# A Diagram Retrieval Method with Multi-label Learning

Songping Fu[a], Xiaoqing Lu[ab*], Lu Liu[a], Jingwei Qu[a] and Zhi Tang[ab]

[a] Institute of Computer Science & Technology, Peking University, Beijing, China;
[b] State Key Laboratory of Digital Publishing Technology, Beijing, China

## ABSTRACT

In recent years, the retrieval of plane geometry figures (PGFs) has attracted increasing attention in the fields of mathematics education and computer science. However, the high cost of matching complex PGF features leads to the low efficiency of most retrieval systems. This paper proposes an indirect classification method based on multi-label learning, which improves retrieval efficiency by reducing the scope of compare operation from the whole database to small candidate groups. Label correlations among PGFs are taken into account for the multi-label classification task. The primitive feature selection for multi-label learning and the feature description of visual geometric elements are conducted individually to match similar PGFs. The experiment results show the competitive performance of the proposed method compared with existing PGF retrieval methods in terms of both time consumption and retrieval quality.

**Keywords:** multi-label learning, multi-class classification, diagram retrieval, shape analysis, plane geometry figure

## 1. INTRODUCTION

The development of computer-aided instructions has led to the digitization of an increasing number of teaching materials. Plane geometric figures (PGFs) (Fig. 1) are important resources for digital education. Several methods have been proposed to retrieve PGF-based geometry questions in the question stem rather than keyword-based geometry questions. In these retrieval systems, additional shape features are adopted to achieve highly accurate search results. However, the matching cost increases rapidly because these methods have to compare a query PGF with all the PGFs in the database by using delicate shape descriptors. One of the radical solution strategies to improve retrieval efficiency is to reduce the amount of matching operations by classification. Thus, if PGFs can be classified before matching, a query PGF only needs to be compared with the PGFs in the same category. Retrieval efficiency and accuracy can be improved by using a narrow comparison range (up to small groups) rather than the whole database.
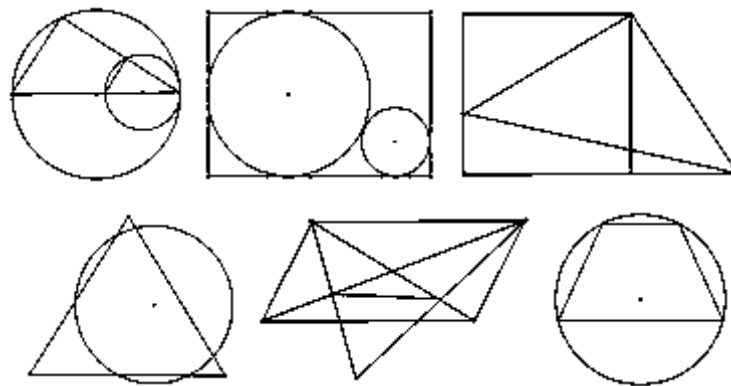


Figure 1. Examples of PGFs.

---

* Further author information: (Send correspondence to Xiaoqing Lu)
Songping Fu: E-mail: fusongping@pku.edu.cn
Xiaoqing Lu.: E-mail: lvxiaoqing@pku.edu.cn, Telephone: 86-13910717378

However, PGF classification is a challenging problem. First, no universally accepted standard exists for PGF classification in computer science or geometry pedagogy. Second, the limited types of geometry elements, such as triangles, rectangles, and circles, lead to highly similar PGFs (Fig. 1). Third, relatively compact PGF structures, which involve various complicated spatial relationships among geometry elements, present deep-seated obstacles for developing effective classification methods.

In this paper, we solve the classification problem of PGFs by multi-label learning. In contrast to the traditional methods of supervised learning with a single classification label, each PGF can be assigned with multiple labels in our system. Although the multi-label method blurs the boundaries of the categories, it provides some indirect information about PGF classification and provides a promotional function for the PGF retrieval system. The rest of this paper is organized as follows. Section 2 discusses existing similarity methods on multi-label classification and PGF retrieval. Section 3 introduces the learning method of the multi-label classification and describes the details of feature selection combined with PGF matching. Section 4 presents our experiments and evaluation results. Section 5 concludes this paper.

## 2. RELATED WORKS

Multi-label classification methods can be categorized into two main categories[1]: i) problem transformation methods and ii) algorithm adaptation methods. Problem transformation methods transform the multi-label classification problem either into one or more single-label classification or regression problem. Representative algorithms include Binary Relevance (BR)[2], Calibrated Label Ranking (CLR)[3], Label PowerSet (LP)[4], Random k-labelsets (RAkEL)[5], and Classifier Chains(CC)[6]. Algorithm adaptation methods extend specific learning algorithms to address multi-label data directly. Representative algorithms include CML[7] and multi-label k-nearest neighbor (ML-kNN)[8].

