Knowledge and Information Systems

An Overview of Cluster-based Image Search Result Organization: Background, Techniques, and Ongoing Challenges

--Manuscript Draft--

Manuscript Number:	
Full Title:	An Overview of Cluster-based Image Search Result Organization: Background, Techniques, and Ongoing Challenges
Article Type:	Survey Paper
Keywords:	Image Retrieval; Information Retrieval; Image Clustering; Search Result Organization; Spatial Arrangement
Corresponding Author:	Joe Tekli, Ph.D. Lebanese American University Byblos, Mount Lebanon LEBANON
Corresponding Author Secondary Information:	
Corresponding Author's Institution:	Lebanese American University
Corresponding Author's Secondary Institution:	
First Author:	Joe Tekli, Ph.D.
First Author Secondary Information:	
Order of Authors:	Joe Tekli, Ph.D.
Order of Authors Secondary Information:	
Funding Information:	
Abstract:	Digital photos and visual data have become increasingly available, especially on the Web considered as the largest image database to date. However, the value of multimedia content depends on how easy it is to search and manage. Thus, the need to efficiently index, store, and retrieve images is becoming evermore important, particularly on the Web where existing image search and retrieval techniques do not seem to keep pace. Most existing solutions return a large quantity of search results ranked by their relevance to the user query. This can be tedious and time consuming for the user, since the returned results usually contain multiple topics mixed together, and the user cannot be expected to have the time to scroll through the huge result list. A possible solution is to better organize the output information (prior or after query refinement), providing a means to facilitate the assimilation of the search results. In this context, image search result organization (ISRO) has been recently investigated as an effective and efficient solution to improve image retrieval quality on the Web. Most methods in this context exploit image clustering as a methodology capable of topic extraction and rendering semantically more meaningful results to the user. This survey paper provides a concise and comprehensive review of the methods related to cluster-based ISRO on the Web. It is made of four logical parts. First, we provide a glimpse on image information retrieval. Second, we briefly cover the background on ISRO. Third, we describe and categorize the various steps involved in cluster-based ISRO, ranging over: image representation, similarity computation, image clustering or grouping, and cluster-based search result visualization. Fourth, we briefly summarize and discuss ongoing research challenges and future directions, including: high-dimensional feature indexing, joint word-image modelling and implicit semantics, describing images based on aesthetics, automatic similarity metric learning, combining ensemble clustering methods, performing adaptive clustering, allowing dynamic trade-off between clustering quality and efficiency, diversifying image search results, integrating user feedback, and adapting results to mobile devices.

An Overview of Cluster-based Image Search Result Organization: Background, Techniques, and Ongoing Challenges

Joe Tekli [a,b]

Abstract

Digital photos and visual data have become increasingly available, especially on the Web considered as the largest image database to date. However, the value of multimedia content depends on how easy it is to search and manage. Thus, the need to efficiently index, store, and retrieve images is becoming evermore important, particularly on the Web where existing image search and retrieval techniques do not seem to keep pace. Most existing solutions return a large quantity of search results ranked by their relevance to the user query. This can be tedious and time consuming for the user, since the returned results usually contain multiple topics mixed together, and the user cannot be expected to have the time to scroll through the huge result list. A possible solution is to better organize the output information (prior or after query refinement), providing a means to facilitate the assimilation of the search results. In this context, image search result organization (ISRO) has been recently investigated as an effective and efficient solution to improve image retrieval quality on the Web. Most methods in this context exploit image clustering as a methodology capable of topic extraction and rendering semantically more meaningful results to the user. This survey paper provides a concise and comprehensive review of the methods related to cluster-based ISRO on the Web. It is made of four logical parts. First, we provide a glimpse on image information retrieval. Second, we briefly cover the background on ISRO. Third, we describe and categorize the various steps involved in cluster-based ISRO, ranging over: image representation, similarity computation, image clustering or grouping, and cluster-based search result visualization. Fourth, we briefly summarize and discuss ongoing research challenges and future directions, including: high-dimensional feature indexing, joint word-image modelling and implicit semantics, describing images based on aesthetics, automatic similarity metric learning, combining ensemble clustering methods, performing adaptive clustering, allowing dynamic trade-off between clustering quality and efficiency, diversifying image search results, integrating user feedback, and adapting results to mobile devices.

Keywords Image Retrieval . Information Retrieval . Image Clustering . Search Result Organization. Spatial Arrangement

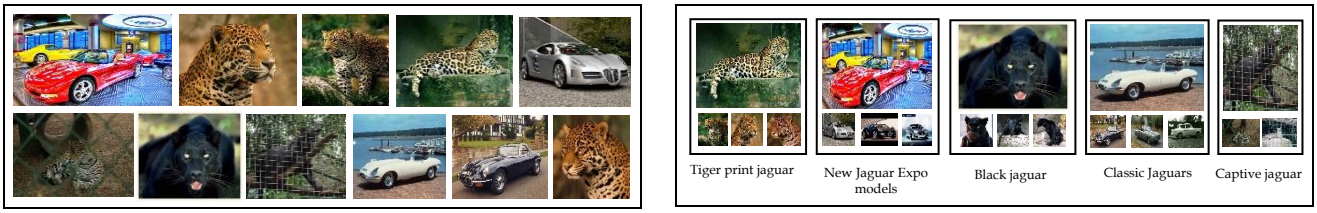
1 Introduction

With the exponential growth of multimedia visual data such as images and videos on the Web, the need for efficient techniques to search and retrieve visual information has become ever more important. Existing Web image search engines (e.g., Google Images[1] and Bing Images[2]) and photo sharing sites (e.g., Flickr[3] and Imgur[4]) return a large quantity of search results, ranked by their relevance to the user query (e.g., Fig. 1.a presents the results of query "Jaguar" submitted to Flickr). This can be tedious and time consuming, since the returned results usually contain multiple topics mixed together, and the user cannot be expected to have the time or patience to scroll through the huge result list. Things become even worse when one topic is overwhelming but is not what the user desires [30]. In addition, users could entirely miss their search goals due to the lack of adapted summarization of the search results [4], leading to information or cognitive overload. Here, it is difficult to decide on the best results to present to the users since we do not know what they are really looking for. One possibility is to help the user reformulate the query by suggesting alternative or more precise search criteria. For instance, query disambiguation and relaxation techniques, e.g., [202, 203], can be used to help narrow down the search result diversity (e.g., suggesting more specific queries such as "Black Jaguar" or "Wild Jaguar", among other alternatives/refinements suggested by Google and Bing for instance). Yet, such techniques do not entirely solve the problem since even the results of refined queries could be difficult to grasp (e.g., refined query "Black Jaguar" produces images of the *black mammal* as well as those of *black cars*).

✉ Joe Tekli
joe.tekli@lau.edu.lb

[a] Associate Professor, E.C.E. Department, School of Engineering, Lebanese American University, 36, Byblos, Lebanon
[b] Adjunct Researcher, Member of the SPIDER research team, LIUPPA Laboratory, University of Pay and Pays Adour, 64600, Anglet, France

[1] https://www.images.google.com [2] www.bing.com/images [3] https://www.Flickr.com [4] https://imgur.com/



a. Sample image search result to query "Jaguar" from Flickr

b. Sample cluster-based ISRO reflecting the main topics of query "Jaguar"

Fig. 1. Examples of clustered Web image search results, adapted from [108]

Another solution is to better organize the output information (prior or after query refinement), providing a means to facilitate the assimilation of the search results by the user. Here, image search result organization (ISRO) has been recently investigated as an effective and efficient solution to improving image retrieval quality on the Web [169, 228]. Most methods in this context exploit image clustering (i.e., identifying groups of mutually similar images) as a methodology capable of topic extraction and rendering semantically more meaningful results to the user [42, 215]. The general assumptions are: i) mutually similar Web images (and documents) tend to be relevant to the same query [46], and ii) semantically similar images (i.e., images with similar underlying topics) tend to be grouped together in some feature space [42] (cf. Fig. 1.b). In other words, identifying the different topics (clusters) of image search results would help the users better understand and navigate the result set, to efficiently identify their search requests.

Nevertheless, performing cluster-based ISRO on the Web faces various challenges, including: i) identifying the most prominent features to assess image similarity (e.g., low-level descriptors, textual annotations, Web links, etc.), e.g., [30, 62], ii) combining multiple feature scores to effectively cluster images (e.g., producing a combined similarity measure, or using separate measures to produce multiple clustering levels), e.g., [30, 169], iii) choosing the best clustering algorithm for the task at hand (e.g., hierarchical, partitioning, spectral, etc.), e.g., [140, 215], iv) identifying and organizing cluster representatives (i.e., cluster visualization and spatial arrangement), e.g., [42, 215], v) performing cluster labeling (i.e., generating textual descriptions to concisely describe the main topic for each cluster), e.g., [140, 228], and vi) evaluating the quality of ISRO methods (i.e., benchmarking, e.g., [90, 91]), among others.

In this paper, we provide a concise review of the methods related to cluster-based ISRO on the Web. The goal of this study is to briefly describe, compare, and categorize the different techniques and methods related to the problem, while illustrating some of the main research challenges and potential directions. To our knowledge, this is the first review study dedicated to cluster-based ISRO, which we hope will foster and guide further research on the subject. The remainder of the paper is organized as follows. Section 2 provides a glimpse on image retrieval. Section 4 briefly describes the background in ISRO. Section 6 reviews and categorizes the literature on cluster-based ISRO techniques, followed by a description of experimental evaluation metrics and test data in Section 5. Sections 6 briefly covers ongoing research challenges and future directions, before concluding the paper in Section 7.

2 A Glimpse on Image Retrieval (ImR)

Many Web-based image retrieval (ImR) systems have been proposed in the literature (cf. survey studies in [47, 95, 114, 124]), and can be roughly categorized as: text-based, content-based, and hybrid methods.

2.1 Text-based versus Content-based ImR

In text-based systems, e.g., [30, 47, 62, 228], images are manually or automatically annotated by text descriptors, which are then used by classic database or text retrieval systems to perform image search [47]. The text-based paradigm has been adopted by most current Web search engines (e.g., Google and Bing) and photo sharing sites (e.g., Flickr and Imgur), due to its well proven scalability in handling the tremendous amounts of images published on the Web. Yet text-based ImR systems are usually characterized by poor result quality, since the search engines are guessing image visual contents using indirect textual clues [228], and are thus usually unable to confirm whether the retrieved images actually contain the desired concepts expressed in

the user queries [62]. In addition, text-based systems usually produce a large quantity of image search results presented in a scrolled list, including multiple underlying topics mixed together. Choosing the desired images from the list is usually tedious and cumbersome to the user, especially when the latter has an abstract or fuzzy search target [30, 228].

In content-based ImR systems, e.g., [43, 124, 126, 188], images are indexed based on their visual content, e.g., color, texture, or shape descriptors, and are consequently processed via search engines specially devised to handle, compute, and compare low-level image feature descriptors (e.g., dominant color, color layout, color and edge histograms, etc.) [124, 126]. Yet, as these descriptors are low-level, they seem effective only in matching images which are almost identical in content [62]. In other words, they seem useful locally, i.e., when applied on a subset of similar images (retrieved a priori through some other means, e.g., using textual or user-provided evidences), but fail when matching relatively disparate images [43, 141]. In addition, low-level features are usually unable to effectively capture the high-level semantic meaning present in images [124], which is known as the *semantic gap* problem [126, 188]: the discrepancy between the descriptive power of low-level image features and the richness of user semantics.

2.2 Hybrid ImR Methods

Various hybrid methods have been developed, integrating both text-based and content-based image processing capabilities. Most methods in this category, e.g., [30, 67, 82, 113, 228], target Web images where both low-level and text-based image clues are available, including: i) the Web links of image files (e.g., URLs) which have a clear hierarchical structure with useful information such as the image Web category [124], as well as ii) Web documents in which images are imbedded (e.g., HTML) including textual metadata, e.g., image title, webpage title, ALT-tag, etc. [30]. Yet, several studies have shown that such metadata can only annotate images to a certain extent (e.g., the image title is usually abbreviated and might be meaningless, the ALT-tag might be missing), and do not utterly solve the *semantic gap* problem [30, 62].

Moving on from traditional low-level image features (e.g., color, shape, and texture features), various studies have investigated *high-level semantics* [47, 124], i.e., deriving semantic descriptions generated automatically based on low-level features. This can be done using different techniques, namely: i) using dedicated ontologies to relate low-level features with high-level concepts (e.g., color ontologies where colors are defined using color names – *red*, *blue*, etc. – linked with numerical representations [137, 166, 192]), ii) using machine learning tools to associate low-level and high-level features using trained classifiers based on sample data provided by experts (e.g., categorizing textures into pre-defined classes – *sea*, *clouds*, *forest*, etc. – based on training numerical spaces [51, 99, 109]), and iii) generating semantic templates to support high-level semantic image retrieval based on low-level features (e.g., retrieval of named events, or of pictures with emotional significance such as "find pictures of a joyful crowd", e.g., [40, 189, 246]). The main premise with this family of hybrid techniques is to try to simulate the visual concept space in terms of lexical concepts as perceived by humans, which remains an inherently complicated task and an ongoing challenge in ImR [47, 59, 124].

2.3 Techniques to Improve ImR Quality

In the past few years, various techniques have been investigated to improve the effectiveness and quality of I_mR systems, ranging over: i) query formulation and refinement, ii) image and object classification, iii) user feedback, and (iv) search result organization (Fig. 2).

Assistance in query formulation: providing suggestions to the users considering their interests and behaviors, in order to perform query refinement and disambiguation [38, 158, 232]. Search requests could be i) narrowed (e.g., query "Mickey Mouse" could be transformed to "Mickey Mouse and Pluto"), ii) expanded (e.g., "Mickey Mouse" → "Disney world"), or simply iii) modified, identifying related queries (e.g., "Mickey Mouse" → "Donald Duck") to better reflect the user's search purpose [38].

Image and object classification: organizing large image collections and digital libraries into predefined categories and providing means for automatic image classification, has been proven central for effective image indexing, browsing, and retrieval [19, 39]. Image classification allows automatic image annotation (i.e., associating predefined labels with images, and performing object detection and recognition) which is central to effective text-based I_mR [71, 252]. It attempts to mirror human

perception in learning the visual appearance of image contents (i.e., identifying meaningful objects), in order to facilitate content-based search [124].

User feedback: allowing the user to interact with the retrieval system by providing information which is relevant to the query [45, 173, 238]. Based on the user's judgments, the system dynamically updates its similarity evaluation model and result sorting functions to give a better approximation of the user's search request. In other words, it brings the user into the retrieval loop to dynamically adapt retrieval results [238].

Most of the above methods exploit supervised learning techniques (e.g. deep neural networks and non-parametric classifiers) [217] which usually require extensive training (large amounts of image training samples and a substantial manual effort) prior to executing the search task [124]. Hence, while efficient on relatively small and static image databases, the latter methods present a serious scalability problem and seem unfit to handle large scale image collections and retrieval tasks on the Web [62, 124].

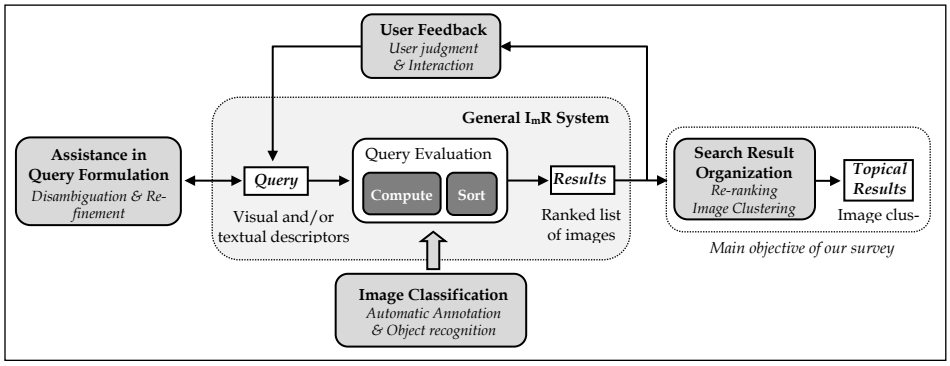


Fig. 2. Outline of I_mR and enhancement techniques

Image search result organization: unlike the above I_mR enhancement techniques based on supervised learning (and involving extensive training), approaches to image search result organization have been gaining importance as simple and efficient solutions to improving I_mR quality on the Web, e.g., [30, 169, 228]. They usually rely on similarity-based clustering, an unsupervised learning paradigm [92] aiming to discover how image search results are organized based solely on the data itself, without any training samples or manual effort [77], thus promoting fast and scalable image retrieval on the Web [67]. We focus on this category of I_mR enhancement techniques in this paper, and further describe them in the following sections.

3 Background on Image Search Result Organization (ISRO)

Transparent to the underlying I_mR system, image search result organization (ISRO) is a search post-processing phase that is introduced to the ImR pipeline to amend: i) *search accuracy*, by re-arranging results to highlight the images which are most relevant to the user [191, 251], ii) *result diversity*, providing a global and diverse view of the result set in order to consider the ambiguous nature of image queries [215, 249], iii) *result visualization*, by grouping together mutually similar images to be presented in a user-friendly manner (e.g., 2-dimentional grid of thumbnails), in comparison with the traditional (1-dimentional) ranked list paradigm [30, 140], and iv) *search speed*, by making users faster in locating a given image or a group of images matching their requirements [168, 169].

While image clustering is arguably the most prominent technique that is used to perform ISRO on the Web [228], other methods have been proposed in the literature. These can be roughly organized in two categories: i) result re-ranking, and ii) similarity-based spatial arrangement.

3.1 Result Re-ranking

Result re-ranking consists in re-organizing the image search results, based on certain visual or semantic criteria, to improve their presentation [191]. The main idea is to re-position images reflecting diverse and more relevant visual contents or textual descriptions in the top raking positions of the result list, in order to provide the user with a better coverage of the search result. Most approaches to image result re-ranking, e.g., [116, 191, 251], analyze the probability distributions of the original image result set in order to identifying salient image characteristics and re-rank images accordingly. In [144], the authors consider manual user refinements in discarding noisy search results and highlighting more relevant images. In [159], the authors consider a pre-defined contrastive class of diversified images, to eliminate near duplicate images from the search result and guide the re-ranking process. A few methods in [171, 199] perform re-ranking through image search result clustering, by i) grouping images in similar clusters, ii) selecting the most representative images from every cluster, and iii) producing a new ranked list by ordering the selected images based on their original ranking and cluster properties. However, while re-ranking methods seem effective in diversifying the search result space [215, 216], their main problem lies in adopting the same ranked list retrieval paradigm used in classic I_mR systems (cf. Section 2.1), which is not always user-friendly and has been shown to be tedious and cumbersome in image search [228]. The problem is particularly aggravated on the Web with the potentially huge size of returned image search results [30]. In other words, users still have to sequentially scan the images – one after the other – to find their request, even after the re-ranking phase [121].

3.2 Spatial Arrangement

Various studies, e.g. [11, 121, 145, 169], have investigated similarity-based spatial arrangement to render the image search result as a set of thumbnails in a similarity-based spatial distribution, such that similar images are positioned closer together. This is based on the premise that (unlike textual documents) the content of an image can be understood at a glance due to its visual nature. The main idea with this family of methods consists in representing inter-object dissimilarities as distances in a high dimensional space, and then approximimg them in a low dimensional (commonly 2D) output configuration. This is achieved using well known dimension reduction techniques such as principal component analysis (PCA), latent semantic analysis (LSA), and multi-dimensional scaling (MDS) [157, 186, 201]. In other words, the similarity matrix between all pairs of images is transformed into a (2-dimensional) configuration of points, where thumbnails of the corresponding images are placed to produce the arrangement. This is done in a way where similar images have their thumbnails placed nearby, such that image dissimilarities are reflected by inter-thumbnail distances (cf. Fig. 3).

