kth neighbor of the point $i = c(u)$ of $\mathcal{X}$ and to the kth neighbor of the uth point of $\mathcal{Y}'$. The Euclidean norm is used in order to avoid the marginalization of $\epsilon(.)$. When $m \neq n$ we have either positive or negative deviations/penalizations from this sum of log-likelihood ratios. Our estimator measures the neighborhood distortion, and this distortion is highly compatible, though not exactly so, with the conditional definition of entropy. For instance, when $\Theta_X = \Theta'_Y$ we have that the conditional entropy is zero as expected. Furthermore, in this way conditional entropy is consistent with the concept of *approximate entropy* insofar as it captures incremental variations [38].

Consequently, we then use the approximation in (20) (replacing $\hat{H}_{m,k,1}$ by $\hat{H}_{m,k,2}$) to compute $S_{NSCE}(\mathcal{G}_X, \mathcal{G}_Y)$ (11).

Finally, the i.i.d. assumption for entropy estimation is dictated by our ignorance about the type of graph to embed and the effect of the embedding function. Recent advances in cryptography [26], where estimating the correct amount of uncertainty is critical, point towards learning techniques that exploit the knowledge available about the random sources (the graphs and the embedding functions). In section 5.5, where we validate the commute time embedding as the most successful embedding function for graph matching/similarity purposes, we will analyze the impact of this choice in the entropy estimator.

5. Experimental results.

5.1. Entropic alignment settings. We refer to the proposed strategy of linearization + similarity as *entropic alignment* (EA). In our experiments, SEgm relies on the CPD (coherent
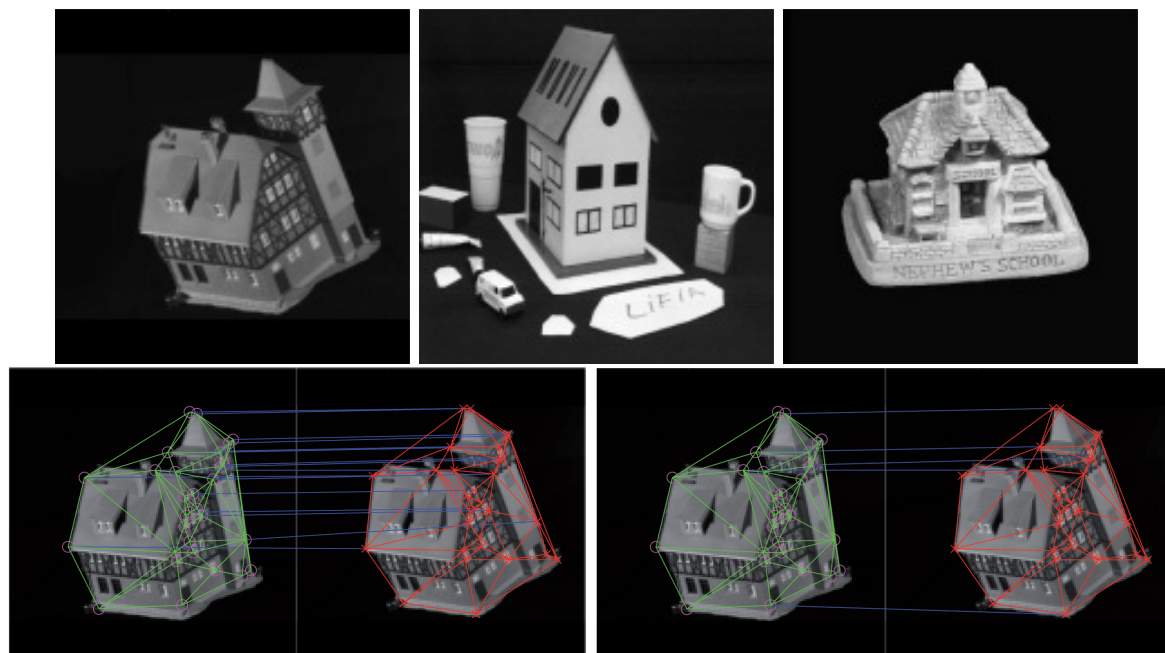
Figure 3. *Houses dataset. Top: From left to right, example frames of CMU, MOVI, and Chalet/York. Bottom: Examples of* inliers *obtained by SEgm when matching frames* 1 *and* 4 *in CMU for different embedding dimensions (d = 4 (left) and d = 11 (right)).*

point drift) algorithm [34] because it generalizes the nonrigid alignment to an arbitrary number of dimensions, say d, of the input data (manifolds in this case). CPD follows a similar approach to that in [25], where the samples are considered the centers of variance-isotropic d-dimensional Gaussian mixtures (GMM). For the Leonenko entropy estimator, which is the key element for measuring S_{NSCE}, we set $k = 4$.

5.2. Houses images dataset. The Houses (CMU+MOVI+Chalet) dataset consists of 10 frames of the CMU-VASC sequence,[1] 10 frames of the INRIA MOVI sequence, and another 10 frames of the Swiss chalet sequence created at the University of York (UK). These sequences have associated with them 30 graphs (Delaunay triangulations) and have been tested (totally or partially) in many papers addressing *pure topological* (attribute-free) graph matching methods (usually of spectral nature) such as [31], [48], [43], and [32]. In Figure 3(top) we show examples of frames for the three categories (a) CMU, (b) MOVI, and (c) Chalet.

We commence our experimental evaluation with this dataset because the topological variability increases from CMU to MOVI and Chalet. CMU has low intraclass variability and high interclass variability, and MOVI can be easily distinguished from CMU but confused with Chalet (it is by far more often confused with Chalet than with CMU). In addition, Chalet is the class with maximal intraclass structural variability and minimal interclass variability. The number of nodes ranges from 30 to 31 in CMU, 130–141 in MOVI and 40–136 in Chalet.

For the EA method, the first question to address is the *supporting quality of the inliers*

---

[1]http://vasc.ri.cmu.edu//idb/html/motion/house

Table 1
Summary of experiments with houses and Gator.

| Algorithm	Complexity	Attributed?	Dataset	AUC
Entropic alignment	$O(n)$	No	Houses	**21.5967**
Factorized graph matching [57]	$O(n^3)$	Yes	Houses	19.4800
Graduated assignment [22]	$O(n^4)$	Yes	Houses	19.4367
Spectral matching with affine const. [13]	$O(n^2)$	Yes	Houses	19.2667
Reweighted random walks [10]	$O(m^2)$	Yes	Houses	18.8167
Kernelized graduated assignment [31]	$O(n^4)$	No	Houses	17.4867
Kernelized graduated assignment [31]	$O(n^4)$	Yes	Houses	20.9600
Entropic alignment	$O(n)$	No	Gator	**59.6056**
PATH algorithm [55]	$O(n^3)$	No	Gator	58.6746
Graduated assignment [22]	$O(n^4)$	No	Gator	39.6155
Kernelized graduated assignment [31]	$O(n^4)$	No	Gator	33.2810
Kernelized graduated assignment [31]	$O(n^4)$	Yes	Gator	53.2810
Spectral matching with affine const. [13]	$O(n^2)$	Yes	Gator	49.0744
Reweighted random walks [10]	$O(m^2)$	Yes	Gator	46.3757
Reweighted random walks [10]	$O(m^2)$	No	Gator	46.0969
Tensor-based matching [14]	$O(n^3 \log n)$	Yes	Gator	50.5456
Caelli–Kosinov [8]	$O(n^3)$	No	Gator	39.8606

provided by SEgm. The quality depends on the dimensionality of the embedding d. In Figure 3(bottom) we show two extremal cases in CMU, where we have ground truth. For a low dimensionality ($d = 4$) we obtain 10 inliers (1/3 of the matchings), whereas for $d = 11$ this number is reduced to 5 (1/6 of the matchings). For each SEgm matching (CMU_i, CMU_j) the number of inliers varies significantly with d; i.e., there is no significant correlation (positive or negative) between d and the number of inliers for a given pair of matched CMU frames, which can be zero. The number of pooled inliers for (CMU_i, CMU_j) is in the range 98–1061 and in the interval 395.7 ± 205.1. In all of these experiments the graph embedding functions $f_X(.)$ and $f_Y(.)$ are given by the CT embedding [39].

Despite the high variance in the number of inliers with respect to d, we have that the S_{NSCE} similarity is quite robust with respect to variations of d in this dataset. In Figure 4(left), we show the evolution of the area under the curve (AUC) of the average recall/retrieval curves for d in the range 1–29. We found that the most discriminative value of d for this dataset is $d = 6$ (we cannot trust estimators of the intrinsic dimension since they tend to overestimate due to the curse of dimensionality).

In Figure 4(right) we show that the *pure topological version* (SEgm based on adjacency matrices) of EA outperforms state-of-the-art graph matching algorithms, such as FGM [57], spectral matching with affine constraint (SMAC) [13], reweighted random walks (RRW) [10], and graduated assignment (GA) [22], when node and/or edge attributes are used and graph similarity relies on their respective cost functions. In Table 1 (where *complexity* refers to complexity per iteration, where applicable) we summarize the results obtained for the AUCs of such algorithms. It is important to stress that although the best result for EA is provided by $d = 6$ (the optimal choice), we have that even with the minimal $d = 2$ EA outperforms the second best alternative (FGM) in terms of AUC. This reveals that the choice of d is not critical in this dataset.
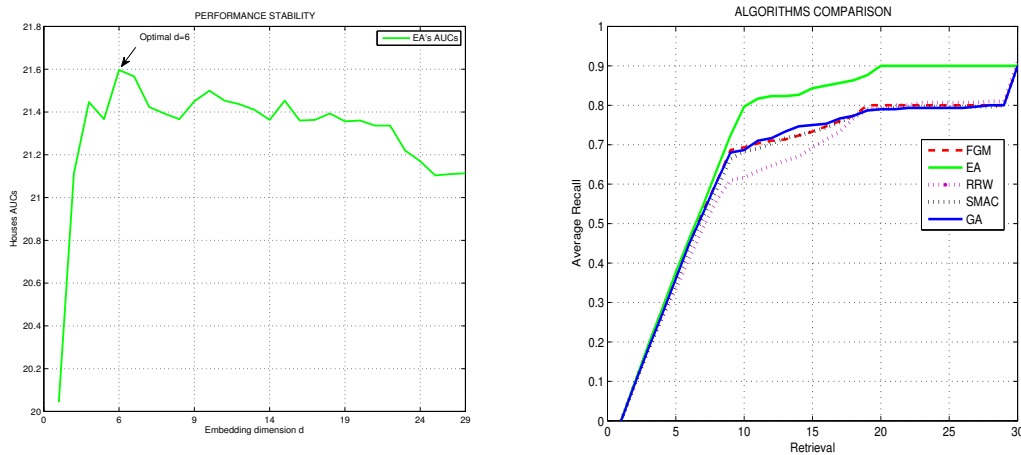
Figure 4. *Performance of EA vs. state-of-the-art alternatives. Left: Stability of Average Precision with respect to the embedding dimension d. Right: Average Precision/Recall curves of EA vs. state-of-the-art algorithms including FGM.*
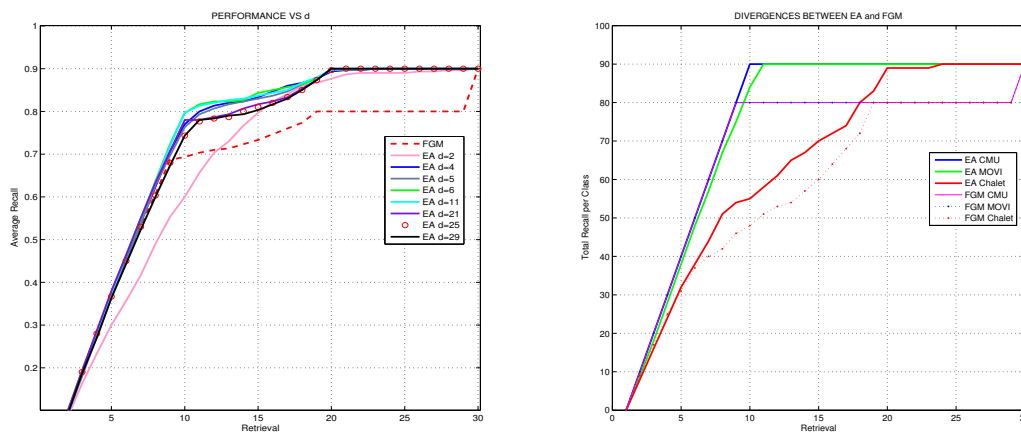
Figure 5. *Performance of EA vs. FGM. Left: Stability of EA's AUCs with respect to the embedding dimension d. Right: Analysis of total recalls for the three houses categories: EA vs. FGM class-by-class analysis.*

A more detailed analysis of the curves in Figure 5(right) reveals that EA with $d = 6$ outperforms FGM even for a small number of retrievals. Since the average recall/retrieval curves show how the performance improves when an increasing number of examples is considered for evaluation, we found that EA begins to improve the FGM method after only three retrievals.

