Pattern Recognition 76 (2018) 596–611

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Product graph-based higher order contextual similarities for inexact subgraph matching

Anjan Dutta[a,*], Josep Lladós[a], Horst Bunke[b], Umapada Pal[c]

[a] Computer Vision Center, Autonomous University of Barcelona, Edifici O, Campus UAB, Bellaterra, Barcelona, 08193 Spain
[b] Institute of Computer Science, University of Bern, Neubrückstrasse 10, Bern, CH-3012 Switzerland
[c] Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, 203, B.T.Road, Kolkata-108, India

A R T I C L E I N F O

Article history:
Received 31 March 2017
Revised 28 August 2017
Accepted 5 December 2017
Available online 7 December 2017

Keywords:
Subgraph matching
Product graph
Random walks
Backtrackless walks
Contextual similarities
Graphic recognition

A B S T R A C T

Many algorithms formulate graph matching as an optimization of an objective function of pairwise quantification of nodes and edges of two graphs to be matched. Pairwise measurements usually consider local attributes but disregard contextual information involved in graph structures. We address this issue by proposing contextual similarities between pairs of nodes. This is done by considering the tensor product graph (TPG) of two graphs to be matched, where each node is an ordered pair of nodes of the operand graphs. Contextual similarities between a pair of nodes are computed by accumulating weighted walks (normalized pairwise similarities) terminating at the corresponding paired node in TPG. Once the contextual similarities are obtained, we formulate subgraph matching as a node and edge selection problem in TPG. We use contextual similarities to construct an objective function and optimize it with a linear programming approach. Since random walk formulation through TPG takes into account higher order information, it is not a surprise that we obtain more reliable similarities and better discrimination among the nodes and edges. Experimental results shown on synthetic as well as real benchmarks illustrate that higher order contextual similarities increase discriminating power and allow one to find approximate solutions to the subgraph matching problem.

1. Introduction

The use of representational models allowing to describe visual objects in terms of their components and their relations has gained interest among the research community. Relations between parts express a notion of context, so objects can be recognized even if they appear in highly distorted or noisy visual conditions. At the heart of structural models, graphs have been used over decades as robust and theoretically solid representations. When objects are represented structurally by attributed graphs, their comparison for recognition and retrieval purposes is performed using some form of *graph matching*. Basically, an *attributed graph* is a 4-tuple $G = (V, E, \alpha, \beta)$ comprising a set V of *vertices* or *nodes* together with a set $E \subseteq V \times V$ of *edges* and two *mappings* $\alpha : V \to \mathbb{R}^m$ and $\beta : E \to \mathbb{R}^n$ which respectively assign attributes to the nodes and edges. Given two attributed graphs, *graph matching* can roughly be defined as a process of finding a mapping between the node and edge sets of two graphs that satisfies some constraints. Graph matching is a very fundamental problem in theoretical computer science and it has been successfully applied to many applications in computer vision and pattern recognition [1–8]. In fact, graphs are used in several domains including chemical compound, social network, biological and protein structure, program flow etc. Many complex data structures can efficiently be represented with a graph in terms of relations among different entities. Hence the theory of graph matching can be applied to different problems in any of the previously mentioned fields.

Graph matching is a very well known but challenging task, which is classified as a *GI-complete* problem [9]. A particular class of algorithms resulting from the consideration of outliers[1] is called *subgraph matching*. Roughly, it can be defined as matching one graph (pattern graph) as part of the other (target graph). Addressing this problem is justified since in computer vision and pattern recognition applications, due to the existence of background, occlusions and geometric transformations of objects, outliers do exist in many scenarios. Most of the modern algorithms formulate (sub)graph matching as an optimization of an objective function

* Corresponding author.
E-mail addresses: adutta@cvc.uab.es (A. Dutta), josep@cvc.uab.es (J. Lladós), bunke@iam.unibe.ch (H. Bunke), umapada@isical.ac.in (U. Pal).

[1] By outliers, we mean the set of nodes and edges in the target graph that are not related to nodes and edges of the pattern graph.

