



Query Rewriting and Semantic Annotation in Semantic-Based Image Retrieval under Heterogeneous Ontologies of Big Data

Baoxian Jia^{1*}, Bin Meng¹, Wunong Zhang², Jia Liu¹

¹Liaocheng University, Liaocheng, 252059, China

²Henan University, Henan 475001, China

Corresponding Author Email: jiabaoxian@lcu.edu.cn

<https://doi.org/10.18280/ts.370113>

ABSTRACT

Received: 8 August 2019

Accepted: 17 November 2019

Keywords:

semantic web, ontology mapping, query rewriting, big data, semantic annotation

In the era of big data, it is of great significance to retrieve the semantic features from images by big data technique. However, most semantic query models perform poorly in actual images, which are distributed heterogeneously. Image ontology mapping provides a solution to the problem. This paper applies the H-Match algorithm to find the mapping relationship between image ontologies in peer-to-peer (P2P) environment, and rewrite user queries for heterogeneous image ontologies. The H-Match algorithm was developed under the framework called Helios evolving interaction-based ontology knowledge sharing (Helios). The weights of semantic annotation were calculated by a novel method, involving word frequency, position and feedback. The research results have great application potentials in various fields.

1. INTRODUCTION

In the era of big data, the number of digital images increases on a daily basis, creating a wealth of image resources. This calls for effective use of big data technology to organize, access, store, and retrieve information from these images. The images both present low-level visual features (e.g. color, texture and shape), and evoke various emotions among the viewers. The viewers' emotions are called high-level semantics of the images.

The computer-based description of the high-level semantics is known as the study of image emotional semantics. For example, an image can be depicted with words like "sad" and "depressed", according to the viewers' emotions induced by the image. The study of image emotional semantics classifies the images emotionally and applies them to content-based image retrieval (CBIR).

The CBIR outputs high-level semantic descriptions of images, which facilitate the query for semantics. The query results are of great importance in art, advertising, and product design. For instance, the semantic query on product design discloses the emotional impact of each design pattern on users, and helps to identify the most emotionally suitable design pattern [1].

Most semantic query models focus on a single knowledge resource, failing to consider the heterogeneous distribution of information in real-world scenarios. The emerging technique of image ontology [2] provides a solution to the problem. Through image ontology mapping, it is possible to share and reuse knowledge effectively among heterogeneously distributed image ontologies.

So far, many scholars have explored deep into image ontology mapping. Some of them discussed the entire mapping process, some focused on the concept of mapping, and some calculated the correlation of entity classes. However, most methods for image ontology mapping require the intervention

of domain experts. This is obviously impractical for engineering applications with a massive amount of image ontologies. Against this backdrop, it is urgent to develop automatic or semi-automatic strategies for image ontology mapping.

Image ontology mapping is one of the two processes in the query about heterogeneously distributed image resources. The other process is rewriting the query based on the image ontology mapping [3], that is, to reformat the local mode-based query as remote-mode based query.

This paper applies the H-Match algorithm to find the mapping relationship between image ontologies in peer-to-peer (P2P) environment. The H-Match algorithm was developed under the framework called Helios evolving interaction-based ontology knowledge sharing (Helios). The H-Match algorithm formulates semantic mapping rules, classifies images semantically through machine learning and clustering, and learns semantic annotations and rules.

2. LITERATURE REVIEW

Emotional semantics, as high-level features, occupy an important position in semantic image retrieval. Retrieving images through emotional semantics can fully satisfy the query demand. Semantic-based methods retrieve images based on the consistency of image content and queries. First, the semantic content of each image is extracted based on the visual features of the image, before setting up a mapping from low-level features to high-level ones; next, the extracted semantics are relied on to retrieve images [4].

Nevertheless, the existing computer image recognition strategies can merely extract visual features like color, texture and shape from the target image. These underlying visual features are insufficient to match the rich semantic expression in user queries [5, 6]. Therefore, three core problems must be

solved in the research of semantic-based image retrieval: semantic description of low-level visual features; mapping low-level features to high-level semantics; analyzing and recognizing the emotions expressed by visual features.

Color is the most emotional feature among all visual features of images. Large color patches can attract the user’s attention, evoking a particular emotion in his/her mind [7]. The color feature is generally expressed by approaches like color histogram, color moment and color set. Nonetheless, these approaches process the color of an image from the statistical perspective, without considering its psychological effects. Shape and contour are another two visual features widely used in emotion research.

