Datasets for the Evaluation of Substitution-Tolerant Subgraph Isomorphism

Pierre Héroux, Pierre Le Bodic, Sébastien Adam

Datasets for the Evaluation of Substitution-Tolerant Subgraph Isomorphism

Pierre Héroux
LITIS EA 4108
Université de Rouen, France
Pierre.Heroux@univ-rouen.fr

Pierre Le Bodic
School of Industrial & Systems Engineering
Georgia Tech, USA
lebodic@gatech.edu

Sébastien Adam
LITIS EA 4108
Université de Rouen, France
Sebastien.Adam@univ-rouen.fr

***Abstract*—Due to their representative power, structural descriptions have gained a great interest in the community working on graphics recognition. Indeed, graph based representations have successful been used for isolated symbol recognition. New challenges in this research field have focused on symbol recognition, symbol spotting or symbol based indexing of technical drawing.**

When they are based on structural descriptions, these tasks can be expressed by means of a subgraph isomorphism search. Indeed, in consists in locating the instance of a pattern graph representing a symbol in a target graph representing the whole document image. However, there is a lack of publicly available datasets allowing to evaluate the performance of subgraph isomorphism approaches in presence of noisy data.

In this paper, we present three datasets that can be used to evaluate the performance of algorithms on several tasks involving subgraph isomorphism. Two of these datasets have been synthetically generated and allow to evaluate the search of a single instance of the pattern with or without perturbed labels. The third dataset corresponds to the structural description of architectural plans and allows to evaluate the search of multiple occurrences of the pattern. These datasets are made available for download. We also propose several measures to qualify each of the tasks.

I. Introduction

Graphs are data structures which have gain a great interest in the document analysis community during the last decade thanks to the computation power of nowadays computers. Indeed, the high computational complexity of algorithms which process graphs is now compensated by computer capacities. It is now possible to leverage the flexibility and the description power of this kind of data structure.

Many approaches in the literature on technical documents use region adjacency graphs in which vertices describe regions while edges express an attributed adjacency relationship [1]. In other representations, vertices is associated to primitive shapes (segments, arcs...) while edges carry informations on the relative position of these shapes [2], [3].

In the mean time, research on technical documents has shifted from recognition of isolated symbols to recognition in context, symbol spotting or indexing through visual words.

When using structural representations, recognize, locate or count the occurrences of a pattern symbol in an image turns into a subgraph isomorphism problem. Indeed, these tasks need to identify, locate or count the occurrences of the structural representation of the searched pattern in the structural representation of the whole document image.

As often in pattern recognition applications, noise may affect the structural representation, that is to say that there exist differences between the pattern graph and each of its searched occurrences. This implies that the subgraph isomorphism must tolerate these differences. This problem is known as error-tolerant subgraph isomorphism.

The differences between the pattern and its occurrences in the target graph can be separated in two categories:

1) differences between labels may occur because the features which label vertices or edges are extracted with non sufficiently robust methods.
2) topological differences occur because of a non robust segmentation, that is to say that a region or a shape may be splitted or merged with an other one, resulting in splitted or merged vertices.

Consequently, the error-tolerant subgraph isomorphism problem may result from two distinct sources : a difference in the labeling or a difference in the topology. The substitution-tolerant subgraph isomorphism refers to the search of a subgraph isomorphism in the only presence of differences on label values, whereas topological differences are tackled by inexact subgraph isomorphism.

The communities working on graphics recognition or structural pattern analysis have always had concerns to propose databases that can be references for benchmarking or performance evaluation of an individual processing or a complete system. For example, the IAPR TC-10 and TC-15 provide several datasets among which we can cite the GREC symbol recognition contests [4], the IAM Graph dataset [5]. However, there is very few data that can be used to test error-tolerant subgraph isomorphism. Moreover, to the best of our knowledge, those existing [6] only contain graphs with nominal labels or no label at all. Hence, there is a need for datasets for the evaluation of error-tolerant subgraph isomorphism methods which process graphs labelled with continuous values.