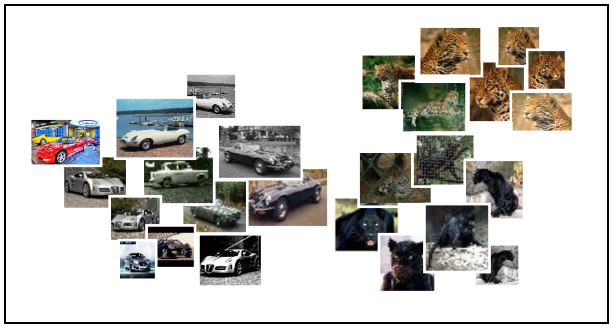


Fig. 3. Example of similarity-based arrangement of image search results for query “Jaguar” from Fig. 1

While similarity-based spatial arrangement seems more user-friendly and efficient in navigating the image search result in comparison with ranked lists, it underlines two main limitations: i) placing similar images next to each other can sometimes cause them to appear to merge, making them less distinctive and ‘eye catching’, where it becomes easier for the user to miss an image all together [169], and ii) thumbnail overlapping is another issue, which makes parts of the images invisible, causing information loss and making it harder for the users to identify their search requests [121]. Overlapping could be resolved by relaxing the similarity distances in order to guarantee the separation of image boundaries. Nonetheless, this is achieved to the expense of accuracy by relaxing/distorting the image similarity/distance relationships [121].

Hence, a technique is required for organizing images based on their mutual similarities (to render more meaningful search results [215]), while simultaneously avoiding both information loss and similarity relationship loss (in comparison with similarity-based spatial arrangement [121]). Both of the above mentioned limitations are addressed with cluster-based ISRO, which we describe in detail in the remainder of this paper.

4 Cluster-based ISRO

Cluster analysis is the organization of a collection of entities (e.g., images) into clusters based on their mutual similarities, such that entities in the same cluster are more similar to each other compared with entities in different clusters [92]. Data clustering has been investigated in a wide spectrum of active research areas, ranging over databases, data-mining, and information retrieval, e.g., [1, 2, 146]. It has been used in two ways: i) standalone, as a separate data processing task, and ii) as a pre-processing step embedded within another data processing task – like data search and retrieval – to improve its performance. In the context of information retrieval, clustering has not been well received as a standalone search paradigm, for two main reasons [73, 93]: i) because it might be too slow for large corpora (clustering the whole document collection for every query can be extremely time-consuming), and ii) because it is not effective in identifying specific search requests (it rather identifies broader search topics). Nonetheless, the task of clustering search results has recently gained importance in information retrieval as a post-processing phase to traditional ranked list search, e.g., [52, 61, 127, 175], where clustering is used for organizing and rendering meaningful search results (cf. Section 4.3.2). The same applies to the domain of I_mR where clustering is mainly used for organizing and visualizing more meaningful image search results (cf. Section 4.3.3).

In the following sub-sections, we describe the various steps and categorize the different techniques involved in cluster-based ISRO, ranging over: i) image representation, ii) similarity computation, iii) image clustering or grouping, and iv) cluster-based search result visualization.

4.1 Image Representation

Unlike classic image retrieval from a fixed database where each image is treated as an independent entity [47], image retrieval on the Web deals with integrated image objects, each contained within its host webpage which could underline a great deal of relevant information about the image itself. In general, the content of Web images is more or less related to the content of their host pages [129]. For instance, a photo of a *Jaguar animal* is more likely to be found in a page talking about wild animals in the Americas, whereas a photo of a *Jaguar car* is likely to appear in a page about luxury European cars. Therefore, Web images could be described not only by their own visual features and textual annotations, but also by their related Web information, which could be exploited to allow more effective indexing, retrieval, and clustering [79].

In short, a Web image can be described in four dimensions: i) visual feature based, ii) textual feature based, iii) link graph based, and iv) region-level based representations.

4.1.1 Low-level Visual Features

Visual feature representation is the basis for content-based I_mR. A typical content-based I_mR system views each image as a collection of low-level visual features, and evaluates the relevance between images w.r.t.[1] their feature similarity [124]. Visual features can be grouped in three main categories: i) color, ii) texture, and iii) shape.

Color descriptors are used to represent the colors present in an image. Different color spaces exist in the literature such as CIE XYZ which attempts to produce a color model based on human eye color perception. Other color spaces include CIE RGB and CIELAB [47, 124]. Various color descriptors have also been proposed including color moments, color histogram, color correlogram, color coherence vector, etc. [123, 132, 192]. The MPEG-7 multimedia metadata description standard has integrated additional descriptors such as dominant color, scalable color, and color layout [133]. The use of color features usually depends on the nature of the images at hand. For instance, for images which do not have an overall homogeneous color, the average or dominant color descriptors might not be very useful. On the other hand, domain knowledge such as color variance and color distribution over all images can be exploited to dynamically assign weights to image pixels [86], allowing to better compute color features. Color descriptors are most commonly used since they are relatively easy to process (compared with

1 With respect to

texture and shape features) and produce good enough results [47]. **Texture descriptors** are intended to capture the granularity and repetitive patterns of surfaces within in an image. They usually consist of spectral features, such as Gabor filtering [134] and wavelet transform [226], as well as statistical features such as the Wold features [118] and Tamura descriptors [198] (which are employed in MPEG-7). However, texture features are not as commonly used as their color counterparts, since they are more easily affected by image distortions and noise [124], and have been proven less effective on images where textures are not very structured and homogeneous (e.g., pictures of natural scenery) [112]. In [243], the authors propose a combined descriptor called Color Texture Moments (CTM), to integrate both color and texture characteristics in a compact form (using color moments and a Fourier transform based texture representation). Experimental results in [243] underline CTM's good performance w.r.t. its classic texture counterparts. **Shape descriptors** allow detecting different shapes and small salient objects in an image, and have been shown to be useful in many applications (especially when dealing with images of synthetic and man-made objects [124]). Shape descriptors include aspect ratio, circularity, Fourier descriptors, consecutive boundary segments, etc. [136]. MPEG-7 has adopted three main descriptors: 3-D shape descriptor derived from 3-D meshes of shape surface, region-based descriptor derived from Zernik moments, and the contour-based descriptor derived from the curvature scale space [133]. However, compared with color and texture, shape features are not so well defined and are not as commonly used [124], and remain relatively marginalized in many I_mR systems, e.g., [86, 137, 184, 192].

4.1.2 High-level Text-based Features

While low-level visual features have been proven efficient in content-based I_mR [124], it is argued that the meaning (i.e., the semantics) of an image remains rarely self-evident [47]. Images which visual features are highly similar to the query image may be very different from the query in terms of user interpretation and intended meaning. That is because human observers do not typically perceive an image in terms of pixel distributions, color patches, or surface features, but rather tend to evaluate images at a higher semantic level, using lexical concepts (e.g., words or expressions) describing salient visual concepts in the image [22, 42]. Hence, a dedicated set of descriptors have been utilized to describe Web images, often referred to as: *high-level features* [47, 114], designating the textual content of the image. Textual descriptors include **tags:** which describe who and how many people are found in a given picture, **place:** label (name) of place where an image was taken, which can be utilized to allow geo-address comparison (using a geo-referenced ontology assigning geographic coordinates with place names [208]), **caption:** title of the image which is usually the most descriptive user-provided textual feature, providing a direct clue to the meaning and context of the image, **comments:** allowing a much larger variety of textual descriptions compared with the previous features, and they are especially useful when captions have not been provided by the user (publisher), and more recently **hash-tags** in social media image posts: adding descriptions provided by the user's online social community. The textual descriptors are then processed for feature representation, including word, phrase, sentence, and document level representations. These range over lexical form (origin of the term), semantic meaning (concept in a reference dictionary), part-of-speech tags (grammar category of the term), n-gram (word associations), syntactic structure (parse tree), and statistical features (e.g., contextual and co-occurrence term frequencies) [60, 190]. The features are subsequently represented as (one or multiple) high-level feature vector(s), where vector weights are computed using legacy term scoring techniques developed in information retrieval[1]. While high-level features attempt to describe the semantics of the image (e.g., who, where, what, etc.) [124], nonetheless, their semantic descriptiveness depends on the quality of the surrounding text portraying the meaning of the image.

Few methods have been recently suggested to generate/semi-automatically enrich the textual descriptions of Web images, using techniques such as probabilistic user-based image tagging (using the tagging logs from the histories of similar users to infer new tags, e.g., [153, 174]), and semi-supervised image annotation based on visual and Web contents (training different machine learning algorithms to annotate new images based on a training image set with predefined labels, e.g., [130, 131]). While promising, yet the latter techniques require training data and training time, which are not always available.

1 The standard *TF-IDF* (*Term Frequency – Inverse Document Frequency*) approach (or one of its variants) from the vector space model [176] is usually used, describing the number of times a term appears in a high-level feature (*TF*) compared with the number of times it appears in all entries of the feature (*IDF*).

4.1.3 Link Graph-based Features

Another kind of information available with Web images is link structure. While textual and visual features reflect the semantics of a single image, link structure reflects the semantic relationships between images [28, 104]. Previous work in Web search has demonstrated that link structure analysis is very effective in identifying relevant webpages [28]. It is based on the premise that a link from page *p* to page *q* is considered as an endorsement of *q* by *p*, and as some form of positive judgment by *p* of *q*'s content. Hence, sophisticated algorithms such as PageRank [28] and HITS [104] have been developed to analyze link structures in order to rank webpages (and are utilized in current search engines such as Google [193] and Yahoo [242]). In short, a document is considered more relevant if it links many "good" pages, and many "good" pages link it. Similar ideas have been applied to Web image retrieval [30, 79, 110], where images are evaluated considering their containing webpages. As a result, an image graph is constructed based on the fact that every image is contained in at least one page, such that the jump from image *i* to image *j* starts from the page containing image *i*, and ends at the page containing image *j*. Once the image graph is constructed, image vector representations can be obtained using spectral graph theory [44, 110], by extracting information contained in the eigenvectors and eigenvalues of the image graph matrix, similarly to traditional Web search approaches, e.g., [28, 48, 80]. However, a single webpage generally contains multiple images, usually describing different semantic topics [32], where images contained in different semantic blocks are likely related to different topics in different pages [79]. As a result, performing effective image link analysis might require decomposing a webpage into finer information units, e.g., page regions or blocks, to correctly analyze the connections between their images.

4.1.4 Region Level based Features

The authors in [32] argue that most webpages contain multiple semantic meanings, with each semantic region (block) linking to different regions in different pages. Hence, they introduce the notion of block-level link analysis to partition each page into different semantic regions and extract link information accordingly. The same approach is consequently extended toward image link analysis on the Web [30, 79]. The main assumptions are: i) images contained in the same region are likely to be related to the same topic, and ii) images contained in pages that are co-cited by a certain region are likely related to the same topic. First, a VIsion based Page Segmentation algorithm (VIPS) [31] is used to extract the semantic structure of a webpage based on its visual representation. It exploits the DOM [233] structure of the HTML document and its page layout features to construct the corresponding semantic tree [183], where each node represents a distinct semantic region (block) in the webpage. As a result, page-to-region *PR*, region-to-page *RP*, and region-to-image *RI* affinity matrices are generated to respectively reflect region containment in pages, region links to pages, and image containment in regions. An image graph is subsequently constructed by multiplying the affinity matrices in order to combine the link and containment relations between pages, regions, and images. Here, image graph construction is based on the fact that every image is contained in at least one region, such that the jump from image *i* to image *j* starts from the page containing the region encompassing image *i*, and ends at the page containing the region encompassing image *j* (in contrast with the page-based link analysis [28, 110] where only atomic pages are considered in the image Web graph construction process). Once the image graph is constructed, image vector representations can be obtained using spectral graph theory [44], similarly to classic page-based link analysis and traditional Web search approaches, e.g., [28, 48, 80]. Despite its usefulness in describing image semantics, a major problem with region-based analysis is identifying semantically meaningful regions. This requires performing automatic segmentation of webpages into regions or blocks describing distinctive semantic topics [31], which is not a trivial task and depends on the quality of the page segmentation techniques used [19, 94]. In other words, region-based image graphs and corresponding vector representations become useful only when the extracted regions can properly isolate webpage semantic contents.

4.2 Image Similarity Computation

After extracting the image features, the next step is to perform feature similarity computation, which is needed to conduct more sophisticated tasks including image clustering and similarity-based retrieval. Image feature similarity can be evaluated separately for every feature representing a different aspect of the image (e.g., visual, textual, linkage), following a feature-specific vector representation (e.g., scalar, histogram, matrix), and using an adapted similarity evaluation method (e.g., Jaccard coefficient, cosine similarity, Minkowski distance, InfoSimba measure). Table 1 presents some of the most common similarity measures used to compare image feature vectors.

Table 1. Common similarity measures between vectors [68, 119, 214]

$$Sim_{Jaccard}(Q,D) = \frac{\sum_{r=1..M} w_Q(n_r) \times w_D(n_r)}{\sum_{r=1..M} w_Q(n_r)^2 + \sum_{r=1..M} w_D(n_r)^2 - \sum_{r=1..M} w_Q(n_r) \times w_D(n_r)} \in [0, 1]$$

where Q and D are the image feature vectors being compared, and M is the total number of dimensions in the common feature space

a. Jaccard coefficient

$$Sim_{Dice}(Q,D) = \frac{2 \times \sum_{r=1..M} w_Q(n_r) \times w_D(n_r)}{\sum_{r=1..M} w_Q(n_r)^2 + \sum_{r=1..M} w_D(n_r)^2} \in [0, 1]$$

b. Dice coefficient

$$Sim_{Cosine}(Q,D) = \frac{\sum_{r=1..M} w_Q(n_r) \times w_D(n_r)}{\sqrt{\sum_{r=1..M} w_Q(n_r)^2 \times \sum_{r=1..M} w_D(n_r)^2}} \in [-1, 1]$$

c. Cosine measure

$$Sim_{PCC}(Q,D) = \frac{\sum_{r=1..M} \left(w_Q(n_r) - \bar{q}\right) \times \left(w_D(n_r) - \bar{d}\right)}{\sqrt{\sum_{r=1..M} \left(w_Q(n_r) - \bar{q}\right)^2 \times \sum_{r=1..M} \left(w_D(n_r) - \bar{d}\right)^2}} \in [-1, 1]$$

where $\bar{q}$ ($\bar{d}$) is the average of the values of vector Q (respectively D)

d. Pearson correlation coefficient (PCC)

$$Sim_{Minkowski}(Q, D) = \frac{1}{1+Dist(Q, D)} \in]0, 1] \quad \text{having}$$

$$Dist(Q, D) = \left(\sum_{r=1..M} \left(w_Q(n_r) - w_D(n_r)\right)^p \right)^{1/p} \in [0, \infty[$$

e. Minkowski distance measure (generalizing Euclidian distance when p=2).

$$Sim_{InfoSimba}(Q, D) = \frac{1}{p^2} \sum_{r=1}^{p} \sum_{s=1}^{p} w_Q(n_r) \times w_D(n_s) \times Sim(n_r, n_s) \in [0,1]$$

where p represents the most representative vector dimensions, and $Sim(D, Q) \in [0, 1]$ can be any similarity measure comparing dimensions n_r and n_s

f. InfoSimba measure

Nonetheless, it is often desired to combine several features simultaneously, in order to benefit from their collective description of the image, which is not a trivial task. Various techniques have been devised to aggregate multiple feature similarities into a single result. They can be organized in various categories such as *means*, *triangular norms*, and *conorms*, among others [20, 202, 206]. In the context of image similarity, most aggregation methods focus on linear combination functions which might not sufficiently explore the inter-dependencies between individual feature similarities [55, 56]. Here, we distinguish between *linear fixed*, *linear adaptive*, and *non-linear* methods [6] as shown in Table 2. Linear fixed methods are most commonly used and include *maximum/minimum* (selecting the largest/smallest similarity of any feature), *weighted sum* (considering relative weights highlighting the importance of each feature similarity, cf. Table 2.a), and *average* (a special case of weighted sum where all feature weights are identical). Linear adapted methods aim to make the feature weight values more adaptable, where the weights are determined through a learning process. Here, the aggregation function comes down to an optimization problem where the weights are chosen to maximize overall image similarity. This can be solved using a number of known techniques that apply linear programming or machine learning in order to identify the best weights for a given problem class, e.g., [12, 13, 85]. Yet, one major limitation of the latter techniques is the time complexity and training time required to compute the weights. In an attempt to solve this problem while allowing dynamic feature weighting, the authors in [214] suggest computing the variance of all image similarity features in the set of retrieved image results, to be used as a weighting and normalization factor in computing aggregate similarity (cf. Table 2.b). In other words, when the variance of a certain feature is small, the images in

the result set would resemble each other in terms of that feature, which make it an importance feature for those images [214]. This brings image similarities according to different features to a similar range, while assigning larger weights to features that are better at evaluating similarity.

Nonetheless, the authors in [6] have criticized both linear fixed and adaptive methods for using linear combination functions and not sufficiently exploring the interdependencies between individual features. In an attempt to address this problem, they suggest using a non-linear combination method (cf. Table 2.c), restricted to the second order since the feature similarity values range between 0 and 1. The first term of the function presents the aggregated similarity (computed as a weighted sum) without considering the interdependencies between individual feature similarities, while the second term presents feature interdependencies (computed as the sum of the pair-wise feature similarity multiplications). The two terms are either added or subtracted depending on the linear similarity value: i) the two terms are added when achieving higher pair-wise similarities between their individual features, indicating that the images are more likely to be similar, and ii) the terms are subtracted in the case of low pair-wise feature similarity where images are not likely to be similar [6]. In other words, the nonlinear aggregation function acts like a contrast filter, amplifying the feature similarities having a higher impact of overall image similarity, while minimizing the other features.

Table 2. Common similarity aggregation measures [6, 68]

$$Sim(Q, D) = \sum_{f \in F} w_f \times Sim_f(Q, D) \quad \in [0, 1]$$ where F is the set of image features, $w_f \in [0, 1]$ the weight of a feature $f \in F$ having $\sum w_f = 1$ and $Sim_f(Q, D) \in [0, 1]$ the similarity between image vectors Q and D following feature f **a.** Linear fixed: weighted sum function
$$Sim(Q, D) \frac{1}{1+Dist(Q, D)} \quad \in]0, 1] \quad \text{having}$$ $$Dist(Q, D) = \frac{1}{f} \sum_{f \in F} \frac{1}{\nabla_f^2} \times Dist_f(Q, D) \quad \in [0, \infty[$$ where ∇_f^2 is the variance according to the f-th feature over all image vectors in the result set. **b.** Linear adaptive: dynamic feature weighting function
$$Sim(Q, D) = \sum_{f \in F} w_f \times Sim_f(Q, D) \pm \sum_{f \in F} \sum_{h \in F} Sim_f(Q, D) \times Sim_h(Q, D) \quad \in [0, 1]$$ **c.** Non-linear aggregation function

Note that identifying the most prominent features to assess image similarity, and choosing the most suitable aggregation function are ongoing challenges in ImR, and depend on user preferences and on the desired properties of the similarity-based application, namely image clustering.

4.3 Image Clustering

Clustering techniques are often utilized to partition entire datasets of images into homogenous clusters grouping similar images together, and therefore are not necessarily suitable for browsing image search results. To perform image search result clustering (ISRC), three main issues need to be considered [29]: i) the algorithm should take as input document snippets (e.g., image feature vectors) instead of the whole documents (e.g., raw images), since the processing of whole images is time consuming, ii) the clustering algorithm should be fast enough for online calculation, since it will run right after initial query execution, and iii) the generated clusters should have representative descriptions (visual or textual) for quick browsing by the user. In the remainder of this section, we first provide a quick overview of legacy data clustering algorithms in Section 4.3.1. We then briefly describe generic search result clustering approaches in Section 4.3.2. We finally describe and categorize ISRC solutions in Section 4.3.3.

4.3.1 Legacy Data Clustering

Data clustering algorithms can be organized in three main categories: i) partitional, ii) hierarchical, and iii) other methods.

