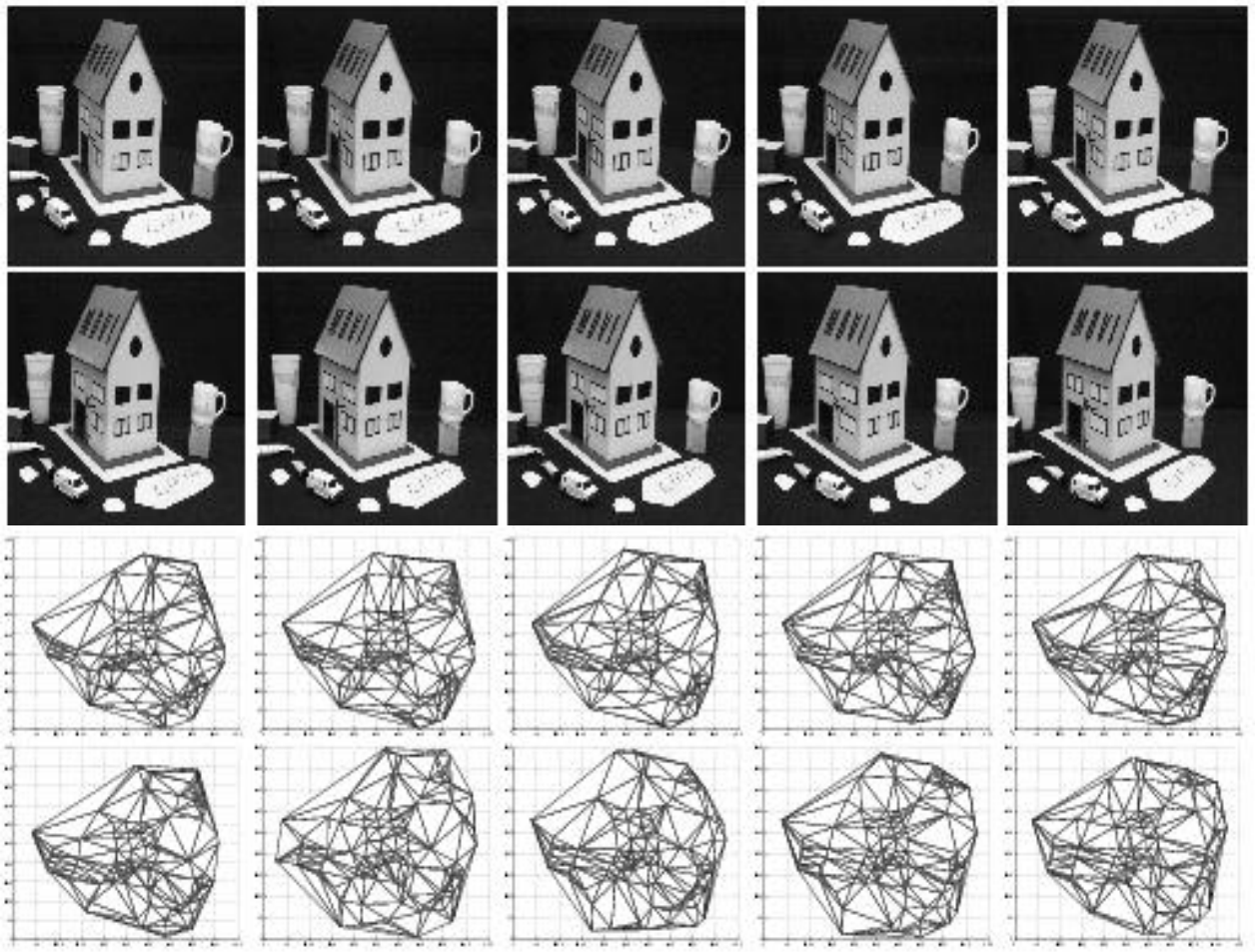
# A Graph Kernel from the Depth-based Representations

Lu Bai, Edwin R. Hancock
Department of Computer Science,
University of York, UK

# Outline

- Background and Motivation
- Approach
  - *h*-layer depth-based representation
  - Depth-based alignment in graphs
  - Depth-based matching kernel
  - Relationship between depth-based matching kernel and all subgraph kernel
- Experiments
- Conclusions