The performance stability of EA with respect to d is detailed in Figure 5(left), where we show the average recall/retrieval curves of EA for different values of d. The curve for EA is only below that for FGM for $d = 2$. Close to the optimal value $d = 6$, AUC is maximal and decreases

slightly for higher dimensions. The performance of the FGM method diverges significantly from that obtained with EA after nine retrievals for $d > 2$, and later (after 12 retrievals) for $d = 2$. A closer *class-by-class* analysis of the divergence (see Figure 5(right)) reveals that FGM is competitive (up to 10 retrievals) when distinguishing between classes CMU and MOVI. After 10 retrievals we find that there is a constant gap between EA and FGM for these classes. This is mainly attributable to the fact that the quadratic cost function of FGM induces significantly more intraclass variability than EA when measuring the similarity between examples of CMU and MOVI. However, the main bulk of the performance divergence comes from the fact that FGM poorly discriminates the most structurally complex class, Chalet, from CMU and MOVI (at least such discrimination is worse than that given by EA). This means that EA is able to deal with high intraclass variability and low interclass variability. FGM, on the other hand, basically relies on the number of correspondences and the associative effects of the rectangle rule and is limited by the size of the smallest graphs in each class. This is why in CMU data, where all graphs have close to 30 nodes, FGM is (to some extent) competitive.

All of the above differences are exacerbated by the fact that EA does not rely on node or edge attributes, whereas FGM exploits this information. In the following section we focus our analysis on the differences between both algorithms *when only topological information is used*. To this end, we need a more complex and challenging dataset, which is provided by the Gator database.

5.3. The Gator dataset. The *Gator_100* Dataset is a *topological version* of the UCF Fish Shape Database.[2] It consists of 100 Delaunay triangulations extracted from images of fishes drawn from 30 different classes (see Table 2, where vertical lines separate examples of different classes). Since the classes are associated to fish genus and not to species, we find high intraclass variability—see Figure 6(a), where the corresponding class has eight species. There are also very similar species from different classes (row (b)) and few homogeneous classes (row (c)). There are 10 classes with one species, but these are not included in the analysis and performance curves. There are 11 with 1–3 individuals, five with 4–6 individuals, and only four classes with more than six species.
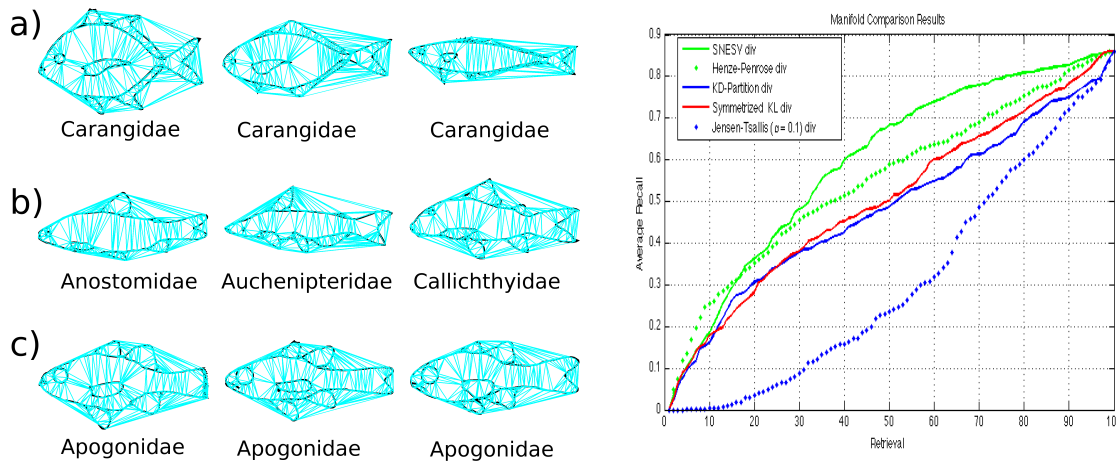
The design of the Gator dataset was motivated by initial shape recognition experiments showing that the S_{NSCE} was very competitive in terms of average recall/retrieval for the standard MPEG7-B 2D shape dataset [18]. These results have encouraged us to explore the same similarity measure for higher dimensions and to compare manifolds coming from graph embedding [16], where the embedding function was the CT.

---

[2]http://www.cise.ufl.edu/~anand/publications.html

Table 2

*Graphs based on Delaunay triangulations for the Gator Database.*

Figure 6. *Examples of the Gator database (left) and average recall curves (right).*

In our shape recognition experiments the most discriminative similarity measure was the Henze–Penrose divergence [36], followed by S_{NSCE} (previously referred to as SNESV or symmetrized normalized entropy squared variation). However, when applied to measure purely structural graph similarity, the most discriminative measure was S_{NSCE}-SNESV followed by Henze–Penrose (see Figure 6(right)). This result suggests that the *distributional* behavior of the similarity is good, in contrast with the 2D setting used for shape recognition. We have also analyzed alternative similarities based on bypass entropy estimators. These include (a) the symmetrized Kullback–Leibler divergence, (b) the Jensen–Tsallis divergence for $q = 0.1$ (both (a) and (b) are estimated through Leonenko's method), and (c) the total variation (L_1) divergence (KDP, k-d partitioning) where the entropy is estimated through k-d tree partitions.

In addition, the Gator dataset is ideal for comparing different choices of Kraskow–Stögbauer–Grassberger estimators, either for the conditional entropy or the MI, used for implementing our structural similarity measure. In Figure 7(left) we show the performance curves for these choices. The best one is the conditional entropy approximation described in (20) (AUC = 59.60). The second best choice consists of taking the distances between the deformed points and their corresponding points as variables for the conditioning. This leads to characterizing the conditional entropy, which controls the smoothness of the optimal matching field. It outperforms the approximation in (20) for a mid-low number of retrievals, which is very promising. However, as the number of retrievals increases, this second approximation is more prone to problems caused by Gator's interclass variability, and this leads to an AUC = 58.16. Finally, when the Kraskow–Stögbauer–Grassberger estimation of MI for joint entropy is used, the performance is very poor, giving an AUC of 44.73 when the optimal transformation is not applied and an AUC = 38.24 when joint entropy relies on pairs of deformed-original corresponding points.

The Gator dataset thus provides an encouraging setting for testing graph matching algorithms by *using only topological information,* i.e., that contained in the Delaunay triangulations, when it is possible. As with the houses dataset, we commenced by analyzing to what extent the embedding dimension d is critical in determining performance. For the Gator
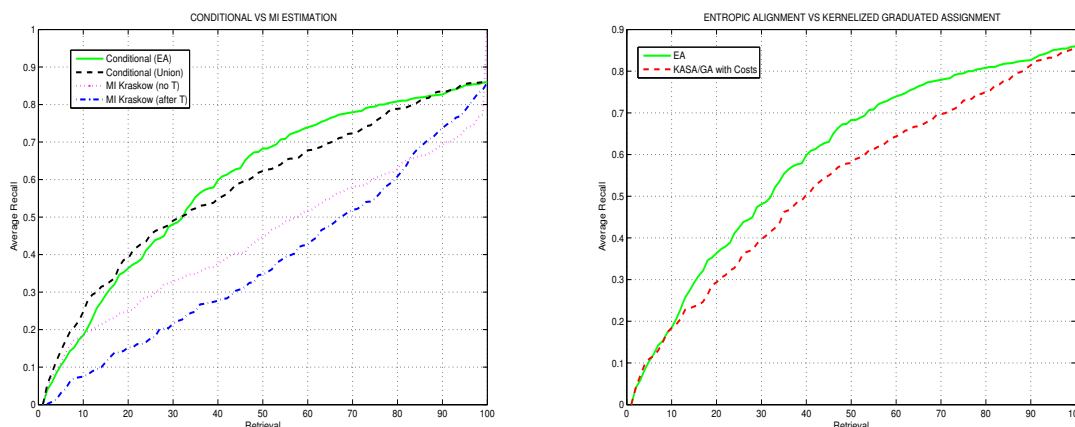
Figure 7. *Left: Comparing Kraskow–Stögbauer–Grassberger estimators: conditional entropy estimation vs. MI estimation. Right: Entropic alignment outperforms our previous method using structural attributes* [31].

dataset we found that the optimal choice was $d = 5$, whereas the estimates of the intrinsic dimension of the data were in the interval (11.6307 ± 2.8846). This overestimation of the intrinsic dimension is due to the curse of dimensionality. For instance, for $d = 10D$ we obtained a near-diagonal average recall/retrieval curve. The number of graph nodes in this dataset is in the range of 20–609, and this scenario of high intraclass variability, together with low-/mid-interclass variability, is significantly more challenging than that explored with the houses dataset.

We compare EA with (a) the classical nonattributed version of the GA method and (b) the attributed and nonattributed versions of RRW. In addition we test the tensor-based (TB) [14] method and the Caelli–Kosinov (CK) spectral method [8]. Tensor computation is not tractable for the raw Gator graphs due to their size. The problem of size also limits the applicability of the RRW method since it relies on a (weighted) association graph. Then, for these comparisons we use Delaunay triangulations obtained by decimating the original point sets by an order of magnitude.

We plot the obtained average precision/retrieval curves in Figure 8(left) and show their associated AUCs in Table 1. The most competitive retrieval strategy is provided by EA (which is nonattributed). The second best choice is the TB method. Here we use the 2D coordinates to compute the triangle, and the relatively good performance is due to the high order information provided by its triangular potentials.

Finally, we compare our EA method with the path-following (PATH) algorithm [55]. In the factorized/deformable graph matching method, the convex-concave relaxation process leading to approximate solutions (doubly stochastic matrices) for the QAP is key to its performance. At each iteration, the Frank–Wolfe algorithm leads to a local optimum. Each iteration takes $O(n^3 + 2m^2)$, where n is the number of nodes and m is the number of edges. The cubic complexity is due to the Hungarian algorithm used to compute the gradient.

The SEgm of EA is driven by coherent point drift [34], which can be done in $O(n)$ when the fast Gaussian transform is applied in conjunction with a linear system solver.
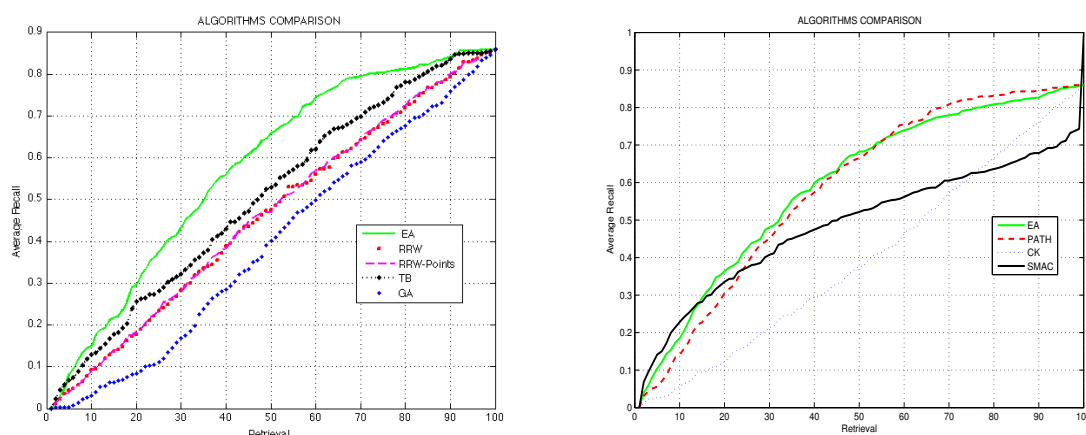
Figure 8. *Left: Comparison of several graph matching algorithms: EA, two versions of RRW, TB, and GA. Right: Comparison with PATH, SMAC, and CK.*

In Figure 8(right) we compare the results obtained with EA, PATH, the CK method, and SMAC. We obtain that EA outperforms PATH in terms of average recall/retrieval. The AUC for EA is 59.6 and for PATH is 58.7. This indicates that the values of the concave cost function of the PATH method capture the main structural similarities between graphs belonging to the same class and discriminates them from graphs belonging to different classes. This is due to the fact that the usual convex quadratic function only dominates the first iterations of the algorithm. The PATH algorithm evolves towards a concave version of this function. This concave function accounts for the spectra and the correlations (Kronecker products) between the Laplacians of the graphs being compared. Since EA, especially the SEgm step, also relies on the Laplacian matrices, we have that the *linearization* step of EA yields a fast approximation that is properly complemented by the S_{NSCE} similarity. As a result, PATH starts outperforming EA after 54 retrievals.