In this paper, we present several ground-truthed databases that we make available. These datasets are intended to evaluate methods for substitution-tolerant subgraph isomorphism when graphs are labeled with continuous values. We also propose the definition of performance measures to numerically qualify the detection of subgraphs.

This paper is structured as follows. The problem statement and main definitions are presented in section II. Section III presents three datasets which are made available and the way the ground-truth is defined. Section IV proposes measures to evaluate the performance of subgraph isomorphism in the context of spotting and counting occurrences of pattern graphs. Finally, section V concludes the paper and draws some perspective to continue this initiative.

II. Substitution-tolerant subgraph isomorphism

Definition 1. A directed attributed multigraph[1] $\mathcal{G}$ is a *4-tuple* $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G}, \mu_\mathcal{G}, \xi_\mathcal{G})$ where $V_\mathcal{G}$ is the set of vertices of $\mathcal{G}$, $E_\mathcal{G}$ is a multiset of ordered pairs $e = (v_1, v_2)$ with $v_1 \in V_\mathcal{G}$ and $v_2 \in V_\mathcal{G}$, i.e. edges of $\mathcal{G}$. $\mu_\mathcal{G} : V_\mathcal{G} \rightarrow L_V$ is a function assigning a *label* to a vertex, L_V being the set of possible labels for vertices. $\xi_\mathcal{G} : E_\mathcal{G} \rightarrow L_E$ is a function assigning a *label* to an edge, L_E being the set of possible labels for edges.

Definition 2. Given a graph $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G}, \mu_\mathcal{G}, \xi_\mathcal{G})$, a subgraph of $\mathcal{G}$ is a graph $\mathcal{G}' = (V_{\mathcal{G}'}, E_{\mathcal{G}'}, \mu_{\mathcal{G}'}, \xi_{\mathcal{G}'})$ such that $V_{\mathcal{G}'} \subseteq V$, $E_{\mathcal{G}'} \subseteq E$, $\forall e = (v_1, v_2) \in E_{\mathcal{G}'}, v_1 \in V_{\mathcal{G}'}, v_2 \in V_{\mathcal{G}'}$ and $\mu_{\mathcal{G}'}$ and $\xi_{\mathcal{G}'}$ are the restrictions of $\mu_\mathcal{G}$ and $\xi_\mathcal{G}$ to $V_{\mathcal{G}'}$ and $E_{\mathcal{G}'}$, i.e. $\mu_{\mathcal{G}'}(v) = \mu_\mathcal{G}(v)$ and $\xi_{\mathcal{G}'}(e) = \xi_\mathcal{G}(e)$

Definition 3. An injective function $f : V_\mathcal{S} \rightarrow V_\mathcal{G}$ is a subgraph isomorphism from a graph $\mathcal{S} = (V_\mathcal{S}, E_\mathcal{S}, \mu_\mathcal{S}, \xi_\mathcal{S})$ to a graph $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G}, \mu_\mathcal{G}, \xi_\mathcal{G})$ if there exists a subgraph $\mathcal{G}'$ of $\mathcal{G}$ such that f is a graph isomorphism from $\mathcal{S}$ to $\mathcal{G}'$:

- $\forall v \in V_\mathcal{G}, f(v) = v' \in V_{\mathcal{G}'}, f^{-1}(v') = v$
- for all $e = (v_1, v_2) \in E_\mathcal{G}$, there exists a distinct edge $e' = (f(v_1), f(v_2)) \in E_{\mathcal{G}'}$

Note that extra edges may exist in $\mathcal{G}'$ between mapped vertices, *i.e.* a subgraph does not need to be induced.