Currently, the semantics of image features are mostly represented by dictionaries or vocabularies. Several representation methods have been developed based on artificial intelligence, e.g. semantic web, frame web and image ontology. Nevertheless, there is not yet a universal representation model suitable for various backgrounds. The key to semantic analysis of image emotions lies in selecting a suitable semantic description method.

In most semantic mapping methods, the machine needs to be trained by prior knowledge or user-provided knowledge. With the growing knowledge and learning ability, the machine becomes increasingly good at producing accurate annotations for new images. Bayesian classifier and support vector machine (SVM) are the most popular methods for semantic mapping. With solid theoretical foundation, the SVM has been successfully applied in image retrieval [8]. In the training phase of semantic mapping, image clustering technology is often adopted to classify target images into meaningful groups. The most common image clustering techniques are k-means clustering (KMC) and its variations [9].

In addition, many methods are available to learn image semantics using feedbacks [10], such as Bayesian networks, neural networks, and so on. Based on tagged image datasets, Han and Gao [11, 12] predicted the semantics of images with a correlation model, and associated visual features with keywords of images through machine learning, realizing multimodal retrieval of images. Once the high-level features are obtained, the target image can be annotated semantically, laying the basis for emotional inference and analysis.

Recent years has seen the introduction of image ontology to emotion research. Effective semantic analyses have been conducted by constructing emotional image ontology, creating an ontology-based inference mechanism, and realizing emotional semantic reasoning. Gao and Han [13, 14] built an ontology generation algorithm based on ant colony optimization (ACO), which can retrieve ontology information quickly and efficiently. Drawing on ant colony system (ACS), Zhou and Gao [15, 16] constructed a semantic retrieval model with features of uncertain description logic. Gao and Zhu [17, 18] combined the genetic algorithm (GA) and the ACO-based clustering into a modular ontology method. Wang and Wang [19, 20] improved the ACO-based clustering to divide the ontology module, and validated the effectiveness of the ontology division method.

3. IMAGE ONTOLOGY MAPPING

The semantic mapping techniques roughly fall into four categories: image classification and clustering, image semantic modelling, feedback-based semantic learning, and

special semantic mapping. However, most of these techniques require lots of prior knowledge for machine training. Otherwise, the new images cannot be annotated accurately. To solve the problem, the mapping relationship between image ontologies was established by the H-Match algorithm under the Helios framework. This framework describes the knowledge of each particle based on particle ontology and P2P interaction, and retrieves information and acquires knowledge through image ontology mapping. Our algorithm works well with a few training sets for manual annotation, and enjoys highly scalable training data and semantic concepts.

3.1 Particle ontology model

Under the Helios framework, image ontology is divided into two layers: the content knowledge on the upper layer and network knowledge on the lower layer. The upper layer provides the knowledge of each particle; each content concept has several properties and a relationship with other content concepts. The lower layer describes the IP address and other network knowledge necessary for particle interaction.

Definition 1. Particle ontology

Particle ontology PO is a quadruple $PO=(C, P, SR, LR)$, where,

$C=CCUNC$, with CC being the content concept on the upper layer and NC being the network concept on the lower layer;

P is a set of properties (e.g. property p of concept C can be expressed as a unary relation $p(C)$);

$SR=\{\text{same-as, kind-of, part-of, contains, associates}\}$ is a set of semantic relationships between concepts (e.g. the semantic relationship sr between content concepts C and D can be expressed as a binary relationship $sr(C, D)$);

LR is a set of positional relationships between content concepts and network concepts (e.g. the position relationship lr between content concept C and network concept D can be expressed as a binary relationship $lr(C, D)$).

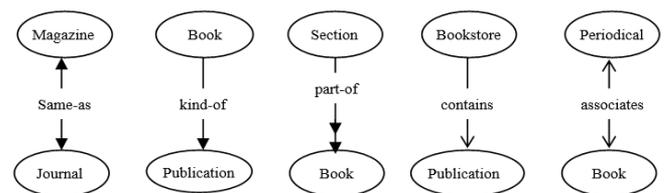


Figure 1. Five semantic relationships

As shown in Figure 1, the five semantic relationships mentioned in the definition of SR can be explained as follows:

(1) Same-as: two concepts are semantical equivalents, that is, they have the same meaning, e.g. same-as (Magazine, Journal).