Our approach therefore combines both elements (good matching and good dissimilarity) by combining information furnished by graph embedding and information theory. Only the PATH algorithm is competitive with our approach, and this is purely topological.[3]

5.4. The importance of topological information. The method proposed in this paper is characterized as exploiting purely structural/topological information. It does not rely on additional attributes associated with the nodes such as distances and/or angles. Our alternative results are partially due to the embedding trick. We analyze the consequences of this trick in detail in section 6. However, there are alternative ways of exploiting topological information. In [31] we *kernelized* the Gold–Rangarajan method. There, we obtained node attributes from different types of graph kernels, mainly from the regularization kernel family (heat kernels, p-step kernels, and so on). When applying this strategy to the houses dataset, we obtain an AUC of 17.48 (see Table 1), a performance similar to that of RRW (AUC = 18.81) with

---

[3]MATLAB code and data for reproducing all of the experiments in this paper can be found at http://sites.google.com/site/scohomepage/ and will be soon submitted to IPOL.

feature-based attributes. This performance reaches an AUC of 20.96 when we add feature-based attributes. Moreover, this method is outperformed by EA, but it slightly outperforms FGM (AUC = 19.48). However, this is not the case with a more complex dataset such as Gator. PATH is still the second best choice, even when feature attributes are considered. EA improves on kernelized GA with feature attributes (Figure 7(right)).

5.5. Embedding comparison. Given a similarity measure like S_{NSCE}, the choice of the embedding is critical for determining the quality of the retrieval results. Here, we consider as alternative embeddings the CT embedding [39], Laplacian eigenmaps (LEM) [5], diffusion maps (DM) [35], heat kernels (HK) [2], and ISOMAP [49] (in this latter case we use the shortest path lengths between nodes as geodesics). The alternative embeddings rely on a function of the eigenvalues (diagonal of Λ) and/or eigenvectors (columns of Φ) of a property matrix. For instance, HK and CT embeddings result from a function $\mathcal{F}(.)$ applied to the Laplacian $\mathcal{F}(\mathbf{L}) = \Phi\mathcal{F}(\Lambda)\Phi^T = \Theta^T\Theta$, where the matrix of embedding coordinates Θ results from the Young–Householder decomposition of the kernel. For CT, $\mathcal{F}(\mathbf{L}) = \sqrt{vol}\Lambda^{-1/2}$, while for HK we have $\mathcal{F}(\mathbf{L}) = \exp\left(-\frac{1}{2}t\Lambda\right)$, where t is time. For DT we have $\mathcal{F}(\mathbf{L}) = \Lambda^t$, where Λ results from a generalized eigenvalue/eigenvector problem as in the case of LEM, where $\mathcal{F}(\mathbf{L}) = \Phi$. Finally, ISOMAP considers the leading eigenvectors of the geodesic distance matrix. Different embeddings yield different point distributions for the same dimensionality. For instance, CT produces denser point clouds than LEM (see [39]). For structural retrieval with a distributional measure such as S_{NSCE}, locating the optimal function is critical and must be determined empirically. Thus, we have obtained the retrieval-recall curves on the Gator database for each of the aforementioned embeddings with the setting $d = 5$. We plot the results in Figure 9(left). The CT embedding outperforms the alternatives. However, reasonable performance is obtained with ISOMAP and DM for $t = 64$ (a time setting that is sufficiently large to give an unfragmented embedding, given the size of the subsampled graphs).

Although CT gives good results, there are recent theoretical results which point to limitations of CT as a global characterization of kNN graphs for point sets resulting from the denseness of the embedding (see, for example, the recent work of von Luxburg, Radl, and Hein [53], [52]). More precisely, when we construct a kNN graph G over a large point set, this implies a high edge density. Under these conditions, we have that the *resistance distance* $R(i,j) = \frac{CT(i,j)}{vol(G)}$ satisfies the condition $R(i,j) \approx \frac{1}{\mathbf{D}(i,i)} + \frac{1}{\mathbf{D}(j,j)}$. In other words, it becomes meaningless as a measure of distance between vertices in a graph, since it depends only on their degree and not their separating path length or edge weights. An experimental means of quantifying proximity to this limit is to analyze the ratio $|R(i,j) - 1/\mathbf{D}(i,i) - 1/\mathbf{D}(j,j)|/R(i,j)$. If we plot the $\log(.)$ of the median of the ratio versus the size of the graphs, this should be monotonically decreasing with the size of the graphs. However, this is not the case for the Delaunay triangulation representations of graphs, since the edge density is relatively low. In fact, for the Gator database the median of the edge densities is $0.3409(34\%)$ and independent of graph size. In Figure 9(right) we show that the ratio defined above is not decreasing. More importantly, the values of the ratio are even higher when $d = 5$. This better performance for the five-dimensional case is explained by the fact that $\widehat{CT}(i,j) \leq CT(i,j)$, where $\widehat{CT}(i,j) = ||f(i) - f(j)||^2$ is the squared Euclidean distance between the d-dimensional embed-
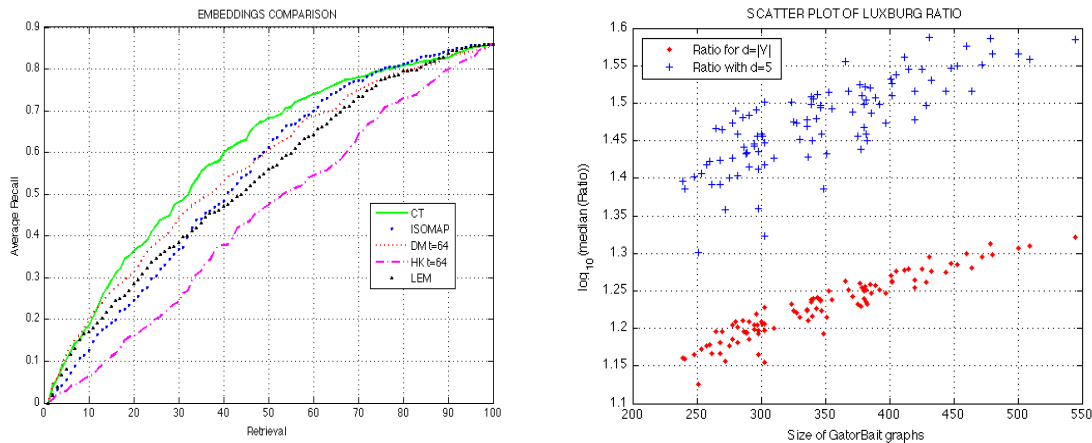
Figure 9. *Embedding analysis. Left: CT vs. ISOMAP, DM, HK, and LEM. Right: von Luxburg–Radl-Hein ratio scatter plots.*

dings of nodes i and j. This has the result of increasing the manifold density without losing the global topology of the graph. This is not the case for the HK embedding, which produces dense but poorly discriminating manifolds. Consequently it yields the poorest retrieval behavior.

5.6. The optimality of the CT embedding. In addition to the deviation of Delaunay triangulations from the von Luxburg law, we conjecture that the better behavior of the CT embedding derives from the reversibility of the embedding (that we explore in section 6). In turn, such reversibility depends on the degree distribution of Delaunay triangulations, since node degree plays an important role in the CT embedding.

Let $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be the points in $\mathbb{R}^n$ to be embedded into a subspace included in $\mathbb{R}^d$ with $d \ll n$, and let $\mathbf{W}_{ij} = \exp(-||\mathbf{x}_i - \mathbf{x}_j||^2/\sigma^2)$ be the similarity matrix. Then the CT embedding is given by the rows of $n \times d$ matrix $\mathbf{Z}$ minimizing [39]

$$(21) \qquad \epsilon' = \frac{\sum_{i=1}^n \sum_{j=1}^d ||\mathbf{Z}(i) - \mathbf{Z}(j)||^2 \mathbf{W}_{ij}}{\sum_{i=1}^n \sum_{j=1}^d \mathbf{Z}_{ij} \mathbf{D}(j,j)} = \mathrm{tr}\left(\frac{\mathbf{Z}^T \mathbf{L} \mathbf{Z}}{\mathbf{Z}^T \mathbf{D} \mathbf{Z}}\right),$$

where $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is the Laplacian matrix and $\mathbf{D}$ is the diagonal degree matrix. Since the denominator of ϵ' relies on the degrees, the optimal embedding can assign large coordinate values to nodes with large degree. This degree-of-freedom allows the scattering of embedded points so that the local structure of the original graph is preserved, because such local structure is determined by the degrees.

Recent studies [24] suggest that the degree distribution of Delaunay triangulations barely follows a power law. Following a power law means that few nodes have a large degree, whereas most of them have small degrees. This produces an exponential decay of the sorted degrees and gives a linear behavior with negative slope in the log-log space. However, as we can see in Figure 10, the slope of the decay is small ($\kappa = -0.3$), which means that the exponential decay is quite moderate. This increases the entropy of the degree distribution with respect to

those with more pronounced decays. Then, since most of the nodes tend to have a moderately high degree, the embedded points can be scattered according to their degrees. This maximizes the distances between the embedded points, at least globally. Since the numerator of (21) must be minimized, the CT embedding tends to map close points (or neighboring nodes, when adjacency matrices are considered instead of weight matrices) to the same cluster. However, the simultaneous minimization of the denominator tends to separate these clusters. In this way, the CT embedding amplifies the distance between tight groups.

This behavior in turn has an impact on entropy estimation since it relies on kNN tests. Nearest neighbors are frequently found in isolated clusters. It is well known that the curse of dimensionality compromises the performance of kNN rules. However, when CTs are used for embedding Delaunay triangulations, entropy estimation is quite robust for high d, as is the case with the houses dataset.

Since the embedded nodes are not i.i.d., the bypass entropy estimator used in this paper tends to underestimate the Shannon entropy. The use of this estimator produces consistent results provided that we do not mix the graphs being compared, which is a relatively mild assumption in the computer vision domain.

The above rationale explains the discriminative power of the CT embedding, since the Laplacian eigenmap, for instance, tends to minimize the numerator of (21) subject to a normalized version of the denominator. In this way, the embedding coordinates are uniformly scaled, which leads to a more entropic distribution of the embedded nodes. This configuration leads to an average retrieval/recall performance close to that for ISOMAP and lower than the performance for DM, which is the best alternative to CTs. Actually CTs come from the integral of diffusion times over time [39]. Consequently, since the nodes embedded by the Laplacian eigenmap are quite uniformly spaced, the embedding is prone to the curse of dimensionality, and then the kNN rules (and in turn the entropy estimator) fail.

As we will see in the next section, the combination of the properties of Delaunay triangulation and the nature of the CT embedding has a significant impact on the reversibility of the embedding.

6. From distances to structure. So far we have analyzed the CT embedding in its *direct* form. It provides a means of transforming the nodes of a graph $G = (V, E)$ into points in a d-dimensional vector space. When $d = |V|$, the Euclidean distance between the point positions of pairs of nodes is equal to the CT between them on the graph. When the embedding is into a subspace, i.e., $d < |V|$, the Euclidean distance is upper bounded by the CT. The embedding allows us to pose the problem of graph matching in terms of nonrigid point set alignment (SEgm), and we then measure graph similarity through the S_{NSCE} of the aligned samples. S_{NSCE} is designed to compare two d-dimensional probability distributions, and implicitly this means that we are representing the graphs to be matched as multidimensional probability distributions. This interpretation opens up additional and intriguing novel perspectives. For instance, *to what extent does the metric information in the embedding encode graph topology?* One way of answering this question is to explore *to what extent metric information is preserved under the embedding and the extent to which it is reversible.* In other words, under what conditions can we recover the original graph from its embedding? Moreover, if this is the case, then can we use the *inverse* of a vectorial generative model for the distribution of points
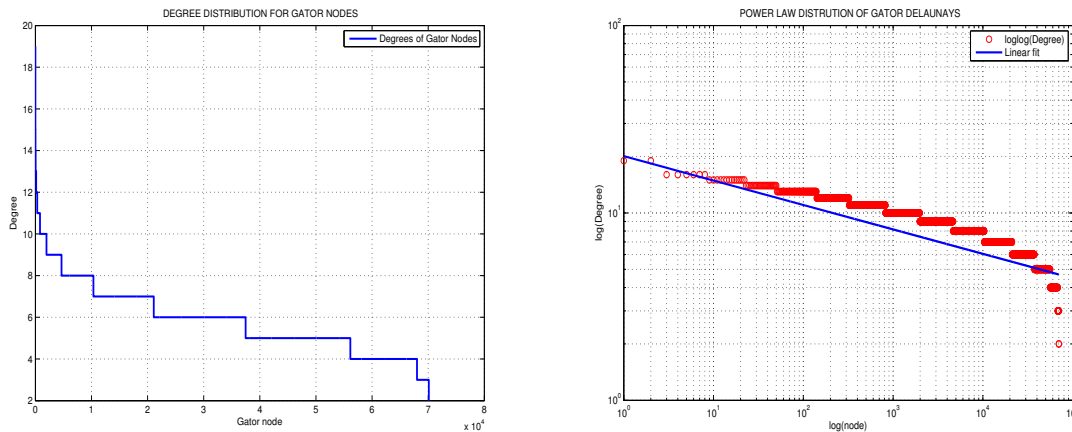
Figure 10. *Slight power law of degree distributions for Delaunay triangulations (Gator graphs). Left: Degree distribution. Right: Log-log curve and linear model fitting the data; slope is $\kappa = -0.3$.*

in the embedding space as a means of sampling graphs? This is of pivotal importance for constructing generative models for graphs, since state-of-the-art methods [23] are subject to the combinatorial constraints associated with the original topological space. We conjecture that these constraints can be bypassed by constructing the prototype in a subspace and then inverting the embedding.