Partitional clustering algorithms attempt to divide data objects (e.g., images) into non-overlapping subsets, i.e., the clusters, such that each data object is in exactly one cluster, by maximizing intra-cluster similarity and minimizing inter-cluster similarity. K-means [125] is one of the most popular algorithms in this category and attempts to recursively minimize the distance between objects in a cluster and a special object designated as the center of the cluster (computed as the average between all objects in the cluster). The clusters are re-computed and adjusted recursively until reaching convergence (where the cluster centers remain unchanged). Similar algorithms such as k-medians, k-medoids, and BSAS (Basic Sequential Algorithmic Scheme) have also been suggested in the literature [27, 155, 220]. On the one hand, partitional algorithms are usually: i) intuitive, ii) easy to implement, and iii) relatively efficient (e.g., k-means is of average $O(n \times k \times i)$ time where n is the number of objects, k is the number of clusters, and i is number of iterations). On the other hand, i) they usually require the user to specify the number of clusters k which is not always known in advance (especially in search result organization), ii) results depend on the initial positions of the cluster centers which are mostly chosen randomly, iii) they are not suitable for identifying clusters of different sizes and densities, and iv) are usually unable to handle noise and outliers. A few algorithms such as x-means [156], affinity propagation [64], and YAK [224] have attempted to counter some of the above limitations, by i) sweeping the search space to compute appropriate statistical functions (e.g., Bayesian information criterion [15]) or energy-based functions (e.g., Bethe free-energy approximation [163]), ii) using the latter results to identify the required number of clusters and their centroids, and then iii) running a traditional partitional process to produce the clusters. Yet, their usage remains limited in the literature.

Hierarchical clustering algorithms generate a set of nested clusters organized in a hierarchy, called *dendrogram*, where the root node of the dendrogram represents the whole dataset and each leaf node represents an individual data object. The cluster hierarchy is produced based on the similarity between individual data objects/clusters. Here, we distinguish between two types of hierarchical clustering solutions: *agglomerative* and *divisive*. Agglomerative clustering starts with each object forming its own cluster, and then finds the best pair to merge into a new cluster, recursively repeating this process until all clusters are fused together. Divisive clustering starts with all the data objects forming a single cluster. It then considers the best way to divide the cluster into two, and recursively repeats the same process on both sides until all clusters are split into individual objects. A stopping rule is used to evaluate the quality and properties of the clusters (e.g., their number, sizes, shapes, inter and intra-cluster similarities, etc., [23, 139, 177]) in order to stop the hierarchical process and determine the best set of output clusters from the dendrogram. On the one hand, hierarchical clustering algorithms: i) usually produce better results compared with their partitional counterparts, ii) do not require the users to specify the number of clusters in advance, and iii) produce a dendrogram structure which describes the clustering process and maps nicely to human intuition. On the other hand: i) they are usually more computationally complex than their partitional counterparts (requiring at least $O(n^2 \times \log(n))$ time where n is number of data objects being clustered), and ii) they tend to break down large clusters into smaller (more homogeneous) ones which might not be always favorable from the user/application's side.

Other clustering approaches include *incremental*, *density-based*, *spectral*, and *fuzzy* clustering algorithms. Incremental clustering considers a stream of data objects (e.g., images) where each data object is processed one at a time and objects are assigned sequentially (as they arrive) to existing clusters. The first object is placed in its own cluster. Then, the next object is compared with the existing cluster and the algorithm decides if the new object should be placed in the same cluster or if a new cluster should be created around it. The process continues in the same manner until all objects have been clustered [36, 74, 254]. While intuitive and seemingly more efficient than partitional algorithms, incremental clustering methods usually produce lower quality results since they depend on the order following which the objects are being processed and clustered [36]. Density-based clustering groups objects that are closely packed together with many nearby neighbors, and marks as outliers objects that lie alone, resulting in clusters that form high-density regions separated by low-density ones. Objects with high-density neighborhoods are identified as core objects, given user chosen neighborhood size and density threshold parameters. Then, objects closely associated with the latter are included in their clusters, while remaining objects are dismissed as low-density noise [35, 181]. While density-based methods can effectively handle noise and outliers given a careful tuning of neighborhood and density parameters, yet they tend to disregard low-density regions all together and thus might disregard relevant data

partitions which occur in low-density regions [34, 107]. Spectral approaches use the spectrum (eigenvalues) of the dataset's pair-wise similarity matrix in order to perform dimension reduction, before running the clustering process (partitional, hierarchical, or other) in the reduced dimensional space. Various approaches such as principal component analysis (PCA) and latent semantic analysis (LSA) can be used to perform dimension reduction [78] [117, 122]. While the reduced feature space is possibly more descriptive than the original object representation, yet it consists of algebraic dimensions (implicit concepts) which are not readable by human users, producing results which might not meet the user's needs [201, 241].

Note that most previously mentioned clustering techniques produce *hard clusters*, where every object is assigned to one single cluster only, such that all clusters are separate and do not intersect. In contrast to the latter paradigm, fuzzy (or soft) clustering allows to associate an object with more than one cluster, producing soft clusters where each object (e.g., image) has a variable degree of membership in each of the output clusters. Notable fuzzy clustering algorithms are fuzzy c-means: a soft adaptation of the k-means partitional algorithm, and FLAME: a more recent density-based fuzzy clustering solution [65].

4.3.2 Search Result Clustering

Search result clustering, also referred to as ephemeral clustering, is the task of discovering clusters from a set of web search results retrieved for a given query, in order to improve the organization and presentation of the query search results e.g., [46, 81, 111, 248]. Search result clustering approaches are designed to deal with small datasets to perform efficient online query post-processing. The search result clusters are presented to the user following query execution, and are then disregarded with the query and do not provide information for future results. A few search engines such as Yippy[1], Carrot[2], and iBoogie[3] support search result clustering through a dynamic generation of query result clustering structures. Most existing approaches in this context follow the same overall process consisting of four main phases: i) search result fetching, ii) result parsing, iii) feature representation, and iv) result clustering. The first phase consists in acquiring the results obtained from a certain Web search engine. The results mainly consist of webpage titles and query-dependent snippets. The latter are assumed to be informative enough to perform result clustering (since they are specifically returned by the search engine to facilitate the user's relevance judgment when manually processing the search results). The second phase consists in parsing the titles and snippets, performing: tokenization, stop-word removal, and stemming (or lemmatization through a semantic knowledge base). The third phase converts the parsed results into feature representations, which are then utilized to perform result clustering in the fourth and final phase. In this context, most search result clustering approaches are distinguished based on: i) the choice or combination of feature representations, and ii) in the underlying clustering algorithms that they adopt.

In [247], the authors introduce the suffix tree clustering algorithm, an adaptation of hierarchical clustering which considers the n-grams of any given length to be inserted into a string-based representation of the textual data, while allowing differing strings to be clustered incrementally in linear order. While the approach in [247] requires linear time complexity, it tends to produce a large number of clusters which can be detrimental to search result organization [52]. In [37], the authors extend the approach in [247] by introducing a criterion to measure the concordance of objects contained in two candidate clusters. The latter is applied on the result of the clustering process, as an optimization phase to decide whether to merge (or not) the candidate clusters based on their mutual information and the average information content of their constituent objects. Results in [37] show improved clustering quality (producing a lesser number of clusters) compared with [247]. In [150], the authors extract frequent phrases from the input text based on suffix-arrays. Then, they apply a spectral clustering approach, performing term-document matrix factorization to discover latent structures of implicit concepts, before matching the textual features with the extracted concepts and grouping them into relevant clusters. In [52], the authors introduce a hierarchical divisive clustering solution, recursively splitting a set of Web snippets based on a variant of the k-means algorithm, combined with a dedicated similarity measure for comparing Web snippets. The approach includes a cluster-labeling step, producing a small set of representative terms describing the produced clusters. In [179], the authors propose to augment the syntactic text representation of documents with a collaborative knowledge-based representation in the form of semantic graphs using a Wikipedia-based annotator [221]. The produced semantic graph representations are then processed through an adapted spectral clustering process, performing term-graph factorization before grouping the transformed graph representations into relevant clusters. Results in [142] show that the method in [52] produced improved results compared with its counterparts in the literature, and it was adapted in [142] to perform ISRC.

[1] http://www.yippy.com [2] http://carrot2.org [3] http://www.iboogie.com

4.3.3 Image Search Result Clustering (ISRC)

A number of approaches have been recently developed to perform ISRC (cf. Table 3). They accept as input the search results produced by a typical Web or image search engine, and then produce as output a clustered organization of the image search results. They can be organized in two main categories: i) single-step and ii) multi-step result processing.

4.3.3.1 Single-step Approaches

Single-step approaches perform a single clustering process on the image search results, applied on one category of image features, namely visual-only features. Methods in this category usually utilize: i) partitional or ii) spectral clustering techniques.

Partitional approaches: the authors in [231] consider as input the images returned by an image search engine (e.g., Google images). They perform region-based image analysis to extract visual codewords (i.e., region-based visual feature representations), and rank the extracted visual codewords based on a regression model learned from human labelled training data. Consequently, they utilize an adaptation of k-means partitional clustering to group images sharing similar visual codewords together. In [119], the authors develop a color-based ISRC approach using a combination of color histogram and color distribution entropy descriptors introduced in [194]. The algorithm starts by selecting an image from the search results as a cluster seed. It measures the similarity between the selected image and every other image, and adds the compared images to the seed image's cluster if their similarities are above a certain threshold, forming new clusters around the images which similarities are below the threshold. The process is repeated until all images are clustered. In [224], the authors describe a text-based approach introducing the YAK algorithm (i.e., Yet Another K-means), designed to overcome k-means' need to specify the number of clusters. YAK decides on the number of clusters computationally, using image similarity and cluster merge thresholds which are defined statistically based on the image search results. After identifying the first centroid randomly, images whose similarities with the centroid are above the threshold are associated with its cluster, and others with similarities below the threshold form the centroids of new clusters. The process is repeated iteratively until all images in the search result have been processed. The resulting clusters are then considered for merging according to a merge threshold, to produce the final clusters.

Aiming to improve the diversity of image search results, the authors in [214] introduce three lightweight partitional clustering algorithms, called *folding*, *maxmin*, and *reciprocal election*. They consider multiple visual features, combined through dynamic weighting (cf. Table 2.b) to capture the visually discriminative aspects of the retrieved images. The folding algorithm first selects a set of cluster representative images from the ranked list of image search results, and forms clusters around the representative images by associating each image with its most similar representative (i.e., using the nearest neighbor rule). The first image from the ranked result list is chosen as the first representative. Then, the remaining image representatives are chosen from the ranked list to be dissimilar enough from the already selected one(s), following a dynamically computed distance threshold (a fuzzy version of the folding algorithm is described in [4], where images are associated general membership degrees w.r.t. every representative, such that an image is assigned to the representative with the maximum membership degree). The maxmin algorithm differs from folding in its cluster representative selection process. It disregards result ranking in the search list and chooses the first representative at random. Then, it selects the remaining representatives as the ones having the largest distances (i.e., minimum similarities) from the already selected one(s). Reciprocal selection interleaves the processes of representative selection and cluster formation. All images cast votes for each other, in the form of reciprocal ranks computed based on visual feature similarity, such that the votes an image receives determine its chances of being elected as representative. When the image with the highest votes is elected to become the first representative, the cluster around it is directly formed by inserting those images that have it in the top of their voting lists. Images in the formed cluster are excluded from the result list, and the process is repeated until every image is selected as representative or is assigned to a cluster. Empirical results show that reciprocal election outperforms its counterparts by improving result organization and diversity, which is attributed to its stability (i.e., minimizing the variation of cluster results) [4].

Spectral approaches: In [63], the authors develop a spectral analysis approach for learning image categories from web search images. They introduce a variation of probabilistic latent semantic analysis (PLSA) [84] applied on a region-based spatial representation of the image search results retrieved from Google Images. The latter are combined with the image textual labels returned by the search engine to perform label-image matrix factorization and implicit concept extraction, before associating the extracted concepts with the corresponding label categories. Another spectral approach in introduced in [42] which starts by querying images based on their visual feature similarity with the user query, and then comparing all the retrieved

images among each other to produce an image search result affinity matrix. The normalized cut spectral clustering algorithm [186] is run on the graph representation of the affinity matrix to identify the image result clusters. A similar spectral approach is described in [68] where the authors utilize kernel principal component analysis (PCA) clustering [180] applied on the image search results returned by Google Images, to group together images sharing similar visual features. Given user-provided constraints (e.g., number of returned results, and dimension size of the kernel matrix), the kernel learning algorithm is run incrementally to update and better approximate the latent image representations, which are then partitioned into multiple visual categories according to the computed kernel matrix.

***Discussion*:** To sum up, single-step clustering methods perform a single clustering process on the image search results, using adaptations of partitional or spectral clustering algorithms. Most methods rely solely on the images' visual features and are usually unable to capture their semantic meaning. Most partitional approaches do not describe how the number of clusters is computed, or randomly select the initial seed images to build the clusters (except for folding and reciprocal election algorithms in [194] where representative images are chosen based on their rankings in the result list). Also, most spectral approaches do not discuss how to pick the number of implicit concepts (i.e., latent dimensions), or how to identify the most informative ones to perform the clustering task [142]. Note that the latter limitations are usually shared with generic partitional and spectral algorithms (cf. Section 4.3.1).

4.3.3.2 Multi-step Approaches

Multi-step approaches run multiple clustering or supervised learning processes on the image search result, where every process is usually applied on a different image feature representation, aiming to leverage the descriptiveness of multiple image features in improving search result organization. Methods in this category mainly utilize i) partitional, ii) hierarchical, or iii) spectral clustering techniques.

Partitional approaches: In [228], the authors develop a two-step text-based processing approach combining clustering with supervised regression analysis. After running the user's original query through a Web search engine, the retrieved search results – consisting of the page titles and their metadata – are concatenated to form a ranked list of textual strings. The strings are then processed using a textual analysis approach based on the salient phrase ranking paradigm [250], producing clusters of similar text phrases using suffix tree clustering [247]. The latter are fed into a supervised regression model learned from human-labeled training data in order to produce a list of candidate cluster labels consisting of individual phrases. The cluster labels are then run through an image search engine (e.g., Google Images), to produce labelled image clusters. A similar approach is described in [258], where the authors integrate visual and textual features using two regression models (i.e., linear multi-variable regression and non-linear support vector regression) as a first step, and then run the affinity propagation algorithm [64] on the combined feature representations – as a second step to produce the image clusters. In [82], the authors introduce a two-step pre-processing approach for image annotation based on image search results. They consider visual features, image tags, and photo-taking metadata in evaluating image similarity. A k-nearest neighbor method is first run on every image in the search result to identify the tags in its neighborhood. The images are then clustered using the maxmin partitional algorithm [214] (cf. Section 4.3.3), and the resulting clusters are processed to filter-out noisy image tags. The resulting image tags are finally recommended as input to the search result clustering process.

Hierarchical approaches: In [256], the authors introduce a three-step agglomerative hierarchical clustering approach considering image metadata (URL of the containing webpage and image anchor text), textual context (surrounding text in the containing webpage), and semantics (matching concepts from the Wikipedia knowledge base). The approach starts by pre-processing the image metadata and textual context, and performs semantic disambiguation [201] to match key terms and phrases with the Wikipedia knowledge base [33]. Then, agglomerative clustering is run in three consecutive steps. The first step clusters the input images based on the concepts extracted from their metadata. The second step accepts as input the clusters produced in the first step and merges them based on the concepts extracted from the images' textual contexts. The third step accepts as input the clusters produced in the second step and expands the context of each cluster in order to merge the ones sharing the most similar contexts. The top concepts in each cluster are then used to represent cluster semantics. A similar hierarchical approach is described in [83] where the authors consider labels added using social tagging, photo taking metadata, and low-

level image features. Constrained agglomerative clustering with must-link constraints is utilized to process the different features. In [4], the authors introduce the folding tree hierarchical algorithm (inspired by the folding algorithm from [214], cf. Section 4.3.3). It starts from the individual images which represent leaf nodes, and merges the most similar ones based on their visual features, to form the first level clusters represented as inner tree nodes. Tree traversal is then performed level-by-level from the leaves to the root. In the next levels, the cluster of each subtree is checked before merging with other clusters or images, and the process continues until it reaches the root of the tree. The main clustering step is followed by three consecutive ones: i) fine-tuning highly dispersed clusters by identifying the farthest images in every cluster and verifying whether they should be assigned to a different one based on image/cluster similarity, ii) merging of small clusters following a certain predefined population average size, and iii) eliminating small clusters that cannot be merged given their dissimilarity from all other clusters.

Table 3. Comparing image search result clustering solutions

Processing	Clustering	Approach	Features	Characteristics
Single-step	Partitional	Wang X. et al. [231]	Visual	- Ranks visual codewords based on trained regression model - Uses adaptation of k-means
		Liu G. and Lee B. [119]	Visual	- Uses color features from [194] - Introduces iterative similarity-based clustering algorithm
		Van Leuken R. et al. [214]	Visual	- Introduces three algorithms: folding, maxmin, and reciprocal election - Uses dynamic weighting to combine multiple features - Reciprocal analysis outperforms its counterparts
		Alamdar F. and Keyvanpour M. [4]	Visual	- Introduces fuzzy folding algorithm (a fuzzified version of the folding algorithm from [214])
		Wang H. et al. [224]	Textual	- Introduces the YAK algorithm (upgrade of k-means) - Determines the number of clusters statistically
	Spectral	Chen Y. et al. [42]	Visual	- Produces image search result affinity matrix and graph - Runs normalized cut spectral clustering [186] on the matrix graph
		Gao Y. et al. [68]	Visual	- Uses kernel PCA clustering [180] - Considers user-provided constraints (e.g., number of returned results, and dimension size of the kernel matrix)
		Fergus R. et al. [63]	Textual	- Uses PLSA variation [84] for image-label matrix factorization - Associates implicit concepts with label categories
Multi-step	Partitional	Wang S. et al [228]	Textual	- Uses a two-step process: i) performs suffix tree clustering [247], ii) performs supervised regression analysis on the clusters to produce cluster labels - Performs textual analysis using salient phrase ranking [250]
		Zhuang Y. et al [258]	Visual & Textual	- Uses a two-step process: i) runs two regression models to process visual and textual features, ii) runs affinity propagation algorithm [64] on combined feature representation to produce image clusters
		Hirota M. et al [82]	Visual & Textual	- Uses a two-step process: i) runs a k-nearest neighbor method to identify tags in the Web image neighborhood and enrich their textual features, ii) runs maxmin partitional clustering [214] on the resulting Web images
	Hierarchical	Zhao K. et al. [256]	Textual	- Uses a three-step agglomerative clustering process: i) clustering input images based on concepts extracted from their metadata, ii) merging clusters produced in the first step based on concepts extracted from the images' textual contexts, and iii) expanding the contexts of clusters from the second step using the Wikipedia knowledge base [33] to represent their semantics
		Hirota M. et al. [83]	Visual & Textual	- Uses a three-step process: runs constrained agglomerative clustering with must-link constraints to process different features in consecutive steps: i) social tags, ii) photo-taking conditions, and iii) visual features.
		Alamdar F. and Keyvanpour M. [4]	Visual	- Introduces folding tree hierarchical clustering algorithm - Uses a four-step process: i) runs folding tree clustering, ii) fine-tunes dispersed clusters by identifying the farthest images in every cluster and verifying whether they should be assigned to a different one based on image/cluster similarity, iii) merges small clusters following a predefined population size, and iii) eliminates small clusters that cannot be merged given their dissimilarity from all others
	Spectral	Gupta G. and Ghosh J. [75]	Visual & Textual	- Uses a two-step process: i) clusters key-phrases using k-lines spectral clustering [24], ii) runs Bergman bubble clustering [17] on images from the first step clusters and groups them based on their visual features
		Cai D. et al. [29]	Visual, Textual, & Link	- Uses a two-step process: i) applies spectral clustering on the combined text and link image affinity matrix, ii) runs a second spectral clustering step on each cluster produced in the first step using the images' visual features

Spectral approaches: In [53], the authors consider a two-step clustering method using textual features in the first step, and then visual features in the second step. First, the approach extracts query key-phrases and clusters them using the k-lines spectral clustering algorithm [24]. The authors consider that query key-phrases capture the semantic topics of the image search

results. Second, images in the resulting clusters which correspond to each key-phrase, are themselves clustered using the Bergman bubble clustering algorithm [17, 75] applied on the images' visual features to improve their visual organization. Another two-step clustering method is introduced in [29], considering text and link-based features in the first step, and then visual features in the second step. The system starts by extracting the text and link information describing every image in its containing webpage. The resulting textual feature vectors are used to compute a pair-wise image affinity matrix, consisting of the cosine similarity values between all pairs of image vectors. Another pair-wise image affinity matrix is computed based on the number of shared links between the images. The scores of the two affinity matrices are then linearly aggregated, and the generalized Eigenvalue problem is solved on the combined matrix, followed by a spectral clustering process applied on the reduced dimensional matrix. A second spectral clustering step is then run on each cluster produced in the first step, this time using the images' visual features to produce visually coherent search result clusters.