# Structural Variations



# Protein-Protein Interaction Networks

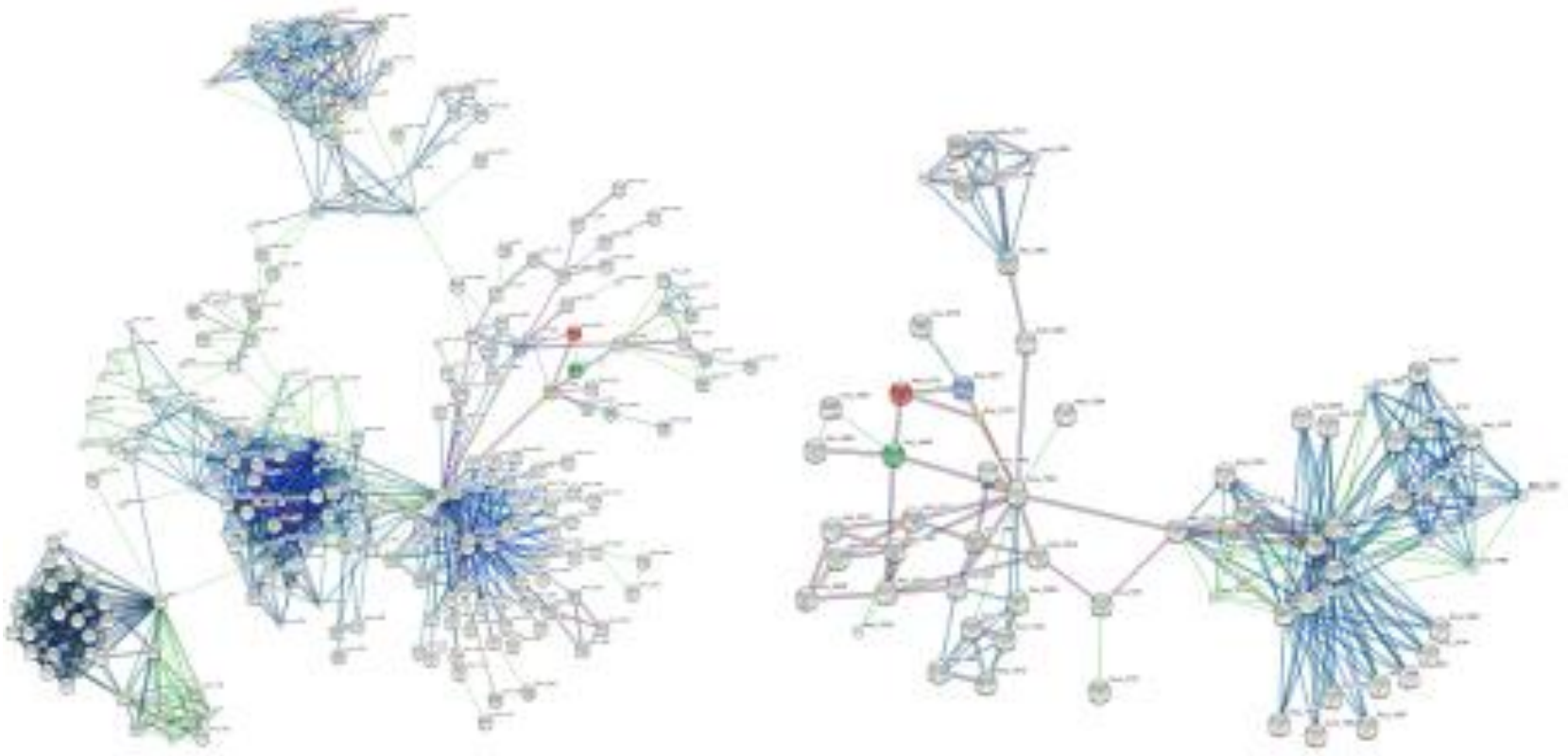


Figure: STRING Protein-Protein Interaction Networks

# Manipulating graphs

- Is structure similar (graph isomorphism, inexact match)?
- Is complexity similar (are graphs from same class but different in detail)?
- Is complexity (type of structure) uniform?

# Goals

- Can we capture determine the similarity of structure using measures that capture their intrinsic complexity.
- Can graph entropies be used for this purpose.
- If they can then they lead naturally to information theoretic kernels and description length for learning over graph data.