In this section we propose an optimization algorithm (*inverse embedding*) to that end and also prove its convergence. Here we extend the formal results presented in [17].

6.1. Inverse embedding. Let $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ be a collection of N-dimensional points in the Euclidean space and generated by a node embedding of an unknown graph $G = (V, E)$ with $|V| = N$ and adjacency matrix $\mathbf{A}$. The problem of learning or inferring the graph G from the latter collection of multidimensional points can be posed as the following optimization problem:

$$
\begin{aligned}
\text{Max} \quad & \sum_{j>i} A_{ij} \\
\text{s.t.} \quad & \Theta_{ij} = ||\mathbf{x}_i - \mathbf{x}_j||^2 \\
& 0 \leq A_{ij} \leq 1 \quad \forall\, i, j\,,
\end{aligned}
$$
(22)

where $\Theta_{ij} = ||\Theta^{(i)} - \Theta^{(j)}||^2 = CT(i, j)$ and $\Theta^{(i)}, \Theta^{(j)}$ are the N-dimensional coordinates of the embedded nodes i and j, respectively. Following [39] we have

$$
CT(i, j) = vol \sum_{z=2}^{N} \frac{1}{\lambda_z} (\phi_z(i) - \phi_z(j))^2\,,
$$
(23)

where vol is the volume of the graph and λ_z, ϕ_z denote the zth eigenvalues and eigenvectors of the *unknown* normalized Laplacian $\mathcal{L}$. The embedding matrix is constructed with

$$
\Theta = \sqrt{vol}\,\Lambda^{-1/2}\Phi^T\,,
$$
(24)

where $\Lambda = \mathrm{diag}(\lambda_1 = 0, \lambda_2, \ldots, \lambda_N)$ and $\Phi = [\phi_1\,\phi_2\,\ldots\,\phi_N]$ is the matrix of eigenvectors which satisfies

$$(25) \qquad ||\Theta^{(i)} - \Theta^{(j)}||^2 = CT(i,j) \,.$$

The maximization of $\sum_{j>i} A_{ij}$ is consistent with finding the closest graph to the complete one—the initial proposal—which satisfies all of the embedding constraints.

Using Lagrange multipliers (one for each constraint), the problem is equivalent to maximizing (26) where the second (entropic) term relies both on an $x \log(x)$ barrier function and on β [41]. The third term contains the $N(N-1)/2 - N$ Lagrange multipliers α_{ij} (one multiplier per constraint).

$$(26) \qquad E(A, \{\alpha_{ij}\}) = \sum_{ij:j>i} A_{ij} + \frac{1}{\beta} \sum_{ij:j>i} A_{ij}(\log A_{ij} - 1)$$
$$+ \sum_{ij:j>i} \alpha_{ij}(\Theta_{ij} - ||\mathbf{x}_i - \mathbf{x}_j||^2).$$

The fixed point equations for updating the A_{ij} are given by

$$\frac{\partial E}{\partial A_{ij}} = 1 + \frac{1}{\beta} \log A_{ij} + \alpha_{ij} \frac{\partial \Theta_{ij}}{\partial A_{ij}}.$$
$$\frac{\partial E}{\partial A_{ij}} = 0 \Rightarrow \frac{1}{\beta} \log A_{ij} = -1 - \alpha_{ij} \frac{\partial \Theta_{ij}}{\partial A_{ij}}$$
$$(27) \qquad \Rightarrow A_{ij} = \exp \beta \left(-1 - \alpha_{ij} \frac{\partial \Theta_{ij}}{\partial A_{ij}} \right) \,,$$

where $\frac{\partial \Theta_{ij}}{\partial A_{ij}}$ (which can be approximated numerically) is the gain, in terms of the squared distance between Θ_i and Θ_j, with respect to the variation of a single component A_{ij}. On the other hand, the update of the multipliers has no available closed form solution and must be performed through gradient ascent, given the previously available estimates of the following multipliers and distances:

$$\frac{\partial E}{\partial \alpha_{ij}} = \Theta_{ij} - ||\mathbf{x}_i - \mathbf{x}_j||^2$$
$$(28) \qquad \Rightarrow \alpha_{ij}^{t+1} = \alpha_{ij}^t + \mu(\Theta_{ij}^t - ||\mathbf{x}_i - \mathbf{x}_j||^2) \,,$$

where $\mu \in [0,1]$ is the learning factor. In practice this factor must be set so that it decreases with the size of the graphs. The convergence of the inverse embedding procedure is dependent on the setting and control of this parameter.

6.2. Deterministic annealing algorithm. The fixed point equations for updating A_{ij} and the gradient ascent equations designed for updating the multipliers α_{ij} motivate the following deterministic annealing algorithm:

Initialize $\beta = \beta_0$, $A_{ij} = 1/N, \alpha_{ij} = 0, j > i, \mu$

Begin: Deterministic Annealing. Do while $\beta \leq \beta_f$

$\quad H \leftarrow ComposeAdjacencyMatrix(\{A_{ij}\})$

$\quad \Theta \leftarrow Embedding(H)$

$\quad \alpha_{ij} \leftarrow \alpha_{ij} + \mu(\Theta_{ij} - ||\mathbf{x}_i - \mathbf{x}_j||^2)$

$\quad \frac{\partial \Theta_{ij}}{\partial A_{ij}} \leftarrow ComputeDerivative(i, j, A)$

$\quad A_{ij} \leftarrow \exp \beta \left(-1 - \alpha_{ij} \frac{\partial \Theta_{ij}}{\partial A_{ij}} \right)$

$\quad \beta \leftarrow \beta \beta_r$

End

$G = MDLCleanup(\{\alpha_{ij}\})$

In this algorithm, the initialization $A_{ij} = 1/N \, \forall \, i \neq j$, that is, a barycenter depending on the complete graph ($A_{ij} = 1 \, \forall \, i \neq j$), ensures that the N-dimensional points of the embedding matrix Θ are initially equally spaced. More precisely, in this case we have that $\mathbf{H} = \frac{1}{N}\mathbf{K}_N$ is the adjacency matrix of the uniformly weighted complete graph with N nodes, and the diagonal degree matrix is $\mathbf{D}_H = \frac{N-1}{N}\mathbf{I}$. These two settings imply that $\mathcal{L}_H = \mathbf{I} - \frac{1}{N-1}\mathbf{K}_N = \mathcal{L}_{K_N}$. Consequently, the CTs for both graphs (encoded by $\mathbf{K}_N$ and $\frac{1}{N}\mathbf{K}_N$) are the same.

In a classic study on random walks [30] Lovász used a power series expansion to prove that in a complete graph of N nodes the hitting time between every pair of nodes is $N - 1$. For this type of graph we therefore have that the hitting times are symmetric, and hence $CT_H(i, j) = 2(N - 1) \, \forall \, i, j \in V_H$. Lovász also derived universal lower and upper bounds for CTs for any type of graph. The bounds are given in (29), where λ_2 is the Fiedler eigenvalue of the normalized Laplacian $\mathcal{L}_G$, that is, the so-called spectral gap of G. Since for a complete graph we have $\lambda_2 = \frac{N}{N-1}$, it is straightforward to prove that $CT_H(i, j) = 2(N - 1)$, where $2(N-1)$ is the upper bound. For any regular graph the lower bound is N. An in-depth analysis by von Luxburg, Radl, and Hein [52] shows that the probability of obtaining an incorrect CT in a kNN graph tends to unity when $k/(\log N) \to \infty$, and this occurs when a single node is connected directly to the remainder. This type of structural pattern may appear in certain clustering problems but does not arise for the types of graphs used in our target domain, i.e., computer vision. Here, planar graphs are typically derived from region adjacency relations or are Delaunay triangulations of points.

In Figure 11(left) we show that the spectral gap decays in a nonlinear way with increasing graph size for the Gator database. This decay results in a large value for the upper bound appearing in (29), and this in turn means that large values of CT are admissible. For each graph in Gator we also show the distribution of the differences between CT and the quantity $2(N - 1)$, where N is the size of the corresponding graph. We observe that the difference is positive and varies approximately linearly with N. This suggests that most of the CTs between pairs of nodes are longer than the expected value for a complete graph of the same size. However, it is highly improbable that this is the case for immediately adjacent nodes. In Figure 11(right) we distinguish the CTs between immediately adjacent nodes and those between the remaining nonadjacent ones. As expected, the median values of $CT(i, j) - 2(N - 1) \, \forall (i, j) \in E$ for the adjacent nodes tend to be negative. However, the distribution of CTs is dominated by $CT(i, j) \, \forall (i, j) \notin E$ for the remaining nodes, and this is why $CT(i, j) - 2(N - 1) \, \forall (i, j) \notin E$ is both highly positive ($\gg 0$) and also increases with N. Finally, we note that
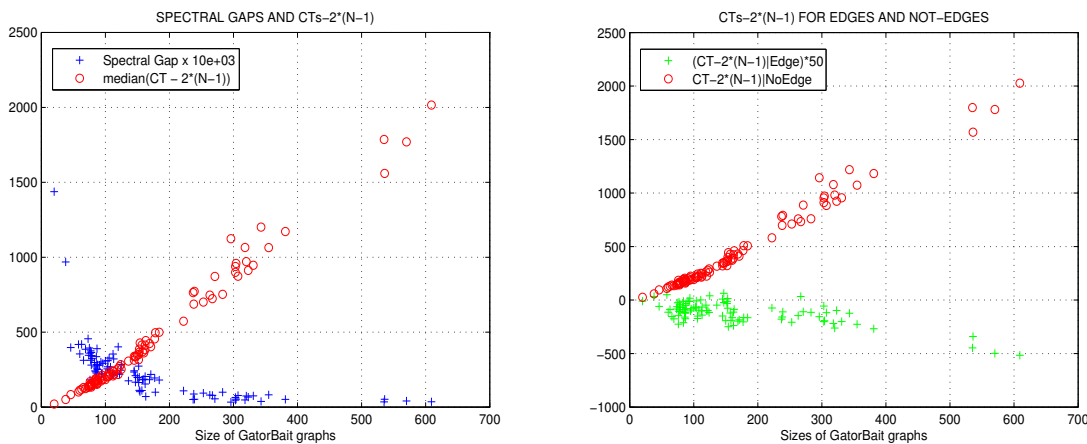
Figure 11. *CT analysis of Gator. Left: $CT - 2(N - 1)$ medians and spectral gaps ($\times 10^3$ for a better visualization). Right: $CT - 2(N - 1)$ modes both for adjacent ($\times 50$) and nonadjacent nodes.*

the CT between nodes i and j is bounded in the following manner, using the individual node degrees $\mathbf{D}(i, i)$ and $\mathbf{D}(j, j)$, the volume of the graph $vol(G)$, and the second smallest Laplacian eigenvalue λ_2:

$$(29) \qquad \frac{vol(G)}{2}\left(\frac{1}{\mathbf{D}(i,i)} + \frac{1}{\mathbf{D}(j,j)}\right) \leq CT(i,j) \leq \frac{vol(G)}{\lambda_2}\left(\frac{1}{\mathbf{D}(i,i)} + \frac{1}{\mathbf{D}(j,j)}\right).$$

The analysis of CT above is key to understanding the dynamics of our inverse embedding method and how to initialize it. We commence with an initialization that ensures equal squared distances between embedded points, i.e., $\Theta_{ij} = 2(N-1)$. More importantly, we have $\Theta_{ij} - ||\mathbf{x}_i - \mathbf{x}_j||^2 = \Theta_{ij} - CT(i, j)$, which is usually negative. In addition to the advantages of CT that we observe in Figure 11, we have also provided a more principled argument for its use in section 5.5. The moderate power law behavior of Delaunay triangulations in combination with the introduction of large distances between clusters in the CT motivates $\Theta_{ij} - CT(i, j) < 0$ in many cases.