In its exact formulation, the subgraph isomorphism must preserve the labelling, *i.e.* $\mu(v) = \mu'(v')$ and $\xi(e) = \xi'(e')$. In pattern recognition applications, where vertices and edges are labeled with measures which may be affected by noise, a substitution-tolerant formulation which allows differences between labels of mapped vertices and edges is mandatory. However, in order to take into account these differences, they are penalized by a non decreasing cost function. Finally, the total cost associated to the mapping between a graph $\mathcal{S} = (V_\mathcal{S}, E_\mathcal{S}, \mu_\mathcal{S}, \xi_\mathcal{S})$ and a subgraph of $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G}, \mu_\mathcal{G}, \xi_\mathcal{G})$ is given by eq. (1).

$$C_M(\mathcal{S}, \mathcal{G}) = \sum_{i \in V_\mathcal{S}} \sum_{k \in V_\mathcal{G}} c_V(i,k) * x_{i,k} + \sum_{ij \in E_\mathcal{S}} \sum_{kl \in E_\mathcal{G}} c_E(ij,kl) * y_{ij,kl} \quad (1)$$

In this equation, $c_V(i,k)$ and $c_E(ij,kl)$ respectively denote the elementary cost for mapping a vertex $i \in V_\mathcal{S}$ to a vertex $k \in V_\mathcal{G}$ and the cost for mapping an edge $ij \in E_\mathcal{S}$ to an edge $kl \in E_\mathcal{G}$. M represents a possible isomorphism f between $\mathcal{S} = (V_\mathcal{S}, E_\mathcal{S}, \mu_\mathcal{S}, \xi_\mathcal{S})$ and a subgraph of $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G}, \mu_\mathcal{G}, \xi_\mathcal{G})$ as a set of binary variables $x_{i,k}$ and $y_{ij,kl}$. $x_{i,k}$ is set to 1 if $f(i) = k$ and equals 0 otherwise. Similarly, $y_{ij,kl}$ is set to 1 if $ij \in V_\mathcal{G}$ is mapped to $kl \in V_\mathcal{S}$ and is set to 0 otherwise. Moreover, the binary variables in M must respect the following constraints in order to ensure that f is an isomorphism.

- Every vertex of $V_\mathcal{S}$ must be matched to a unique vertex of $V_\mathcal{G}$:
$$\sum_{k \in V_\mathcal{G}} x_{i,k} = 1 \quad \forall i \in V_\mathcal{S} \quad (2)$$
- Every edge of $E_\mathcal{S}$ must be matched to a unique edge of $E_\mathcal{G}$:
$$\sum_{kl \in E_\mathcal{G}} y_{ij,kl} = 1 \quad \forall ij \in E_\mathcal{S} \quad (3)$$
- Every vertex of $V_\mathcal{G}$ must be matched to at most a vertex of $E_\mathcal{S}$:
$$\sum_{i \in V_\mathcal{S}} x_{i,k} \leq 1 \quad \forall k \in V_\mathcal{G} \quad (4)$$
- If two vertices are matched together, an edge originating the vertex of $\mathcal{S}$ must be matched with an edge originating the vertex of $\mathcal{G}$:
$$\sum_{kl \in E_\mathcal{G}} y_{ij,kl} = x_{i,k} \quad \forall k \in V_\mathcal{G}, \forall ij \in E_\mathcal{S} \quad (5)$$
- If two vertices are matched together, an edge targeting the vertex of $\mathcal{S}$ must be matched with an edge targeting the vertex of $\mathcal{G}$:
$$\sum_{kl \in E_\mathcal{G}} y_{ij,kl} = x_{j,l} \quad \forall l \in V_\mathcal{G}, \forall ij \in E_\mathcal{S} \quad (6)$$

III. Datasets

In this section, we present the three datasets that are made available at http://litis-ilpiso.univ-rouen.fr [2]. These datasets have been used to evaluate the substitution-tolerant subgraph isomorphism approach described in [7].