***Discussion*:** While early ISRC solutions mainly perform single-step image search result processing, more recent solutions mostly perform multi-step processing. Most multi-step methods make use of text features to run a first clustering of the result images, where the main meanings and facets of the user queries are discovered. Then, the clustering is improved by introducing visual or weblink features in subsequent clustering steps, to improve the visual organization and diversification of result images and facilitate user browsing. Note that partitional and spectral methods usually suffer from the same limitations mentioned with single-step methods (cf. Section 4.3.3.1) while hierarchical methods are usually more computationally expensive compared with their counterparts (similarly to generic clustering solutions in Section 4.3.1). The main characteristics of single-step and multi-step ISRC solutions are summarized in Table 3.

While promising results have been produced in the literature, yet most existing (single-step and multi-step) ISRC solutions share several common challenges. Most methods are developed separately and do not compare their results against each other. They suggest different ways of presenting the resulting clusters (using cluster representatives, hierarchical structures, or spatial arrangements) which makes it difficult to draw clear conclusions on the quality of the proposed solutions. Also, there is a lack of common evaluation datasets and metrics, which limits the reproducibility and comparison of the evaluation results. Another key challenge is the interplay between *relevance* and *diversity*: focusing on producing relevant results may produce many near duplicate images, while adding diversification may result in losing relevant result images [90, 91]. We further discuss these challenges along with other future directions in the following sections.

4.4 Cluster-based Search Result Visualization

Organizing image search results into clusters becomes more useful for image browsing and retrieval if the clusters can be visualized properly. As a result, different visualization techniques have been proposed in the literature. We organize them in three main categories: i) cluster representatives, i) hierarchical arrangement, and ii) spatial arrangement.

4.4.1 Cluster Representatives

Once generated by the clustering algorithm, clusters of images can be abstracted and represented as a bunch of: i) representative images, or iii) representative labels.

Representative Images: Clusters can be described visually as a set of representative images describing the visual properties and diversity of their containing clusters. Here, we distinguish between three main approaches: i) image subset, ii) image selection, and ii) image synthesis. The first approach consists in presenting a subset of the cluster's images re-ranked and organized to highlight the cluster's visual properties. A subset of the cluster's images is first selected following certain criteria targeting relevance, diversity, or their combination (cf. Section **Error! Reference source not found.**). Then, the image subset is ranked following similar criteria (relevance, diversity, or a combination of both) for presentation to the user [225, 239]. The second approach can be viewed as a more selective version of the first, where only one (or a small number of) image representative(s) is selected to represent the cluster. The representative image(s) can be selected as: i) the most relevant w.r.t. the user query, where the original image ranking or a re-ranking of the cluster's images is utilized to identify the top-most image(s), ii) the most similar to all other images in the cluster, by comparing each image with all others (pair-wise image similarity scores that were computed during the clustering process and can be straightforwardly utilized if available), iii) the most dissimilar from all other images in the cluster: mainly utilized when selecting more than one cluster representative such that the main

focus is on diversifying the representatives [42, 214]. The third approach consists in producing a synthetic image that would sum-up the cluster's visual features. The latter is usually computed as the *average image* that only exists in the visual feature space, and is built by aggregating all the cluster images into one canonical image. This is done using mathematical aggregation functions (e.g., arithmetic mean, maximum, weight sum, cf. Table 2) applied on the visual features' histogram-like vector representations producing a synthetic feature representation describing the cluster [4, 96, 214]. A hybrid approach is described in [161] were the authors first produce the synthetic image as the average of all image features in the search result, and then sort all images according to the new similarity values computed after subtracting the synthetic image features from the original ones. The re-ranked images are then clustered using a k-means approach, and a new synthetic image is computed for each cluster allowing to re-rank the images within the cluster according to their similarities w.r.t. to the synthetic image, such that every cluster is represented by its top ranked images.

Representative Labels: Another way of describing clusters would be to assign them meaningful labels that describe their semantic contents. Typical topic extractions approaches can be utilized, e.g., [96, 187], applied on the images' textual features to extract representative terms or labels. Here, we distinguish between two main approaches: i) image-based label extraction and ii) cluster-based label extraction. The first approach consisting in processing the images' textual features separately, such as representative labels are identified for every image. Then, the image's representative labels are combined to select the top labels that best describe the cluster [256]. The second approach consisting in combining the cluster images' textual features together, and performing topic extraction on the combined textual features in one pass, to produce the labels that best describe the cluster [224]. Dedicated measures such as information gain, label importance, cluster text compactness, and cluster overlap entropy [231, 235, 240] have been suggested to extract the most descriptive cluster labels. More recent approaches have suggested using external knowledge sources such as Wikipedia or Yago [162, 213], where the labels representing the clusters are identified based on their semantic meaning, and might be selected from outside the clusters (e.g., term *sports car* might be selected to describe a cluster containing terms *Ferrari* and *Corvette*, such that *sports car* never appears in the cluster).

Note that representative labels can be combined with or extracted from representative images, providing an integrated visual and textual abstraction of the cluster [66, 222]. A recent approach in [76] clusters images of touristic attractions from Flickr, and then runs multiple neural network models to extract representative images of the main attractions. Photos with similar geotags are used to detect place-relevant tags, and are then used to merge and extend the clusters according to the similarity between pairs of tag embeddings. Noisy images are then filtered out using a single-shot multi-box detector model, before selecting the cluster representative images using an integrated ranking model.

4.4.2 Hierarchical Arrangement

Another way of visualizing image clusters is by arranging them in a hierarchical structure, i.e., a taxonomy consisting of a set of nodes and a set of hierarchical links connecting the nodes together. Hierarchical arrangement techniques can be distinguished following: i) the kind of taxonomy, and ii) the construction process being used.

Image taxonomies: We distinguish between two kinds of image taxonomies: i) visual and ii) semantic. Visual taxonomies consist of nodes representing clusters of images, and hierarchical links representing the *containment* relationships between the clusters. The taxonomy's root node represents the largest cluster consisting of all images in the search result set, and *contains* one or many sub-clusters represented as the root node's children. The latter also *contain* their own sub-clusters, and the same structure is repeated until reaching the taxonomy's leaf nodes which are made of individual image clusters. Visual taxonomies allow decomposing the output space into several layers of highly similar images which allows the user to visually browse the search results and identify the images of interest [160]. They can also help in object recognition by navigating the branches of the taxonomy which contain relevant image clusters, in order to avoid irrelevant possibilities (e.g., navigating the branch which includes clusters of images containing *car* objects means the user is probably interested in *car* or *vehicle*-related images) [72]. A visual taxonomy can also be used as a data mining tool, providing insights on the visual patterns and correlations between images in the search result set [72].

Semantic taxonomies consist of nodes representing image labels, and links representing the semantic relationships between the labels. Labels are extracted from the search result images' textual features, and allow to associate the images with the taxonomy. The most common hierarchical semantic relationships are hypernymy/hyponymy (IsA/HasA, e.g., *Jaguar*-IsA-*car*) and meronymy/holonymy (PartOf/HasPart, e.g., *chassis*-PartOf-*car*) [195, 196]. Semantic taxonomies can be used to filter-out

irrelevant or non-descriptive labels in the search result set. They can guide the user by offering images not just from the query keywords, but also from their semantically similar labels in the taxonomy [72]. Moreover, a semantic taxonomy can serve as a data mining tool, allowing to learn the visual appearance of textual labels in the image result set, thus narrowing the gap between visual and textual features to help solve the *semantic gap* problem [135].

Construction process: The taxonomy construction process involves two phases: i) generating a hierarchical organization of nodes (i.e., image clusters or labels) and ii) associating the images with the nodes. The latter can be undertaken using two kinds of construction processes: i) semi-automated, or ii) fully-automated. Following the semi-automated approach, either the first phase, the second phase, or both phases of the taxonomy construction process are done manually by humans. Various works for semi-automatic generation of Web document and image hierarchies have been developed in the literature, e.g., [72, 135, 164, 253]. They mainly rely on labelled data or predefined categories applied on self-organizing maps [54, 149] to produce the node hierarchy. Visual or textual categorization techniques [18, 72, 135] are then used to associate the images with the produced hierarchy. In this context, result images can be classified against existing taxonomies where relevant taxonomic extracts are returned to the user. While the semi-automated approach usually produces reliable structures and associations, yet it is time and effort consuming and depends on the quality and expressiveness of the manual taxonomies or labeled data used as reference.

The fully-automated approach for taxonomy construction consists in building a hierarchical organization of search result images without human intervention or prior knowledge of the structure or predefined categories. Most methods in this category rely on hierarchical clustering algorithms, e.g., [72, 185], where the taxonomy consists of an adaptation of the dendrogram of nested hierarchical image clusters. The latter can be post-processed to extract representative images or labels for every cluster (cf. Section 4.4.1) and append them to the taxonomy. Compared with semi-automated methods, the fully-automated approach requires minimal human intervention in the form of tuning certain parameters of the hierarchical clustering algorithms which might affect the produced clusters (e.g., choice of stopping rule [23, 139, 177] to halt the hierarchical process and determine the best set of output clusters that form the dendrogram).

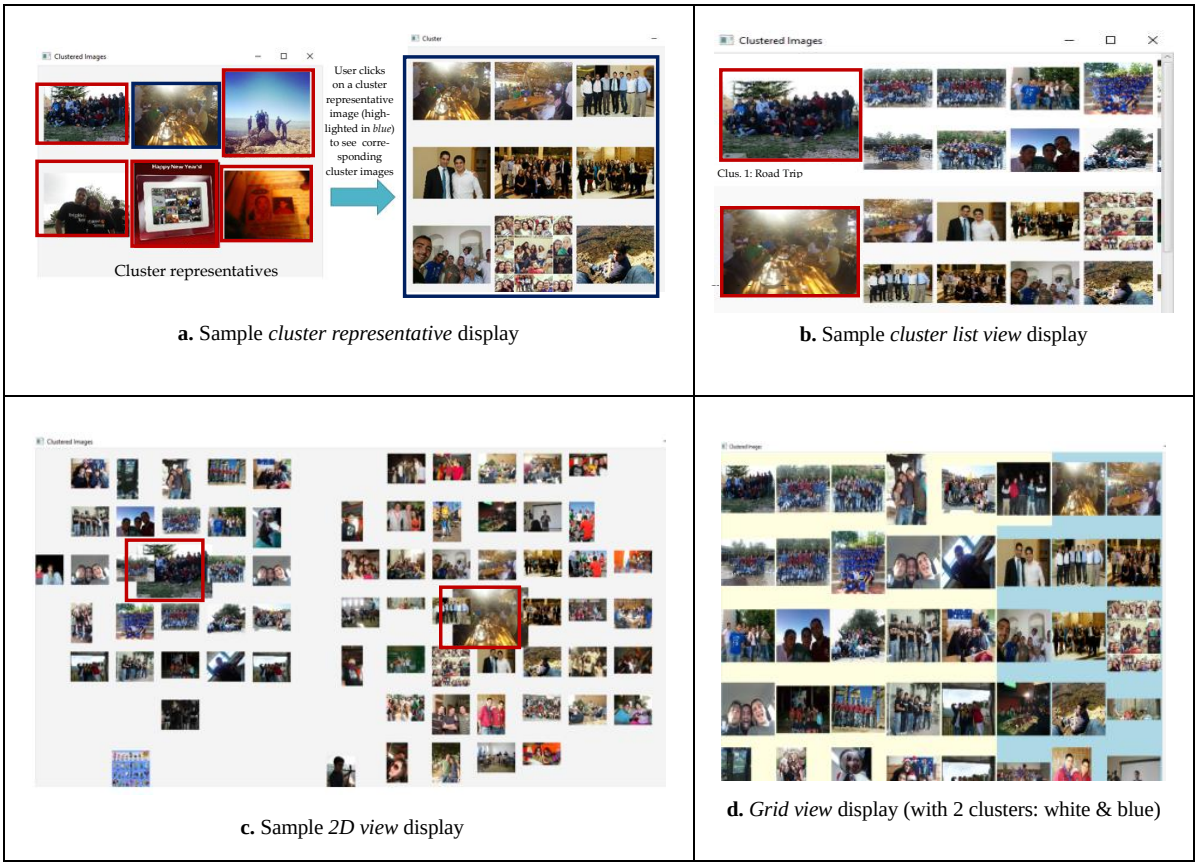
4.4.3 Spatial Arrangement

Various studies have investigated similarity-based spatial arrangements that present the image search results as a set of thumbnails in a spatial distribution, e.g., [11, 145, 169, 228]. The main idea consists in measuring the pair-wise similarities between images in the search result set, and then transforming the image similarity matrix into a 2-dimensional configuration of points, where the thumbnails of the corresponding images are properly placed to produce the arrangement. This is done in a way where similar images are positioned closer together. While spatial arrangement techniques can be used as stand-alone search result organization techniques (cf. Section 2.3), they can also be applied to visualize clustering results. In this context, several visualization techniques have been suggested to answer different user preferences, including: i) representative display, ii) cluster list view display, iii) 2D display, iv) grid view display, and v) fish-eye view display.

Representative display: Following this layout, the representative image of each cluster is displayed at first, and then when the user clicks on one of the images, a new window opens containing the images of the corresponding cluster [11]. This is done by looping through the first image in the array list for each cluster, and displaying the images in one window. Then, each image can be selected to retrieve the corresponding cluster and display the images in that cluster. The main advantage of this view is its speed in displaying the result images since it only requires initially displaying the representative images without having to display the rest of the images in each cluster. In other words, there is no need to load all cluster images unless the user chooses to do so explicitly for a given (number of) clusters. A disadvantage of this view is that it might not be very intuitive in displaying the clustering results, since the user cannot easily visualize how the clusters are organized w.r.t. image similarities/distances, or how close/far away clusters are from each other (cf. Fig. 4.a).

Cluster list view display: It consists of a list in which each item represents a cluster. For each cluster, the representative image is displayed in large, and then the rest of the images are displayed in smaller size and are placed next to the representative [169, 228]. Each image can be enlarged (upon user selection) to display it in its actual size, while allowing the user to change the cluster representative (choosing another image from the cluster to serve as its representative). Changing cluster representatives not only affects the visualization of clusters, but can also replace the old cluster representatives upon the user's request. The main advantage of this layout is that it allows the user to view all clusters at once in an organized manner and allows the

user to change representative images. A disadvantage is that it can be time and memory consuming due to the fact that a new instance should be created for every cluster, and all image search results will be loaded into memory upon display (cf. Fig. 4.b).



a. Sample *cluster representative* display

b. Sample *cluster list view* display

c. Sample *2D view* display

d. *Grid view* display (with 2 clusters: white & blue)

Fig. 4. Sample spatial arrangement displays (from [11, 169]).

2D display: It presents images in a 2-dimensional plane, where each cluster of images is separated from the rest (using different color indicators), and within each cluster of images: the representative image is placed in the middle, and the rest of the images are placed around it according to the similarity between the images and the representative image [11, 169]. In a given cluster, images that are most similar to the cluster's representative are displayed closest to the center, while those that are least similar are displayed farther from the center. This is done by creating a specific display pane for each cluster in which the images are laid out according to the above description. Then, each display pane is added to the main grid which separates the clusters from each other. The main advantage of this approach is that it helps visualize clusters while highlighting their intra-cluster similarities and inter-cluster similarity-based spatial organization. A disadvantage of this display is that it is more computationally expensive and time-consuming, compared with the previous two displays (cf. Fig. 4.c).

Grid view display: It places all the images in a 2-dimensional grid such that images in the same cluster are placed as close as possible to each other. The difference between the grid view and the 2D display is that grid view places images in an ordered manner as tiles next to the cluster representative, whereas 2D display places images in a spiral shape around the representative. Different clusters are distinguished using background colors assigned to each cluster [11, 169]. To do this, the representative image is placed first in the grid, and then the next image in the cluster having the highest similarity to the representative is placed as close to it as possible. For each new image, the system identifies the next tile in the grid which can be filled taking into account the similarity/distance w.r.t. the representative (i.e., trying to keep the maximum similarity/minimum distance). The main advantage of this view is that images can be displayed in a 2-dimentional manner while requiring less computation and time compared with the 2D display approach. A disadvantage of this display is that it seems less expressive of the intra- and inter-organization of the clusters in comparison with 2D display (e.g., the distances among images and among clusters cannot be easily spotted with the grid view display, compared with 2D display where these are clearly highlighted, cf. Fig. 4.d).

Fish-eye view display: This is similar to 2D display with one major difference: the sizes of images surrounding the cluster representatives decrease as their similarities w.r.t. the representatives decrease, causing the images that are farther away from the representatives to appear smaller. It carries the already mentioned advantages and limitations of 2D display.

***Discussion*:** Spatial arrangement techniques applied on search result clusters seem to be more user-friendly compared with other cluster visualization approaches [11, 228]. Yet, they also suffer from the same limitations of stand-alone spatial arrangement methods described in Section 2.3, namely: ii) difficulty in distinguishing many images placed next to each other, and ii) potential information loss due to the overlapping of certain thumbnails or images with certain visualizations. The choice of the best cluster visualization technique depends on the user's needs and the application at hand.

5 Evaluation Methodology

As for empirical evaluation, we identify two main issues facing cluster-based ISRO approaches: i) the lack of common evaluation metrics to compare empirical results, and ii) the lack of common evaluation datasets. Existing evaluation studies are usually carried out on particular and closed datasets which limits the reproducibility and comparison of their results.

5.1 Test Measures

The effectiveness (i.e., quality) of a cluster-based ISRO solution can be evaluated based on the relevance and diversity of the search results. In this context, most existing approaches suggest to i) first manually solve the clustering task, and then ii) use the results as a reference to evaluate the quality of the clusters produced by the system [91].

The *precision* (*PR*) and *recall* (*R*) evaluation measures adopted from the field of information retrieval [176] can be utilized to compare user and system generated clusters [204, 205]. For an extracted cluster C_i that corresponds to a given ground truth cluster G_i:

- a_i is the number of images in C_i that indeed correspond to G_i (correctly clustered images).
- b_i is the number of images in C_i that do not correspond to G_i (miss-clustered).
- c_i is the number of images not in C_i, although they correspond to G_i (images that should have been clustered in C_i).

Consequently, given n the total number of generated clusters:

$$PR = \frac{\sum_{i=1}^{n} a_i}{\sum_{i=1}^{n} a_i + \sum_{i=1}^{n} b_i} \in [0,1] \quad \text{and} \quad R = \frac{\sum_{i=1}^{n} a_i}{\sum_{i=1}^{n} a_i + \sum_{i=1}^{n} c_i} \in [0,1] \tag{1}$$

High *precision* denotes that the clustering task achieved high accuracy, grouping together images that actually correspond to the ground truth clusters. High *recall* means that very few images are not in the appropriate cluster where they should have been. In addition to comparing one approach's *precision* improvement with another's *recall*, it is a common practice to consider *f-value*, which represents the harmonic mean of *precision* and *recall*:

$$F\text{-}value = \frac{2 \times PR \times R}{PR + R} \in [0,1] \tag{2}$$

Therefore, as with traditional information retrieval evaluation, high *precision* and *recall*, and thus high *f-value* characterize a good clustering approach. Other similar cluster-based evaluation measures have been proposed in the data clustering literature, such as Davies-Boulding and Dunn indices among others [21, 50, 58], and can be adapted to evaluate the relevance of cluster-based ISRO methods. Yet, an aspect which is not directly evaluated by the latter measures is the impact of search result ordering in evaluating the diversity of the clustering results. To that end, the authors in [154] utilize *precision at X* (*PR@X*) and introduce *cluster recall at X* (*CR@X*) as two measures that can evaluate both image cluster relevance and diversity. *Precision at X* measures the number or relevant images among the top *X* results, and *cluster recall at X* measures how many clusters from the ground truth are represented among the top *X* results provided by the system to assess result diversity [91]:

$$PR@X = \frac{a_X}{X} \in [0,1] \quad \text{and} \quad CR@X = \frac{|C_X|}{|G|} \in [0,1] \tag{3}$$

where a_X is the number of relevant images from the first X ranked results, $|C_X|$ the number of image clusters represented in the first X ranked images, and $|G|$ the total number of image clusters from the ground truth. To perform and overall assessment of both relevance and diversity, *f-value* at X (*f-value@X*) is introduced in [91] as the harmonic mean of the two measures:

$$F\text{-}value@X = \frac{2 \times PR@X \times CR@X}{PR@X + CR@X} \in [0,1] \quad \textbf{(4)}$$

High *precision at X* and *cluster recall at X*, and thus high *f-value at X* characterize relevant and diverse image clusters and a good cluster-based ISRO approach [91, 154].