The deterministic annealing (DA) algorithm progresses by maximizing (26), which is dominated by the second term $\frac{1}{\beta}\sum_{ij:j>i} A_{ij}(\log A_{ij} - 1)$ for low values of β. However, the elements of the adjacency matrix A_{ij} depend on the Lagrange multipliers α_{ij} (see (27)), which in turn depend on $\Theta_{ij} - ||\mathbf{x}_i - \mathbf{x}_j||^2 = \Theta_{ij} - CT(i, j)$ (28). As a result, in most cases the quantity $\Theta_{ij} - CT(i, j) < 0$ plays an important role in the dynamics of the algorithm. More precisely, the optimization process is focused on how the less negative multipliers emerge as β increases so that at convergence these multipliers will be associated to edges of the recovered graph. In Appendix B we detail the proof of convergence.

Once the algorithm has converged, we must extract the edges from the less negative multipliers. We address this task using an MDL (minimum description length) approach. We do not know in advance how many edges the hidden graph contains. We assume that it has a single connected component. Therefore, it seems reasonable to postprocess the multipliers so

that we select the minimum number such that the resulting graph is connected. This procedure does not preclude us from using statistics regarding the number of edges, should these become available. We use the *blind* inverse embedding and sort the multipliers in ascending order according to their absolute value. We commence by selecting the first $k = N - 1$ multipliers to check whether we have found a connected graph of N vertices ($N-1$ is the minimal number of edges that give a connected component on N vertices). If the multiplier α_{ij} satisfies this condition, then (i, j) is selected as an edge, i.e., $A_{ij} = 1$ and $A_{ji} = 1$. If the condition is not satisfied, we set $A_{ij} = A_{ji} = 0$ to the edges not selected. If the resulting graph is not singly connected, we make $\mathbf{A} = 0$ and repeat the latter procedure for $k + 1$ until convergence to a single component (the number of connected components is detected using spectral graph theory). This part of the algorithm is called $MDLCleanup(\{\alpha_{ij}\})$ and returns the MDL maximization of the objective function. The computational cost of the DA algorithm is $O(N^2 \times N^3) = O(N^5)$, since for each iteration we update a quadratic number of multipliers. Each update requires the computation of an embedding and thus the computation of eigenvalues and eigenvectors, which takes $O(N^3)$. However, as we will see in the experimental results for this part of the paper, for practical purposes the speed of convergence is very fast.

6.3. Results for Gator. We have successfully tested the proposed DA inverse embedding on several types of graphs, including linear ones. Linear graphs are difficult to obtain due to the fact that they are characterized by a small number of constraints (just $(N-1)$ sufficiently large multipliers are required). In general, each type of graph requires a different value of the parameter μ (for example, $\mu = 0.00001$ for a linear graph of $N = 50$ nodes and $\mu = 0.01$ for a grid graph with 10×4 nodes, each with a maximum of 4 neighbors). In order to determine whether the required original (hidden) structure is recovered, we define a reconstruction error measure. We have used $\mathcal{E} = \sum_{ij} \frac{|A_{ij} - A_{ij}^*|}{vol(G)}$, where G is the known graph (adjacency matrix) and G^* is the recovered adjacency matrix through inverse embedding. We consider both (i, j) and (j, i) as different edges, and thus we normalize by the volume of the graph. As a result, $\mathcal{E}$ defines a relative error. For instance, the linear graph was recovered with zero error, whereas the grid-like graph was recovered with $\mathcal{E} = 0.3647$ (36.47%). These preliminary results encouraged us to test our method on the challenging Gator database as a proof-of-concept of the usefulness of CT inverse embeddings to decode metric relations which are encoded by CT *direct* embeddings.

In Figure 12(top-left) we show the inverse embeddings of two example graphs of Gator. In both cases we set $\mu = 0.000000001 = 10^{-8}$, $\beta_0 = 0.5$, $\beta_r = 1.075$, and $\beta_f = 10$. In Figure 12(top-right) we show the convergence of the concave energy function. Errors for $Gator\#1$ and $Gator\#5$ are 34.30% and 39.53%, respectively. Most of the topology is consistently recovered (see Figure 12(bottom)). However, the numeric results may be misleading because we apply an MDL criterion in the reconstruction, and our method halts as soon as we detect a connected graph. This means that we may recover a graph which, while very closely related to the original one, may be somewhat simpler in structure. Entropic graph matching provides a way of testing this hypothesis, and it is the underpinning mechanism for learning prototypical manifolds, and thus generative models, in the future.
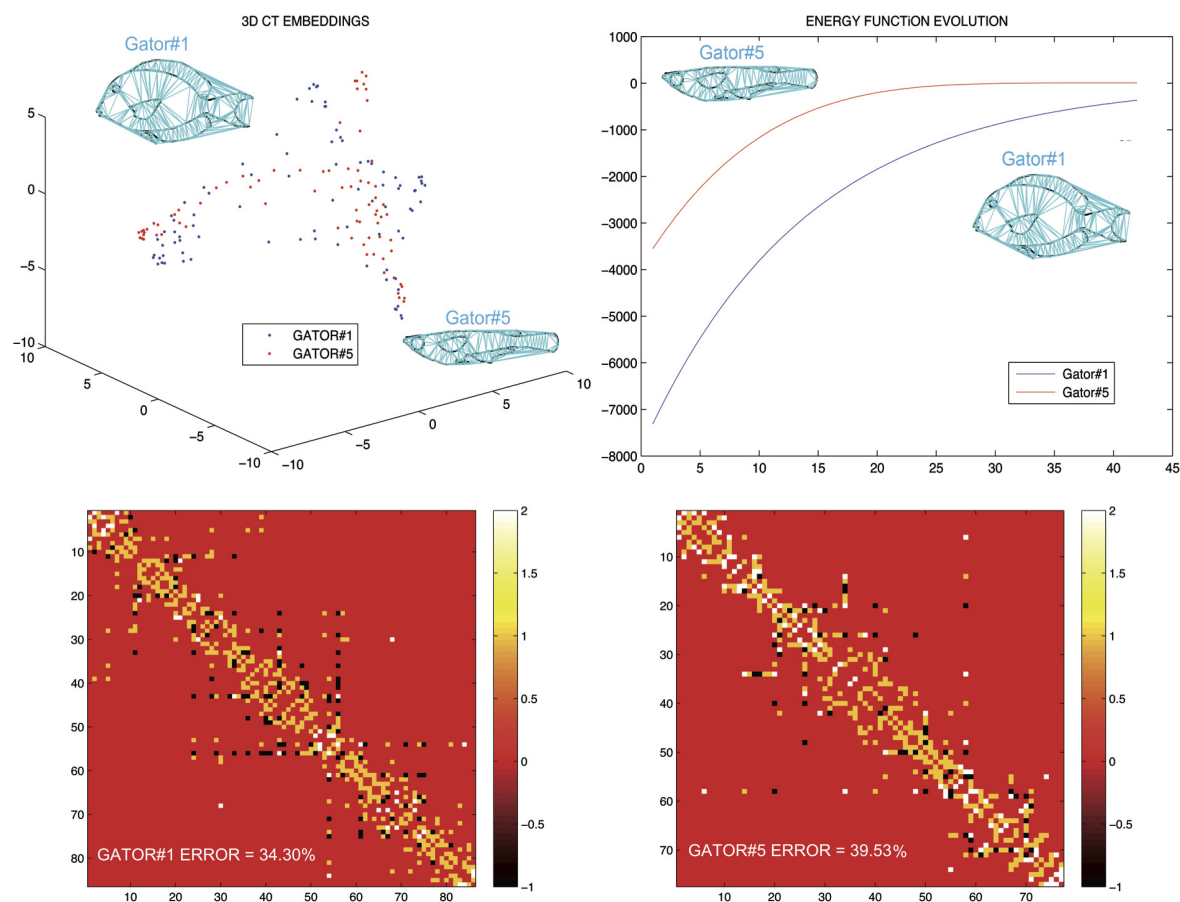
Figure 12. *Inverse embedding in Gator. Top-left: Examples of Gator#1 and Gator#5 CT embeddings for three dimensions (for visualization purposes, since the complete dimensions are used in both cases). Top-right: Evolution of the respective concave energy functions. Convergence speed is very fast. Bottom: Comparison between the adjacency matrix A and the inferred graph A^*; in each image we represent the following: $2A - A^*$ so that coincident edges have value $+1$, edges in A but not in A^* have value $+2$, and edges in A^* but not in G have value -1. Most of the values are $+1$ with errors 34.30% for Gator#1 and 39.53% for Gator#2.*

7. Conclusion. This paper decouples the measurement of graph similarity into two sequential steps. The first step is the linearization of the quadratic assignment problem (QAP) in a low-dimensional space, given by the embedding trick. The second step is the evaluation of an information-theoretic (IT) distributional measure which relies on deformable manifold alignment. Manifolds are obtained from the commute time (CT) embedding of Delaunay graphs. The proposed IT-based measure, the symmetrized normalized squared conditional entropy (S_{NSCE}), induces a positive definite (pd) kernel between manifolds and thus between graphs. Moreover, we have successfully tested the S_{NSCE} on two datasets and compared the CT with alternative state-of-the-art methods for embedding. We have also compared our approach with alternative competitive graph matching algorithms, including factorized/deformable graph matching (FGM and DGM) and path following (PATH). Our

algorithm outperforms most of them and is very competitive with FGM and PATH, despite relying only on topological information. Finally, we have addressed to what extent the original topology of the graph can be recovered from Euclidean distances (inverse embedding—see proof of convergence of the proposed deterministic annealing algorithm in Appendix B) as well as the impact of reversibility in the high discriminability of CTs.

Future work includes establishing formal links with graph edit distance. We are also investigating the joint role of entropic alignment (EA) and S_{NSCE} in learning prototypical manifolds from input exemplars. We are also developing alternative IT dissimilarities.

Appendix A. Kozachenko–Leonenko entropy estimators. In a multidimensional setting, where a random variable $\mathcal{X}$ is given by a set of i.i.d. samples (points) $\mathbf{x}_1, \ldots, \mathbf{x}_N$ in $\mathbb{R}^d$, a well-known *bypass* estimation of the Shannon entropy $H(\mathcal{X}) = -\int_{\mathcal{X}} p(\mathcal{X}) \log p(\mathcal{X}) d\mathcal{X}$ consists of approximating the density $p(\mathbf{x}_i)$ in terms of the distribution of the kNN neighbors of $\mathbf{x}_i$. In [27], for instance, $P_k(\epsilon)d\epsilon$ is the chance that (1) there is *one* point within distance $r \in [\epsilon/2, \epsilon/2 + d\epsilon/2]$ from $\mathbf{x}_i$, (2) there are $k - 1$ additional points at smaller distances, and (3) there are $N - k - 1$ points with larger distances from $\mathbf{x}_i$. These three conditions lead to a trinomial model for $P_k(\epsilon)d\epsilon$,

$$(30) \qquad P_k(\epsilon)d\epsilon = \frac{(N-1)!}{1!(k-1)!(N-K-1)!} \frac{dp_i(\epsilon)}{d\epsilon} d\epsilon \times p_i(\epsilon)^{k-1} \times (1 - p_i(\epsilon))^{N-k-1} ,$$

where $p_i(\epsilon) = \int_{||\xi - \mathbf{x}_i|| < \epsilon/2} p(\xi) d\xi$ is the mass of the ϵ ball centered at $\mathbf{x}_i$. Using the formal link between Dirichlet-like distributions and digamma functions, we obtain the expectation of $\log p_i(\epsilon)$ for point i:

$$(31) \qquad E(\log p_i(\epsilon)) = \int_0^\infty \log p_i(\epsilon) P_k(\epsilon) d\epsilon$$
$$= k \binom{N-1}{k} \int_0^1 p_i(\epsilon)^{k-1} \times (1 - p_i(\epsilon))^{N-k-1} \log p_i(\epsilon) dp_i$$
$$= \Psi(k) - \Psi(N) ,$$

where the expectation is taken over the positions of all remaining $N - 1$ points with $\mathbf{x}_i$ fixed, and where $\Psi(x) = \frac{\Gamma'(x)}{\Gamma(x)} = \frac{d}{dx} \log \Gamma(x)$ is the digamma function, whose properties are similar to those of the natural logarithm. Assuming that the density $p(\mathbf{x}_i)$ is constant inside the ϵ ball, we have $p_i(\epsilon) \approx V_d \epsilon^d p(\mathbf{x}_i)$, where V_d is the volume of the d-dimensional unit ball in $\mathbb{R}^d$ ($V_d = 1$ for the maximum norm and $V_d = \Gamma(1 + d/2)/2^d$ for the Euclidean norm). Then, following the asymptotic equipartition property (a consequence of the law of large numbers) we have that $-E(\log(p(\mathbf{x}_i))) = (-1/N) \sum_{i=1}^N \log(p(\mathbf{x}_i)) \to H(\mathcal{X})$ as $N \to \infty$. This leads to the following estimator of the Shannon entropy:

$$(32) \qquad \hat{H}_{N,k,1} = -\Psi(k) + \Psi(N) + \log V_d + \frac{d}{N} \sum_{i=1}^N \log \epsilon(i)$$

and to its Rényi-like counterpart [28],

$$(33) \qquad \hat{H}_{N,k,2} = -\frac{\Psi(k)}{N} + \frac{\log(N-1)}{N} + \log V_d + \frac{d}{2N} \sum_{i=1}^N \log \epsilon(i) ,$$

where $\epsilon(i)$ is twice the distance to the kth nearest neighbor of $\mathbf{x}_i$. Then, both $\hat{H}_{N,k,1}$ and $\hat{H}_{N,k,2}$ can be understood as the result of quantifying entropy in terms of the kNN distances and then adding correction terms related to the digamma function.