# Background and Motivation

- Graph based representations are important and widely used in pattern recognition and computer vision.
- One challenge is to compactly represent a graph in a way that best preserves its structure.
  - 1) embedding graphs into a vector space,
  - 2) defining a graph kernel for graphs.

# Background and Motivation (Graph Embedding)

- Graph based pattern recognition is potentially more powerful and expressie than vector based pattern recognition.
  - Drawback of graph based approach: lack of standard machine learning algorithms *[Bunke et al., 2010].*
  - Can be overcome by embedding into a vector space *[Hancock et al., 2012].*
- Graph embedding methods
  - 1) Dissimilarity embedding *[Bunke et al., 2011, PR].*
  - 2) Ihara zeta function of graphs *[Ren et al., 2011, TNN].*
  - 3) Graph pattern vectors from the algebraic graph theory *[Wilson et al., 2009, PAMI].*
  - 4) Skew spectrum and graphlet spectrum vectors of graphs *[Kondor and Borgwardt, 2008, ICML] and [Kondor and Borgwardt, 2009, ICML]*
  - 5) Depth-based complexity traces of graphs *[Bai and Hancock., 2014, Pattern Recognition].*
- Drawbacks: **Information loss**

# Background and Motivation (Graph Kernels)

- Graph Kernels: **Why use graph kernel**?
  - ***Kernel offers an elegant solution to the cost of computation on high dimensional feature space*** *[K. Riesen, and H. Bunke, 2009, Pattern Recognition]*
- Existing Graph Kernels from the R-convolution *[Haussler, 1999]*
  - Random walk based kernels
    - Product graph kernels *[Gartner et al., 2003, ICML]*
    - Marginalized kernels on graphs *[Kashima et al., 2003, ICML]*
  - Path based kernels
    - Shortest path kernel *[K. m. Borgwardt, 2005, ICDM]*
  - Restricted subgraph or subtree based kernels
    - Weisfeiler-Lehman subtree kernel *[Shevashidze et al., 2010, JMLR]*
    - Graphlet count kernel *[Shevashidze et al., 2009, ICML]*
    - Neighborhood subgraph kernel *[Costa and Grave, 2010, ICML]*

# Background and Motivation (Graph Kernels)

- Drawbacks of the existing R-convolution kernels
  - **1)** Definitions of R-convolution kernels: for a pair of graph *Gp* and *Gq*, assume $\{\mathcal{S}_{1;1},\ldots,\mathcal{S}_{1;n_1},\ldots,\mathcal{S}_{1;N_1}\}$ and $\{\mathcal{S}_{2;1},\ldots,\mathcal{S}_{2;n_2},\ldots,\mathcal{S}_{q;N_2}\}$ are their substructure sets respectively, then the R-convolution kernel is

$$k_R(G_1,G_2)=\sum_{n_1=1}^{N_1}\sum_{y=1}^{N_2}\delta(\mathcal{S}_{1;n_1},\mathcal{S}_{2;n_2}),$$

    where

$$\delta(\mathcal{S}_{1;n_1},\mathcal{S}_{2;n_2})=\begin{cases}1 \text{ if } \mathcal{S}_{1;n_1}\simeq\mathcal{S}_{2;n_2},\\ 0 \text{ otherwise.}\end{cases}$$

  - **2)** Graph kernels with limited sized substructures can only reflect restricted topological characteristics of a graph.
  - **3)** Do not indicate the relative locations between substructures within a graph - cannot establish precise structural correspondence between pairs of vertices.
- Developing efficient and effective graph kernels still remains a challenge.

# Idea

- Decompose graph into layered subgraphs from centroid.
- Use Shannon entropy to compare subgraphs.
- Find consistent arrangement of subgraph correspondences using Hungarian algorithm.
- Construct kernel over subgraphs.