Multi-label learning has been widely applied to different problems, such as text categorization[9], music classification[10], and semantic image classification[2,11,12,13]. For multi-label image classification problem, several multi-label learning approaches have been proposed recently. Bang et al.[11] propose a high-order label-correlation-driven active learning approach that allows the iterative learning algorithm to select the informative example-label pairs in which the algorithm learns; thus, the algorithm learns an accurate classifier with less annotation efforts. Furthermore, they tackle image classification as a multi-label batch mode active learning problem. Zhang et al.[12] propose a new image multi-label annotation method based on double-layer probabilistic latent semantic analysis (PLSA). The new double-layer PLSA model is constructed to bridge the low-level visual features and high-level semantic concepts of images for effective image understanding. Sun et al.[13] propose a mthod to learn a sparse structure of label dependency. The obtained sparse label dependency structure discards the outlying correlations between labels, thus making the learned model generalizable for future samples.

Several methods with better performance than previous multi-label classifiers have recently been proposed. Wang[14] proposes a method to enhance multi-label classification by modeling dependencies among labels by using a Bayesian network. Enrique Sucar[15] introduces a method for chaining Bayesian classifiers; the method combines the strengths of classifier chains and Bayesian networks for multi-label classification. Yu et al.[16] present two novel multi-label classification algorithms based on variable precision neighborhood rough sets; these algorithms are called multi-label classification with rough sets (MLRS) and MLRS with local correlation. These algorithms consider two important factors that affect prediction accuracy, namely, the correlation among labels and the uncertainty that exists within the mapping between the feature space and label space.

The shape-based PGF retrieval problem has gradually attracted considerable attention. Li et al.[17] propose a method to analyze and decompose a PGF into separate special quadrangles. Their method can extract rectangles, parallelograms, and trapezoids in PGFs. Feng et al.[18] propose a structure analysis method to find distinguishing compound shapes to describe overlapped PGFs. This structure analysis method can reflect the structures and layouts of figures compared with classical low-level shape descriptors. Liu et al.[19] propose a PGF image retrieval method based on a new shape descriptor, bag of shapes. Seo et al.[20] present a diagram understanding method that identifies the visual elements in a diagram while maximizing the agreement between textual and visual data. Their ultimate goal is to build an automated system that can solve geometry questions.

# 3. METHOD

## 3.1 Outline of the PGF Retrieval System

Our PGF retrieval system contains two workflows (Fig. 2). The first workflow is used to establish the classification model of multi-label PGFs and involves preprocessing, feature extraction, and multi-label training. The second workflow retrieves query PGFs and involves preprocessing, feature extraction, label prediction, and matching.
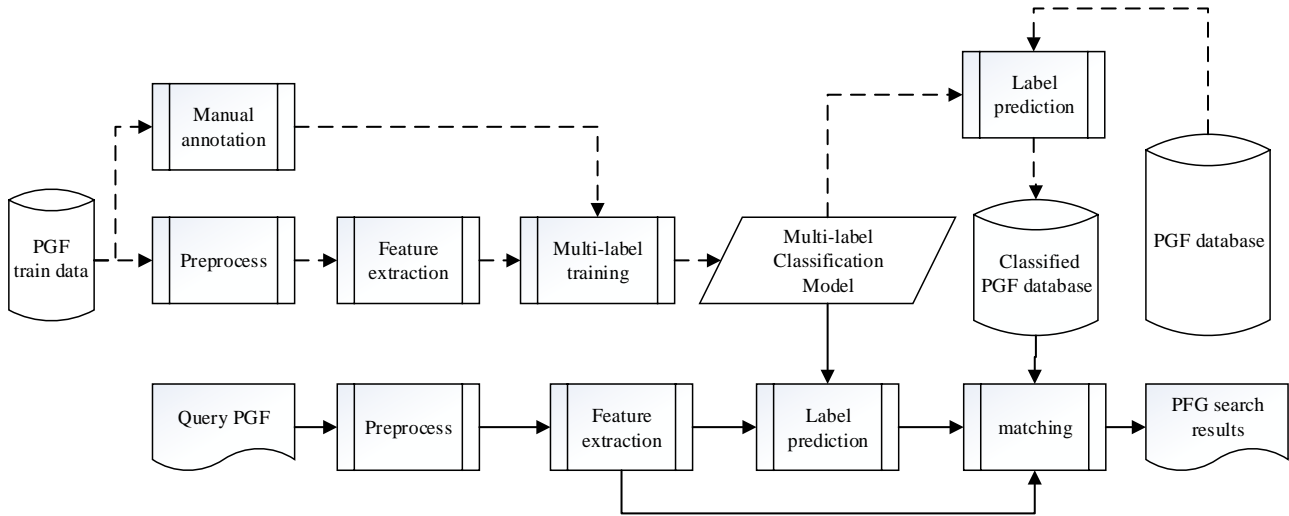


Figure 2. Flowchart of the proposed retrieval system of PGFs.

The classification model of PGFs plays an important role in the proposed PGF retrieval system. An effective multi-label learning method and supporting features need to be explored and adopted. The model is trained with ground truth and then used to assign suitable labels automatically for all PGFs in the database offline. By using the online prediction of the labels of a query PGF, the relative candidate PGFs can be filtered according to at least one common label shared by the PGFs. Further details on multi-label classification is described in Sec. 3.2.

In the PGF matching part, elaborate features are used to accurately compare a query PGF and every candidate PGF. These various features are extracted by different descriptors, and all of these descriptors are represented as a unified form of vectors. The differences of the feature vectors (FVs) are calculated by using cosine similarity. This topic is expanded in Sec. 3.3.