We can follow a similar rationale for estimating MI $I(\mathcal{X}, \mathcal{Y})$ (see details in [27]). In this case, a two-dimensional joint space must be constructed since $I(\mathcal{X}, \mathcal{Y}) = H(\mathcal{X}) + H(\mathcal{Y}) + H(\mathcal{X}, \mathcal{Y})$. Let $\mathbf{z}_i = (\mathbf{x}_{c(i)}, \mathbf{y}_i)$ be the samples of $\mathcal{Z} = (\mathcal{X}, \mathcal{Y})$ resulting from stacking the samples $\mathbf{x}_i$ and $\mathbf{y}_{c(i)}$ according to a correspondence function $c : \mathbb{N} \to \mathbb{N}$. The correspondence function is typically given beforehand; i.e., the notation $\mathbf{z}_i = (\mathbf{x}_i, \mathbf{y}_i)$ assumes that the samples of $\mathcal{Y}$ have been previously reordered with respect to those of $\mathcal{X}$ or vice versa. For the sake of simplicity, we follow the $\mathbf{z}_i = (\mathbf{x}_i, \mathbf{y}_i)$ notation in this appendix.

When using the maximum norm, the construction of the joint probability distribution $P_k(\epsilon_x, \epsilon_y)$ relies on hyper rectangles of sides $\epsilon_x(i)$ and $\epsilon_y(i)$, where $\epsilon_x(i)$ is twice the distance of $\mathbf{x}_i$ to the kth nearest neighbor from the set of samples of $\mathcal{X}$, and $\epsilon_y(i)$ is similarly defined, in this case with respect to $\mathbf{y}_i$ and $\mathcal{Y}$. Then we have

$$(34) \quad P_k(\epsilon_x, \epsilon_y) = \binom{N-1}{k} \frac{d^2[q_i^k]}{d\epsilon_x d\epsilon_y} (1 - p_i(\epsilon))^{N-k-1} + (k-1) \binom{N-1}{k} \frac{d^2[q_i^k]}{d\epsilon_x d\epsilon_y} (1 - p_i(\epsilon))^{N-k-1} ,$$

where $q_i = q_i(\epsilon_x, \epsilon_y)$ is the mass of the rectangle of size $\epsilon_x \times \epsilon_y$ centered at $(\mathbf{x}_i, \mathbf{y}_i)$, and $p_i(\epsilon)$ is the mass of the square of size $\epsilon = \max\{\epsilon_x, \epsilon_y\}$. Then $E(\log q_i)$ is given by

$$(35) \quad E(\log q_i(\epsilon_x, \epsilon_y)) = \int_0^\infty \int_0^\infty \log q_i(\epsilon_x, \epsilon_y) P_k(\epsilon_x, \epsilon_y) d\epsilon_x d\epsilon_y$$
$$= \Psi(k) - \frac{1}{k} - \Psi(N) ,$$

which leads to the following estimator of $I(\mathcal{X}, \mathcal{Y})$:

$$(36) \quad \hat{I}_{N,k,1} = \Psi(k) - \frac{1}{k} - \Psi(N) - \frac{1}{N} \sum_{i=1}^{N} (\Psi(n_x(i)) + \Psi(n_y(i))) ,$$

where $n_x(i)$ and $n_y(i)$ are, respectively, the number of points with $||\mathbf{x}_i - \mathbf{x}_j|| < \epsilon_x(i)/2$ and $||\mathbf{y}_i - \mathbf{y}_j|| < \epsilon_y(i)/2$.

Appendix B. Proof of deterministic annealing convergence. Following the methodology in [42], in order to prove the convergence of the DA method proposed for maximizing (26) we must find a Lyapunov function $\Delta E = E^{t+1} - E^t > 0$ so that the increment of energy between iteration t and iteration $t + 1$ is always positive. Let $\phi(A_{ij}) = A_{ij}(\log A_{ij} - 1)$ be the barrier function. Then we have (37). The convexity of $\phi(A_{ij})$ implies (38). For the A_{ij} we have that $\frac{1}{\beta}(\log A_{ij}) = -1 - \alpha_{ij} \frac{\partial \Theta_{ij}}{\partial A_{ij}}$, and therefore setting $d_{ij}^2 = ||\mathbf{x}_i - \mathbf{x}_j||^2$ we obtain (40).

Proving $\Delta E > 0$ involves proving the following in turn:
1. Negative increment: $\Delta A < 0$, that is, $A_{ij}^{t+1} < A_{ij}^t$;
2. mostly positive products involving multipliers and derivatives: $\alpha_{ij} \frac{\partial \Theta_{ij}}{\partial A_{ij}}$;
3. positive increment of the sum of products involving multipliers and degree of constraint satisfaction: $\sum_{ij:j>i} \alpha_{ij}(\Theta_{ij} - ||\mathbf{x}_i - \mathbf{x}_j||^2)$.

First, considering how the A_{ij} are updated, equality

$$A_{ij} = \exp \beta \left(-1 - \alpha_{ij} \frac{\partial \Theta_{ij}}{\partial A_{ij}}\right) \geq 0$$

implies that for μ small enough we have that $A_{ij}^t = \exp \beta_t(-1 - \epsilon^t) \approx \exp(-\beta_t) \;\; \forall t > 0$ for $\epsilon^t = \alpha_{ij}^t \frac{\partial \Theta_{ij}^t}{\partial A_{ij}^t} \ll 1$. Consequently, under these conditions $A_{ij}^{t+1} < A_{ij}^t \; \forall t > 1$, since $A_{ij}^0 = 1/N$ and β_0 is a free parameter. If we choose β_0 so that $\exp \beta_0(-1) < 1/N$ we will have a positive increment, but the function is dominated by the barrier function, and the overall energy will increase with respect to $t = 0$. In any case we must set $\beta_0 \gg \mu$ (which is a mild assumption since $\mu \in [0,1]$) to ensure $\Delta A < 0$. Then, as β increases with the number of iterations, $\Delta A < 0$ is also ensured for large values of β. Therefore we may assume that $A_{ij}^{t+1} \approx \exp(-\beta_{t+1}) = \exp(-\beta_t \beta_r)$, because the perturbations induced by ϵ^{t+1} are attenuated by β_{t+1}. Summarizing, μ should be small enough for setting $\epsilon^t \ll 1$, but it should also be large enough to provide significant updates of the multipliers.

The change in energy is given by

$$(37) \quad \begin{aligned} \Delta E(A, \{\alpha_{ij}\}) = &\sum_{ij:j>i} \Delta A_{ij} + \frac{1}{\beta} \sum_{ij:j>i} \phi(A_{ij}^{t+1}) - \frac{1}{\beta} \sum_{ij:j>i} \phi(A_{ij}^t) \\ &+ \sum_{ij:j>i} \alpha_{ij}^{t+1}(\Theta_{ij}^{t+1} - \|\mathbf{x}_i - \mathbf{x}_j\|^2) - \sum_{ij:j>i} \alpha_{ij}^t(\Theta_{ij}^t - \|\mathbf{x}_i - \mathbf{x}_j\|^2). \end{aligned}$$

$$(38) \quad \sum_{ij:j>i} \phi(A_{ij}^{t+1}) - \sum_{ij:j>i} \phi(A_{ij}^t) \geq \sum_{ij:j>i} \phi'(A_{ij}) \Delta A_{ij} \equiv \sum_{ij:j>i} (\log A_{ij}) \Delta A_{ij} .$$

(39)

$$\begin{aligned} \Delta E(A, \{\alpha_{ij}\}) \geq &\sum_{ij:j>i} \Delta A_{ij} + \sum_{ij:j>i} \left(-1 - \alpha_{ij} \frac{\partial \Theta_{ij}}{\partial A_{ij}}\right) \Delta A_{ij} \\ &+ \sum_{ij:j>i} \alpha_{ij}^{t+1}(\Theta_{ij}^{t+1} - d_{ij}^2) - \sum_{ij:j>i} \alpha_{ij}^t(\Theta_{ij}^t - d_{ij}^2) \\ = &\sum_{ij:j>i} -\left(\alpha_{ij} \frac{\partial \Theta_{ij}}{\partial A_{ij}}\right) \Delta A_{ij} + \sum_{ij:j>i} \alpha_{ij}^{t+1}(\Theta_{ij}^{t+1} - d_{ij}^2) \\ &- \sum_{ij:j>i} \alpha_{ij}^t(\Theta_{ij}^t - d_{ij}^2) > 0. \end{aligned}$$

Second, we must prove that $\epsilon^t > 0$ in most of the cases, although we set μ so that $\epsilon^t \ll 1$. Since $\epsilon^t = \alpha_{ij}^t \frac{\partial \Theta_{ij}^t}{\partial A_{ij}^t}$, we must find many coincidences between the sign of the multipliers and that of the derivatives. Multipliers are mostly negative along the process because $\alpha_{ij}^t = \alpha_{ij}^{t-1} + \mu(\Theta_{ij}^{t-1} - \|\mathbf{x}_i - \mathbf{x}_j\|^2) = \alpha_{ij}^{t-1} + \mu(\Theta_{ij}^{t-1} - CT(i,j))$ and $\Theta_{ij}^{t-1} - CT(i,j)$ are usually negative. Therefore, we must prove that the derivatives are mostly negative. In order to do

so, we approximate the derivatives at any time by $\Delta\Theta_{ij} = \Theta_{ij}^{A_{ij}+h} - \Theta_{ij}$, where $\Theta_{ij}^{A_{ij}+h}$ is the *perturbed* Θ_{ij} after replacing A_{ij} by $A_{ij} + h$ (the same for A_{ji}) and then computing the CT embedding. Let β be the inverse temperature corresponding to a given iteration, and let us approximate the current adjacency matrix as suggested above: $\mathbf{A} = \frac{1}{r}(\mathbf{1}\mathbf{1}^T - \mathbf{I})$, where $r = \exp(\beta)$. Let $\mathbf{E}$ be the so-called $N \times N$ perturbation matrix defined by

$$(40) \qquad E_{ab} = \begin{cases} h & \text{if } a = i \text{ and } b = j, \\ 0 & \text{otherwise,} \end{cases}$$

where $h > 0$. Then $\hat{\mathbf{A}} = \mathbf{A} + \mathbf{E}$ is the linearly perturbed adjacency matrix. It is straightforward to verify that $\mathcal{L}_A = \mathcal{L}_{K_N}$. Consequently, from the eigendecomposition $\mathcal{L}_A = \Phi_{\mathcal{L}_A}\Lambda_{\mathcal{L}_A}\Phi_{\mathcal{L}_A}^T$ we have

- Spectrum: $\lambda_{\mathcal{L}_A}^{(1)} = 0 < \lambda_{\mathcal{L}_A}^{(2)} = \cdots = \lambda_{\mathcal{L}_A}^{(N)} = \frac{N}{N-1}$ (see [12]).
- Eigenvectors: $\phi_{\mathcal{L}_A}^{(1)} = \alpha\mathbf{D}_A^{1/2}\mathbf{1}$, where $\alpha \in \mathbb{R}$ and $\mathbf{D}_A$ is the degree matrix of $\mathbf{A}$. Eigenvectors are orthonormal and also satisfy $\sum_{i=1}^N \phi_{\mathcal{L}_A}^{(z)}(i) = 0$ for $z \geq 2$.