(2) Kind-of: a concept embodies another concept, e.g. kind-of (Book, Publication).

(3) Part-of: a concept is part of another concept, e.g. part-of (Section, Book).

(4) Contains: a concept contains another concept, e.g. contains (Bookstore, Publication).

(5) Associates: two concepts are correlated but not in the manner of any of the four-above relationship, e.g. associations (Periodical, Book).

Definition 2. Concept context

Given a concept c , $P(c)=\{p_i|p_i(c)\}$ is the property set of c , and $SR(c)=\{c_j|sr_j(c,c_j)\}$ means: for all concepts having a

semantic relationship with c , the context $Ctx(c)$ of c is defined as the union of $P(c)$ and $SR(c)$, that is, $Ctx(c)=P(c)cSR(c)$.

Under the Helios framework, the semantic relevance between two concepts can be derived from their term relationships and semantic relationships. The H-Match algorithm computes semantic relevance based on linguistic relevance and contextual relevance.

The linguistic relevance refers to the relevance of two terms in the thesaurus. Under the Helios framework, the WordNet [6] is often employed to compute the linguistic relevance between concepts. The linguistic relevance obtained by WordNet is typically expressed as: {SYN (Synonym-of), BT (Broader term), NT (Narrower term), RT (Related term)}, where SYN, BT, NT and RT correspond to Synonym, Hypernym, Hyponym and Meronym in WordNet, respectively.

The contextual relevance is defined as the correlation of the semantic relationship between a given concept and its neighbor concept, which is called “relational relevance” RA (r, r'). It is a method to evaluate the relationship between semantic relationships, or between semantic relationships and properties. Note that r and r' either both represent the semantic relationships, or one represents the semantic relationship while the other represents the property.

Table 1. Lists the weights of term relationship and semantic relationship in our algorithm

	Relationship	Weight
Linguistic relevance	SYN	1.0
	BT/NT	0.8
	RT	0.5
Contextual relevance	Property	1.0
	Same-as	1.0
	Kind-of	0.7
	Part-of	0.6
	Contains	0.5
	Associates	0.4

3.2 Semantic parameters

Through image ontology mapping, each user query can be rewritten as different queries based on the specific mode of image ontology; these queries can be filtered by semantic parameters again, to eliminating the queries that do not satisfy the semantic parameters.

In this paper, the H-Match algorithm is adopted to compute the semantic relevance between two concepts. The semantic relevance falls in the interval of [0, 1]. The greater the semantic relevance, the two concepts are more similar. If the user is not interested in a low semantic relevance, then the low value will serve as the filter threshold.

For example, if a user queries the concept Book and wants a semantic relevance of Book of 0.45, then our algorithm will filter out those concepts whose semantic relevance with Book is smaller than 0.45.

Let t_1 and t_2 be the names of two concepts. The linguistic relevance between t_1 and t_2 is known as lexical relevance, and computed in the same way as the linguistic relevance. The lexical relevance falls in the interval of [0, 1]. If lexical relevance is 1, the two concepts are synonyms with the semantic relationship of SYN. If the user cares about both semantic relevance and lexical relevance, then the rewritten queries will be further filtered by the lexical relevance. In this paper, the lexical relevance is set to either zero or one.

4. QUERY REWRITING

Definition 3. Semantic problem

A semantic problem can be expressed as a quaternion (S, m, sa, la), where S is the image ontology structure, m is a particle in S , sa is the semantic relevance, and la is the lexical relevance.

As shown in Figure 2, the user submits a query to an image ontology O_i . Then, O_i rewrites the query in a form that its neighbor O_j can understand and process, according to the ontology mapping, semantic relevance and lexical relevance between O_i and O_j . Then, the query is submitted to O_j , which will repeat the actions of O_i . After the query passes through all particles, the result will be returned to the user.