# A *h*-layer Depth-based Representation

- A Shannon Entropy of a graph
  - Consider a graph $G(V,E)$, the adjacency matrix has elements

$$A(i,j)=\begin{cases}1 \text{ if}(i,j)\in E;\\ 0 \text{ otherwise.}\end{cases}$$

  - The vertex degree matrix of $G(V,E)$ is given by

$$D(v_i,v_i)=d(i)=\sum_{j\in V}A(i,j).$$

  - Shannon entropy: The probability of a random walk on $G(V,E)$ visiting vertex $v_i$ is $P_G(i)=d(i)/\sum_{j\in V}d(j)$. The Shannon entropy of $G(V,E)$ is

$$H_S(G)=-\sum_{i=1}^{|V|}P_G(i)\log P_G(i)$$

# The Depth-Based Representation of A Graph

- Subgraphs from the Centroid Vertex
  - For graph *G(V,E)*, construct shortest path matrix matrix *SG* whose element *SG(i, j)* are the shortest path lengths between vertices *vi* and *vj* . Average-shortest-path vector *SV* for *G(V,E)* is a vector with element $S_V(i) = \sum_{j=1}^{|V|} S_G(i,j)/|V|$ from vertex *vi* to the remaining vertices.
  - Centroid vertex for *G(V,E)* as

$$\hat{v}_i = \arg\min_i \sum_{j=1}^{|V|} [S_G(i,j) - S_V(i)]^2$$

  - The *K*-layer centroid expansion subgraph

$$\mathcal{G}_K(\mathcal{V}_K;\mathcal{E}_K) \quad \begin{cases} \mathcal{V}_K = \{u \in N_{\hat{v}_C}^K\}; \\ \mathcal{E}_K = \{(u,v) \subset N_{\hat{v}_C}^K \mid (u,v) \in E\}. \end{cases}$$

where $N_{\hat{v}_C}^K = \{u \in V \mid S_G(\hat{v}_C, u) \leq K\}$

# Depth-Based Representation

- For a graph *G*, we obtain a family of centroid expansion subgraphs $\{\mathcal{G}_1,\cdots,\mathcal{G}_K,\cdots,\mathcal{G}_L\}$ , the depth-based representation of *G* is defined as

$$D(G)=\{H(\mathcal{G}_1),\cdots,H(\mathcal{G}_K),\cdots,H(\mathcal{G}_L)\}$$

where $H(\cdot)$ is either the Shannon entropy or the von Neumann entropy.

*Measures complexity via variation of entropy with depth*

# The Depth-Based Representation

- An example of the depth-based representation for a graph from the centroid vertex

