Digital Signal Processing 31 (2014) 1–12

Contents lists available at ScienceDirect

Digital Signal Processing

www.elsevier.com/locate/dsp

Component evolution analysis in descriptor graphs for descriptor ranking

Levente Kovács *,[1], Anita Keszler [1], Tamás Szirányi [1]

Distributed Events Analysis Research Laboratory, Institute for Computer Science and Control Hungarian Academy of Sciences (MTA SZTAKI), Kende u. 13-17, 1111 Budapest, Hungary

ARTICLE INFO

Article history:
Available online 5 May 2014

Keywords:
Descriptor evaluation
Feature extraction
Feature selection
Graph representation
Graph components

ABSTRACT

This paper presents a method based on graph behaviour analysis for the evaluation of descriptor graphs (applied to image/video datasets) for descriptor performance analysis and ranking. Starting from the Erdős–Rényi model on uniform random graphs, the paper presents results of investigating random geometric graph behaviour in relation with the appearance of the giant component as a basis for ranking descriptors based on their clustering properties. We analyse the phase transition and the evolution of components in such graphs, and based on their behaviour, the corresponding descriptors are compared, ranked, and validated in retrieval tests. The goal is to build an evaluation framework where descriptors can be analysed for automatic feature selection.

1. Introduction

Content based retrieval in large video/image datasets is highly dependent on the choice of discriminating features and efficient index structures. Recent approaches involve graph clustering, clique searching, and component analysis methods. Open issues remain how to build the graphs (selection of edges and weights), and how to navigate them efficiently (neighbourhood search). We propose and work towards proving that graph theoretic approaches can be useful in content based retrievals, for descriptor evaluation and automatic feature selection. We build our approach on the investigation of entity difference distributions according to several descriptors and analysing their relations and behaviour during component formulation and the appearance of the so called giant component in the graphs of the descriptors. As we will detail later, as the novelty of our approach, our goal is to exploit the inherent properties of the graph representations to evaluate descriptors based on the behaviour of their graphs during the formulation of the giant component, analysing their discrimination capabilities. The presented method has some connections to graph clustering methods in the sense that the effects of the descriptors on the structure of their graphs is related to their clustering properties.

When searching for similar content in video/image datasets, we need to apply feature extractors that gather information about the content and structure of the stored data, and use that information to create a searchable index for the dataset, which in turn will be the basis of searching for similar content. However, there are a lot of different descriptors, and usually it is very hard to select those, that perform well for a given dataset, when using them to produce retrieval results. Our purpose is to help this process by providing a means to evaluate a set of descriptors for a given set of classes and data, and to find a combination of descriptors that perform better. This information can then be used to create more efficient indexes and produce higher precision retrievals.

Feature selection in the presence of irrelevant features (noise) is presented in [1], taking into consideration sample data points in 2D for boundary selection and investigating the distribution of feature weights in high dimensions. A method for feature selection [2] is based on approximately 1000 features on real videos, using heuristics for feature retention, using the sort-merge approach for selecting ranked feature groups. A method for sport video feature selection is presented in [3]; Setia and Burkhardt [4] present a method for automatic image annotation based on a feature weighting scheme and machine learning; Guldogan and Gabbouj [5], Li et al. [6] present similar approaches for feature selection based on mutual information and principal component analysis. Zhang et al. [7] presents a query by example approach where histograms of point distances are investigated as a basis to show that with increased dimensions the distance distributions tend to be narrower (poor discrimination), and SIFT feature distribution histograms are used to improve clustering and retrieval.

* Corresponding author.
E-mail addresses: levente.kovacs@sztaki.mta.hu (L. Kovács), keszler.anita@sztaki.mta.hu (A. Keszler), sziranyi@sztaki.mta.hu (T. Szirányi).
[1] *URL:* http://web.eee.sztaki.hu

http://dx.doi.org/10.1016/j.dsp.2014.04.013
1051-2004/

Graphs are a natural way of representing data structures, describing interconnections and internal structures of datasets, visualising relations and distances of elements, and finding subsets, clusters and communities in such structures. Graphs have been widely used for clustering applications, including spectral clustering [8] for graph partitioning, MST (minimum spanning tree) based clustering [9], dense sub-graph mining [10], etc. The uses of graph clustering approaches are various, from generic pattern recognition (e.g., [11,12]), to the recently highly researched community detection approaches in graphs representing social structures [13–16].

Contrary to other approaches, we do not use artificial feature weighting or a priori clustering, instead we use real data with multiple features and weigh the built graphs by the points' differences according to features, and investigate the behaviour of the distributions. The goal is to show that this method is a good alternative to previous ones for finding features with higher discriminative properties. In our earlier work [17] we have proposed the use of descriptor graphs for descriptor ranking, and we produced a fitness function for providing such a rank [18]. This work extends these previous results by deeper investigation of the properties of such graph structures, regarding similarity in behaviour and topography, and the use of such intrinsic properties for feature selection.