A. Exact synthetic dataset

The `ILPIso_exact_synth` dataset is a synthetic dataset which provides 180 pattern-target graph couples. The graph couples have been synthetically generated according to the following procedure. First, a random graph $\mathcal{S}$ is generated according to the Erdös-Rényi model [8] whose parameters are $n_\mathcal{S}$, the number of vertices and p which is the probability that a directed edge between two distinct vertices exists. Vertices and edges are labeled with a random numerical value according a uniform probability distribution in $[-100, 100]$. Then, a graph $\mathcal{G}_0$ is created as an exact copy of $\mathcal{S}$. Finally, $\mathcal{G}_0$ is completed to form a graph $\mathcal{G}$ with vertex and edge insertions (with the same random model for labels) according the Erdös-Rényi model until its size is $n_\mathcal{G}$.

The following parameters have been chosen:

- Size of $\mathcal{G}$: $|V_G| = n_\mathcal{G} \in \{50, 100, 250, 500\}$
- Size of $\mathcal{S}$: $|V_S| = n_\mathcal{S} \in \{10, 25, 50\}$
- Probability that an edge connects two vertices : $p \in \{0.01, 0.05, 0.1\}$

[1] In the remaining of the paper, the term graph denotes a directed attributed multigraph.

[2] We also intend to propose theses datasets on the TC-10 and TC-15 websites.

The `ILPIso_exact_synth` dataset is composed of five instances of pattern-target graph couples for each combination of $(n_{\mathcal{G}}, n_{\mathcal{S}}, p)$.

During this procedure, the mapping between the vertices of $\mathcal{S}$ and $\mathcal{G}$ is tracked to finally constitute the ground-truth. Even if new non-tracked isomorphisms can be added during the completion of $\mathcal{G}_0$ to $\mathcal{G}$, their cost cannot be lower than the ground-truth one whose cost is 0. Moreover, it has been experimentally checked that the isomorphism in the groundtruth information is the only one whose cost is 0.

This dataset is mainly intended to check whether a unique instance of an exact subgraph isomorphism is successfully found with a tested algorithm.

B. Noisy synthetic dataset

The `ILPIso_noisy_synth` dataset is also a synthetic dataset. It also contains 180 pattern-target graph couples. It has been created in the same manner than the `ILPIso_exact_synth` dataset (with the same combination of values for $(n_{\mathcal{G}}, n_{\mathcal{S}}, p)$) but an additional step has been introduced. Before its completion to $\mathcal{G}$, $\mathcal{G}_0$ has been modified by editing vertices and edges labels. Each label has been added a random value according a gaussian distribution with $m = 0$ and $\sigma^2 = 5$.

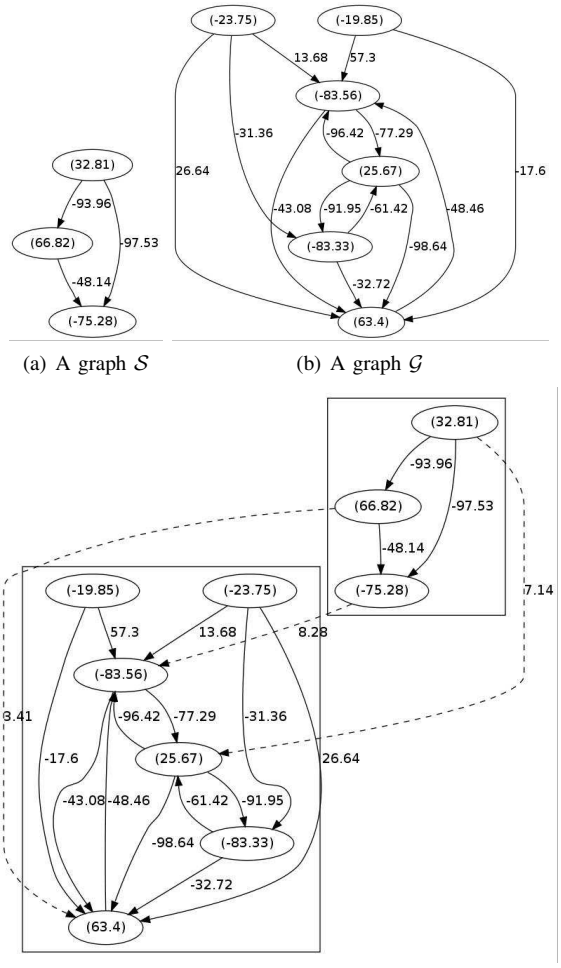
This dataset is intended to evaluate the performance of a substitution-tolerant subgraph isomorphism search program where the mapping with the lower cost is searched. As for the `ILPIso_exact_synth` dataset, new non tracked isomorphisms are added when $\mathcal{G}_0$ is completed to $\mathcal{G}$, but it has been checked that the groundtruth corresponds to the isomorphism with the lowest cost.