In order to relate the spectrum and eigenvectors of $\mathcal{L}_A$ with those of $\mathcal{L}_{\hat{A}}$, our first intuition is to exploit *matrix perturbation theory* [47]. This theory relies on pessimistic bounds. For instance, the Bauer–Fike theorem [4] states that if $\mathcal{L}_{\hat{A}} = \Phi_{\mathcal{L}_{\hat{A}}}\Lambda_{\mathcal{L}_{\hat{A}}}\Phi_{\mathcal{L}_{\hat{A}}}^T$ and $\lambda_{\mathcal{L}_{\hat{A}}}$ is an eigenvalue of the perturbed Laplacian, we have

$$(41) \qquad \min_{\lambda_{\mathcal{L}_A}} \left|\lambda_{\mathcal{L}_{\hat{A}}} - \lambda_{\mathcal{L}_A}\right| \leq \left\|\Phi_{\mathcal{L}_A}\right\|_p \left\|\Phi_{\mathcal{L}_A}^{-1}\right\|_p \left\|\mathbf{E}\right\|_p = \kappa_p(\Phi_{\mathcal{L}_A})\left\|\mathbf{E}\right\|_p$$

for eigenvalues $\lambda_{\mathcal{L}_A}$ (not necessarily all), where p is the type of norm used (for example, $p = 1, p = 2, p = \infty$) and $\kappa_p(.)$ is the so-called condition number for such a norm. Assuming $p = 2$, we have that $\kappa_p(\Phi_A) = 1$ for $\Phi_A^{-1} = \Phi_A^T$ (matrix Φ_A is orthonormal). Therefore $\min_{\lambda_{\mathcal{L}_A}} \left|\lambda_{\mathcal{L}_{\hat{A}}} - \lambda_{\mathcal{L}_A}\right| \leq \|\mathbf{E}\|_2$. Since $\lambda_E^{(1)} = -h$, $\lambda_E^{(2)} = \cdots = \lambda_E^{(N-1)} = 0$, and $\lambda_E^{(N)} = h$ we have $\left|\lambda_{\mathcal{L}_{\hat{A}}} - \lambda_{\mathcal{L}_A}\right| \leq h$. In addition, the fact that h is small implies that any eigenvalue of $\mathcal{L}_{\hat{A}}$ is very similar to some eigenvalue of $\mathcal{L}_A$. However, the latter theorem does not provide a way of predicting which value of $\lambda_{\mathcal{L}_{\hat{A}}}$ is the most divergent. At this point we complement the matrix perturbation analysis with the spectral analysis of graph-cuts. It is well known that the Fiedler vector of the normalized Laplacian encodes the bipartition of the graph. When we change A_{ij} to $A_{ij} + h$, we induce the partition $\{i,j\} \bigcup V - \{i,j\}$ in the complete attributed graph. The existence of this partition implies a reduction of the spectral gap ($\lambda_{\mathcal{L}_{\hat{A}}}^{(1)} = 0$ for any Laplacian and also $\phi_{\mathcal{L}_{\hat{A}}}^{(1)} = \alpha'\mathbf{D}_{\hat{A}}^{1/2}\mathbf{1}$). Then we have $\lambda_{\mathcal{L}_A}^{(2)} > \lambda_{\mathcal{L}_{\hat{A}}}^{(2)} = \lambda_{\mathcal{L}_A}^{(2)} - \gamma > 0$. The larger h is, the smaller the gap (resp., the larger the γ) until a minimal nonzero gap is reached independently of h. In addition, $\phi_{\mathcal{L}_A}^{(2)}$ is perturbed in such a way that the following hold:

- Different value: $\phi_{\mathcal{L}_{\hat{A}}}^{(2)}(i) = \phi_{\mathcal{L}_{\hat{A}}}^{(2)}(j) \neq \phi_{\mathcal{L}_{\hat{A}}}^{(2)}(k)$ $k \notin \{i,j\}$.
- Different sign: $sign(\phi_{\mathcal{L}_{\hat{A}}}^{(2)}(i))sign(\phi_{\mathcal{L}_{\hat{A}}}^{(2)}(j)) = +1$ and $sign(\phi_{\mathcal{L}_{\hat{A}}}^{(2)}(i))sign(\phi_{\mathcal{L}_{\hat{A}}}^{(2)}k)) = -1$ $k \notin \{i,j\}$.
- Same value and sign: Both $\phi_{\mathcal{L}_{\hat{A}}}^{(2)}(k) = \phi_{\mathcal{L}_{\hat{A}}}^{(2)}(l)$ and $sign(\phi_{\mathcal{L}_{\hat{A}}}^{(2)}(k))sign(\phi_{\mathcal{L}_{\hat{A}}}^{(2)}(l)) = +1$ for $k, l \in V - \{i,j\}$.
- Nonzero sum: $\sum_{i=1}^N \phi_{\mathcal{L}_{\hat{A}}}^{(2)}(i) \neq 0$.

As far as the rest of the spectrum of $\mathcal{L}_{\hat{A}}$ is concerned, we have that $\lambda_{\mathcal{L}_{\hat{A}}}^{(3)} = \cdots = \lambda_{\mathcal{L}_{\hat{A}}}^{(N-1)} = \frac{N}{N-1}$, as is the case in $\mathcal{L}_A$. It is straightforward to check that $\frac{N}{N-1}$ is a root of $\left| \mathcal{L}_A - \lambda_{\mathcal{L}_{\hat{A}}} \mathbf{I} \right| = 0$ with multiplicity $N - 2$. Consequently, $\lambda_{\mathcal{L}_{\hat{A}}}^{(N)} = \lambda_{\mathcal{L}_A}^{(N)} + \gamma$, so that $\text{tr}(\mathcal{L}_{\hat{A}}) = N$. This means that the spectral perturbation induced by h is confined to both the Fiedler eigenvalue and the highest one. This result is consistent with the Bauer–Fike theorem in the sense that there is a correspondence of eigenvalues from the third to the $(N - 1)$th. In this particular case we have $\min_{\lambda_{\mathcal{L}_A}} \left| \lambda_{\mathcal{L}_{\hat{A}}} - \lambda_{\mathcal{L}_A} \right| = 0$. However, for $\lambda_{\mathcal{L}_A}^{(2)}$ and $\lambda_{\mathcal{L}_A}^{(N)}$ we cannot find an eigenvalue of $\mathcal{L}_{\hat{A}}$ satisfying the Bauer–Fike bound unless $h \to 0$.

Given the latter eigenvalues $\lambda_{\mathcal{L}_{\hat{A}}}^{(3)} = \cdots = \lambda_{\mathcal{L}_{\hat{A}}}^{(N-1)} = \frac{N}{N-1}$ and the orthonormality requirements, we have that the corresponding eigenvectors are of the form $\phi_{\mathcal{L}_{\hat{A}}}^{(z)}(k) = 0$ for $k = i$ and $k = j$. Considering that a given i and j induce a partition $i, jUV - i, j$ as stated above, we have

$$(42) \qquad \left| \phi_{\mathcal{L}_{\hat{A}}}^{(N)}(k) \right| = \begin{cases} 0 & \text{if } k \neq i \text{ and } k \neq j, \\ \frac{\sqrt{2}}{2} & \text{otherwise}. \end{cases}$$

More precisely $sign(\phi_{\mathcal{L}_{\hat{A}}}^{(N)}(i)) sign(\phi_{\mathcal{L}_{\hat{A}}}^{(N)}(j)) = -1$, and this form is consistent with the similar form of the eigenvectors of $\mathbf{E}$.

Given the spectral analysis described above, now we exploit the spectral definition of CT in order to prove that $\Delta \Theta_{ij} = \Theta_{ij}^{A_{ij}+h} - \Theta_{ij}$ is negative provided that $\mathbf{A}$ encodes a uniformly weighted complete graph. Let us rename $\Theta_{ij}^{A_{ij}+h}$ and Θ_{ij} as follows: $CT_{\hat{A}}(i,j) = \Theta_{ij}^{A_{ij}+h}$ and $CT_A(i,j) = \Theta_{ij}$. Our purpose is to prove that $CT_{\hat{A}}(i,j) - CT_A(i,j) < 0$. We have

$$(43) \qquad CT_{\hat{A}}(i,j) = vol_{\hat{A}} \sum_{z=2}^{N} \frac{1}{\lambda_{\mathcal{L}_{\hat{A}}}^{(z)}} \left(\frac{\phi_{\mathcal{L}_{\hat{A}}}^{(z)}(i)}{\sqrt{\mathbf{D}_{\hat{A}}(i,i)}} - \frac{\phi_{\mathcal{L}_{\hat{A}}}^{(z)}(j)}{\sqrt{\mathbf{D}_{\hat{A}}(j,j)}} \right)^2 .$$

However, due to the facts that $\phi_{\mathcal{L}_{\hat{A}}}^{(2)}(i) = \phi_{\mathcal{L}_{\hat{A}}}^{(2)}(j)$ (Fiedler vector components) and $\phi_{\mathcal{L}_{\hat{A}}}^{(z)}(i) = \phi_{\mathcal{L}_{\hat{A}}}^{(z)}(j) = 0$ for $z = 3 \ldots N - 1$ (largest eigenvalue components), (43) is reduced to only one summand,

$$
\begin{aligned}
CT_{\hat{A}}(i,j) &= \frac{vol_{\hat{A}}}{\lambda_{\mathcal{L}_{\hat{A}}}^{(N)}} \left(\frac{\phi_{\mathcal{L}_{\hat{A}}}^{(N)}(i)}{\sqrt{\mathbf{D}_{\hat{A}}(i,i)}} - \frac{\phi_{\mathcal{L}_{\hat{A}}}^{(N)}(j)}{\sqrt{\mathbf{D}_{\hat{A}}(j,j)}} \right)^2 \\
&= \frac{vol_{\hat{A}}}{\lambda_{\mathcal{L}_{\hat{A}}}^{(N)}} \left(2 \frac{\phi_{\mathcal{L}_{\hat{A}}}^{(N)}(i)}{\sqrt{\mathbf{D}_{\hat{A}}(i,i)}} \right)^2 \\
(44) \qquad &= \frac{vol_{\hat{A}}}{\lambda_{\mathcal{L}_{\hat{A}}}^{(N)}} \left(\frac{2}{\mathbf{D}_{\hat{A}}(i,i)} \right) .
\end{aligned}
$$

Considering that $\mathbf{D}_{\hat{A}}(i,i) = \frac{1}{r}(N-1) + h$, we have that $vol_{\hat{A}} = \frac{N(N-1)}{r} + 2h$. Since $\lambda_{\mathcal{L}_{\hat{A}}}^{(N)} = \lambda_{\mathcal{L}_A}^{(N)} + \gamma = \frac{N}{N-1} + \gamma$, we have

$$
\begin{aligned}
CT_{\hat{A}}(i,j) &= \frac{\frac{N(N-1)}{r} + 2h}{\frac{N}{N-1} + \gamma} \left(\frac{2}{\frac{1}{r}(N-1) + h} \right) \\
&= \left(\frac{1}{r} \right) \frac{N(N-1) + 2hr}{\frac{N}{N-1} + \gamma} \left(\frac{2r}{(N-1) + hr} \right) \\
&= \frac{N(N-1) + 2hr}{\frac{N}{N-1} + \gamma} \left(\frac{2}{(N-1) + hr} \right) \\
&< \frac{N(N-1) + 2hr}{(N-1) + hr} \left(\frac{2}{\frac{N}{N-1} + \gamma} \right) \\
&< N \left(\frac{2}{\frac{N}{N-1} + \gamma} \right) < N \left(\frac{2}{\frac{N}{N-1}} \right) \\
&= N \left(\frac{2(N-1)}{N} \right) = 2(N-1) \ .
\end{aligned}
$$
(45)

Therefore $CT_{\hat{A}}(i,j) < CT_A(i,j)$, that is, $\Delta\Theta_{ij} = \Theta_{ij}^{A_{ij}+h} - \Theta_{ij}$ is negative provided that both $\mathbf{A}$ encodes a uniformly weighted complete graph and $h > 0$, $\gamma > 0$. However, since $A_{ij}^t = \exp\beta_t(-1 - \epsilon^t)$ with $\epsilon \ll 1$, it is possible to find some positive increments, but most of them are negative. Therefore $\sum_{ij:j>i} -(\alpha_{ij} \frac{\partial\Theta_{ij}}{\partial A_{ij}})\Delta A_{ij} > 0$ (first term of (40)) for most of the multipliers α_{ij} are negative.

Finally, since we encode emerging edges of the true graph with the less negative (ideally zero) multipliers $\alpha_{ij}^{t+1} = \alpha_{ij}^t + \mu(\Theta_{ij}^t - ||\mathbf{x}_i - \mathbf{x}_j||^2)$ this means that we must evolve from an initial situation (low values of $\beta > \beta_0$) where almost all multipliers are negative towards a state where some of them are zero or positive. Therefore, negative multipliers always dominate nonnegative ones. This is due to the fact that we seek to meet $\frac{vol(G)}{2} = |E|$ constraints with the highest degree of satisfaction and typically $|E| \ll \frac{N(N-1)}{2}$ in computer vision. Consequently $\sum_{ij:j>i} \alpha_{ij}^t(\Theta_{ij} - ||\mathbf{x}_i - \mathbf{x}_j||^2) > 0 \ \forall t > 0$.