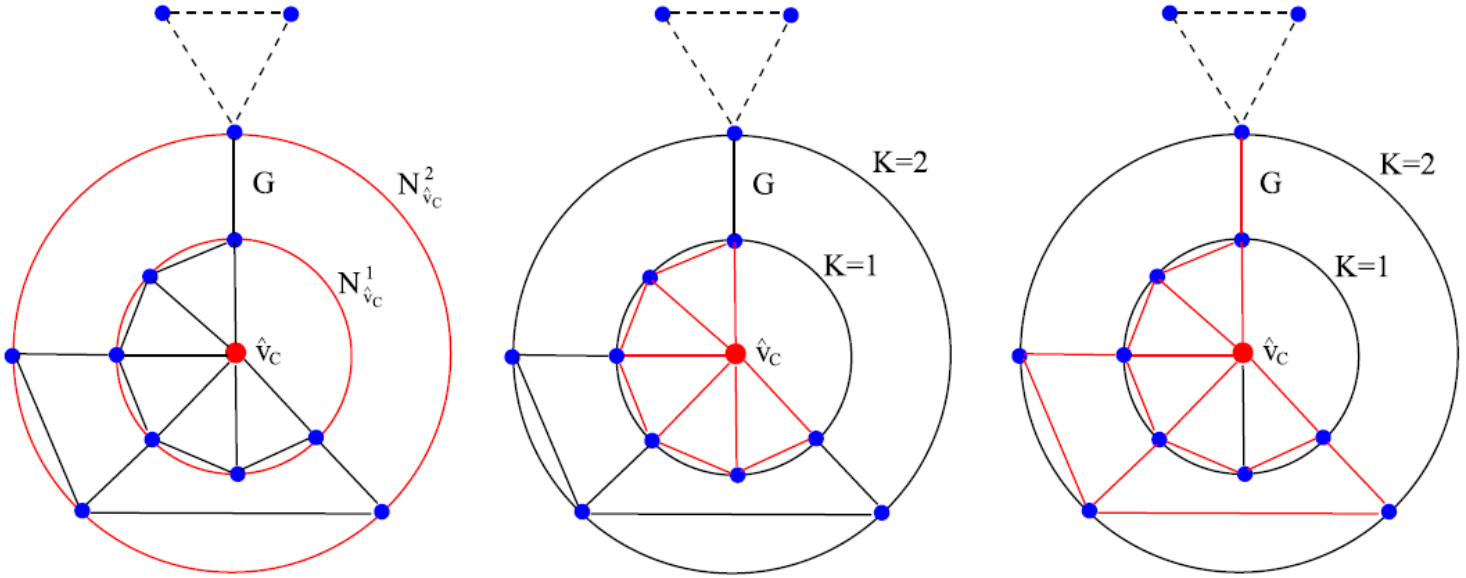



Figure 1: The left-most figure shows the determination of $K$-layer centroid expansion subgraphs for a graph $G(V, E)$ which hold $|N^1_{\hat{v}_C}| = 7$ and $|N^2_{\hat{v}_C}| = 11$ vertices. While the middle and the right-most figure show the corresponding 1-layer and 2-layer subgraphs regarding the centroid vertex $\hat{v}_C$, and are depicted by red-colored edges.

# Exploiting Depth Arrangement of Subgraphs

- Depth-based graph matching
  - Use Euclidean distance between the depth-based entropy representations $D^h_{G_p}(v_i)$ *and* $D^h_{G_q}(u_j)$ *as* distance measure for the pair of vertices $v_i$ *and* $u_j$ in graphs $G_p(V_p,E_p)$ *and* $G_q(V_q,E_q)$, respectively. The affinity matrix element R(i,j) is

$$R(i,j) = \sqrt{[D^h_{G_p}(v_i) - D^h_{G_q}(u_j)]^T [D^h_{G_p}(v_i) - D^h_{G_q}(u_j)]}.$$

- If R(i,j) is the smallest element both in row *i* and in column *j*, there should be a one-to-one correspondence between the vertex $v_i$ of $G_p(V_p,E_p)$ and the vertex $u_j$ of $G_q(V_q,E_q)$.

$$C(i,j) = \begin{cases} 1 & \text{if } R(i,j) \text{ is the smallest element} \\ & \text{both in row } i \text{ and in column } j; \\ 0 & \text{otherwise.} \end{cases}$$

  - Hungarian algorithm on the *C* matrix: assign each vertex a matched vertex at most.

## Properties of the Kernel

- The depth-based (DB) matching kernel for graphs

$$k_{DB}^{(h)}(G_p, G_q) = \sum_{i=1}^{|V_p|} \sum_{j=1}^{|V_q|} C(i,j).$$

- The DB kernel is positive definite.

$$k(G_p, G_q) = k_{DB}^{(h)}(G_p, G_q) = \sum_{v_i \in V_p} \sum_{u_j \in V_q} \delta(v_i, u_j).$$

where

$$\delta(v_i, u_j) = \begin{cases} 1 & \text{if } C(i,j) = 1; \\ 0 & \text{otherwise.} \end{cases}$$

# Time Complexity

- Subgraph kernel graphs for graphs with *n* vertices and *m* edges, has time complexity *O(n^2L + mn)*, where *L* is the size of the largest layer of the expansion subgraph.

  **Depth–based representation** is *O(n^2L+mn)*.
  **Hungarian method (Munkres)** *O(n^3)*.

# Relationship with All-Subgraphs Kernel

- Relationship between the DB kernel and the all subgraph kernel.
  - The classical all subgraph kernel

$$k_{sub}(G_p, G_q) = \sum_{S_p \sqsubseteq G_p} \sum_{S_q \sqsubseteq G_q} k_{iso}(S_p, S_q),$$

where

$$k_{iso}(S_p, S_q) = \begin{cases} 1 & \text{if } S_p \simeq S_q, \\ 0 & \text{otherwise.} \end{cases}$$