## 3.2 PGF Multi-Label Classification

### 3.2.1 PGF labels

Considering the main visual elements and their relationship with the PGF position, we select two types of labels, i.e., main element and position relation labels. Main element labels include various shapes, such as circles, triangles, and quadrangles. Position relation labels include intersection and inclusion relations. Thus, the labels selected for PGFs are $L = \{circle, triangle, quadrangle, others, \ intersection \ relation, inclusion \ relation\}$. Some PGF examples with labels are shown in Tab. 1.

### 3.2.2 Multi-label classification method

The performance of multi-label classification relies on exploiting the underlying label correlations. A brief analysis of the correlations among PGF labels is as following. No more than one main element label exists for PGFs with dominant visual elements. For example, if an instance is labeled with a circle, this instance is unlikely to be labeled with other main element labels. Moreover, a label is likely to co-occur with certain types of labels. If a PGF is labeled with inclusion relation, such a PGF is likely to be labeled with a circle.

A natural approach to address the multi-label classification of PGFs is to transform the multi-label learning problem into a multi-class classification problem. LP[4] considers each distinct label combination in a multi-label training set as an

individual class. Some examples of these classes are shown in Tab. 2 with their corresponding label sets and PGFs. By introducing the aforementioned new classes, the multi-label classification is converted into a single-label classification.
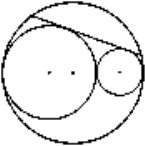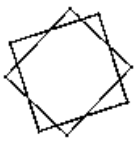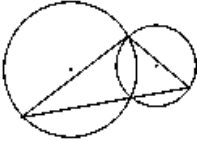
Table 1. Examples of PGFs and their labels.
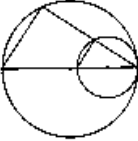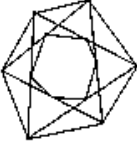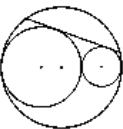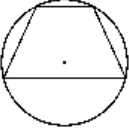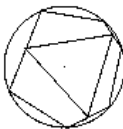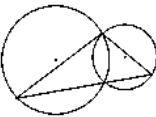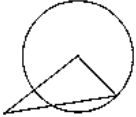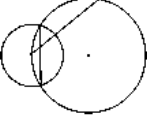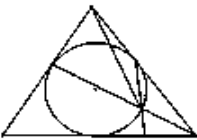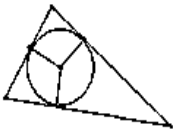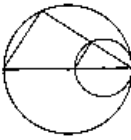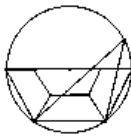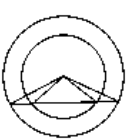
| PGFs | labels | PGFs | labels | PGFs | labels |
|------|--------|------|--------|------|--------|
|  | Circle, inclusion relation |  | Quadrangle, intersection relation |  | Triangle, inclusion relation |
|  | Circle, intersection relation |  | Circle, intersection relation, Inclusion relation |  | Others, intersection relation, Inclusion relation |

Table 2. Class examples of LP with the corresponding label sets and instances

| Classes | Label sets | Instances |
|---------|-----------|-----------|
| … | …… | …… |
| Class i | Circle, inclusion relation |  ………… |
| Class i+1 | Circle, intersection relation |  ………… |
| Class i+2 | Triangle, inclusion relation |  ………… |
| Class i+3 | Circle, inclusion relation, intersection relation |  ………… |
| … | …… | …… |

The alternative classification based on LP has obvious drawbacks: the large number of label sets increases the computational cost. Furthermore, many labels associating with very few training examples make the learning quite difficult. To overcome these limitations, the method based on RAkEL[5] is adopted. The main strategy is to randomly break the initial set of labels into $m$ subsets. Each subset has $k$ labels, called $k$-labelset. The LP method is employed to train the $m$ corresponding multi-label classifiers. For the multi-label classification of a new instance, its labels are the combination of all the LP classifiers, which calculates the mean of these predictions for each label and outputs a final decision.

To classify PGFs, the label set $L$ introduced in Sec. 3.2.1 is redefined as $= \{\lambda_i : i = 1 \dots |L|\}$ . For the training process, given the number of the LP classifiers $m$ and size of the k-labelset, the algorithm iteratively employs LP on m random k-labelset, each training a multi-label classifier. For each iteration, a k-labelset $R_i$ is randomly selected from $L^k$ (the set of all distinct k-labelsets of $L$) without replacement. The multi-class training set $D^+$ is then constructed on the basis of the original multi-label training set $D$ and $R_i$. Finally, a LP classifier $h_i$ is trained on $D^+$. For the multi-label classification of an unlabeled PGF $\vec{x}$, each LP classifier $h_i$ yields the binary predictions $h_i(\vec{x}, \lambda_j)$ for each label $\lambda_j$ in the corresponding k-labelset $R_i$. For each label $\lambda_j$, two measurements are calculated by Eqs. (1) and (2).

$$\phi(\vec{x}, \lambda_j) = \sum_{i=1}^{m} [\![\lambda_j \in R_i]\!] \quad (1 \le j \le |L|), \tag{1}$$

$$\varphi(\vec{x}, \lambda_j) = \sum_{i=1}^{m} [\![(\lambda_j \in R_i) \&\& h_i(\vec{x}, \lambda_j)]\!] \quad (1 \le j \le |L|), \tag{2}$$

where $\phi(\vec{x}, \lambda_j)$ counts the number of $k$-labelsets containing $\lambda_j$, $\varphi(\vec{x}, \lambda_j)$ counts the number of $k$-labelsets containing $\lambda_j$, and the corresponding classifier outputs a positive prediction. Accordingly, the predicted label set of the PGF corresponds to the following:

$$\lambda = \{\lambda_j | \varphi(\vec{x}, \lambda_j) / \phi(\vec{x}, \lambda_j) > 0.5, 1 \le j \le |L| \}. \tag{3}$$

The pseudo code for the multi-label classification of PGFs is presented in Fig. 3.