Besides evaluating *effectiveness* (i.e., quality), note that evaluating the *efficiency* (i.e., time and/or space performance) of ISRO methods is almost completely dismissed in existing approaches, and needs to be addressed in upcoming studies.

5.2 Test Data

Few studies have published information regarding image datasets and manually annotated tag names that can be used for search result evaluation, and which can form the seed for an integrated baseline for future evaluation and comparative studies. The existing dataset characteristics are summarized in Table 4.

The authors in [171] use a collection of 207,000 Flickr images captured around 207 locations in Paris (i.e., 100 images per location) to assess the diversity of visual summaries of geographic areas. They determine the ground truth labels by exploiting the geographical coordinates accompanying the images, using an affinity propagation clustering of the latitude and longitude coordinates, and thus do not require manual user input. In [199], the authors address the diversification problem in the context of populating the YAGO knowledge base, using 2 million labelled images (e.g., people, buildings, mountains, lakes, etc.) gathered from Wikipedia. The authors suggest using mean average precision (MAP) and other related measures to evaluate the usefulness (i.e., information gain) of images based on their positions in the result list. In [214], the authors use a dynamic dataset of 3,750 images produced from the results of 75 randomly selected queries from Flickr logs for which only the top 50 results are retained. They acquire manual annotations from human assessors to group the data into clusters with similar visual appearances, and evaluate performance using the Folwkes-Mallows metric (i.e., a measure equivalent to the f-value metric, cf. Formula 2). In [106], the authors introduce a dataset of 71,478 Web images produced from the top ranked search results of 353 image search queries, along with their associated meta-data. The metadata consists of the original textual query, the image URL, the URL of the webpage containing the image, the page title, the image's alternative text, the 10 words before and after the image in the containing webpage, as well as user-provided manual labels describing the images' relevance to the search query. In [154], the authors introduce the ImageCLEF benchmark, including the Photo Retrieval task with a dataset of 498,920 news photograph images and caption texts classified into sub-topics (e.g., locations and animals). The authors conduct an empirical evaluation exercise on a total of 50 topics associated with a certain number of clusters, and suggest the use of *precision at X* (=20) images and *cluster recall at X* (=20) to evaluate the percentage of different clusters represented in the top X (=20) results. In a recent study in [90, 91], the authors introduce a new dataset dedicated to evaluating ISRO, consisting of 43,418 Flickr ranked photos of 396 geographic location landmarks that are manually annotated for both relevance and diversity. While smaller in size than the ImageCLEF collections [154, 212], yet the dataset proposed in [90, 91] contains images that are already associated with topics by Flickr, and can be straightforwardly used to evaluate information retrieval and search result organization. Also, unlike ImageCLEF and other datasets which target generic ad-hoc retrieval scenarios, the dataset in [90, 91] considers a focused real-world scenario (i.e., tourism), to evaluate search diversification quality [89]. Another recent study in [170] introduces the SubDiv17 dataset that is specifically designed to evaluate visual diversification in ISRO. It consists of 57,326 images retrieved from Flickr based on 200 general-purpose queries sampled from the word-wide Google query trends[1]. The dataset includes manual annotations by 33 human annotators describing the relevance and diversity of the images w.r.t. their queries, to facilitate the investigation of the quality and subjectivity aspects in ISRO. The proposed dataset was validated in the MediaEval 2017 Retrieving Diverse Social Images task using *PR@X*, *CR@X*, and *f-value@X* measures (cf. Section 5.1) to reflect the relevance and diversification levels of image search results.

1 http://trends.google.com/

Table 4. Characteristics of test datasets

Approach	# of images	Source	Scenario	Application	Evaluation
van Leuken R. et al. [214]	3,750	Flikr	Generic – based on user queries	Image clustering	Individual
Krapac J. et al. [106]	71,478	Web	Generic – based on user queries	Search Result Clustering	Individual
Paramita M. et al. [154]	498,920	Web	Generic – news photographs	Search Result Diversification	Comparative
Taneva B. et al. [199]	2,000,000	Wikipedia	Generic – from Wiki pages	Populating a knowledge base	Individual
Rudinac S. et al. [171]	207,000	Flikr	Specific - geolocations	Search Result Diversification	Individual
Ionescu B. et al. [90, 91]	43,418	Flikr	Specific - tourism	Search Result Clustering	Comparative
Rohm M. et al. [170]	57,326	Flick	Generic – based on Google trends	Search Result Diversification	Comparative

***Discussion*:** Many datasets in Table 4 have been developed to evaluate individual solutions introduced by the authors themselves, which limits the reproducibility of the results. Recent studies by Paramita M. et al. [154], Ionescu B. et al. [90, 91], and Rohm M. et al. [170] introduce different datasets which are specifically developed to perform comparative evaluation studies. A main challenge in this context is to integrate the latter datasets in a unified benchmark with common evaluation measures to be used as a gold standard data repository for future image search result organization studies.

6 Discussion and Research Challenges

To wrap up, we discuss some of the research challenges facing existing ISRO methods, and outline some ongoing and future directions. We organize the challenges following the main steps involved in ISRO, including i) image representation, ii) similarity computation, iii) search result clustering, and iv) search result presentation and visualization. For more on general ImR research challenges, the reader can refer to recent survey studies in [95, 114].

6.1 Image Representation

6.1.1 High-dimensional Indexing of Image Features

Despite the importance of evaluating ISRO quality (effectiveness) as mentioned previously, ISRO efficiency remains a central factor that contributes to the adoption or dismissal of a given ISRO approach, especially when applied on huge image collections. A basic challenge that would affect the whole ISRO pipeline is how to handle high-dimensional image features. In this context, multi-dimensional image indexing has been investigated as an offline solution to improve the representation of image features, in order to allow faster access, processing, and retrieval. Most existing solutions fall into three main categories: i) tree-based indexing, ii) hashing-based indexing, and iii) visual words based indexing. Tree-based indexing solutions successively partition the image search space and form hierarchical tree structures. The inner-nodes represent groups (clusters) of images or image regions, and the leaf nodes represent the images or the regions that are indexed. Image or region partitioning is conducted using adapted image clustering algorithms, e.g., k-means and hierarchical k-means used for region-based cluster indexing in [88, 230]. While relatively efficient to produce, tree-based indexing solutions are usully inefficient when the number of feature dimensions exceeds 20 [3]. This requires the image features to be pre-processed for dimension reduction prior to applying tree-based indexing, or to be divided into sub-vectors of lower dimensionality that are (separatly) suitable for tree-based indexing, which might not be always feasible. Hash-based indexing solutions project image features from high dimensions to low dimensions using hash functions. Different approaches have been proposed, including Locality Sensitive Hashing (LSH) [57] to construct a family of spectral hashing functions where the probability of collision is higher for images that are close to each compared with those which are far apart in the reduced dimensional space. Nonetheless, hash-based indexing is usually inefficient on sparse features, and requires the features representations to be pre-processed into dense vectors before being indexed [3]. This adds significant overhead to the index construction process, and might lead to reduced indexing quality due to the features' transformation into dense vectors representations. Visual words based indexing solutions extract the local features from images, and quantize them into their closest visual words (codebook) based on a pre-learned training set. Consequently, a visual word-based vector is produced and is represented as an inverted file to allow for fast identification of all images containing the visual word entries, and subsequently fast feature processing and similarity computation. A major challenge with visual words based indexing is how to produce semantically relevant visual words that are more discriminative of the image contents compared with its initial feature representation. Another major challenge is how to produce indexing structures which are robust to adding or removing images from the search space in order to allow fast and dynamic index updates [3], especially that the index would need to updated regularly and on-the-fly for ISRO.

6.1.2 Joint Word-Image Modeling and Implicit Image Semantics

Word embeddings have recently proven to be an important tool for the representation of word meanings in large text corpora. Their effectiveness rests on the distributional hypothesis that *words occurring in the same context carry similar semantic information*. This leads to the creation of so-called *implicit concepts*, i.e., *synthetic* concepts generated by extracting latent relationships between terms in a document collection, or by calculating probabilities of encountering terms, such that the generated concepts do not necessarily align with any human-interpretable concept [97, 201]. This is different from conventional concept-based semantic analysis, which utilizes *explicit concepts* representing *real-life* entities/notions defined following human perception (e.g., concepts defined within a conventional knowledge base such as WordNet or Wikipedia) [255]. In this context, various recent approaches have investigated the adaptation of word embeddings and related techniques to perform joint word-picture modeling and extracting implicit concepts from image objects and regions. The main hypothesis with this family of methods is that *images* (*or regions*) *containing similar objects* (*or subregions*) *are considered to be semantically related* [209]. As a result, few approaches have been put forward to learn object and region embeddings from an image corpora. In [69, 148], the authors study the co-occurrence of both visual and textual features using PLSA to produce a combined vectored data representation of both modalities. They extend the PLSA to higher order to become applicable to more that two observable variables (visual and textual), and utilize cross-modal dependencies learned from corpora of tagged images to approximate the join distribution of the two variables. In [210], the authors adapt LSA and word2vec's skipgram and CBOW models to generate embeddings from object co-occurrences in images and subregions, and show that the produced embeddings improve typical object classification models by an average 3-to-4.5% top 1 accuracy. In [41], the authors introduce a Dual Path Recurrent Neural Network (DP-RNN) which processes images and sentences symmetrically by a deep learning model. Given an input image-text pair, the model reorders the image objects based on the positions of their most related words in the text. Similarity to extracting the hidden features from word embeddings, the model leverages the RNN to extract high-level object features from the reordered object inputs, producing similar representations in describing semantically related objects. The proposed approach produces state-of-the-art retrieval quality results compared with typical ImR techniques.

While extracting implicit semantics from images is promising, yet existing methods suffer from various limitations, namely: i) implicit concepts are difficult to understand and evaluate by human users, ii) the number of generated implicit concepts depends on statistical analysis/deep learning rather than the actual meaning of the visual objects, iii) deep learning based methods rely on object or region annotations which are not always available, iv) image semantics are not limited to the objects or regions that they contain, but often depend on the spatial relationships between the objects and regions, where considering spatial context embeddings could help improve feature vector quality.

6.1.3 Describing Images based on Aesthetics

The focus of most ISRO solutions thus far has been on image contents (visual) and meta-data (textual and linkage). Another way to organize and distinguish among images is following their *quality*. Quality can be perceived at two levels, one involving concrete image parameters like size, aspect ratio, and color depth, and the other involving human perception which is denoted as *aesthetics* [47, 151]. While it is trivial to organize images based on their parameters, their differences may not be significant enough to use as discriminative criteria. On the other hand, aesthetics designate the way people perceive images, as good quality (like) or bad quality (dislike). Yet given the vagueness and subjectivity of human perception, how to aesthetically organize pictures remains an open challenge. In a sense, this is similar to the problem of *semantic gap* [114], where the aesthetics gap can be viewed as the lack of relationship between the information that can be extracted from the image features and the interpretation of the human perception of image quality [47]. One way to model aesthetics is to study image rating trends in photo-sharing sites such PhotoNet[1] which supports the peer-rating of photographs based on their aesthetics [100]. This has generated a sizeable database of ratings corresponding to the over 5 million photographs. Another major attempt is the Aesthetic and Attributes Database (AADB) consisting of 10k images with ratings of various aesthetic attributes including *interesting content*, *object emphasis*, *good lighting*, and *color harmony*, among others [105]. Recent studies have utilized Convolutional Neural Networks (CNNs) and related deep learning algorithms to learn to assess image aesthetics by training on the above mentioned image databases and others, e.g., [165, 197]. Nonetheless, major concerns in this field include: i) the lack of a common large-

[1] https://www.photo.net/

scale image aesthetics benchmark for model training and evaluation, ii) the lack of common attributes and feature representations to describe aesthetics, iii) the lack of objective measures to evaluate the quality of aesthetic features in ImR systems, as well as iv) the subjective nature of image aesthetics and the impact of human psychology (e.g., personality, taste, history, preferences) on the reliability of the rating process [151].

6.2 Image Similarity Computation

6.2.1 Automatic Learning of a Similarity Metric

While machine learning techniques have been used to help solve some of basic problems of ImR, namely feature selection, query processing, as well as result ranking and presentation (e.g., cluster-based ISRO), few studies have investigated the adaptation of machine learning algorithms to automatically learn a similarity metric from ground-truth image data. One approach to achieve this is to learn a generalized Mahalanobis distance defined over the input image feature space using equivalence constraints. Unlike image labels, equivalent constraints can be automatically obtained without the need for human intervention, and allow to modify the representation of the feature space, leading to improved clustering and classification performance [200, 223]. Under certain assumptions, the problem of learning a generalized Mahalanobis distance can be viewed as a Maximum Likelihood estimation of the within class covariance matrix. In [244], the authors use a boosting approach to learn a improved distance measure for similarity estimation based on statistical analysis of distribution models and distance functions. Based on the assumption that a single isotropic distribution model is often inappropriate, the authors use a boosted distance measure framework that finds multiple distance measures, which fit the distribution of selected feature elements that are better suited for accurate similarity estimation in the consider feature spare representation. Experimental results on stereo matching and motion tracking in video sequence show a more accurate feature representation model compared with legacy Euclidean and Manhattan distances [182, 244]. Another approach to learn a similarity metric is to perform kernel-based learning by capturing the non-linear relationships among the meta-data and contextual information provided by the data, e.g., [147, 236, 237]. Multiple Kernel Learning (MLK) algorithms have also be used to address the problems associated with kernel selection, and have been shown to produce improved classification quality [237]. For instance, the authors in [147] introduce three new similarity-based MKL algorithms to classify remote-sensing images. They introduce three kernel-based similarity measures (kernel alignment, norm of kernel difference, and Hilbert-Schmidt independence criterion), and then solve the optimization problem associated with each similarity measure using heuristic and convex optimization methods. The proposed algorithms identify the optimal combination of kernels by maximization compared with an ideal kernel. While promising, yet most of the above-mentioned solutions for automatic similarity learning suffer from the same limitations of general machine learning algorithms, namely the need for adequate training time and data which are not always available. In addition, most methods have produced good results in separate and specific application domains, and need to be further investigated and evaluated within common scenarios and benchmarks.

6.3 Clustering

6.3.1 Combining Multiple Clustering Methods

Given the number of different clustering algorithms that have been used to perform ISRO, and considering the relevance and diversification issues discussed in the previous paragraphs, a major challenge in this context is how to choose the most adequate algorithm or combination of algorithms that would best organize the image search results. Here, we can identify various sub-challenges: i) there is no prior knowledge about the underlying structure of the search results and the user might not have a clear idea of what to consider as a good solution, ii) different clustering algorithms may produce different clustering results for the same data, by imposing a particular structure or computation process onto the data, iii) there is no single clustering algorithm that can perform reliably well for different scenarios or criteria, and iv) there are no clear guidelines to follow for choosing individual clustering algorithms [9].

In this context, a possible solution that is worth investigating is the use of *consensus clustering*, as a method for aggregating different results from multiple clustering algorithms [152, 257]. Also referred to as *cluster ensembles*, consensus clustering consists in reconciling the clustering results produced for the same data by different clustering algorithms or different runs of the same algorithm [218]. More specifically, it consists in combining multiple clustering solutions (clusters or partitions) into a single consolidated solution [7]. While the ensemble method has been well-studied in the field of supervised learning due to

its successful applications in classification tasks [70], yet, the transition from supervised learning to unsupervised learning is not straightforward. Few studies have recently attempted to apply the same paradigm to the unsupervised learning field, particularly to clustering problems, e.g., [8, 9], aiming to produce more consistent, reliable, and accurate clustering results. A main challenge here is how to combine the clusters that are generated by the individual clustering solutions in an ensemble, as this cannot be done through simple voting or averaging as in classification [8]. Few cluster aggregation functions have been suggested in the literature. For instance, the authors in [8] propose a three-step adaptive consensus function: i) transforming the member clusters into binary representations, ii) measuring the similarity between initial clusters and adaptively merging the most similar ones to produce *k* consensus clusters, and iii) identifying the candidate clusters and evaluating their quality. The process is repeated iteratively until reaching convergence where a minimal number of elements change clusters. The authors in [9] attempt to improve on the previous approach by applying an adaptive method to choose the number of clusters *k* and fine-tune the convergence parameters. Yet, the authors state that the literature still lacks an effective and scalable consensus function that can be commonly utilized for practical applications [9], including solutions adapted for cluster-based ISRO.

6.3.2 Performing Adaptive Clustering

Adaptive clustering uses external feedback to improve cluster quality. External feedback can take different shapes such as direct user feedback, or reward values computed based on successive data clustering runs by memorizing what worked well in the past. It provides a means of exploring multiple paths when searching for good clusters, starting from the features extracted and the measures used for feature similarity computation, to the usage of the clustering algorithms and their tunable parameters. For instance, if certain features are not useful in producing good quality clusters in certain contexts, they will be considered as less rewarding than other features, and will be transformed or neglected when measuring image similarity to perform the clustering task [5]. Statistical and entropy-based measures can be used to evaluate the relatedness between feature representations and result clustering quality [16, 178]. For instance, the authors in [229] generate adaptable multi-histogram features and utilize fuzzy c-means clustering to optimize the original feature set and adaptively determine the optimal number of clusters for low embedding rates. In [211], the authors introduce an approach for feature extraction based on Grassmann manifolds, and run multiple clustering tasks under different subspace views. They consequently and adaptively learn the neighborhood relationships from the obtained coefficient matrix. In [14], the authors propose an adaptive feature representation model based on the Common Spatial Pattern (CSP), and introduce a generalized eigendecomposition method by Recursive Least Squares updates of the CSP filter coefficients. They describe an incremental self-training classification algorithm using density clustering to select high-confidence samples to update the spatial filters and classifier accordingly [98, 128]. Another approach to improve the robustness of selecting spatial constraint parameters is described in [234], where the author introduce a new symmetric regularizing considering the correlation between pixels and their neighbors using adaptive weighting fusion of local mean information, and embedding the maximum weight entropy constraint in parameter selection. The local spatial information is then utilized to modify the fuzzy partition information and adapt fuzzy c-means clustering centers accordingly. In [138, 227], the authors address the problem of image segmentation and introduce an adapted fuzzy c-means clustering algorithm for image segmentation with adaptive noise reduction capability. They apply a bilateral filter to acquire the image's local spatial information, and compute the difference between the original image and the bilateral filtered image. The reciprocal of the difference images and the difference images themselves are processed using fuzzy c-means, and the membership degrees within every cluster are aggregated to produce an objective function for spatial features. In [128], the authors address the problem of clustering imbalanced data, and introduce a so-called self-adaptive competitive cluster learning for imbalanced clusters. They utilize multiple sub-clusters to represent each cluster with an automatic adjustment of the number of sub-clusters. Then, the sub-clusters are merged into the final clusters based on an adapted separation measure to determine the number of final clusters during the merging process.

While most existing approaches highlight the quality and potential of adaptive methods in improving clustering quality, nonetheless, they also point out the added complexity and overhead required to perform the necessary computations, as well as the probability of producing inadequate results and generating big noise if certain specific constraints are not met. Hence, adaptive clustering remains a hot and promising research area.