  - For a pair of graphs $G_p$ and $G_q$, the $h$-layer depth-based representations around a vertex $v$ of $G_p$ and a vertex $u$ of $G_q$ are $D^h_{G_p}(v) = \{\mathcal{G}^1_{p;v}, \cdots, \mathcal{G}^K_{p;v}, \cdots, \mathcal{G}^h_{p;v}\}$ and $D^h_{G_q}(u) = \{\mathcal{G}^1_{q;u}, \cdots, \mathcal{G}^K_{q;u}, \cdots, \mathcal{G}^h_{q;u}\}$. If $v$ and $u$ are matched, we approximately have that $\mathcal{G}^h_{p;v} \simeq \mathcal{G}^h_{q;u}$, thus the DB kernel can be re-written as

$$k^h_{DB}(G_p, G_q) = \sum_{S_p \sqsubseteq G_p} \sum_{S_q \sqsubseteq G_q} k_{iso}(S_p, S_q),$$

where

$$k_{iso}(S_p, S_q) = \begin{cases} 1 & \text{if } S_p = \mathcal{G}^h_{p;v} \textit{ and } S_q = \mathcal{G}^h_{q;u}, \\ & \textit{and } v \textit{ and } u \textit{ are matched}, \\ 0 & \text{otherwise.} \end{cases}$$

# Advantages of Depth-based Kernel

- The relationship indicates the following difference between the depth-based
  - a) For the depth-based graph kernel, only the subgraphs around a pair of matched vertices having a maximum topology distance K are evaluated with respect to isomorphism. While for the all subgraph kernel, ALL PAIRS of subgraphs are evaluated for identifying the isomorphism.
  - b) The depth-based graph kernel overcomes the NP-hard problems of measuring all possible pairs of subgraphs arising in the all subgraph kernel.
  - c) For the depth-based graph kernel, isomorphic subgraphs are identified by a pair of matched vertices. Hence depth arrangement determines correspondence between the isomorphic subgraphs with respect to their global location. This is not the case for the all subgraph kernel.

# Depth-Based Matching Kernel for Unattributed Graphs

- The relationship indicates the following difference between the depth-based
  - a) For the depth-based graph kernel, only the subgraphs around a pair of matched vertices having a maximum topology distance K are evaluated with respect to isomorphism. While for the all subgraph kernel, ALL PAIRS of subgraphs are evaluated for identifying the isomorphism.
  - b) The depth-based graph kernel overcomes the NP-hard problems of measuring all possible pairs of subgraphs arising in the all subgraph kernel.
  - c) For the depth-based graph kernel, isomorphic subgraphs are identified by a pair of matched vertices. Hence depth arrangement determines correspondence between the isomorphic subgraphs with respect to their global location. This is not the case for the all subgraph kernel.

# Depth-Based Matching Kernel for Unattributed Graphs

- The relationship indicates the following difference between the depth-based
  - a) For the depth-based graph kernel, only the subgraphs around a pair of matched vertices having a maximum topology distance K are evaluated with respect to isomorphism. While for the all subgraph kernel, ALL PAIRS of subgraphs are evaluated for identifying the isomorphism.
  - b) The depth-based graph kernel overcomes the NP-hard problems of measuring all possible pairs of subgraphs arising in the all subgraph kernel.
  - c) For the depth-based graph kernel, isomorphic subgraphs are identified by a pair of matched vertices. Hence depth arrangement determines correspondence between the isomorphic subgraphs with respect to their global location. This is not the case for the all subgraph kernel.

# Experiments on Bioinformatics Datasets (new)

- Experimental results on graph datasets from the bioinformatics: MUTAG, D&D, PTC(MR), CATH1, CATH2, and PPIs.
  - Alternative kernels: The Weisfeiler-Lehman subtree kernel (**WL**) *[Shervashidze et al., JMLR, 2010]*, the shortest path kernel (**SPGK**) *[Borgwardt and Kriegel, ICDM, 2005]*, and the graphlet count kernel with graphlet of size 3 (**GCGK**) *[Shervashidze et al., ICML, 2009]*
  - Classification with C-SVM

Table I
INFORMATION OF THE GRAPH-BASED DATASETS

| Datasets | MUTAG | D&D | PTC(MR) | CATH1 | CATH2 | PPIs |
|---|---|---|---|---|---|---|
| Max # vertices | 28 | 5748 | 109 | 568 | 568 | 232 |
| Min # vertices | 10 | 30 | 2 | 44 | 143 | 3 |
| Mean # vertices | 17.9 | 284.3 | 25.6 | 205.7 | 308.0 | 109.6 |
| # graphs | 188 | 1178 | 344 | 712 | 190 | 86 |
| # disjoint graphs | 0 | 21 | 0 | 18 | 7 | 0 |
| Proportion of disjoint graphs | 0% | 1.7% | 0% | 2.53% | 3.68% | 0% |
| # classes | 2 | 2 | 2 | 2 | 2 | 2 |