---

**Multi-label classification of PGFs**

**Input**: the labels of PGF classification $L = \{\lambda_i : i = 1 \dots |L|\}$, multi-label training set $D$, number of Label Powerset classifiers $m$, labelset size $k$, unlabeled instances $X$.

**Output:** predicted label sets for unlabeled PGFs

1.   **for** $i \leftarrow 1$ **to** $m$ **do**
2.       randomly choose a $k$-labelset $R_i$ without replacement;
3.       construct multi-class training set $D^+$ based on $D$ and $R_i$;
4.       train a LP classifier $h_i$ based on $D^+$;
5.   **end for**
6.   **for** each unlabeled instance $\vec{x}$ in $X$ **do**
7.       **for** $j \leftarrow 1$ **to** $|L|$ **do**
8.          calculate $\phi(\vec{x}, \lambda_j)$ according to Eq. (1);
9.          calculate $\varphi(\vec{x}, \lambda_j)$ according to Eq. (2);
10.      **end for**
11.      obtain the predicted label set according to Eq. (3);
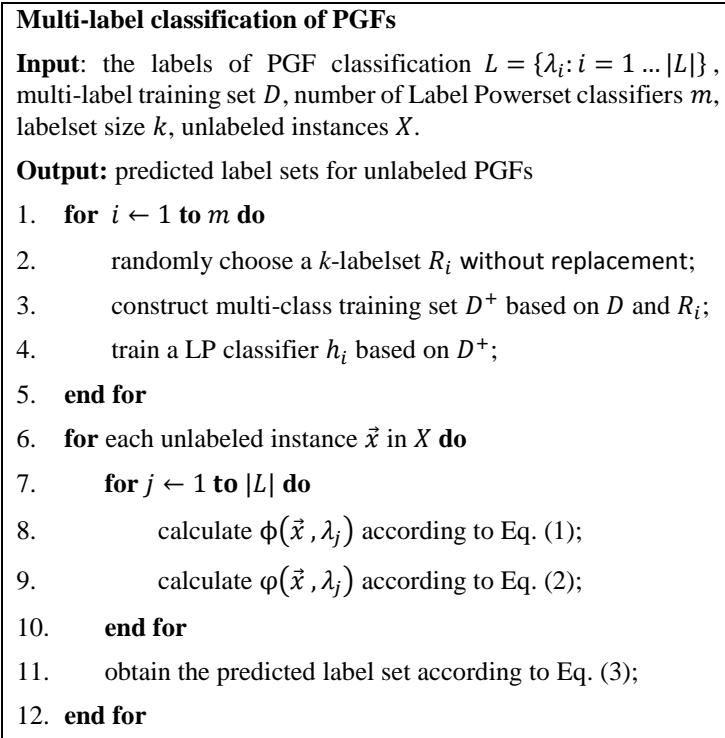12. **end for**

---

Figure 3. Multi-label classification of PGFs

### 3.2.3 Feature selection for multi-label classification

Feature selection plays an important role in classification. Following four types of features are selected, and then compose a feature vector for multi-label learning.

(1)   Contour feature

To predict the main elements accurately, we employ the boundary information in PGFs. The boundary information of a PGF is obtained by the envelope extraction method proposed by Song[22]. Moreover, the curvature is adopted to represent the boundary information because the efficiency of the curvature, as well as the invariance of scaling, rotation, and translation, which is verified by experiments for shape recognition. Finally, a shape descriptor based on curvatures is used as one of the features for multi-label classification.

(2)   Area ratio average

A PGF consists of a number of graphic primitives, such as line segments and arcs. Each primitive is only a small part of a certain PGF and has a negligible effect on shape analysis. However, some statistic information based on all primitives are helpful for multi-label classification. The following formula computes the area ratio average of all graphic primitives as a type of feature:

$$FV_{Area} = \frac{\sum_{j=1}^{n} A_j}{A_{BB}*n},$$  (4)

where $A_j$ is the area of primitive $S_j$ in image $I$, $j = 1,2,...,n$, $n$ is the number of primitives in image $I$, and $A_{BB}$ is the area of the bounding box of $I$.

(3)   Circumference ratio average

Similar to the area ratio average, we select the circumference ratio average as another type of feature:

$$FV_{Circumference} = \frac{\sum_{j=1}^{n} C_j}{C_{BB}*n},$$  (5)

where $C_j$ is the circumference of primitive $S_j$ in image $I$, $j = 1,2,...,n$, $n$ is the number of primitives in image $I$, and $C_{BB}$ is the circumference of the bounding box of $I$.