Figure 1 illustrates an output produced by the synthetic data generation in the presence of noise.

C. `ILPIso_real` dataset

The `ILPIso_real` dataset is a dataset that contains structural representations of document images of architectural plans. It also contains graphs modeling 16 isolated graphical symbols. Complete plan images which lead to the building of `ILPIso_real` dataset are 200 images extracted from the `floorplan` section of the SESYD dataset[9]: 20 first images for each of the 10 templates. Each symbol may occurs once or several times or not occur at all on each floorplan instance. Considering the structural representations, the problem of locating one or several instances of a symbol in a complete floorplan turns into the search of a subgraph isomorphism problem.

The structural representation for both complete plans and isolated symbol models are region adjacency graphs (RAG), which are extracted according to the following process. A vertex is created for each white connected component after a skeletonization process [10] which has reduced the width of black strokes to 1 pixel. Two directed edges are created between the vertices which describe adjacent white regions. Vertices are labelled with a feature vector corresponding to a set of the 24 first Zernike Moments (ZM)[11] and edges are labelled with *relative scale* and *relative distance*. The algorithm used for extracting such RAGs is fully described in



(a) A graph $\mathcal{S}$ (b) A graph $\mathcal{G}$

(c) Ground-truth for the mapping between $\mathcal{S}$ and $\mathcal{G}$.

Fig. 1. Illustration of the generation of synthetic data for the substitution-tolerant subgraph isomorphism search with the following parameters : $n_{\mathcal{S}} = 3$, $n_{\mathcal{G}} = 6$, $p = 0.3$, labels in [-100,100] and a Gaussian noise ($\sigma^2 = 5$). The mapping cost is 24.93

[1]. The whole `ILPIso_real` dataset contains 5609 symbol instances, with an average of 28 instances per document image. The graphs corresponding to symbol instances contain 4 vertices and 7 edges on average, whereas the structural representations of the plans contain 121 vertices and 525 edges on average.

Figure 2 shows two examples of plans and the corresponding RAGs. Figure 3 represent the 16 symbol models.

Even if the document images have been synthetically generated, the graph dataset which has been produced can be considered as a real dataset. It can be used to evaluate the search of one or multiple instances of substitution-tolerant subgraph isomorphism. Indeed, since the images are synthetic there is not any problem with splitted or merged regions which would have generated topological differences between the pattern graph and its occurrences in the target graph. The only differences concern vertices and edges labels.

The ground-truth information associated to this dataset gives for each target graph, the number of occurrences for each of the 16 pattern graph. For each occurrence, the vertices