For proving that $\sum_{ij:j>i} \alpha_{ij}^{t+1}(\Theta_{ij}^{t+1} - d_{ij}^2) - \sum_{ij:j>i} \alpha_{ij}^t(\Theta_{ij}^t - d_{ij}^2) > 0$ (second term of (40)), we exploit the facts $\Theta_{ij}^{t+1} \approx 2(N-1)$ and $\Theta_{ij}^t \approx 2(N-1)$. Therefore, $\Theta_{ij}^{t+1} \approx \Theta_{ij}^t$, and the latter term is reduced to $\sum_{ij:j>i}(\alpha_{ij}^{t+1} - \alpha_{ij}^t)(\Theta_{ij}^{t+1} - d_{ij}^2)$. Since $\alpha_{ij}^{t+1} - \alpha_{ij}^t = \mu(\Theta_{ij}^t - d_{ij}^2)$ are usually negative, the complete term is positive.

Therefore we have proved that (40) satisfies $\Delta E(A, \{\alpha_{ij}\}) > 0$ for $t > 0$, and it defines a Lyapunov function. The proposed DA algorithm converges. ∎

REFERENCES

[1] S. Agarwal, J. Lim, L. Zelnik-Manor, P. Perona, D. Kriegman, and S. Belongie, *Beyond pairwise clustering*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '05), vol. 2, San Diego, CA, 2005, pp. 838–845.

[2] X. BAI, E. HANCOCK, AND R. WILSON, *Geometric characterization and clustering of graphs using heat kernel embeddings*, Image Vision Comput., 28 (2010), pp. 1003–1021.

[3] H. G. BARROW AND R. J. POPPLESTONE, *Relational descriptions in picture processing*, in Machine Intelligence, vol. 6, B. Meltzer and D. Michie, eds., Edinburgh University Press, Edinburgh, UK, 1971, pp. 377–396.

[4] F. BAUER AND C. FIKE, *Norms and exclusion theorems*, Numer. Math., 2 (1960), pp. 137–141.

[5] M. BELKIN AND P. NIYOGI, *Laplacian eigenmaps for dimensionality reduction and data representation*, Neural Comput., 15 (2003), pp. 1373–1396.

[6] K. L. BOYER AND A. C. KAK, *Structural stereopsis for 3-d vision*, IEEE Trans. Pattern Anal. Mach. Intell., 10 (1988), pp. 144–166.

[7] H. BUNKE, *A relation between graph edit distance and maximum common subgraph*, Pattern Recognition Lett., 18 (1997), pp. 689–694.

[8] T. CAELLI AND S. KOSINOV, *An eigenspace projection clustering method for inexact graph matching*, IEEE Trans. Pattern Anal. Mach. Intell., 26 (2004), pp. 515–519.

[9] M. CHERTOK AND Y. KELLER, *Efficient high order matching*, IEEE Trans. Pattern Anal. Mach. Intell., 32 (2010), pp. 2205–2215.

[10] M. CHO, J. LEE, AND K. LEE, *Reweighted random walks for graph matching*, in Proceedings of the European Conference on Computer Vision (ECCV 2010), Lecture Notes in Comput. Sci. 6315, Springer, Berlin, 2010, pp. 492–505.

[11] W. CHRISTMAS, J. KITTLER, AND M. PETROU, *Structural matching in computer vision using probabilistic relaxation*, IEEE Trans. Pattern Anal. Mach. Intell, 17 (1995), pp. 749–764.

[12] F. CHUNG, *Spectral Graph Theory*, CBMS Regional Conf. Ser. in Math. 92, American Mathematical Society, Providence, RI, 1997.

[13] T. COUR, P. SRINIVASAN, AND J. SHI, *Balanced graph matching*, in Proceedings of Advances in Neural Information Processing Systems 19 (NIPS 2006), MIT Press, Cambridge, MA, 2007, pp. 313–320.

[14] O. DUCHENNE, F. BACH, I.-S. KWEON, AND J. PONCE, *A tensor-based algorithm for high-order graph matching*, n Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009), Miami, FL, 2009, pp. 1980–1987.

[15] O. DUCHENNE, A. JOULIN, AND J. PONCE, *A graph-matching kernel for object categorization*, in Proceedings of the IEEE International Conference on Computer Vision (ICCV 2011), Barcelona, Spain, pp. 1792–1799.

[16] F. ESCOLANO, E. HANCOCK, AND M. LOZANO, *Graph matching through entropic manifold alignment*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2011), Colorado Springs, CO, 2011, pp. 2417–2424.

[17] F. ESCOLANO AND E. R. HANCOCK, *From points to nodes: Inverse graph embedding through a Lagrangian formulation*, in Proceedings of Computer Analysis of Images and Patterns (CAIP), Part I, Lecture Notes in Comput. Sci. 6854, Springer, Berlin, 2011, pp. 194–201.

[18] F. ESCOLANO, M. LOZANO, B. BONEV, AND P. SUAU, *Bypass information-theoretic shape similarity from non-rigid points-based alignment*, in Proceedings of Computer Vision and Pattern Recognition Workshops (CVPRW), San Francisco, CA, 2010, pp. 37–44.

[19] G. FERRATE, T. PAVLIDIS, A. SANFELIU, AND H. BUNKE, EDS., *Syntactic and Structural Pattern Recognition*, NATO ASI Ser. 45, Springer-Verlag, Berlin, 1988.

[20] A. FINCH, R. WILSON, AND E. HANCOCK, *An energy function and continuous edit process for graph matching*, Neural Comput., 10 (1998), pp. 1873–1894.

[21] M. FISCHLER AND R. ELSCHLAGER, *The representation and matching of pictorial structures*, IEEE Trans. Comput., C-22 (1973), pp. 67–92.

[22] S. GOLD AND A. RANGARAJAN , *A graduated assignment algorithm for graph matching*, IEEE Trans. Pattern Anal. Mach. Intell., 18 (1996), pp. 377–388.

[23] L. HAN, R. C. WILSON, AND E. R. HANCOCK, *Generative graph prototypes from information theory*, IEEE Trans. Pattern Anal. Mach. Intell., 37 (2015), pp. 2013–2027.

[24] J. HOBBY AND G. TUCCI, *Traffic Analysis in Random Delaunay Tessellations and Other Graphs*, preprint, https://arxiv.org/abs/1203.4863, 2012.

[25] B. JIANG AND B. VEMURI, *A robust algorithm for point set registration using mixture of Gaussians*, in Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV 2005), vol. 2, Beijing, China, 2005, pp. 1246–1251.

[26] J. KELSEY, K. MCKAY, AND M. S. TURAN, *Predictive models for min-entropy estimation*, in Proceedings of Cryptographic Hardware and Embedded Systems (CHES 2015), T. Güneysu and H. Handschuh eds., Lecture Notes in Comput. Sci. 9293, Springer, Berlin, 2015, pp. 373–392.

[27] A. KRASKOV, H. STÖGBAUER, AND P. GRASSBERGER, *Estimating mutual information*, Phys. Rev. E, 69 (2004), 066138.

[28] N. LEONENKO, L. PRONZATO, AND V. SAVANI, *A class of Renyi information estimators for multidimensional densities*, Ann. Statist., 36 (2008), pp. 2153–2182.

[29] M. LEORDEANU AND M. HEBERT, *A spectral technique for correspondence problems using pairwise constraints*, in Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV 2005), vol. 2, Beijing, China, 2005, pp. 1482–1489.

[30] L. LOVÁSZ, *Random walks on graphs: A survey*, in Combinatorics, Paul Erdös is Eighty, Vol. 2 (Keszthely, 1993), Bolyai Soc. Math. Stud., 2, János Bolyai Math. Soc., Budapest, 1996, pp. 353–397.

[31] M. LOZANO AND F. ESCOLANO, *Graph matching and clustering using kernel attributes*, Neurocomput., 113 (2013), pp. 177–194.

[32] B. LUO, R. WILSON, AND E. HANCOCK, *Spectral embedding of graphs*, Pattern Recognition, 36 (2003), pp. 2213–2230.

[33] A. MARTINS, N. SMITH, E. XING, P. AGUIAR, AND M. FIGUEIREDO, *Nonextensive information theoretic kernels on measures*, J. Mach. Learn. Res., 10 (2009), pp. 935–975.

[34] A. MYRONENKO AND X. B. SONG, *Point-set registration: Coherent point drift*, IEEE Trans. Pattern Anal. Mach. Intell., 32 (2010), pp. 2262–2275.

[35] B. NADLER, S. LAFON, R. COIFMAN, AND I. KEVREKIDIS, *Diffusion maps, spectral clustering and eigenfunctions of Fokker-Planck operators*, in Proceedings of Advances in Neural Information Processing Systems 18 (NIPS 2005), MIT Press, Cambridge, MA, 2005, pp. 955–962.

[36] H. NEEMUCHWALA, A. HERO, AND P. CARSON, *Image matching using alpha-entropy measures and entropic graphs*, Signal Process., 85 (2005), pp. 277–296.

[37] E. PARZEN, *On estimation of a probability density function and mode*, Ann. Math. Statist., 33 (1962), p. 1065–1076.

[38] S. PINCUS, *Approximate entropy as a measure of system complexity*, Proc. Nat. Acad. Sci. U.S.A., 88 (1991), pp. 2297–2301.

[39] H. QIU AND E. HANCOCK, *Clustering and embedding using commute times*, IEEE Trans. Pattern Anal. Mach. Intell., 29 (2007), pp. 1873–1890.

[40] A. RAJWADE, A. BANERJEE, AND A. RANGARAJAN, *Probability density estimation using isocontours and isosurfaces: Applications to information-theoretic image registration*, IEEE Trans. Pattern Anal. Mach. Intell., 31 (2009), pp. 475–491.

[41] A. RANGARAJAN, S. GOLD, AND E. MJOLSNESS, *A novel optimizing network architecture with applications*, Neural Computat., 8 (1996), pp. 1041–1060.

[42] A. RANGARAJAN, A. YUILLE, AND E. MJOLSNESS, *Convergence properties of the softassign quadratic assignment algorithm*, Neural Comput., 11 (1999), pp. 1455–1474.

[43] A. ROBLES-KELLY AND E. HANCOCK, *Graph edit distance from spectral seriation*, IEEE Trans. Pattern Anal. Mach. Intell., 27 (2005), pp. 365–378.

[44] A. ROBLES-KELLY AND E. HANCOCK, *A Riemannian approach to graph embedding*, Pattern Recognition, 40 (2007), pp. 1042–1056.

[45] A. SANFELIU AND K. FU, *A distance measure between attributed relational graphs for pattern recognition*, IEEE Trans. Systems Man Cybernet., 13 (1973), pp. 353–363.

[46] L. G. SHAPIRO AND R. M. HARALICK, *Structural description and inexact matching*, IEEE Trans. Pattern Anal. Mach. Intell., 3 (1981), pp. 504–519.

[47] G. STEWARD AND J. G. SUN, *Matrix Perturbation Theory*, Academic Press, New York, 1990.

[48] J. TANG, B. JIANG, A. ZHENG, AND B. LUO, *Graph matching based on spectral embedding with missing value*, Pattern Recognition, 45 (2012), pp. 3768–3779.

[49] J. B. TENENBAUM, V. DE SILVA, AND J. C. LANGFORD, *A global geometric framework for nonlinear dimensionality reduction*, Science, 290 (2000), pp. 2319–2323.

[50] A. TORSELLO AND E. HANCOCK, *Learning shape-classes using a mixture of tree-unions*, IEEE Trans. Pattern Anal. Mach. Intell., 28 (2006), pp. 954–967.

[51] S. UMEYAMA, *An eigendecomposition approach to weighted graph matching problems*, IEEE Trans. Pattern Anal. Mach. Intell., 10 (1988), pp. 695–703, https://doi.org/10.1109/34.6778.

[52] U. VON LUXBURG, A. RADL, AND M. HEIN, *Getting lost in space: Large sample analysis of the commute distance*, in Proceedings of Advances in Neural Information Processing Systems 23 (NIPS 2010), MIT Press, Cambridge, MA, 2010, pp. 2622–2630.

[53] U. VON LUXBURG, A. RADL, AND M. HEIN, *Hitting and commute times in large random neighborhood graphs*, J. Mach. Learn. Res., 15 (2014), pp. 1751–1798.

[54] R. WILSON AND E. HANCOCK, *Structural matching by discrete relaxation*, IEEE Trans. Pattern Anal. Mach. Intell., 19 (1997), pp. 634–648.

[55] M. ZASLAVSKIY, F. BACH, AND J.-P. VERT, *A path following algorithm for the graph matching problem*, IEEE Trans. Pattern Anal. Mach. Intell., 31 (2009), pp. 2227–2242.

[56] R. ZASS AND A. SHASHUA, *Probabilistic graph and hypergraph matching*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2008), Anchorage, AK, 2008, pp. 1–8.

[57] F. ZHOU AND F. D. LA TORRE, *Factorized graph matching*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2012), Providence, RI, 2012, pp. 127–134.

[58] F. ZHOU AND F. D. LA TORRE, *Deformable graph matching*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2013), Portland, OR, 2013, pp. 2922–2929.