(4)   Zernike Moments

Region-based shape descriptors use the whole area information in a shape and can be applied to our classification. Zernike Moments (ZMs) are selected as another type of learning feature because ZMs are commonly used in the field of pattern recognition. ZMs are orthogonal moments[23] derived from orthogonal Zernike polynomials:

$$V_{nm}(x,y) = V_{nm}(r\cos\theta, \sin\theta) = R_{nm}(r)exp(jm\theta),$$  (6)

where $R_{nm}(r)$ is the orthogonal radial polynomial:

$$R_{nm}(r) = \sum_{s=0}^{(n-|m|)/2}(-1)^s \frac{(n-s)!}{s!\times\left(\frac{n-2s+|m|}{2}\right)!\left(\frac{n-2s-|m|}{2}\right)!} r^{n-2s},$$  (7)

n = 0, 1, 2, . . . ; $0 \le |m| \le$ n; n − |m| is even.

Zernike polynomials are a complete set of complex valued functions that are orthogonal over the unit disk, i.e., $x^2 + y^2 \le 1$. The ZM of order n with repetition m of the shape region f(x, y) is given by the following:

$$Z_{nm} = \frac{n+1}{\pi}\sum_r \sum_\theta f(r\cos\theta, r\sin\theta) R_{nm}(r)exp(jm\theta) \quad r \le 1.$$  (8)

## 3.3 PGF Matching

We apply the two-step retrieval framework: the candidate results are first obtained and then ranked according to matching similarity. For a query image, the proposed multi-label classification results are used as a filter to obtain PGF candidates. Thereafter, we retrieve PGFs in the predicted categories instead of the whole PGF database. In this section, we initially introduce the features selected for PGF matching and then discuss in detail the similarity measure for matching.

### 3.3.1 Feature selection for PGF matching

By using the multi-label filtered candidate sets, we employ the classic vector similarity for ranking. The features applied for the PGF matching process are as follows:

(1) Single shape feature

A PGF is composed of several visual geometric elements. We consider 10 types of basic shapes: line, triangle, trapezoid, rectangle, regular polygon (5–8 sides), parallelogram, and circle. The feature value is the shape frequency.

(2) Dual-shape feature

Considering that basic single shape alone is insufficient to describe the PGF layout, 41 strongly correlated compound shapes are selected as dual-shape structures.

(3) Main visual geometric element feature

The dominant geometric element is considered more salient than small elements. We introduce the shape attributes of the top 3 largest (in area) visual geometric elements as the main geometric element feature.

(4) Global feature

To improve feature robustness, six types of global metrics are selected to build the global feature. These metrics are the centroid distance average, centroid distance average inner type, circularity, compactness, minimum/maximum axis length average, and normalized radial length.

Thus, a feature vector (FV) is composed of four types of features: single-shape feature, dual-shape feature, main visual geometric element feature, and global Feature (Eq. (9)). More details can be found in Ref. 19.

$$FV = [FV_{single}, FV_{dual}, FV_{main}, FV_{global}]. \tag{9}$$

### 3.3.2 Similarity measure

The cosine similarity for matching is adopted in our retrieval system. The cosine similarity between two PGF feature vectors, namely, $FV_x = (x_1, x_2, \cdots, x_t)$ and $FV_y = (y_1, y_2, \cdots, y_t)$, is calculated by Eq. (10):

$$\cos(\theta) = \frac{FV_x \cdot FV_y}{|FV_x||FV_y|} = \frac{\sum_{k=1}^{t}(x_k \cdot y_k)}{\sqrt{\sum_{k=1}^{t} x_k^2} \cdot \sqrt{\sum_{k=1}^{t} y_k^2}}. \tag{10}$$

## 4. EXPERIMENTS

In this section, we first introduce the dataset and then present the multi-label classification experiment results. Finally, we compare our method with related and state-of-the-art methods to validate the effectiveness and efficiency of our proposed PGF retrieval method.

Our experiments are conducted on a computer with an Intel Core i5 (3.20 GHz) processor, 4 GB RAM, and Windows 7 OS. We use Mulan[24] for the multi-label learning experiments and evaluation.

### 4.1 Dataset

The PGF database contains 267 black/white PGFs. We establish a $267 \times 267$ matrix to score the similarity between each PGF pair. The similarity between each PGF pair is scored between 0 and 5. A higher score indicates a higher similarity.

For the multi-label classification experiment, each PGF is associated with a set of labels instead of a single label. The ground truth labeling of our PGF dataset is presented in Tab. 3.

Table 3. Ground truth labeling of our PGF dataset

| label | Number of examples |
| --- | --- |
| circle | 156 |
| triangle | 78 |
| quadrangle | 43 |
| intersection relation | 78 |
| inclusion relation | 75 |

### 4.2 Multi-label Classification Experiment

#### 4.2.1 Evaluation metrics

Evaluation metrics in multi-label classification are more complicated than the traditional single-label classification, because each example can be associated with multiple labels simultaneously. Therefore, a number of evaluation metrics that are specific to multi-label learning are proposed. These evaluation metrics can be generally categorized into two

groups, i.e., example-based metrics and label-based metrics[4]. We adopt both kinds of metrics in this work.