6.3.3 Performance Trade-off between Clustering Quality and Efficiency

Another research issue is the performance of the clustering process, which is specifically important in ISRO since clustering is run on-the-fly during query execution time. When the size of the result set and the number of images to be processed for result clustering increase, most current approaches do not scale well. A possible solution that can be investigated is to employ two similarity measures in the cluster evaluation process: the first one would be a coarse-grained measure able to quickly identify the images that are dissimilar in order to distinguish them into broad clusters, and the second one would be a fine-grained (and probably more time consuming) measure that processes the results of the first measure to produce the final set of image clusters. A key challenge is the choice of the two measures. They should be compatible where the first measure serves as a filter function for the second measure, while complying with certain conditions inspired from query-filter architectures, e.g., [101, 207], namely: allowing to produce an initial clustering result which is as close as possible to the final result, while satisfying the *completeness* property of the filtering phase [207]. The *completeness* property underlines that the filter measure must not allow any false clustering or dropouts, such that all images that are deemed similar following the second measure are also deemed similar according to the first filter measure. In addition, the performance of the clustering task involves multiple aspects including clustering quality (accuracy), clustering efficiency (speed), and other cluster properties (size, shape, density, etc.). Optimizing one aspect generally affects the other aspects. The trade-off between the different aspects can be addressed as a multi-objective optimization problem [5], allowing to identify a combined measure as a compromise between them. For instance, considering a set of image search results to be clustered, and considered two clustering solutions A and B, where system A is more effective than B, while system B is more efficient than A, the questions here are: i) which system will be used to solve the problem? [5], and ii) how can solutions A and B be combined into a hybrid solution C that fulfills both requirements?

6.4 Search Result Presentation

6.4.1 Diversifying Image Search Results

An efficient ISRO solution should be able to summarize and provide a global view the search space, identifying results that are both relevant and that cover diverse aspects of the query. Most queries involve many intents and target multiple sub-topics (e.g., animals are of different species, cars are of different types and manufacturers) [26]. By widening the pool of possible results, one can increase the likelihood of the retrieval system to provide the user with the information needed and thus to increase its effectiveness. In this context, the problem of result diversification was initially addressed for text-based retrieval, and typically involves two steps [219]: i) first, a ranking candidate set with elements that are most relevant to the user query is retrieved, ii) then, a subset of the relevant results is computed by retaining only the most diverse elements. In fact, the main idea behind search result diversification is to mitigate relevance and diversity [90], which in general tends to be antinomic [91], i.e., the improvement of one of them usually results in a degradation of the other. Too much diversification may result in losing relevant elements while increasing relevance only tends to provide many near duplicates [91]. Considering multimedia data and more specifically Web and social images, the diversification problem receives the additional challenge of dealing with different (multimodal) feature representations, e.g., visual, textual, link-based, and region-based (cf. Section 4.1). Due to the heterogeneous nature of the features, Web images tend to be more complex and difficult to handle than text data.

Some approaches have attempted to simplify the task by transposing certain feature representations into more simple (numeric) representations using content descriptors and fusion schemes [91], where diversification is then carried out in these multi-dimensional feature spaces with strategies that mainly involve cluster-based ISRO. For instance, the authors in [214] address the visual diversification of image search results using a lightweight partitional clustering technique in combination with a dynamic weighting function of visual features to best capture the discriminative aspects of image results. Diversification is achieved by selecting a representative image from each cluster. In [49], the authors make use of dynamic programming techniques to produce an optimized image ranking scheme combining both relevance and diversity in the search result. In [199], the authors populate a database with high precision and diverse photos of different entities by re-evaluating the relatedness between the entities. They use a model parameter that is estimated from a small set of training entities, and handle visual diversity using the classic scale-invariant feature transform (SIFT). In [171], the authors address the problem of image diversification in the context of automatic visual summarization of geographic areas, and exploit user-contributed images and related explicit and implicit metadata collected from popular content-sharing websites. They introduce an approach based on a Random Walk scheme with runs over a graph modeling the relations between images, considering their visual features, associated text,

as well as the metadata information about the uploader and annotators. In [26], the authors use relevance feedback techniques to add the user in the loop, by harvesting feedback about the relevance of the search results. This information is used as ground truth for re-computing a better representation of the data, and diversifying the search results. Search result diversification has also been recently addressed in the context of Web video retrieval [87], where the authors use a video near-duplicate graph that represents visual similarity relationships among videos on which near-duplicate clusters are identified and ranked based on cluster properties and inter-cluster links. Diversification is then achieved by selecting a representative video from each ranked cluster.

In short, adapting or expanding cluster-based IRSO solutions to better address search result diversification, while maintaining highly relevant results, would provide more adapted solutions and thus improve retrieval quality.

6.4.2 Integrating User Feedback

Adding feedback about the quality of the search results and their organization could improve their relevance and provide the user with more personalized results. One of the earliest and most successful relevance feedback solutions is Rocchio's algorithm [167], which is based on the assumption that most users have a general conception of which documents are relevant or non-relevant. Using the latter judgments acquired from a certain relevance feedback window, the user's search query is revised by adding the features of positive examples and subtracting the features of negative examples from the original feature representation. In [172], the authors introduce the relevance feature estimation (RFE) algorithm which assumes that some specific features might be more important than others for a given query, according to the user's subjective judgment. They introduce a re-weighting strategy which analyzes the relevant objects in order to understand which dimensions are more important than others in determining more relevant results. Features with higher variation with respect to the relevant queries lead to lower importance factors compared with elements having less variation. More recently, machine learning techniques have been used to allow relevance feedback, by formulating the problem either: i) as a two-class classification of the negative and positive samples, or ii) as a one-class classification problem, separating positive samples by negative ones [26]. Most methods use legacy classifiers such as support vector machines [115], nearest neighbor approaches [214], classification trees (e.g., Random Forests) [25], or boosting techniques (e.g., AdaBoost) [245]. Following the training phase, all the results are ranked according to the classifier's confidence level [115], and are classified as relevant or irrelevant depending on some output function [245].

Nonetheless, most existing relevance feedback techniques focus exclusively on improving the relevance of the results, completely neglecting the issue of result diversification. In a recent study in [26], the authors introduce a pseudo-relevance approach where they combine the concept of relevance feedback with result diversification. They automatically simulate user feedback by selecting relevant and non-relevant sample images from the initial query results. Consequently, they utilize hierarchical clustering to re-group images according to their visual features, and re-rank the cluster representatives to diversify the final search result. Note that user feedback can be integrated at the different phases of cluster-based ISRO, starting from the choice of the image features and the similarity measures used to compare the features, to the clustering algorithms and cluster search result visualization techniques used, allowing to improve the relevance and diversification of the results and better adapt them to the user's needs.

6.4.3 Adapting Result Presentation to Mobile Devices

In recent years, the growing number of hand-held mobile devices which are connected to the Internet has changed the way users access and interact with Web contents [102, 103]. Yet Web image search on mobile phones is still conducted in a way similar to desktop computers, where a list or grid of ranked image results is returned to the user [142]. Previous works on human-computer interaction have shown that the needs of mobile users are different from those of desktop users [102]. In particular, ranked image lists are not suitable for the exploration and selection of relevant results on mobile devices, as they involve repeated scrolling, sliding, and zooming actions, which are not always practical and might become overwhelming. In [10], the authors conduct a study of multiple interfaces for Web image search, including a large scale analysis of search logs based on a set of 55 million queries. They conclude that Web image searchers view more pages of search results, spend more time looking at those pages, and click on more results compared with webpage searchers. According to the authors in [10], one of the main reasons for this observation is the fact that there is often no absolute answer to a query, which means that the sought after image could be one of many.

In this context, a few recent studies have suggested the use of adapted cluster-based ISRO methods as a potential solution to this problem [120, 142, 143]. Different from typical Web-based solutions, the mentioned studies particularly focus on the trade-off between clustering accuracy and used space-interface on the mobile phone screen. The authors in [142] consider that cluster-based ISRO systems should combine two criteria: i) maximum cluster accuracy and ii) minimum wasted space-interface. In other words, the mobile interface is expected to clearly show the most relevant images that are sought by the user, and present them in a compact way to properly fit the limited-size mobile screen. To do so, the authors address the issues or relevance and diversification, and introduce a new metric to quantitatively measure the compactness of a given interface: evaluating the mismatch of the used space-interface between the ground truth and the cluster distribution generated for ISRO. The authors highlight the high divergences between clustering accuracy and used space maximization, and conclude that the trade-off between relevance, diversification, and ease of exploration of Web image search results on mobile devices is difficult to define. This opens the door for much needed innovations and improvements in this area.

6.4.4 Creating and Experimental Benchmark

Last but not least, a major challenge for ISRO studies is to develop a common experimental benchmark: i) implementing existing cluster-based ISRO methods to be used in comparative studies, enabling the users to evaluate the effectiveness and efficiency of various algorithms in different scenarios, and allowing them to choose the one that is most adapted to their needs, ii) implementing dedicated test measures (e.g., *precision*, *recall*, *coverage*, *diversity*, *aesthetics*) for evaluating the effectiveness of different methods, iii) providing readily available test data with predefined queries and manually vetted search result clusters and cluster representatives, serving as a baseline (gold standard) for testing, and iv) allowing testers to easily append their own algorithms, test measures, and test data in order to dynamically extend the benchmark for future empirical evaluations. Providing an experimental benchmark would facilitate future empirical studies and thus foster further research in the area.

7 Conclusion

In this survey paper, we have given an overview of current research related to cluster-based image search result organization (ISRO). We have provided a glimpse on image information retrieval (ImR) and have briefly covered the background on ISRO. We have described and categorized the various steps involved in cluster-based ISRO, ranging over: image feature representation (visual, textual, link-based, and region-based), similarity computation (vector-based and aggregation measures), image clustering (legacy data clustering, ephemeral clustering, and image search result clustering), and search result visualization (cluster representatives, hierarchical arrangement, and spatial arrangement). We have presented the main evaluation metrics and existing datasets that can be used for empirical testing. We have also summarized and discussed ongoing research challenges and future directions, including: high-dimensional feature indexing, joint word-image modelling and implicit semantics, describing images based on aesthetics, automatic similarity metric learning, combining multiple clustering methods, performing adaptive clustering, allowing dynamic trade-off between cluster quality and efficiency, diversifying image search results, integrating user feedback, adapting results to mobile devices, and creating an experimental benchmark. We hope that the unified presentation of cluster-based ISRO in this paper will contribute to strengthen further research on the subject.

References

[1] Aggarwal C.C. and Reddy C.K., *Data Clustering: Algorithms and Applications.* CRC Press, ISBN 978-1-46-655821-2, p. 49, 2014.

[2] Ahmad A. and Khan S., *Survey of State-of-the-Art Mixed Data Clustering Algorithms.* IEEE Access 2019. 7: 31883-31902.

[3] Ai L., et al., *High-dimensional Indexing Technologies for Large-scale Content-based Image Retrieval: a Review.* Science Journal of Zhejiang University, 2013. 14(7): 505-520.

[4] Alamdar F. and Keyvanpour M., *Effective Browsing of Image Search Results via Diversified Visual Summarization by Clustering and Refining Clusters.* Signal, Image and Video Processing, 2014. 8(4): 699-721.

[5] Algergawy A., et al., *XML Data Clustering: An Overview.* ACM Computing Survey 2011. 43(4):25.

[6] Algergawy A., Nayak R., and Saake G., *Element Similarity Measures in XML Schema Matching.* Information Sciences, 2010. 180(24): 4975-4998

[7] Alguliyev R., Aliguliyev R., and Sukhostat L., *Weighted Consensus Clustering and its Application to Big Data.* Expert Systems with Applications, 2020. 150: 113294.

[8] Alqurashi T. and Wang W., *A New Consensus Function based on Dual-similarity Measurements for Clustering Ensemble.* International Conference of Data Science and Advanced Analytics (DSAA'15), 2015. pp. 149–155.

[9] Alqurashi T. and Wang W., *Clustering Ensemble Method.* International Journal of Machine Learning and Cybernetics, 2019. 10: 1227–1246.

[10] Andre P., et al., *Designing Novel Image Search Interfacesby Understanding Unique Characteristics and Usage.* In: Gross, T., Gulliksen, J., Kotz´e, P., Oestreicher, L., Palanque, P., Prates, R.O., Winckler, M. (eds.) INTERACT, 2009. 5727: 340–353.

[11] Ayoub I., Codouni K., and Tekli J., *Personalized Social Image Organization, Visualization, and Querying Tool using Low- and High-Level Features.* IEEE Inter. Conf. on Computational Science and Engineering (CSE'16), 2016. Paris, France.

[12] Azar D., Fayad K., and Daoud C., *A Combined Ant Colony Optimization and Simulated Annealing Algorithm to Assess Stability and Fault-Proneness of Classes Based on Internal Software Quality Attributes.* International Journal of Artificial Intelligence, 2016. 14:2.

[13] Azar D. and Vybihal J., *An Ant Colony Optimization Algorithm to Improve Software Quality Predictive Models.* In Journal of Information and Software Technology, 2011. 53(4): 388-393.

[14] Bagherjeiran A., et al., *Adaptive Clustering: Obtaining Better Clusters Using Feedback and Past Experience.* IEEE International Conference on Data Mining (ICDM'05), 2005. pp. 565-568.

[15] Baimuratov I., et al., *A Bayesian Information Criterion for Unsupervised Learning Based on an Objective Prior.* International Conference on Computational Science and Its Applications (ICCSA), 2019. pp. 707-716.

[16] Balzanella A. and Verde R., *Histogram-based Clustering of Multiple Data Streams.* Knowledge and Information Systems, 2020. 62(1): 203-238.

[17] Banerjee A., Dhillon I. S., and Ghosh J., *Clustering with Bregman Divergences.* Journal of Machine Learning Research (JMLR), 2005. 6: 1705–1749.

[18] Barghout L., *Hypernym and Spatial-Taxon Hierarchy. A Cognitive Informatics & Fuzzy Logic Approach to Combining Linguistic and Image Taxonomies.* IEEE International Conference on Cognitive Informatics and Cognitive Computing (ICCI*CC), 2018. pp. 575-582

[19] Barnard K. and Forsyth D.A., *Learning the Semantics of Words and Pictures.* IEEE Conference on Computer Vision, 2001. Vol 2, pp. 408-415.

[20] Beliakov;, G., A. Pradera;, and T. Calvo;, *Aggregation Functions: A Guide for Practitioners.* Studies in Fuzziness and Soft Computing, 2007. vol. 221.

[21] Bezdek J. C. and Pal N. R., *Some New Indexes of Cluster Validity.* IEEE Transactions on Systems, Man, and Cybernetics ~ NPART B: CYBERNETICS, 1998. 28(3):301-315.

[22] Black J.A. Jr; Kahol K.; Kuchi P.; Fahmy G. and Panchanathan S., *Characterizing the High-Level Content of Natural Images Using Lexical Basis Functions.* Human Vision and Electronic Imaging VIII, SPIE, 2003. pp.378-391.

[23] Boberg J. and Salakoski T., *General Formulation and Evaluation of Agglomerative Clustering Methods with Metric and Non-metric Distances.* Pattern Recognition, 1993. 26:1395–1406.

[24] Bobrowski L., *K-Lines Clustering with Convex and Piecewise Linear (CPL) Functions.* IFAC Proceedings Volumes, 2012. 45(2):108-111.

[25] Bosch A., Zisserman A., and Munoz X., *Image Classifcation using Random Forests and Ferns.* IEEE International Conference on Computer Vision (ICCV'07), 2007. pp. 1–8.

[26] Boteanu B., Mironica I., and Ionescu B., *Hierarchical Clustering Pseudo-Relevance Feedback for Social Image Search Result Diversification.* International Conference on Content-Based Multimedia Indexing (CBMI'15) 2015. pp. 1-6.

[27] Bradley P., Mangasarian O., and Street W., *Clustering via Concave Minimization.* Advances in Neural Information Processing Systems, vol. 9, M. C. Mozer, M. I. Jordan, and T. Petsche, Eds. Cambridge, Massachusetts: MIT Press, 1997. pp. 368–374.

[28] Brin S. & Page L., *The Anatomy of a Large Scale Hypertextual Web Search Engine.* In Computer Networks & ISDN Systems, 1998. 30 (1-7):107-117.

[29] Cai D., et al., *Hierarchical Clustering of WWW Image Search Results using Visual, Textual and Link Information.* Proceedings of the International ACM Multimedia Conference, 2004. pp. 952-959.

[30] Cai D.; He X.; Li Z.; MA W.-Y. and Wen J.-R., *Hierarchical Clustering of WWW Image Search Results using Visual, Textual and Link Information.* Proceedings of the International ACM Multimedia Conference, 2004. pp. 952-959.

[31] Cai D.; He X.; Wen J.R. and Ma W.Y., *VIPS: a Vision-based Page Segmentation Algorithm.* Microsoft Technical Report, MSR-TR-2003-79, 2003.

[32] Cai D.; Yu S.; Wen J.R. and Ma W.Y., *Block-level Link Analysis.* Proceedings of the International ACM SIGIR Conference, 2004. pp. 440 - 447.

[33] Cai Z., et al., *Wikification via Link Co-occurrence.* Inter. Conf. on Information and Knowledge Management (CIKM), 2013. pp. 1087–1096.

[34] Campello R., et al., *A Framework for Semi-supervised and Unsupervised Optimal Extraction of Clusters from Hierarchies.* Data Mining and Knowledge Discovery, 2013. 27(3):344.

[35] Campello R., et al., *Hierarchical Density Estimates for Data Clustering, Visualization, and Outlier Detection.* ACM Transactions on Knowledge Discovery from Data, 2015. 10(1):1–51.

[36] Can F., *Incremental Clustering for Dynamic Information Processing.* ACM Transactions on Information Systems, 1993. 11(2):143–164.

[37] Carpineto C. and Romano G., *Optimal Meta Search Results Clustering* In 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), 2010. pp. 170–177.

[38] Carpineto C.; de Mori R.; Romano G. and Bigi B., *An Information-Theoretic Approach to Automatic Query Expansion.* ACM Transactions on Information Systems (TOIS), 2001. 19(1):1-27.

[39] Chang E.; Goh K.; Sychay G. and Wu G., *CBSA: Content-based Soft Annotation for Multimedia Image Retrieval using Bayes Point Machines.* IEEE Transactions on Circuits and Systems for Video Technology, Special Issue on Conceptual and Dynamic Aspects of Multimedia Concent Description, 2003. (13):26-38.

[40] Chang S. F , Chen W., and Sundaram H., *Semantic Visual Templates: Linking Visual Features to Semantics.* International Conference on Image Processing (ICIP), Workshop on Content Based Video Search and Retrieval,, 1998. Vol 3, pp. 531–534.

[41] Chen T. and Luo J., *Expressing Objects Just Like Words: Recurrent Visual Embedding for Image-Text Matching.* AAAI Conference on Artificial Intelligence (AAAI'20), 2020. pp. 10583-10590.

[42] Chen Y.; Wang J. and Krovetz R., *Content-based Image Retrieval by Clustering.* Proceedings of the ACM International Conference on Multimedia Information Retrieval (MIR'03), 2003. pp. 193-200.

[43] Chua T.S.; Zhao Y.; Chaisorn L.; Koh C.-K.; Yang H.; Xu H. and Tian Q., TREC 2003 Video Retrieval and Story Segmentation Task at NUS PRIS., 2003. http://www-nlpir.nist.gov/projects/tv.pubs.org.

[44] Chung F., *Spectral Graph Theory.* Regional Conference Series in Mathematics, 1997. American Mathematical Society, pp. 212.

[45] Cox I.J.; Miller M.L.; Minka T.P.; Papathomas T.V. and Yianilos P.N., *The Bayesian Image Retrieval System, PicHunter: Theory, Implementation, and Psychophisical Experiments.* IEEE Transactions on Image Processing, 2000. 9(1):20-37.

[46] Cutting D.R.; Karger D.R.; Pedersen J.O. and Tukey J.W., *Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections.* Proceedings of the ACM SIGIR International Conference on Research and Development in Information Retrieval, 1992. pp. 318-329.

[47] Datta R.; Joshi D.; Li J. and Wang J.Z., *Image Retrieval: Ideas, Influences and Trends of the New Age.* ACM Computer Surveys, 2008. 40(2):1-60.

[48] Deerwester S.; Dumais S.; Furnas G.; Landauer T. and Harshman R., *Indexing by Latent Semantic Analysis.* Journal of the American Society for Information Science, Special Topic XML/IR, 1990. 41(6):391–407.

[49] Deselaers T., et al., *Jointly Optimising Relevance and Diversity in Image Retrieval* ACM International Conference on Image and Video Retrieval (CIVR'09), 2009. pp. 1–8.

[50] Desgraupes B., *Clustering Indices - Package clusterCrit for R.* University Paris Ouest, Lab Modal'X, 2017. 33 p.

[51] Dhanalakshmi K. and Rajamani V., *An intelligent mining system for diagnosing medical images using combined texture-histogram features.* International Journal of Imaging Systems and Technology, 2013. 23(2):194–203.