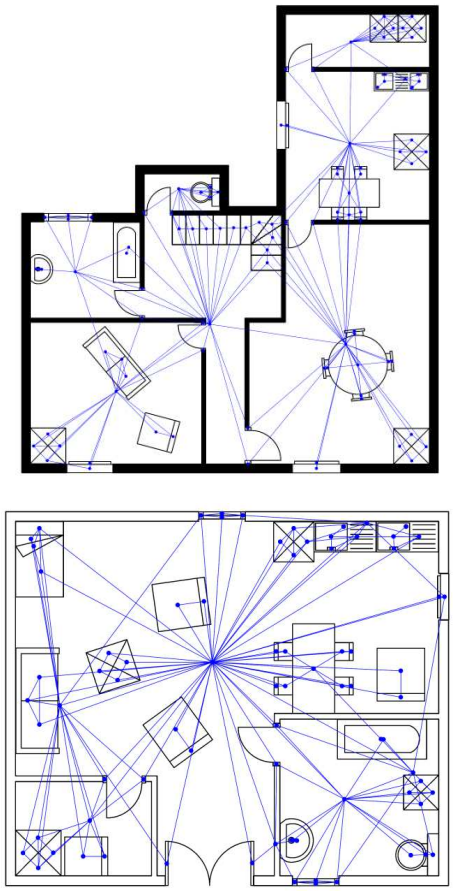


Fig. 2. Examples of plans from the `floorplan` dataset with the corresponding RAGs

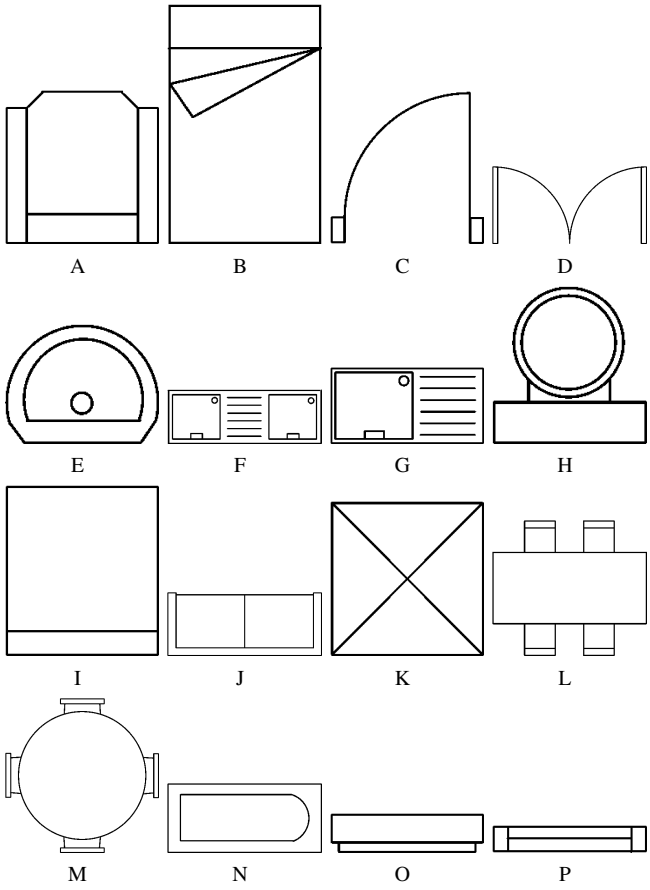


Fig. 3. Symbol models

identifier from the target graph involved are listed. The induced subgraph represents the subgraph of the target graph which is isomorphic with the pattern graph. However, due to symmetry phenomenons, there can not be an exact vertex-to-vertex mapping between a pattern graph and its isomorphic subgraph in the target. This impacts the definition of performance measures which can only be done at a subgraph level and not at the vertex level.

IV. Performance measures

In this section, we propose some performance measures which can be used with the databases presented in section III to evaluate substitution-tolerant subgraph method.

The `ILPIso_exact_synth` dataset can be used to evaluate the ability of an approximate algorithm to detect an exact subgraph isomorphism. As there is a unique instance of pattern-target matching for each graph couple, this measure can only be done at the database level. The most objective measure is the detection rate

$$\text{detection rate} = \frac{\text{\#detected mapping}}{\text{\#graph couples}}$$

The same measure can be used with the `ILPIso_noisy_synth` dataset for which the objective is to find the lowest cost mapping. For approximate search algorithm able to provide a ranked list of mapping, this measure can be extended to top n results.

$$\text{top n detection rate} = \frac{\text{\#detected mapping in the n first results}}{\text{\#graph couples}}$$