Table II
ACCURACY AND CPU RUNTIME COMPARISONS ON GRAPH DATASETS

| Datasets | MUTAG | D&D | PTC(MR) | CATH1 | CATH2 | PPIs |
|---|---|---|---|---|---|---|
| **DB** | 86.11% | **78.29%** | 57.64% | **99.15%** | **86.31%** | **85.50%** |
| WL | 82.05% | **78.29%** | 58.10% | 94.36% | 74.32% | 81.50% |
| SPGK | **88.13%** | — | **58.46%** | 98.59% | 84.56% | 66.14% |
| GCGK | 75.61% | 75.13% | 55.19% | 97.61% | 66.84% | 51.95% |
| RWGK | 80.72% | — | 51.40% | — | — | 69.70% |
| Datasets | MUTAG | D&D | PTC(MR) | CATH1 | CATH2 | PPIs |
| **DB** | 1” | 40′ | 21” | 9′21” | 1′50” | 12” |
| WL | 1” | 6′21” | 8” | 2′41” | 51” | 8” |
| SPGK | 1” | $> 1day$ | 2” | 6′ | 2′55” | 10” |
| GCGK | 3” | 21′51” | 6” | 19′9” | 7′42” | 15” |
| RWGK | 8” | $> 1day$ | 7′25” | $> 1day$ | $> 1day$ | 27′21” |

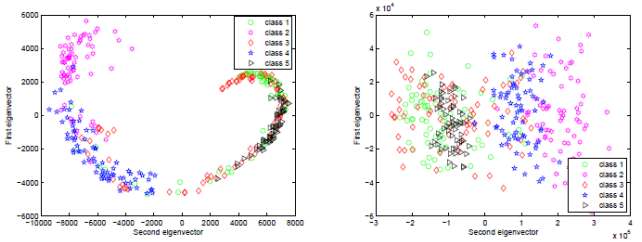
# Experiments on Computer Vision Datasets

- Experimental results on COIL5 (from images), and BAR31, BSPHERE31 and GEOD31 (from 3D shapes). Other kernels include: WL, SPGK and GCGK.
  - Classification with C-SVM and Rand Index for K-means from kernel Principle Component Analysis (kPCA)

| Datasets | DB | WL | SPGK | GCGK |
|---|---|---|---|---|
| COIL5 | **74.22** ± .41 | 33.16 ± 1.01 | 69.97 ± .92 | 67.00 ± .55 |
| BAR31 | **69.40** ± .56 | 58.53 ± .53 | 55.73 ± .44 | 22.96 ± .65 |
| BSPHERE31 | **69.10** ± .66 | 58.00 ± .52 | 55.40 ± .35 | 21.66 ± .65 |
| GEOD31 | **69.43** ± .48 | 58.53 ± .89 | 55.93 ± .53 | 22.93 ± .48 |

| Datasets | DB | WL | SPGK | GCGK |
|---|---|---|---|---|
| COIL5 | **0.4129** | 0.3417 | 0.4019 | 0.3958 |
| BAR31 | **0.2319** | 0.2047 | 0.1734 | 0.1638 |
| BSPHERE31 | **0.2319** | 0.2047 | 0.1734 | 0.1638 |
| GEOD31 | **0.2319** | 0.2047 | 0.1734 | 0.1638 |

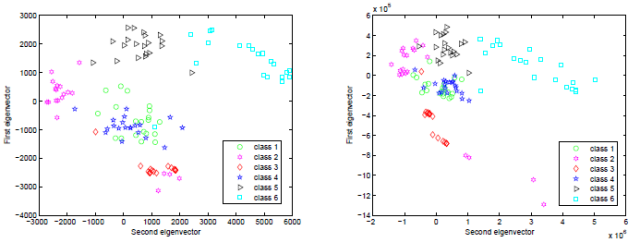
  - Embedding Results with kPCA




(a) For DB Kernel (b) For WL Kernel

(c) For SPGK Kernel (d) For GCGK Kernel

Clusters of Graphs from the COIL5 Dataset.