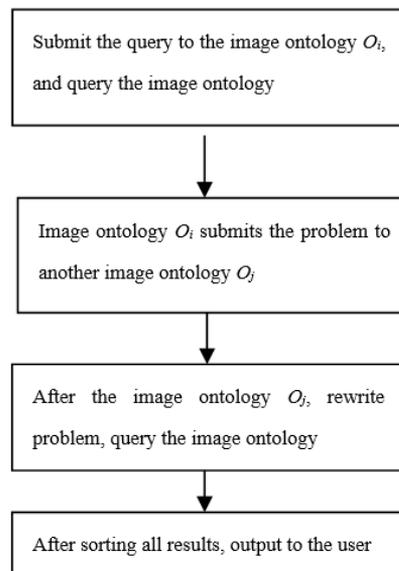


Figure 2. Handling of user queries

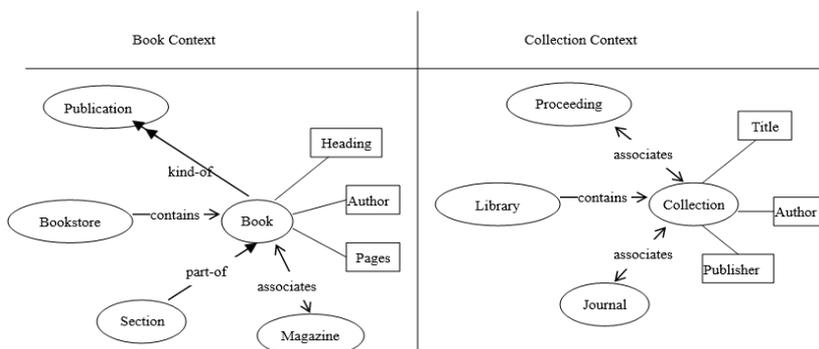


Figure 3. Example of query rewriting in different domain

Definition 4. Query rewriting

Let SA be the source structure, SB be the destination structure, and M be the mapping between SA and SB. Then, the rewriting of query QA can be expressed as a quaternion $\langle m, n, d, ld \rangle$, where particle $m \in SA$, particle $n \in SB$, d is the semantic relevance of m and n , $d \geq sa$, ld is the lexical relevance of m and n , $ld \geq la$.

Taking Figure 3 for example, the source structure and the destination structure are on the left and right, respectively. To query the concept Book, the minimum semantic relevance is 0.3 and the lexical relevance is 1. According to the H-Match algorithm, the semantic relevance between the concepts Collection and Book in the target structure is 0.4, and the lexical relevance is 1, which satisfies user requirements. Then, the query can be rewritten as $\langle Book, Collection, 0.4, 1.0 \rangle$.

5. SEMANTIC ANNOTATION

Semantic annotation refers to the analysis of file content by certain rules and procedures, and then assigns a certain number of content tags to each file, laying the basis for storage and retrieval.

The information files from the Web are generally HTMLs, which are unstructured data. To facilitate retrieval, the HTML files should be analyzed in the following steps: adding semantic information to each file in the light of existing domain ontology; extracting the semantic information from the file with an information extraction agent (e.g. resource description framework (RDF) crawler), and saving the information into the semantic metadata along with the URL of file.

The semantic metadata of a Web file, organized according to the ontology structure, not only reflects the internal information of the file (e.g. title, author and keywords), but also records the relationship between the file and other files (e.g. same file and similar file).

The traditional index-based retrieval technique can reflect some features of the file, but cannot demonstrate the file position in the entire domain or its relationship with other files. Considering human perception and psychological knowledge, this paper analyzes the emotions of images expressed on multiple low-level features, including color, shape and fluctuation.

Firstly, these low-level features were fused with the method proposed by Sun et al., and combined with psychological responses to illustrate the emotions on the image. In this way, the emotions evoked by the image can be described in a comprehensive and accurate manner. Next, the ACO was introduced to perform edge search, creating a gray boundary image. Finally, the gray boundary image was superimposed with the original image into an enhanced image. Through the above steps, the semantic features can be retrieved quickly without sacrificing the accuracy of edge extraction.

Formal ontology can be serialized into RDF statements. The RDF is a universal language to describe any type of metadata. Each RDF statement is a triple $\langle S, P, O \rangle$, where the subject S (Subject) has the property P (Property) with the value O (Object); S and P are the resource URIs, and O is a resource URI or trivial text. A set of RDF statements forms an RDF graph of multiple particles and arcs. Each particle is a resource URI or property value, and each edge is an arc. The resource is represented as an ellipse, the property value as a text or a resource URI, and the text as a rectangle.

The tag weights were calculated based on word frequency, vocabulary location, and the feedback of professional users. Among them, word frequency was selected following the ideal of the traditional weighting algorithm: term frequency-inverse document frequency (TF-IDF); the greater the frequency of a word (i.e. the concept in image ontology) in the text, the more likely for the word to convey the main meaning of the text. As for vocabulary location, the vocabulary is more important in the title, introduction and conclusions than in the conclusions. The feedback of professional users is used to manually adjust the weight of the annotation, making up the lack of understanding of automatically extracted semantics.