[52] Dias G., Cleuziou G., and Machado D., *Informative Polythetic Hierarchical Ephemeral Clustering.* Web Intelligence, 2011. pp. 104-111.

[53] Ding H., Liu J., and Lu H., *Hierarchical Clustering-based Navigation of Image Search Results.* ACM Multimedia, 2008. pp. 741-744.

[54] Dittenbach M., Merkl D., and Rauber A., *Using Growing Hierarchical Self-organizing Maps for Document Classification.* The European Symposium on Artificial Neural Networks (ESANN), 2000. pp. 7-12.

[55] Do H. and Rahm E., *Matching Large Schemas: Approaches and Evaluation.* Information Systems, 2007. 32(6): 857-885.

[56] Domshlak C., Gal A., and Roitman H., *Rank Aggregation for Automatic Schema Matching.* IEEE Transactions on Knowledge and Data Engineering, 2007. 19(4):538–553.

[57] Durmaz O., B.H., *Fast Image Similarity Search by Distributed Locality Sensitive Hashing.* Pattern Recognition Letters 2019. 128: 361-369.

[58] Everitt B.S., Landau S., and Leese M., *Cluster Analysis, 5th Ed.* Arnold, London, 2011. 346 p.

[59] Fadzli S. A. and Setchi R., *Concept-based indexing of annotated images using semantic DNA.* Journal of Engineering Applications of Artificial Intelligence 2012. 25(8):1644-1655.

[60] Fares M., et al., *Unsupervised Word-level Affect Analysis and Propagation in a Lexical Knowledge Graph.* Elsevier Knowledge-Based Systems, 2019. 165: 432-459.

[61] Favory X., Font F., and Serra X., *Search Result Clustering in Collaborative Sound Collections.* International Conference on Multimedia Retrieval (ICMR'20) 2020. pp. 207-214.

[62] Feng H.; Shi R. and Chua T.-S., *A Bootstrapping Framework for Annotating and Retrieving WWW Images.* Proceedings of the International ACM Multimedia Conference, 2004. pp. 960-967.

[63] Fergus R., et al., *Learning Object Categories from Google's Image Search.* IEEE Inter. Conf. on Computer Vision (ICCV), 2005. pp. 1816-1823.

[64] Frey B.J. and Dueck D., *Clustering by Passing Messages between Data Points* Science journal, 2007. 315(5814):972–976.

[65] Fu L. and Medico E., *FLAME: a Novel Fuzzy Clustering Method for the Analysis of DNA Microarray Data.* BMC Bioinformatics, 2007. 8(3).

[66] Gali N., Tabarcea A., and Fränti P., *Extracting Representative Image from Web Page.* International Conference on Web Information Systems and Technologies (WEBIST), 2015. pp. 411-419.

[67] Gao B.; Liu T.-Y.; Qin T.; Zheng X.; Cheng Q.-S. and Ma W.-Y., *Web Image Clustering by Consistent Utilization of Visual Features and Surrounding Texts.* Proceedings of the International ACM Multemedia Conference, 2005. pp. 112-121.

[68] Gao Y., et al., *A Novel Approach for Filtering Junk Images from Google Search Results.* Conf. on Multimedia Modeling (MMM'08), 2008. pp. 1-12.

[69] Giouvanakis S. and Kotropoulos C., *Saliency Map Driven Image Retrieval Combining the Bag-of-words Model and PLSA.* International Conference on Digital Signal Processing (DSP'14), 2014. pp. 280-285.

[70] Gomes H., B.J.P., Enembreck F., and Bifet A., *A Survey on Ensemble Learning for Data Stream Classification.* ACM Computing Surveys, 2017. 50(2): 23:1-23:36.

[71] Grauman K. and Darrel T., *The Pyramid Match Kernel: Discriminative Classification with Sets of Image Features.* Proceedings of the Internation Conference on Computer Vision (ICCV), 2005. pp. 1458-1465.

[72] Griffin G. and Perona P., *Learning and Using Taxonomies for Fast Visual Categorization.* IEEE CVPR 2008, 2008.

[73] Griffiths A.; Luckhurst H.C. and Willett P., *Using Inter-Document Similarity Information in Document Retrieval Systems.* Journal of the American Society for Information Science, 1986. 37:3-11.

[74] Guha S., et al., *Clustering Data Streams.* Proceedings of the Annual Symposium on Foundations of Computer Science (FOCS), 2000. pp. 359–366.

[75] Gupta G. and Ghosh J., *Bregman Bubble Clustering: A Robust Framework for Mining Dense Clusters.* ACM Transactions on Knowledge Discovery from Data, 2008. 2(2): 8:1-8:49.

[76] Han S., et al., *Extracting Representative Images of Tourist Attractions from Flickr by Combining an Improved Cluster Method and Multiple Deep Learning Models.* . ISPRS International Journal of Geo-Information, 2020. 9(2): 81.

[77] Hastie T., T.R. and Friedman J., *The Elements of Statistical Learning: Data Mining, Inference, and Prediction - Second Edition.* Spinger, New York, 2008. pp. 763.

[78] He X., G.T., Roqueiro D., and Borgwardt K., *Kernel Conditional Clustering and Kernel Conditional Semi-supervised Learning* Knowledge and Information Systems, 2020. 62(3): 899-925.

[79] He X.; Cai D.; Wen J.R.; Ma W.Y and Zhang H.J., *ImageSeer: Clustering and Searching WWW Images Using Link and Page Layout Analysis.* Microsoft Technical Report - MSR-TR-2004-38, 2004.

[80] He X.; Ma W.Y. and Zhang H. J., *ImageRank: Spectral Techniques for Structural Analysis of Image Database.* IEEE International Conference on Multimedia and Expo, , 2003. Vol 2, pp. 25 - 28.

[81] Hearst M.A.; Karger D.R. and Pedersen J.O., *Scatter/Gather as a Tool for the Navigation of Retrieval Results.* The AAAI Symposium on AI Appliations in Knowledge Navigation and Retrieval, 1995. Cambridge, MA.

[82] Hirota M., et al., *A Robust Clustering Method for Missing Metadata in Image Search Results.* Journal of Information Processing, 2012. 20(3):537-547.

[83] Hirota M.; Yokoyama S.; Fukuta N. and Ishikawa H., *Constraint-based Clustering of Image Search Results using Photo Metadata and Low-level Image Features.* Proceedings of the 9th IEEE/ACIS International Conference on Computer and Information Science (ICIS'10), 2010.

[84] Hofmann T., *Probabilistic Latent Semantic Indexing.* SIGIR Forum, 2017. 51(2): 211-218.

[85] Hopfield J. and Tank D., *Neural Computation of Decisions in Optimization Problems.* . Biological Cybernetics, 1985. 52(3):52–141.

[86] Hua K.A.; Vu K. and Oh J.H., *Proceedings of the 7th ACM Intertional Multimedia Conference (ACM MM'99)* pp. 225-234, SamMatch: A Flexible and Efficient Sampling-based Image Retrieval Technique for LArge Image Databases

[87] Huang Z., et al., *Mining Near-duplicate Graph for Cluster-based Re-ranking of Web Video Search Results.* ACM Transactions on Information Systems (TOIS), 2010. 28: 22:1-22:27.

[88] Hussain S. and Haris M., *A K-Means based Co-Clustering (kCC) Algorithm for Sparse, High-dimensional Data.* Expert Systems and Applications, 2019. 118: 20-34.

[89] Ionescu B., et al., *Retrieving Diverse Social Images at MediaEval 2013: Objectives, Dataset and Evaluation.* Working Notes Proceedings MediaEval 2013Workshop, Eds. Larson M. et al., co-located with ACM Multimedia, Barcelona, Spain, 2013. Vol 1043.

[90] Ionescu B., et al., *Div150Cred: A Social Image Retrieval Result Diversification with User Tagging Credibility Dataset.* ACM Multimedia Systems (MMSys), 2015. DOI: 10.1145/2713168.2713192.

[91] Ionescu B., et al., *Result Diversification in Social Image Retrieval: A Benchmarking Framework.* Multimedia Tools and Applications (MTAP), 2014. pp. 1–31.

[92] Jain A.K.; Murty M.N. and Flynn P.J., *Data Clustering: A Review.* ACM Computing Surveys, 1999. 31(3):264-323.

[93] Jardine N. and van Rijsbergen C.J., *The Use of Hierarchical Clustering in Information Retrieval.* Information Storage and Retrieval, 1971. 7:217-240.

[94] Jeon J.; Lavrenko V. and Manmatha R., *Automatic Image Annotation and Retreival using Cross Media Relevance Models.* Proceedings of the International ACM SIGIR Conference, 2003. pp. 119-126.

[95] Ji Z., et al., *A Survey of Personalised Image Retrieval and Recommendation.* . National Conference on Theoretical Computer Science (NCTCS'17) 2017. pp. 233-247.

[96] Jia Y., et al., *Finding Image Exemplars using Fast Sparse Affinity Propagation.* In Proceedings of the ACM Multimedia, 2008. pp. 639–642.

[97] Jiang C., L.J., et al., *Implicit Semantics Based Metadata Extraction and Matching of Scholarly Documents.* . Journal of Database Management, 2018. 29(2): 1-22.

[98] Jiang Q., et al., *An Adaptive CSP and Clustering Classification for Online Motor Imagery EEG.* IEEE Access, 2020. 8: 156117-156128.

[99] Jisha K.P., *An image retrieval technique based on texture features using semantic properties.* International Conference on Signal Processing Image Processing & Pattern Recognition (ICSIPR), 2013. pp. 248 - 252

[100] Joshi D., D.R., et al., *Aesthetics and Emotions in Images.* IEEE Signal Processing Magazine, 2011. 28(5): 94-115.

[101] Kailing K., et al., *Efficient Similarity Search for Hierarchical Data in Large Databases.* Proceedings of the International Conference on Extending Database Technology, 2004. pp. 676-693.

[102] Kamvar M., et al., *Computers and Iphones and Mobile Phones, oh my!: a Logs-based Comparison of Search Users on Different Devices.* 18th International World Wide Web Conference (WWW), 2009. pp. 801–810.

[103] Kamvar M. and Baluja S., *A Large Scale Study of Wireless Search Behavior: Google Mobile Search.* In Proceedings of the SIGCHI Conference on Computer Human Interaction, 2006. pp. 701–709.

[104] Kleinberg J., *Authoritative Sources in a Hyperlinked Environment.* Journal of ACM, 1999. 46(5):604–632.

[105] Kong S., et al., *Photo Aesthetics Ranking Network with Attributes and Content Adaptation.* European Conference on Computer Vision (ECCV'16), 2016. 1:662-679.

[106] Krapac J., et al., *Improving Web Image Search Results using Query-relative Classifiers* Computer Vision and Pattern Recognition (CVPR), 2010. pp. 1094-1101.

[107] Kriegel H. P., Schubert E., and Zimek A., *The (Black) Art of Runtime Evaluation: Are we Comparing Algorithms or Implementations?* Knowledge and Information Systems, 2016. 52(2):341.

[108] Krishnan A., et al., *Leveraging Semantic Resources in Diversified Query Expansion.* World Wide Web journal, 2018. 21:1041–1067.

[109] Kulkarni S. and Verma B., *Fuzzy Logic for Texture Queries in CBIR.* Proc. of the International Conference on Computational Intelligence and Multimedia Applications (ICCIMA), 2003. pp. 223-226.

[110] Lempel R. and Soffer A., *PicASHOW: Pictorial Authority Search by Hyperlinks on the Web".* Proceedings of the 10th International World Wide Web Conference, 2001. pp. 438-448.

[111] Leouski A.V. and Croft B., *An Evaluation of Techniques for Clustering Search Results.* TEchnical Report IR-76, 1996. Computer Science Department, University of Massachusetts.

[112] Leow W.K. and Lai S.Y., *Scale and Orientationi-Invariant Texture Matching for Image Retrieval* In Pietikainen (Ed.) Texture Analysis in Machine Vision, 2000. pp. 151-163, World Scientific, Singapore.

[113] Li P., Zhang L., and Ma J., *Dual-ranking for web image retrieval.* CIVR, 2010. pp. 166-173.

[114] Li X., et al., *Socializing the Semantic Gap: A Comparative Survey on Image Tag Assignment, Refinement, and Retrieval.* ACM Computing Surveys 2016. 49(1): 14:1-14:39.

[115] Liang S. and Sun Z., *Sketch Retrieval and Relevance Feedback with Biased SVM Classification.* Pattern Recognition Letters, 2008. 29: 1733-1741.

[116] Lin W.H.; Jin R. and Hauptmann A., *Web Image Retrieval Re-Ranking with Relevance Model.* Proceedings of the IEEE Conference on Web Intelligence (WIC'03), 2003. pp. 242-249.

[117] Liu B., et al., *Encrypted Data Indexing for the Secure Outsourcing of Spectral Clustering.* Knowledge and Information Systems, 2019. 60(3): 1307-1328.

[118] Liu F. and Picard R.W., *Periodicity, Directionality, and Randomness: Wold Features for Image Modelling and Retrieval.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 1996. 18(7):722-733.

[119] Liu G. and Lee B., *A Color-based Clustering Approach for Web Image Search Results* International Conference on Hybrid Information Technology (ICHIT'09), 2009. pp. 481-484.

[120] Liu H., et al., *Clustering-based Navigation of Image Search Results on Mobile Devices.* In: Myaeng, S.-H., Zhou, M., Wong, K.-F., Zhang, H.-J. (eds.) AIRS, 2005. 3411: 325–336.

[121] Liu H.; Xie X; Tang X.O.; Li Z.W. and Ma W.Y., *Effective Browsing of Web Image Search Results.* Proceedings of the ACM SIGMM International Workshop on Multimedia Information Retrieval, 2004. pp. 84-90.

[122] Liu M., et al., *A New Local Density and Relative Distance based Spectrum Clustering.* Knowledge and Information Systems, 2019. 61(2): 965-985.

[123] Liu Y.; Zhang D.; Lu G. and Ma W.-Y., *Region-based Image Retrieval with Perceptual Colors.* Proceedings of the Pacific-Rim Multimedia Conference (PCM), 2004. pp. 931-938.

[124] Liu Y.; Zhang D.; Lu G. and Ma W., *A Survey of Content-Based Image Retrieval with High-Level Semantics* Pattern Recognition, 2006. 40(1):262-282.

[125] Lloyd S., *Least Squares quantization in PCM.* IEEE Transactions on Information Theory, 1982. 28(2):129-137.

[126] Long F.; Zhang H.J. and Feng D.D., *Fundamentals of Content-based Image Retrieval.* In: D. Feng (Ed.), Multimedia Information Retrieval and Management, Springer, Berlin., 2003. pp. 1-26.

[127] Lozada C., et al., *Clustering of Web Search Results based on the Cuckoo Search Algorithm and Balanced Bayesian Information Criterion.* Information Sciences, 2014. 281: 248-264.

[128] Lu Y., Cheung Y., and Tang Y., *Self-Adaptive Multiprototype-Based Competitive Learning Approach: A k-Means-Type Algorithm for Imbalanced Data Clustering.* IEEE Transactions on Cybernetics, 2021. 51(3): 1598-1612.

[129] Luo B.; Wang X.G. and Tang X.O., *A World Wide Web based Image Search Engine using Text and Image Content Features.* Proceedings of IS&T/SPIE Electronic Imaging, 2003.

[130] Ma L., et al., *Learning Efficient Binary Codes From High-Level Feature Representations for Multilabel Image Retrieval.* IEEE Transactions on Multimedia, 2017. 19(11): 2545-2560.

[131] Madduma B. et al., *Image Retrieval based on High Level Concept Detection and Semantic Labelling.* Intelligent Decision Technologies, 2012. 6(3): 187-196.

[132] Manjunath B.S., *Color and Texture Descriptors.* IEEE Transactions on Circuits and Systems for Video Technology (CSVT), 2001. 6:703-715.

[133] Manjunath B.S., *Introduction to MPEG-7.* Wiley, New York, 2002. pp. 412.

[134] Manjunath B.S. and Ma W.Y., *Texture Features for Browsing and Retrieval of Image Data.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 1996. 18(8):837-842.

[135] Marszalek M. and Schmid C., *Semantic Hierarchies for Visual Object Recognition.* Computer Vision and Pattern Recognition (CVPR), 2007. DOI: 10.1109/CVPR.2007.383272.

[136] Mehrotra R. and Gary J.E., *Similar-Shape Retrieval in Shape Data Managment* IEEE Computer Society Press, 1995. 28(9):57-62.

[137] Mezaris V.; Kompatsiaris I. and Strintzis M.G., *Proceedings of the International Conference on Image Processing* (*ICIP*). Vol. 2, pp. 511-514, An Ontology Approach to Object-based Image Retrieval

[138] Miao J., Zhou X., and Huang T., *Local Segmentation of Images using an Improved Fuzzy C-means Clustering Algorithm based on Self-adaptive Dictionary Learning.* Applied Soft Computing, 2020. 91: 106200.

[139] Milligan G. and Cooper M., *An examination of procedures for determining the number of clusters in a data set.* Psychometrika, 1985. 50(52):159-179.

[140] Moellic P.A.; Haugeard J.E. and Pittel Guillaume, *Image Clustering based on a Shared Nearest Neighbors Approach for Tagged Collections.* Proceedings of the International Conference on Image and Video Retrieval (CIVR), 2008. pp. 269-278.

[141] Moraveji N. et al., *Analyzing and Searching Broadcast News Video* Informedia at TRECVID'03, 2003. http://www-nlpir.nist.gov/projects/tv.pubs.org.

[142] Moreno J. and Dias G., *Using Text-Based Web Image Search Results Clustering to Minimize Mobile Devices Wasted Space-Interface.* European Conference on Information Retrieval (ECIR), 2013. pp. 532-544.

[143] Moreno J.G. and Dias G., *Using Ephemeral Clustering and Query Logs to Organize Web Image Search Results on Mobile Devices.* International ACM Workshop on Interactive Multimedia on Mobile and Portable Devices (IMMPD'11), 2011. pp. 33–38.

[144] Morsillo N.; Pal C. and Nelson R., *Mining the Web for Visual Concepts.* Proc. of the 9th International Workshop on Multimedia Data Mining (in conjuction with ACM SIGKDD). pp. 18-25.

[145] Nguyen G.P. and Worring M., *Interactive access to large image collections using similarity-based visualization.* Journal of Visual Languages and Computing, 2008. 19(2):203-224

[146] Nguyen H., Woon Y., and Ng W., *A Survey on Data Stream Clustering and Classification.* Knowledge and Information Systems 2015. 45(3): 535-569.

[147] Niazmardi S., Safari A., and H. S., *Similarity-Based Multiple Kernel Learning Algorithms for Classification of Remotely Sensed Images.* IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2017. 10(5): 2012-2021.

[148] Nikolopoulos S., et al., *High Order pLSA for Indexing Tagged Images.* Signal Processing, 2013. 93(8): 2212-2228.

[149] O'Connell C., Kutics A., and Nakagawa A., *Layered Self-Organizing Map for Image Classification in Unrestricted Domains.* International Conference on Image Analysis and Processing (ICIAP), 2013. pp 310-319.

[150] Osinski S., Stefanowski J., and Weiss D., *Lingo: Search Results Clustering Algorithm based on Singular Value Decomposition.* In Intelligent Information Systems Conference (IIPWM), 2004. pp. 369–378.

[151] Osman T., et al., *An Algorithmic Approach to Estimate Cognitive Aesthetics of Images Relative to Ground Truth of Human Psychology through a Large User Study.* Journal of Information and Telecommunication, 2019. 3(2): 156-179.

[152] Panwong P., Boongoen T., and Iam-on N., *Improving Consensus Clustering with Noise-induced Ensemble Generation.* Expert Systems and Applications 2020. 146: 113138

[153] Papapanagiotou V., Diou C., and Delopoulos A., *Improving Concept-Based Image Retrieval with Training Weights Computed from Tags.* ACM Transactions on Multimedia Computing, Communications, and Applications, 2016. 12(2): 32:1-32:22.

[154] Paramita M., Sanderson M., and Clough P., *Diversity in Photo Retrieval: Overview of the ImageCLEFPhoto Task 2009.* Conference and Labs of the Evaluation Forum (CLEF), 2009. pp 45-59.