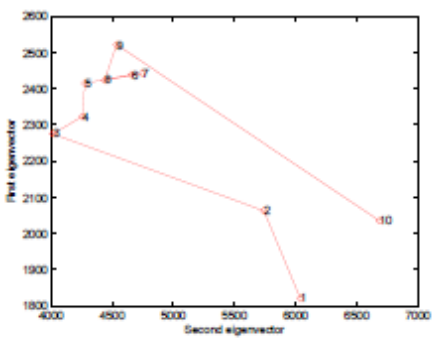

(a) For DB Kernel (b) For WL Kernel

(c) For SPGK Kernel (d) For GCGK Kernel

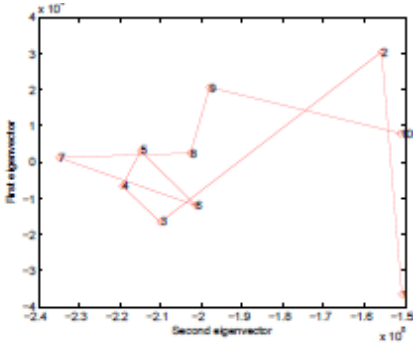
Clusters of Graphs from the BAR31 Dataset.

# Depth-Based Matching Kernel for Unattributed Graphs

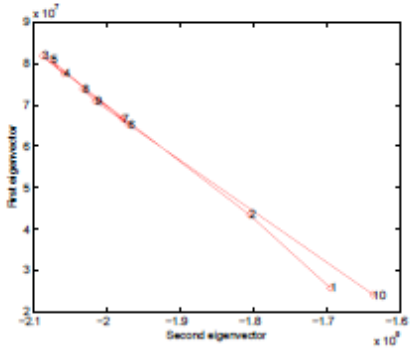
- A) Embedding Results with kPCA (continuous)
- B) Comparisons with increasing *h*



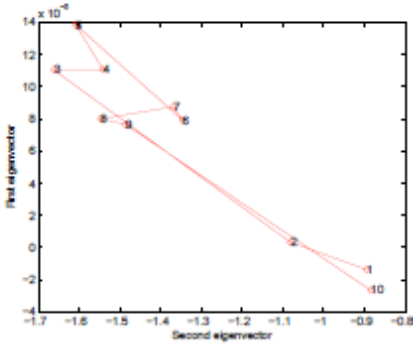

(a) For DB Kernel




(b) For WL Kernel

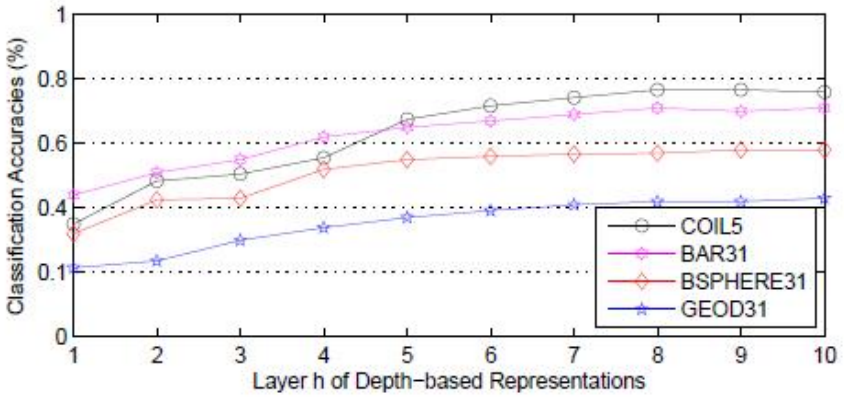



(c) For SPGK Kernel




(d) For GCGK Kernel

Eigenprojections of Graphs.



# Conclusion and Further Work

- Exploit subgraph arrangement and entropic measure of their structural complexity to construct new subgraph kernel.
- More efficient than all-subgraphs kernel.
- Outperforms alternatives on bioinformatics and vision datasets.
- Future work: information theoretic measures of subgraph similarity.

# Acknowledgments


- Prof. **Edwin R. Hancock** is supported by a ***Royal Society Wolfson Research Merit Award***.

- We thank Prof. **Karsten Borgwardt** and Dr. **Nino Shervashidze** for providing the Matlab implementation for the various graph kernel methods, and Dr. **Geng Li** for providing the graph datasets. We thank Dr. **Peng Ren** for the constructive discussion and suggestion about the further work.

Thank you!