For the definitions of these metrics in this work, we consider an evaluation dataset $(X_i, Y_i, i = 1 \dots m)$, where $Y_i \subseteq L$ is the set of true labels and $L = \{\lambda_i : i = 1 \dots q\}$. Given instance $X_i$, the predicted label set is denoted as $Z_i$.

The example-based evaluation metrics adopted in this work are as follows.

$$Hamming - Loss = \frac{1}{m} \sum_{i=1}^{m} \frac{|Y_i \cap Z_i|}{|L|}$$

$$Accuracy = \frac{1}{m} \sum_{i=1}^{m} \frac{|Y_i \cap Z_i|}{|Y_i \cup Z_i|}$$

$$Recall = \frac{1}{m} \sum_{i=1}^{m} \frac{|Y_i \cap Z_i|}{|Y_i|}$$

$$F_1 = \frac{1}{m} \sum_{i=1}^{m} \frac{2|Y_i \cap Z_i|}{|Z_i| + |Y_i|}$$

As for label-based metrics, Let $tp_\lambda, fp_\lambda, tn_\lambda, fn_\lambda$ be the number of true positives, false positives, true negatives, false negatives respect to a label $\lambda$. The micro-averaged and macro-averaged versions of a binary classification metric $B(tp, fp, tn, fn)$ are obtained as follows:

$$B_{micro} = B\left(\sum_{\lambda=1}^{q} tp_\lambda, \sum_{\lambda=1}^{q} fp_\lambda, \sum_{\lambda=1}^{q} tn_\lambda, \sum_{\lambda=1}^{q} fn_\lambda\right)$$

$$B_{macro} = \frac{1}{q} \sum_{\lambda=1}^{q} B(tp_\lambda, fp_\lambda, tn_\lambda, fn_\lambda)$$

### 4.2.2 Multi-label learning results and discussion

We compare the proposed method with the following multi-label classification algorithms: BR2, CLR3, ML-kNN8, Ensemble of Classifier Chains(ECC)6. AdaboostM1 with the C4.5 decision tree is adopted as the base classifier for BR, CLR, ECC, and the proposed method based on RAkEL. Ensemble iterations are set to 50 in ECC. For our method, the subset size k is set to 3 and the number of models m is set to 2q (q is the number of labels). For ML-kNN, the number of neighbors is set to 10 and the smoothing factor is set to 1. Ten-fold cross-validation experiments are performed for the evaluation. Table 4 compares the predictive performance of the 5 competing multi-label classification methods on our PGF dataset.

Table 4 shows that the proposed method and ECC yield superior predictive performances than the other three methods. ECC performs best under Hamming loss, accuracy, and micro-F1. The proposed method performs best under the remaining five measures that we take into consideration. We notice that the multi-label learning methods that perform well on our PGF dataset consider the correlations among labels. BR and ML-kNN are based on the first-order strategy, which ignores the correlations that exist among labels. Due to this loss of information, the predictive performance of the two methods is somewhat unsatisfactory. CLR is a multi-label learning method based on the second-order strategy, which considers the pairwise relations between labels. ECC and the proposed method are based on the high-order strategy, which has stronger correlation-modeling capabilities than the first order and second-order strategies.

Given that the multi-label classification results are used as a filter to obtain PGF candidates, we focus on the recall measure of the classification results. The performances of different methods under the recall measure illustrate that the proposed multi-label method is indeed appropriate for PGF classification.
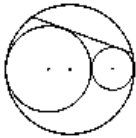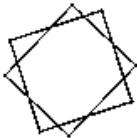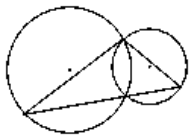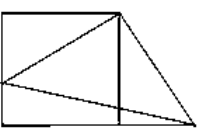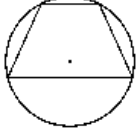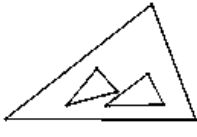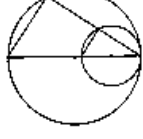
We present some classification results of several PGFs in Tab. 5. The first five PGFs are classified correctly, whereas the label prediction results of the last three PGFs are incorrect. We believe that the incorrect classification results of the sixth and eighth PGFs may be caused by the insufficiency of the training dataset because the specific position relation of both PGFs, i.e., a triangle included by a triangle and a circle included by a quadrangle, do not appear in the training dataset.

The inexactly correct labeling results of the seventh image may be caused by the selected training features. These are the two issues that we will further study next.