Finally, the most complex evaluation can be performed with the `ILPIso_real` dataset as it contains 0, 1 or several occurrences of the 16 pattern graphs in the 200 target graphs. This dataset can first be used to evaluate a structural information retrieval system where the objective is to select among the target graphs those containing occurrences of query patterns. As for classical IR techniques, the evaluation can be measured in terms of precision P and recall R.

$$P = \frac{\text{\#relevant retrieved pattern graphs}}{\text{\#retrieved pattern graphs}}$$

$$R = \frac{\text{\#relevant retrieved pattern graphs}}{\text{\#relevant pattern graphs}}$$

Some algorithms are able to compute a numerical value (cost or probability) which can be thresholded to decide whether there is an occurrence of the query or not. Several values of the P/R trade-off can be obtained by varying the threshold resulting in a precision-recall curve.

But this dataset can also be used to evaluate a subgraph isomorphism as a spotting system. Indeed, in each structural representation of a complete floorplan, several sets of vertices have been identified. The induced subgraph for each vertex set is considered as an instance of the structural representation of a symbol model.

For a single query, the system under evaluation should return its results as a set that contains the identifiers of the vertices in the target graph that are involved in the searched instance. It may also be the case that the system decides that no instance of the pattern graph occurs in the pattern. When comparing the result with the groundtruth information, several configurations can happen :

- There can be a perfect mapping between the detected set of identifiers and the set given in the groundtruth (good detection).
- There can also be a decision of reject when to instance of the searched pattern exists in the target graph (good reject).
- We consider a false reject decision, when the system decides a reject whereas an instance of the pattern graph exists in the groundtruth.
- On the other hand, we denote as a false alarm the event of detecting a pattern graph instance whereas the groundtruth indicates that it does not exist.

Moreover, besides these classical definitions, we also consider :

- an erroneous detection when a pattern instance is detected but the corresponding vertex identifier set has an empty intersection with the one defined in the groundtruth.
- a partial detection when a pattern instance is detected but the corresponding vertex identifier set has a non empty intersection with the one defined in the groundtruth.

The `ILPIso_real` dataset contains 16 pattern graphs and 200 target graphs allowing to test a system according the criterions on 3200 single occurrence searches. As mentioned before, the reject decision may be taken according to a threshold on the cost of the mapping. As a consequence, several trade-offs between the measures can be achieved with different threshold values.

Finally, the `ILPIso_real` dataset can also be used to evaluate a system in its ability to find multiple occurrences of a pattern in the target graph. For this purpose, the measures detailed above can be used for the evaluation performance with some adjustments. Indeed, when a system tries to find the n^{th} occurrence of a pattern graph, it should be considered that $n-1$ occurrences have already been, at least, partially found. So, the n^{th} detected occurrence should be considered as a false alarm if and only if the real number of occurrences given by the groundtruth information is lower than n. Otherwise, it should be considered as a perfect, partial or erroneous detection. If the system returns a reject decision in the n^{th} whereas not all occurrences have already been detected, it should be classified as a non detection. A good rejection decision is taken if the n^{th} is a reject whereas all occurrences given in the groundtruth have been detected. For a complete evaluation, the number of searched instances should be greater than the maximum number of real instances of a pattern in target graph, which is 13.

V. Conclusion

In this paper, three datasets are presented that can be used for several tasks involving the search for subgraph isomorphisms. We have also proposed several measures allowing performance evaluation on different tasks.

Two synthetic datasets can be used to benchmark the search for a single instance of a pattern in a target graph. In the first one, no perturbation is brought to labels whereas the second one has numerical labels which have been modified with the add of a gaussian noise. The measure defined for this task quantify the detection rate for a subgraph isomorphism tool, and the second dataset serves at evaluating its robustness to noise on numerical labels.