[155] Park H, Lee J., and Jun C., *A K-means-like Algorithm for K-medoids Clustering and Its Performance.* Proceedings of the 36th CIE Conference on Computers & In-dustrial Engineering, 2006. pp.1222-1231.

[156] Pelleg D. and Moore A., *X-means: Extending k-means with Effcient Estimation of the Number of Clusters.* In International Conference on Machine Learning (ICML), 2000. pp. 727-734.

[157] Philbin J., Sivic J., and Zisserman A., *Geometric Latent Dirichlet Allocation on a Matching Graph for Large-scale Image Datasets.* International Journal of Computer Vision 2011. 95(2):138-153.

[158] Picsearch. http://www.picsearch.com [March 2012].

[159] Popescu A.; Mollic P.; Kanellos I. and Landais R., *Lightweight Web Image ReRanking.* Proceedings of the 17th ACM International Conference on Multimedia, 2009. pp. 657-660.

[160] Punera K., Rajan S., and Ghosh J., *Automatic Construction of N-ary Tree Based Taxonomies.* IEEE International Conference on Data Mining (ICDM) Workshops, 2006. pp. 75-79.

[161] Radu A-L, et al., *A Hybrid Machinecrowd Approach to Photo Retrieval Result Diversification.* Multimedia Model, 2014. LNCS 8325:25–36.

[162] Rajendran T. and Gnanasekaran T., *Multi-level Object Relational Similarity based Image Mining for Improved Image Search using Semantic Ontology.* Cluster Computing, 2019. 22: 3115-3122.

[163] Rangan S., et al., *Inference for Generalized Linear Models via Alternating Directions and Bethe Free Energy Minimization.* IEEE Trans. Inf. Theory, 2017. 63(1): 676-697.

[164] Recio B., et al., *A Taxonomy Generation Tool for Semantic Visual Analysis of Large Corpus of Documents.* Multimedia Tools and Applications (MTAP), 2019. 78(23): 32919-32937.

[165] Reddy G., Mukherjee S., and Thakur M., *Measuring Photography Aesthetics with Deep CNNs.* IET Image Processing, 2020. 14(8): 1561-1570.

[166] Ren J.; Shen Y. and Guo L., *A Novel Image Retrieval Based on Representative Colors.* Image Vision and Computing Conf., pp. 102-107.

[167] Rocchio J., *Relevance Feedback in Information Retrieval.* Smart Retrieval System Experiments in Automatic Document Processing, Prentice Hall, Englewood Cliffs NJ, 1971. pp. 313–323.

[168] Rodden K.; Basalaj W.; Sinclair D. and Wood K.R., *Evaluating a Visualization of Image Similarity as a Tool for Image Browsing.* Proceedings of the IEEE Symposium on Information Visualization, 1999. pp. 36-43.

[169] Rodden K.; Basalaj W.; Sinclair D. and Wood K.R., *Does Organization by Similarity Assist Image Browsing?* Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2001. pp. 190-197

[170] Rohm M., et al., *Subdiv17: a Dataset for Investigating Subjectivity in the Visual Diversification of Image Search Results.* ACM SIGMM Conference on Multimedia Systems (MMSys'18), 2018. pp. 444-449.

[171] Rudinac S., Hanjalic A., and Larson M.A., *Generating Visual Summaries of Geographic Areas Using Community-Contributed Images.* IEEE Transactions on Multimedia, 2013. 15(4): 921–932.

[172] Rui Y. and C.S.-F. Huang T., *Image Retrieval: Current Techniques, Promising Directions and Open Issues.* Visual Communication and Image Representation, 1999. 10(1): 39–62.

[173] Rui Y.; Huang T.S.; Ortega M. and Mehrotra S., *Relevance Feedback: A Power Tool for Interactive Content-Based Image Retrieval.* IEEE Transactions on Circuits and Video Technology, 1998. 8(5):644-655.

[174] Ruocco M. and Ramampiaro H., *Event-related Image Retrieval: Exploring Geographical and Temporal Distribution of User Tags* International Journal of Multimedia Information Retrieval, 2013. 2(4): 273-288.

[175] Salah Z., et al., *A Methodology to Refine Labels in Web Search Results Clustering.* International Journal of Computational Intelligence Systems, 2018. 12(1): 299-310.

[176] Salton G. and Mcgill M.J., *Introduction to Modern Information Retrieval,.* 1983. McGraw-Hill, Tokio.

[177] Salvador S. and Chan P., *Determining the Number of Clusters/Segments in Hierarchical Clustering/Segmentation Algorithms.* In Proceedings of the 16th IEEE InternationalConference onTools with Artificial Intelligence (ICTA), 2004. pp. 576–584.

[178] Sanz I., et al., *An Entropy-Based Characterization of the Heterogeneity of XML Collections.* International Conference on Database and Expert Systems Applications (DEXA'08) Workshops, 2008. pp. 238-242.

[179] Scaiella U., et al., *Topical Clustering of Search Results.* In 5th ACM International Conference on Web Search and Data Mining (WSDM). pp. 223–232.

[180] Scholkopf B., Smola A.J., and Muller K.-R., *Kernel Principal Component Analysis.* Neural Computation, 1998. 10(5):1299–1319.

[181] Schubert E., et al., *DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN.* ACM Transactions on Database Systems (TODS), 2017. 42(3):19:1–19:21. .

[182] Sebe N., et al., *Similarity Matching in Computer Vision and Multimedia.* Computer Vision and Image Understanding, 2008. 110(3): 309-311.

[183] Semaan B., et al., *Toward Enhancing Web Accessibility for Blind Users through the Semantic Web.* Proceedings of the International Conference on Signal Image Technology and Internet based Systems (SITIS'13), 2013, 2013. Kyoto, Japan, pp. 247-256.

[184] Sethi I.K. and Coman I.L., *Mining Association Rules Between Low-level Image Features and High Level Concepts.* Proceedings of the SPIE Data Mining and Knowledge Discovery 2001. Vol 3, pp. 279-290.

[185] Setia L. and Burkhardt H., *Learning Taxonomies in Large Image Databases.* Proceedings of the ACM SIGIR Workshop on Multimedia Information Retrieval, 2007. Amsterdam, Holland.

[186] Shi J. and Malik J., *Normalized Cuts and Image Segmentation.* IEEE Transactions on Pattern Analysis and Machine Intelligence (IEEE TPAMI), 2000. 22(8):888-905.

[187] Shin Y., Ryo C.Y., and Park J., *Automatic Extraction of Persistent Topics from Social Text Streams.* World Wide Web journal, 2014. 17(6): 1395-1420.

[188] Smeulders A.; Worring M.; Santini S.; Gupta A. and Jain R., *Content-based Image Retrieval at the End of the Early Years.* IEEE Transactions of Pattern Analysis and Machine Intelligence, 2000. 22(12):1349–1380.

[189] Smith J.R., L.C.S., *Decoding image semantics using composite region templates.* IEEEWorkshop on Content-Based Access of Image and Video Libraries (CBAIVL-98), 1998. pp. 9–13.

[190] Soares V. et al., *Combining Semantic and Term Frequency Similarities for Text Clustering.* Knowledge and Information Systems, 2019. 61(3): 1485-1516.

[191] Song K.; Tian Y.; Gao W. and Huang T., *Diversifying the Image Retrieval Results.* Proceedings of the 14th Annual ACM International Conference on Multimedia, 2006. pp. 707-710.

[192] Stanchev P.L.; Green D. Jr. and Dimitrov B., *High Level Color Similarity Retrieval.* International Journal on Information Theory and Applications, 2003. 10(3):363-369.

[193] Sugihara K., *Using Complex Numbers in Website Ranking Calculations: A Non-ad hoc Alternative to Google's PageRank.* . Journal of Software, 2019. 14(2): 58-64.

[194] Sun J., et al., *Image Retrieval based on Color Distribution Entropy.* Pattern Recognition Letters, 2006. 27: 1122-1126.

[195] Taddesse F.G., et al., *Semantic-based Merging of RSS Items.* World Wide Web Journal: Internet and Web Information Systems Journal Special Issue: Human-Centered Web Science., 2010. 13(1-2): 169-207, Springer Netherlands.

[196] Taddesse F.G., et al., *Relating RSS News/Items.* Proceedings of the 9th International Conference on Web Engineering (ICWE'09), LNCS, 2009. pp. 44-452, San Sebastian, Spain.

[197] Takimoto H., Omori F., and Kanagawa A., *Image Aesthetics Assessment Based on Multi-stream CNN Architecture and Saliency Features.* Applied Artificial Intelligence, 2021. 35(1): 25-40.

[198] Tamura H.; Mori S. and Yamawaki T., *Texture Features Corresponding to Visual Perception.* IEEE Transactions on Systems, Man, and Cybernetics, 1978. 8(6):460-473.

[199] Taneva B., Kacimi M., and Weikum G., *Gathering and Ranking Photos of Named Entities with High Precision, High Recall, and Diversity.* ACM Web Search and Data Mining, 2010. pp. 431–440.

[200] Tao F., et al., *A Novel KA-STAP Method based on Mahalanobis Distance Metric Learning.* Digital Signal Processing 2020. 97.

[201] Tekli J., *An Overview on XML Semantic Disambiguation from Unstructured Text to Semi-Structured Data: Background, Applications, and Ongoing Challenges.* IEEE Transactions on Knowledge and Data Engineering (IEEE TKDE), 2016. 28(6): 1383-1407.

[202] Tekli J., et al., *SemIndex+: A Semantic Indexing Scheme for Structured, Unstructured, and Partly Structured Data.* Elsevier Knowledge-Based Systems, 2019. 164: 378-403.

[203] Tekli J., et al., *Full-fledged Semantic Indexing and Querying Model Designed for Seamless Integration in Legacy RDBMS.* Data and Knowledge Engineering, 2018. 117: 133-173.

[204] Tekli J., Chbeir R., and Yétongnon K., *A Fine-grained XML Structural Comparison Approach.* Proceedings of the 26th International Conference on Conceptual Modeling (ER), 2007. LNCS 4801, pp. 582-598.

[205] Tekli J., Chbeir R., and Yétongnon K., *Structural Similarity Evaluation between XML Documents and DTDs.* Proceedings of the 8th International Conference on Web Information Systems Engineering (WISE), 2007. pp. 196-211.

[206] Tekli J., Chbeir R., and Yétongnon K., *Minimizing User Effort in XML Grammar Matching.* Elsevier Information Sciences Journal, 2012. 210:1-40.

[207] Tekli J., Damiani E., and Chbeir R., *Using XML-based Multicasting to Improve Web Service Scalability.* International Journal on Web Services Research (IJWSR), 2012. 9(1):1-29.

[208] Tekli J.; Chbeir R.; Ferri F. and Grifoni P., *Toward Approximate GML Retrieval Based on Structural and Semantic Characteristics.* Proceedings of the International Conference on Web Engineering (ICWE'09), 2009. pp. 16-34.

[209] Tian D., *Research on PLSA Model based Semantic Image Analysis: A Systematic Review.* Journal of Information Hiding and Multimedia Signal Processing, 2018. 9(5): 1099-1113.

[210] Treder M., Mayor-Torres J., and Teufel C., *Deriving Visual Semantics from Spatial Context: An Adaptation of LSA and Word2Vec to generate Object and Scene Embeddings from Images.* CoRR abs/2009.09384, 2020.

[211] Trokicic A. and Todorovic B., *Constrained Spectral Clustering via Multi-layer Graph Embeddings on a Grassmann Manifold.* International Journal of Applied Mathematics and Computer Science, 2019. 29(1): 125-137.

[212] Tsikrika T. and P.A. Kludas J., *Building Reliable and Reusable Test Collections for Image Retrieval: the Wikipedia Task at Imageclef.* IEEE Multimedia, 2012. 19(3):24–33.

[213] Tu N.A., Khan K., and Lee Y., *Featured Correspondence Topic Model for Semantic Search on Social Image Collections.* Expert Systems Applications, 2017. 77: 20-33.

[214] Van Leuken R. H., Garcia L., and Olivares X., *Visual Diversification of Image Search Restuls.* Proceedings of the International World Wide Web Conference, 2009. pp. 341-350.

[215] Van Leuken R. H.; Garcia L. and Olivares X., *Visual Diversification of Image Search Restuls.* Proceedings of the International World Wide Web Conference, 2009. pp. 341-350.

[216] Van Zwol R.; Murdock V; Pueyo L.G. and Ramirez G., *Diversifying Image Search with User Generated Content.* Proceedings ot the ACM International Conference on Multimedia Information Retrieval, 2008. pp. 67-74.

[217] Vapnik V.N., *Statistical Learning Theory.* Wiley, New York, 1998. pp. 768.

[218] Vega-Pons S. and Ruiz-Shulcloper J., *A Survey of Clustering Ensemble Algorithms.* International Journal of Pattern Recognition and Artificial Intelligence, 2011. 25 (3): 337–372.

[219] Vieira M., et al., *On Query Result Diversification* IEEE International Conference on Data Engineering (ICDE'11), 2011. 11(16): 1163–1174.

[220] Villena-Román J., Lana-Serrano S., and González-Cristóbal J.C., *MIRACLE-GSI at ImageCLEFphoto 2009: Comparing Clustering vs. Classification for Result Reranking.* CLEF (Working Notes), 5 p., 2009.

[221] Vitale D., Ferragina P., and Scaiella U., *Classification of Short Texts by Deploying Topical Annotations.* In: Baeza-Yates, R., de Vries, A.P., Zaragoza, H., Cambazoglu, B.B., Murdock, V., Lempel, R., Silvestri, F. (eds.) ECIR 2012. LNCS, 2012. 7224: 376–387.

[222] Vyas K. and Frasincar F., *Determining the Most Representative Image on a Web Page.* Information Sciences, 2020. 512: 1234-1248.

[223] Wang C., et al., *KPML: A Novel Probabilistic Perspective Kernel Mahalanobis Distance Metric Learning Model for Semi-supervised Clustering.* International Conference on Database and Expert Systems Applications (DEXA'20), 2020. 2: 259-274.

[224] Wang H., et al., *Context-Based Clustering of Image Search Results.* Deutsche Jahrestagung für Künstliche Intelligenz (KI), 2009. pp. 153-160.

[225] Wang J. et al., *Interactive Browsing via Diversified Visual Summarization for Image Search Results.* Multimedia Systems, 2011. 17(5): 379-391.

[226] Wang J.Z.; Li J. and Wiederhold G., *SIMPLIcity: Semantics-Sensitive Integrated Matching for Picture Libraries.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2001. 23(9):947–963.

[227] Wang Q., et al., *Robust Fuzzy C-means Clustering Algorithm with Adaptive Spatial & Intensity Constraint and Membership Linking for Noise Image Segmentation.* Applied Soft Computing, 2020. 92: 106318.

[228] Wang S., et al., *IGroup: Presenting Web Image Search Results in Semantic Clusters.* Proceedings of the Computer-Human Interaction Conference, 2007. pp. 587-596.

[229] Wang W., et al., *Improving Multi-Histogram-Based Reversible Watermarking Using Optimized Features and Adaptive Clustering Number.* IEEE Access, 2020. 8: 134334-134350.

[230] Wang X., Chen R., and Yan F., *High-dimensional Data Clustering Using K-means Subspace Feature Selection.* Journal of Network Intelligence, 2019. 4(3): 80-87.

[231] Wang X.J.; Ma W.Y.; He Q.C. and Li X., *Grouping Web Image Search Results.* Proceedings of the International ACM Conference on Multimedia (ACM MM'04), 2004. pp. 436-439.

[232] Weinberger K.; Slaney M. and van Zwol R., *Resolving Tag Ambiguity.* Proceedings of the 16th International Conference on Multimedia (MM'08), 2008. pp. 111-120, Vancouver, Canada.

[233] World Wide Web Consortium. *The Document Object Model*. http://www.w3.org/DOM [March 2021].

[234] Wu C. and Chen Y., *Adaptive Entropy Weighted Picture Fuzzy Clustering Algorithm with Spatial Information for Image Segmentation.* Applied Soft Computing, 2020. 86.

[235] Wu F., et al., *Clustering Results of Image Searches by Annotations and Visual Features.* Telematics Informatics, 2014. 31(3): 477-491.

[236] Wu G., Chang E., and Panda N., *Formulating Context-dependent Similarity Functions.* ACM Multimedia, 2005. pp. 725-734.

[237] Wu L. and Wang Y., *Robust Hashing for Multi-view Data: Jointly Learning Low-rank Kernelized Similarity Consensus and Hash Functions.* Image and Vision Computing, 2017. 57: 58-66.

[238] Xu X.S. and Huang T.S., *Relevance Feedback in Image Retrieval: A Comprehensive Review.* Multimedia Systems, 2003. 8(6):536-544.

[239] Yang X., et al., *Web Image Search Re-Ranking With Click-Based Similarity and Typicality.* IEEE Trans. on Image Processing, 2016. 25(10): 4617-4630.

[240] Yang Y. and Pedersen J.O., *A Comparative Study on Feature Selection in Text Categorization.* Proceedings of the Fourteenth International Conference on Machine Learning (ICML), 1997. pp. 412–420.

[241] Yi X. and Allan J., *A comparative study of utilizing topic models for information retrieval.* Proceedings of the 31st European Conference on IR Research (ECIR'09), 2009. pp. 29–41.

[242] Yin D., et al., *Ranking Relevance in Yahoo Search.* Knowledge Discovery and Data Mining (KDD), 2016. 323-332.

[243] Yu H.; Li M.; Zhang H.J. and Feng J., *Color Texture Moments for Content-based Image Retrieval.* Proceedings of the International Conference on Image Processing (ICIP), 2002. pp. 24-28.

[244] Yu J., et al., *Distance Learning for Similarity Estimation.* IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI), 2008. 30(3): 451-462.

[245] Yu J., et al., *Integrating Relvance Feedback in Boosting for Content-based Image Retrieval.* IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'07), 2007. pp. 965–968.

[246] Yuan J., Luo J., and Wu Y., *Mining Compositional Features From GPS and Visual Cues for Event Recognition in Photo Collections.* IEEE Transactions on Multimedia, 2010. 12(7):705-716.

[247] Zamir O. and Etzioni O., *Web Document Clustering: A Feasibility Demonstration.* In 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), 1998. pp. 46–54.

[248] Zamir O. and Etzioni O., *Grouper: A Dynamic Clustering Interface to Web Search Results.* Proceedings of the International World Wide Web Conference, 1999. pp. 1361-1374.

[249] Zeigler C.N.; McNee S.M.; Konstan J.A. and Lausen G., *Improving Recommentation Lists Through Topic Diversification.* Proceedings of the 14th International Conference on the World Wide Web, 2005. pp. 22-32.

[250] Zeng H.J., et al., *Learning to Cluster Web Search Results.* Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), 2004. pp. 210-217.

[251] Zhang B.; Li H.; Liu Y.; Ji L.; Xi W.; Fan W.; Chen Z. and Ma W.Y., *Improving Web Search Results using Affinity Graph.* Proceedings of the 28th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2005. pp. 504-511, NY.

[252] Zhang J.; Marszalek M.; Lazebnik S. and Schmid C., *Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study.* International Journal of Computer Vision, 2007. 73(2):213-238.

[253] Zhang L., et al., *InfoAnalyzer: a Computer-aided Tool for Building Enterprise Taxonomies.* International Conference on Information and Knowledge Management (CIKM), 2004. pp. 477-483.

[254] Zhang T., Ramakrishnan R., and Linvy M., *BIRCH: An Efficient Data Clustering Method for Very Large Databases.* Proceedings of the ACM SIGMOD Conference on Management of Data, 1996. 25(2):103–114.

[255] Zhao G., et al., *Entity Disambiguation to Wikipedia using Collective Ranking.* Information Processing and Management 2016. 52(6): 1247-1257.

[256] Zhao K., et al., *Clustering Image Search Results by Entity Disambiguation.* European Conf. on Machine Learning (ECML/14), 2014. (3): 369-384.

[257] Zhong X., X.X., *Clustering-based Method for Large Group Decision Making with Hesitant Fuzzy Linguistic Information: Integrating Correlation and Consensus.* Applied Soft Computing, 2020. 87: 105973.

[258] Zhuang Y., et al., *Personalized Clustering for Social Image Search Results Based on Integration of Multiple Features.* International Conference on Advanced Data Mining and Applications (ADMA), 2012. pp. 78-90.