Table 4. Performance of the competing multi-label classification methods

| Evaluation Metrics | | BR | CLR | MLkNN | ECC | proposed |
|---|---|---|---|---|---|---|
| Example-Based | Hamming Loss | 0.2035 | 0.2111 | 0.2271 | **0.1677** | 0.2013 |
| | Accuracy | 0.5438 | 0.5382 | 0.5036 | **0.6299** | 0.6245 |
| | Recall | 0.6144 | 0.6297 | 0.5431 | 0.6648 | **0.7111** |
| | F1 | 0.6059 | 0.6037 | 0.5534 | 0.6811 | **0.6820** |
| Label-based | Micro-Recall | 0.6151 | 0.6326 | 0.5605 | 0.6501 | **0.7067** |
| | Micro-F1 | 0.6596 | 0.6594 | 0.6134 | **0.7132** | 0.6945 |
| | Macro-Recall | 0.5257 | 0.5438 | 0.4443 | 0.5386 | **0.6213** |
| | Macro-F1 | 0.5534 | 0.5570 | 0.4570 | 0.5670 | **0.6051** |

Table 5. Classification results of several PGFs

| PGFs | labels | | PGFs | labels | |
|---|---|---|---|---|---|
| | Ground truth | Predicted | | Ground truth | Predicted |
|  (1) | Circle, inclusion relation | Circle, inclusion relation |  (2) | Quadrangle, intersection relation | Quadrangle, intersection relation |
|  (3) | Circle, intersection relation | Circle, intersection relation |  (4) | Quadrangle, intersection relation | Quadrangle, intersection relation |
|  (5) | Circle, inclusion relation | Circle, inclusion relation |  (6) | Triangle, inclusion relation | Triangle |
|  (7) | Circle, intersection relation, inclusion relation | Circle, inclusion relation |  (8) | Quadrangle, inclusion relation | Quadrangle, intersection relation |

## 4.3 PGF Retrieval Experiment

On the basis of the multi-label classification experiment, we select 40% of the PGFs (107 images) as the training dataset and then predict the labels on the remaining 60% of the PGFs (160 images), which are set as the test dataset for the following retrieval experiment.

We use three classical evaluation metrics to evaluate the retrieval quality: precision, mean average precision (MAP), and normalized discount cumulative gain (NDCG)[25]. To verify the effectiveness and efficiency of our methods for image retrieval, we compare our method with the method proposed in Ref. 19. The precision, MAP, and NDCG curve against recall and rank are presented in Fig. 4. The retrieval results in Fig. 4 show the competitive performance of our method, particularly on the MAP measure, which emphasizes that relevant results should be ranked before irrelevant ones.

The average time consumed for a query image to search for the top 50 relevant images is compared with the method proposed in Ref. 19. The average retrieval time (in seconds) is reduced dramatically from 0.0068 s to 0.0024 s.

Therefore, the retrieval results illustrate the competitiveness of our method against existing methods in terms of retrieval quality and time consumed.
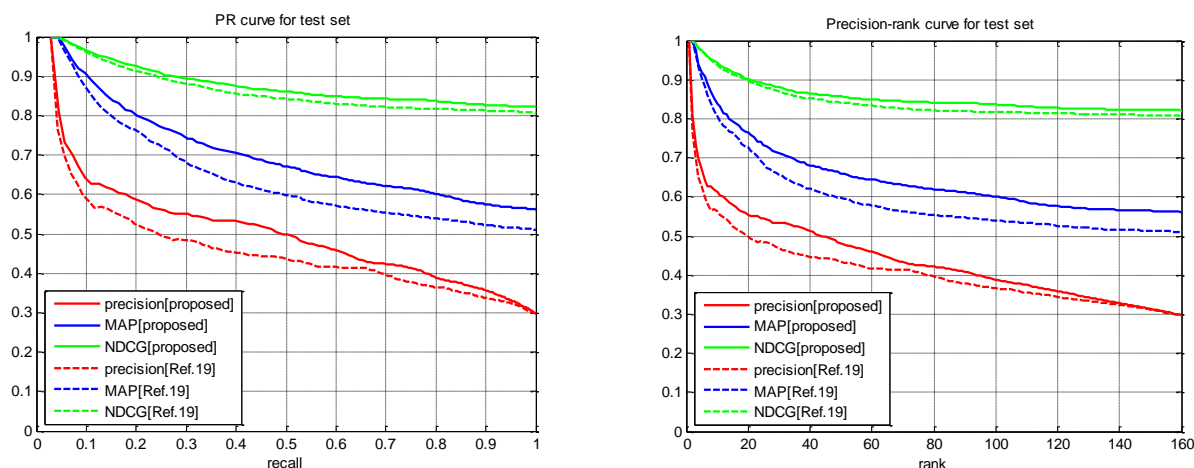


(a) Precision, MAP and NDCG curve against recall          (b) Precision, MAP and NDCG curve against rank

Figure 4.  Precision, MAP and NDCG curve

## 5. CONCLUSION

This study aims to retrieve the plane geometry figures effectively and efficiently. We introduce multi-label learning to classify PGFs. First, the multi-label classification of PGFs is performed to obtain the candidate set, considering the label correlations among PGFs. For a query PGF, we then retrieve the PGFs within the candidate set, only compare it with PGFs in the predicted categories instead of the whole PGF database. The PGF classification results show the significant performance of multi-label learning. The retrieval results illustrate that our method outperforms state-of-the-art PGF retrieval methods significantly both in terms of retrieval quality and time consumption.

We assume that the incorrect classification results of some parts of the PGFs may be caused by the extracted training features and the insufficiency of the training set. In the future, we will extract more appropriate multi-label training features and create a larger dataset to overcome the current shortages. More learning techniques can be adopted in future works to manage multi-label classification.