https://doi.org/10.1016/j.patcog.2017.12.003
0031-3203/

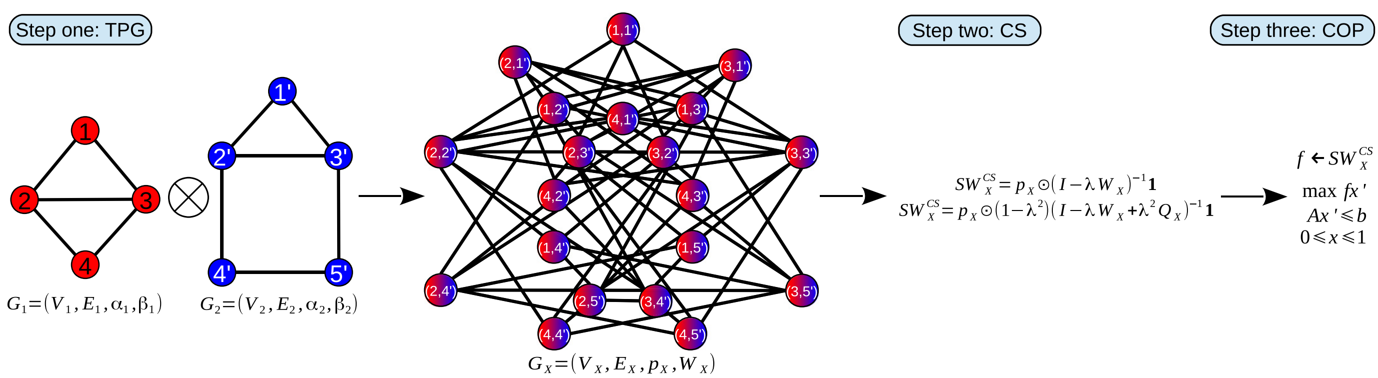

Fig. 1. Outline of the proposed method: Step one: computation of the tensor product graph (TPG) G_X of two operand graphs G_1 and G_2, here $\otimes$ denotes the tensor product operation of two graphs, Step two: algebraic procedure to obtain contextual similarities (CS), Step three: constrained optimization problem (COP) for subgraph matching.

of pairwise quantification of the nodes and edges. But pairwise measurements usually consider local attributes and do not encode context information *i.e.* neighbourhood information of nodes and edges. As a result, pairwise quantifications are less reliable, especially in case of subgraph matching scenarios [10]. This is because of the existence of large amounts of outliers, which most of the graph matching algorithms do not take into account [11–13]. This fact is also reported for many real world scenarios where higher order similarities that employ contextual information have been proposed as a solution [14]. We have also experienced quite similar phenomena with pairwise measurements in the presence of outliers. This fact will be presented in the experimental study in Section 6 (symbol spotting experiment) as a benefit of the proposed contextual similarities (see Fig. 8).

The main motivation of this work is the use of contextual information of nodes (*i.e.* neighbouring structures) to make subgraph matching more robust and efficient in large scale visual recognition scenarios. A second key component is the formulation of subgraph matching with approximate algorithms. Therefore, in this paper we propose an inexact subgraph matching methodology based on *tensor product graph* (TPG). Given two attributed graphs, it is quite straight forward to get the pairwise similarities and assign them as weights on the edges of TPG (step one in Fig. 1). Next, one can think of having a random walk from node to node considering the weights on the edges as the probabilities to proceed to the next node. Finally, we accumulate the probabilities of finishing a walk at each of the vertices, which we refer to as *contextual similarities* (CS), where the context of each node is the set of its neighbouring nodes. Throughout this article, by contextual similarities we mean higher order affinities between two nodes, each coming from two graphs to be matched. The procedure of obtaining CS essentially diffuse the pairwise similarities in the context of neighbouring nodes. Since the edges of TPG contain the pairwise similarities between nodes and edges, the consideration of longer random walks on TPG reevaluates the pairwise similarities between the nodes of the operand graphs in the context of other nodes. This random walk procedure essentially takes into account higher order similarity information, which can be obtained by simple algebraic operation (discounted exponentiation and summation) of the adjacency (or weight) matrix of the product graph (step two in Fig. 1). A similar phenomenon is termed *diffusion on tensor product graph*, which is well known to capture the geometry of data manifold and higher order contextual information between two objects [14,15]. All these works show that considering the manifold structure or context together with pairwise comparisons significantly improves ranking/retrieval performance [14], which acts as our further incentive towards this direction. Thereafter, we formulate subgraph matching as a node and edge selection problem in TPG. To do that, we use those contextual similarities and formulate a constrained optimization problem (COP) to optimize a function constructed from the higher order similarity values (step three in Fig. 1). We solve the optimization problem with linear programming (LP) which is solvable in polynomial time. Depending on whether we need a discrete solution or not, we may add a discretization step. In this paper we will show that the higher order contextual similarities allow us to relax the constrained optimization problem in real world scenarios.

The main contribution of this work is to propose a random walk based approach to obtain contextual similarities between nodes that capture higher order information. In the literature, there are methods that formulate graph matching explicitly using higher order information of nodes and edges [16,17]. They often embed the higher order relations of nodes by hypergraph matching formulations. Usually the affinity of feature tuples are modelled using tensor and later approximate algorithms are proposed to obtain the correspondences [16]. Our work takes into account the higher order context information by propagating the pairwise similarities with random walk formulation. For that, we utilize TPG, the procedure counts the number of incoming walks to the nodes of TPG. Here the general hypothesis is that a particular node in TPG, that constitutes a pair of nodes of the operand graphs likely to be matched, should receive a reasonably high amount of random walks. This idea is also supported by other works [14–16,18–20], where the idea of random walk has been used for obtaining similarity of pairs of nodes. Particularly, Cho et al. [20] employed a random walk based procedure for iteratively updating the matching matrix **X** by considering the context information propagated through random walks. Our formulation and interpretation are completely different than the others. Here, we have used the classical algebraic formulation to obtain the similarities between nodes whereas most of the previously mentioned algorithms have used an approximated iterative procedure to obtain the similarities or the node to node correspondences depending on the requirement. Apart from that, we have revealed that backtrackless walks [21] can also be utilized for obtaining contextual similarities of pairs of nodes. Later, we use these contextual similarities to formulate a subgraph matching procedure as a node and edge selection procedure in TPG. This is an optimization problem and here we are motivated by the ILP formulation of Le Bodic et al. [22]. However, Le Bodic et al. solved the problem with an integer programming, whereas, we solved it with a linear program followed by a discretization step, which is far efficient than [22]. We have proved the effectiveness of the proposal with a detailed experimental study. Throughout the paper, we use **I** to denote the