The weight ($weight_i$) assigned to the image ontology concept i can be computed by:

$$weight_i = fren_i + loc_i + \delta_i \quad (1)$$

where, $fren_i$ is the word frequency of image ontology concept i ; loc_i is the local concept i as the tagged position; δ_i is the feedback.

Firstly, the idea of TF-IDF was adopted to calculate the word frequency, aiming to slow down the growth of word frequency. Let n be the number of certain words in the specified file, N be the number of words segmented from the specified file, C is the total number of files in the file set, and $df(w_i)$ is the number of files containing the current word w_i . Then, the word frequency of image ontology concept i can be calculated by:

$$fren_i = \frac{n}{N} \times \log \frac{C}{df(w_i)} \quad (2)$$

The next step is to determine the position (loc_i). In most articles, the first paragraph is the introduction, and the last paragraph is conclusions. The author sums up the entire article in the title, introduction and conclusions. Thus, the same concept appearing in these three positions is more important than that appearing in the text. Here, the word frequency is fine-tuned: One appearance of a concept in the three positions is equivalent to m appearances in the text. The m value was empirically set to 3. Otherwise, the loc_i will be zero. Hence, the loc_i can be computed by:

$$loc_i = \frac{n+3}{N} \times \log \frac{c}{df(w_i)} - fren_i \quad (3)$$

The final step is to compute the feedback (δ_i). The higher the credibility of the professional users, the more correct the expression of the article meanings. Hence, the δ_i was assigned the maximum weight. The δ_i value is the sum of the maximum value obtained by the TF-IDF, and the original value calculated by the TF-IDF:

$$\delta_i = \sum_1^n \text{Max}(fren_j) + fren_i \quad (4)$$

where, n is the number of words segmented from the file; $fren_j$ is the weight calculated by the TF-IDF corresponding to the j -th word.

Substituting the values of word frequency, position and feedback of concept i into formula (1), the tag weight of the word can be obtained.

6. CONCLUSIONS

In the era of big data, the images are a treasure trove of semantic knowledge. To make effective use of this knowledge, this paper uses the H-Match algorithm to realize the mapping relationship between image ontology in P2P environment. Several semantic parameters were selected to rewrite the local mode-based query submitted by the user to the remote mode-based query. Image ontology was constructed to facilitate the implementation of the H-Match algorithm.

The research results have great application potentials in the following fields: the CBIR-based image query based on emotional semantics; art, advertising, clothing and product design; the optimization of design pattern based on psychological needs; image analysis in special fields like remote sensing detection, weather forecasting, and medicine. The future research will focus on the application of image ontology mapping.

ACKNOWLEDGMENT

Shandong Province Graduate Education Quality Improvement Program (SDYY18183).

REFERENCES

[1] Vasconcelos, N. (2007). From pixels to semantic spaces: advances in content-based image retrieval. *Computer*, 40(7): 20-26. <https://doi.org/10.1109/MC.2007.239>

[2] Jia, B.X., Huang, X., Jiao, S. (2018). Application of semantic similarity calculation based on knowledge graph for personalized study recommendation service. *Educational Sciences: Theory & Practice*, 18(6): 2958-2966. <https://doi.org/10.12738/estp.2018.6.195>

[3] Doan, A., Madhavan, J., Domingos, P., Halevy, A. (2002). Learning to map between ontologies on the semantic web. In *Proceedings of the 11th international conference on World Wide Web*, pp. 662-673. <https://doi.org/10.1145/511446.511532>

[4] Bouquet, P., Kuper, G.M., Scoz, M., Zanobini, S. (2004). Asking and answering semantic queries. In *Meaning Coordination and Negotiation Workshop (MCNW-04) in conjunction with Int. Semantic Web Conf (ISWC-04)*.

[5] Castano, S., Ferrara, A., Montanelli, S. (2003). H-match: an algorithm for dynamically matching ontologies in peer-based systems. In *Proceedings of the First International Conference on Semantic Web and Databases*, pp. 218-237.