## ACKNOWLEDGMENTS

## REFERENCES

[1]  Tsoumakas, G. and Katakis, I., "Multi-label classification:An overview," International Journal of Data Warehousing and Mining, 3(3), pp. 1-13, 2007.
[2]  Boutell, M. R., Luo, J., Shen, X., and Brown, C. M., "Learning multi-label scene classification," Pattern Recognition, 37(9), pp. 1757-1771, 2004.
[3]  Fürnkranz, J., Hüllermeier, E., Mencía, E. L., and Brinker, K., "Multilabel classification via calibrated label ranking," Machine Learning, 73(2), pp. 133-153, 2008.
[4]  Tsoumakas, G., Katakis, I., and Vlahavas, I., "Mining multi-label data," Data mining and knowledge discovery handbook, pp. 667-685, 2010.
[5]  Tsoumakas, G., Katakis, I., and Vlahavas, I., "Random k-labelsets for multilabel classification," Knowledge and Data Engineering, IEEE Transactions on, 23(7), pp. 1079-1089, 2011.

[6] Read, J., Pfahringer, B., Holmes G., et al, "Classifier chains for multi-label classification," Machine learning, 85(3), pp. 333-359, 2011.

[7] Ghamrawi, N. and McCallum, A., "Collective multi-label classification," in Proceedings of the 14th ACM International Conference on Information and Knowledge Management, pp. 195-200, Bremen, Germany, 2005.

[8] Zhang, M. and Zhou, Z., "ML-KNN: A lazy learning approach to multi-label learning," Pattern Recognition, 40(7), pp. 2038-2048, 2007.

[9] McCallum, A., "Multi-label text classification with a mixture model trained by EM," In AAAI'99 Workshop on Text Learning, pp. 1-7, 1999.

[10] Trohidis, K., Tsoumakas, G., Kalliris, G., and Vlahavas, I. P., "Multi-Label Classification of Music into Emotions," In ISMIR, pp. 325-330, 2008.

[11] Bang, Z., Yang, W., and Fang, C., "Multilabel image classification via high-order label correlation driven active learning," IEEE transactions on image processing: a publication of the IEEE Signal Processing Society, 23(3), pp. 1430-1441, 2014.

[12] Zhang, J., Li, D., Hu, W., Chen, Z., and Yuan, Y., "Multilabel Image Annotation Based on Double-Layer PLSA Model," The Scientific World Journal, 2014.

[13] Sun, F., Tang, J., Li, H., Qi, G. J., and Huang, T. S., "Multi-Label Image Categorization With Sparse Factor Representation," IEEE Transactions on Image Processing, 23(3), pp. 1028-1037, 2014.

[14] Wang, S., Wang, J., Wang, Z., and Ji, Q., "Enhancing multi-label classification by modeling dependencies among labels," Pattern Recognition, 47(10), pp. 3405–3413, 2014.

[15] Sucar, L. E., Bielza, C., Morales, E. F., Hernandez-Leal, P., Zaragoza J. H., and Larrañaga, P., "Multi-label classification with Bayesian network-based chain classifiers," Pattern Recognition Letters, 41(5), pp. 14–22, 2014.

[16] Yu, Y., Pedrycz, W., and Miao, D., "Multi-label classification by exploiting label correlations," Expert Systems with Applications, 41(6): pp. 2989-3004, 2014.

[17] Li, K., Lu, X., Ling, H., Liu, L., Feng T., and Tang, Z., "Detection of Overlapped Quadrangles in Plane Geometric Figures," 12th International Conference on Document Analysis and Recognition (ICDAR 2013), Washington DC, U.S.A, Aug 25-28, 2013.

[18] Feng, T., Lu, X., Liu, L., Li, K., and Tang, Z., "Structure Analysis for Plane Geometry Figures," in Document Recognition and Retrieval XXI(DRR 2014), San Francisco, CA, USA, Feb.02-06, 2014.

[19] Liu, L., Lu, X., Li, K., Qu, J., Gao, L., and Tang, Z., "Plane Geometry Figure Retrieval with Bag of Shapes," 11th IAPR Workshop on Document Analysis Systems (DAS), 2014.

[20] Seo, M. J., Hajishirzi, H., Farhadi, A., and Etzioni, O., "Diagram Understanding in Geometry Questions," The AAAI Conference on Artificial Intelligence (AAAI-2014), Québec City, Québec, Canada, July 27-31, 2014.

[21] Nayef, N. and Breuel, T. M., "Efficient symbol retrieval by building a symbol index from a collection of line drawings," in Document Recognition and Retrieval, San Francisco, CA, U.S.A., Feb 5-7, 2013.

[22] Song, J., Lu, X., Ling, H., Wang, X., and Tang, Z., "Envelope extraction for composite shapes for shape retrieval," in 21st International Conference on Pattern Recognition, 2012.

[23] Celebi, M. E. and Aslandogan, Y. A., "A comparative study of three moment-based shape descriptors," in Proc. Of the International Conference of Information Technology: Coding and Computing, pp. 788-793, 2005.

[24] Tsoumakas, G., Spyromitros-Xioufis, E., Vilcek, J., and Vlahavas, I., "Mulan: A Java Library for Multi-Label Learning," Journal of Machine Learning Research, 12, pp. 2411-2414, 2011.

[25] Jarvelin, K. and Kekalainen, J., "IR evaluation methods for retrieving highly relevant documents," in Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 41-48, 2000.