We will start by introducing basic concepts and random geometric graphs (Section 2), followed by the description of the proposed parameters for ranking based on phase transition and component behaviour of descriptor graphs (Section 3), then the presentation of the used datasets and descriptors (Section 4), and finally the presentation of the ranking function and the performed evaluations (Section 5).

2. Component analysis of random graph models

In this section we overview the properties of two frequently applied random graph models and their component structures. Based on the results corresponding to random graphs, we get a better understanding of the properties of real-world graph structures. Let us start with some definitions of important terminology.

Definition 1. An *undirected graph* is a $G = (V, E)$ pair, where V denotes the set of vertices (or nodes) and E denotes the set of edges. $E \subseteq V \times V$ is a symmetric binary relation on V. The edges represent connections between the vertices of the graph, $e_{ij} \in E$ being an edge connecting vertices v_i and v_j.

Definition 2. The *neighbourhood* of a vertex $v \in V$ is $N(v) = \{w : (v, w) \in E\}$. The *degree* $d(v)$ of a vertex v is the number of its neighbours.

Definition 3. Graph $G' = (V', E')$ is a *sub-graph* of G if $V' \subseteq V$, $E' \subseteq E$ and if $e_{ij} \in E'$ then $v_i, v_j \in V'$.

Definition 4. If $W : E \mapsto \mathbb{R}$ is a *weight* function on $G = (V, E)$, then we say that the graph is *weighted* and a w_{ij} weight value corresponds to an edge e_{ij}.

Definition 5. A $G = (V, E)$ graph is *connected*, if there is a path between any two vertices. A *path* is a sequence of vertices in the graph, where neighbouring vertices of the sequence are adjacent in the graph, and a vertex appears only once in the sequence.

Definition 6. C is a *component* of $G = (V, E)$, if C is a sub-graph of G and it is connected. The size of a component is the number of vertices it contains.

Definition 7. A *random geometric graph* (RGG) is obtained as follows. We pick n random node position values as $X_1, X_2, \ldots, X_n \in \mathbb{R}^d$ (according to a probability distribution ν on $\mathbb{R}^d$, where d is the number of dimensions). We connect two nodes v_i and v_j $(i \neq j)$ if their distance $\|X_i - X_j\| < r_n$, the radius of the graph.

The theory of random graphs has an important role in discrete mathematics since the early 60's. Besides the theoretically interesting problems, random graphs have proven to be useful in engineering applications as well. Although real-world datasets are usually too complex to mimic each of their properties with synthetic datasets, some important parameters of their structure can be exposed by analysing random graphs. Famous examples are social networks [13,16] and web graph analysis [19,20].

The network parameters frequently modelled by random graphs are: the probability of the existence of certain edges of the real graph, the degree constraints, or – in case of weighted graphs –, the weights' distribution. After the model is built, some structural patterns get revealed, such as the number or size of components, cliques, or the occurrence of some special sub-graphs.

In our case, random graphs are used to analyse the number and size of components in real graphs. We aim to compare graphs built from test datasets based on a well known phenomenon in random graphs, namely the appearance of the giant component (defined in Section 2.1, Theorem 1). Our test results provide evidence of the existence of a component in these graphs with similar behaviour to the giant component (GC) in random graphs. Besides the properties of the GC in real graphs, we are also interested in the size and number of the components as well.

2.1. The Erdős–Rényi model

Erdős and Rényi analysed the properties of random graphs with uniformly distributed edges [21]. They considered the evolution of components, while adding randomly selected edges to the graph. The process starts with n vertices and 0 edges, and in each step a randomly selected new edge is added, independently of the already chosen edges. After each step, the size and number of components are studied. During the evolution of the graph, connected components start to appear and, when reaching a critical point, they merge into a so called giant component (GC).

The Erdős–Rényi model (ER-model) was originally described by the number of vertices and edges at a given step of the evolution: $G(n, e)$, where n denotes the number of vertices, and e is the number of edges. Recent results connected to this problem are formulated using the number of vertices and the p probability of the existence of an edge $G(n, p)$. If the edges are selected independently, this formulation gives the same result (as the above (n, e) description), and p is usually described as a function of the number of vertices: $p = c/n$, where c is a constant. A complete graph with n vertices has $n(n-1)/2$ edges, that is a $G(n, p = 1/n)$ graph $(c = 1)$ corresponds to the ER-model with $n/2$ edges.

One of the most interesting results of Erdős and Rényi is a theorem [21] that can be formulated as follows:

Theorem 1 *(Erdős–Rényi). The behaviour of the ER graph model can be divided into three important phases, from the point of view of component sizes (where the size of the largest component is denoted by C_{max}):*

1. $c < 1$: $C_{max} = O(\ln n)$ *(the graph contains only small components);*
2. $c = 1$: $C_{max} = O(n^{2/3})$;
3. $c > 1$: $C_{max} = O(n)$ *(the giant component appears), and all other components have size* $O(\ln n)$.

The results presented in [21] also deal with the complexity of the components, but now we are interested in their sizes. The important consequence of this theorem is that after a given number of edges, a unique giant component (GC) appears. Below this