The third dataset is a real dataset composed from the structural descriptions by means of attributed RAGs of architectural plans. These target graphs contain several instances of several pattern graph describing symbols occurring in the plans. Thanks to the associated measures, this dataset can be used to benchmark a retrieval system based on structural description or the search of multiple occurrences of pattern graph in a target in presence of perturbed numerical labels.

This work could be extended in several directions. First, in order to have a better review of the robustness to label perturbation, it could be considered to offer several noise models with several levels. Otherwise, the real dataset is currently composed of attributed RAGs. Despite the important labeling effort this would require, it could interesting to propose alternative structural descriptions such as graphs of graphical primitives [2].

In the longer term, the scope of evaluated tasks could be extended by the integration of error-tolerant subgraph isomorphism, where topological differences are allowed between the pattern and its occurrences in the target graph. But, this extension is not manifest since it raises some issues on the groundtruth definition. Indeed, defining if a vertex in the target graph belongs or not to the pattern instance is questionable. Moreover, this may require to define one-to-one, many-to-one, one-to-many and many-to-many mappings between vertices.

The authors hope that the presented work could help the community. Any suggestion would be welcome and will be considered.

References

[1] P. Le Bodic, H. Locteau, S. Adam, P. Héroux, Y. Lecourtier, and A. Knippel, "Symbol detection using region adjacency graphs and integer linear programming," in *Proceedings of the International Conference on Document Analysis and Recognition (ICDAR'09)*, 2009, pp. 1320–1324.

[2] R. L. Qureshi, J.-Y. Ramel, D. Barret, and H. Cardot, "Spotting symbols in line drawing images using graph representations," in *Graphics Recognition. Recent Advances and New Opportunities*, ser. Lecture Notes in Computer Science, 2008, pp. 91–103.

[3] H. Locteau, S. Adam, E. Trupin, J. Labiche, and P. Héroux, "Symbol spotting using full visibility graph representation," in *Proceedings of the seventh International Workshop on graphics Recognition*, 2007, pp. 49–50.

[4] E. Valveny, M. Delalandre, R. Raveaux, and B. Lamiroy, "Report on the symbol recognition and spotting contest," in *Graphics Recognition. New Trends and Challenges*, ser. Lecture Notes in Computer Science, Y.-B. Kwon and J.-M. Ogier, Eds. Springer Berlin Heidelberg, 2013, vol. 7423, pp. 198–207.

[5] K. Riesen and H. Bunke, "Iam graph database repository for graph based pattern recognition and machine learning," in *Structural, Syntactic and Statistical Pattern Recogntion*, ser. Lecture Notes in Computer Science, N. Vitoria Lobo, T. Kasparis, F. Roli, J. Kwok, M. Georgiopoulos, G. Anagnostopoulos, and M. Loog, Eds. Springer Berlin Heidelberg, 2008, vol. 5342, pp. 287–297.

[6] P. Foggia, C. Sansone, and M. Vento, "A database of graphs for isomorphism and sub-graph isomorphism benchmarking," in *CoRR*, 2001, pp. 176–187.

[7] P. Le Bodic, P. Héroux, S. Adam, and Y. Lecourtier, "An integer linear program for substitution-tolerant subgraph isomorphism and its use for symbol spotting in technical drawings," *Pattern Recognition*, vol. 45, no. 12, pp. 4214 – 4224, 2012.

[8] P. Erdös and A. Rényi, "On random graphs," *Publicationes Mathematicae*, vol. 6, pp. 290–297, 1959.

[9] M. Delalandre, E. Valveny, T. Pridmore, and D. Karatzas, "Generation of synthetic documents for performance evaluation of symbol recognition; spotting systems," *International Journal on Document Analysis and Recognition*, vol. 13, pp. 187–207, 2010.

[10] G. S. di Baja and E. Thiel, "Skeltonization algorithm running on path-based distance maps," *Image and Vision Computing*, vol. 14, pp. 47–57, 1996.

[11] M. Teague, "Image analysis via the general theory of moments," *Journal of the Optical Society of America*, vol. 70, no. 8, pp. 920–930, 1980.