[6] Castano, S., Ferrara, A., Montanelli, S., Zucchelli, D. (2003). Helios: a general framework for ontology-based knowledge sharing and evolution in P2P systems. In *14th International Workshop on Database and Expert Systems Applications, Proceedings*, pp. 597-603. <https://doi.org/10.1109/DEXA.2003.1232087>

[7] Miller, G.A. (1995). WordNet: a lexical database for English. *Communications of the ACM*, 38(11): 39-41. <https://doi.org/10.1145/219717.219748>

[8] Gao, K., Yang, F., Zhou, M., Pan, Q., Suganthan, P.N. (2018). Flexible job-shop rescheduling for new job insertion by using discrete Jaya algorithm. *IEEE transactions on cybernetics*, 49(5): 1944-1955. <https://doi.org/10.1109/TCYB.2018.2817240>

[9] Pan, Q.K., Tasgetiren, M.F., Liang, Y.C. (2008). A

discrete particle swarm optimization algorithm for the no-wait flowshop scheduling problem. *Computers & Operations Research*, 35(9): 2807-2839. <https://doi.org/10.1016/j.cor.2006.12.030>

[10] Gao, K., Cao, Z., Zhang, L., Chen, Z., Han, Y., Pan, Q. (2019). A review on swarm intelligence and evolutionary algorithms for solving flexible job shop scheduling problems. *IEEE/CAA Journal of Automatica Sinica*, 6(4): 904-916. <https://doi.org/10.1109/JAS.2019.1911540>

[11] Han, Y.Y., Pan, Q.K., Li, J.Q., Sang, H.Y. (2012). An improved artificial bee colony algorithm for the blocking flowshop scheduling problem. *The International Journal of Advanced Manufacturing Technology*, 60(9-12): 1149-1159. <https://doi.org/10.1007/s00170-011-3680-0>

[12] Gao, K.Z., Suganthan, P.N., Pan, Q.K., Chua, T.J., Cai, T.X., Chong, C.S. (2016). Discrete harmony search algorithm for flexible job shop scheduling problem with multiple objectives. *Journal of Intelligent Manufacturing*, 27(2): 363-374. <https://doi.org/10.1007/s10845-014-0869-8>

[13] Gao, K.Z., Suganthan, P.N., Pan, Q.K., Chua, T.J., Cai, T.X., Chong, C.S. (2014). Pareto-based grouping discrete harmony search algorithm for multi-objective flexible job shop scheduling. *Information Sciences*, 289: 76-90. <https://doi.org/10.1016/j.ins.2014.07.039>

[14] Han, Y.Y., Gong, D.W., Sun, X.Y., Pan, Q.K. (2014). An improved NSGA-II algorithm for multi-objective lot-streaming flow shop scheduling problem. *International Journal of Production Research*, 52(8): 2211-2231. <https://doi.org/10.1080/00207543.2013.848492>

[15] Zhou, G., Wang, L., Xu, Y., Wang, S. (2011). An effective artificial bee colony algorithm for multi-objective flexible job-shop scheduling problem. In *International Conference on Intelligent Computing*. Springer, Berlin, Heidelberg, pp. 1-8. https://doi.org/10.1007/978-3-642-25944-9_1

[16] Gao, K.Z., Suganthan, P.N., Pan, Q.K., Tasgetiren, M.F. (2015). An effective discrete harmony search algorithm for flexible job shop scheduling problem with fuzzy processing time. *International Journal of Production Research*, 53(19): 5896-5911. <https://doi.org/10.1080/00207543.2015.1020174>

[17] Gao, K.Z., Suganthan, P.N., Chua, T.J., Chong, C.S., Cai, T.X., Pan, Q.K. (2015). A two-stage artificial bee colony algorithm scheduling flexible job-shop scheduling problem with new job insertion. *Expert Systems with Applications*, 42(21): 7652-7663. <https://doi.org/10.1016/j.eswa.2015.06.004>

[18] Zhu, S., Zhu, C., Wang, W. (2018). A novel image compression-encryption scheme based on chaos and compression sensing. *IEEE Access*, 6: 67095-67107. <https://doi.org/10.1109/ACCESS.2018.2874336>

[19] Wang, W., Qian, Y. (2015). Adaptive $L_{1/2}$ Sparsity-Constrained NMF with half-thresholding algorithm for hyperspectral unmixing. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(6): 2618-2631. <https://doi.org/10.1109/JSTARS.2015.2401603>

[20] Wang, W., Qian, Y., Tang, Y.Y. (2016). Hypergraph-regularized sparse NMF for hyperspectral unmixing. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, pp. 1-14. <https://doi.org/10.1109/JSTARS.2015.2508448>