

# EVALUATION OF LOW-LEVEL FEATURES BY DECISIVE FEATURE PATTERNS

Wei Wang and Aidong Zhang

Department of Computer Science and Engineering  
State University of New York at Buffalo  
Buffalo, NY 14260, USA

## ABSTRACT

In content-based image retrieval (CBIR), the effectiveness of the low-level features depends on their capabilities in describing the high-level semantic concepts. How to properly evaluate such an effectiveness remains a challenge. In this paper, we address the evaluation problem by using the decisive feature patterns of the low-level features. Intuitively, a decisive feature pattern is a combination of low-level feature values that are unique and significant for describing a semantic concept. An evaluation study on three low-level features shows that our method can tackle the evaluation problem well. That is, the decisive feature patterns can properly characterize the low-level features' capabilities in describing the semantic concepts.

## 1. INTRODUCTION

Although research on content-based image retrieval (CBIR) has been actively pursued for more than a decade, there are still no highly satisfactory methods for evaluating the effectiveness of the low-level features in describing the high-level semantic concepts (referred as **evaluation problem**).

Suppose a CBIR method  $T$  is using low-level feature  $F$  and feature distance measure  $D$  to retrieve images. To evaluate the performance of  $F$ , presently, typical steps include: (1) set up a test image database, extract  $F$  for all the images in the database; (2) set up a query set as well as the ground truth; (3) for each query, retrieve top-ranked images from database by  $D$ ; (4) after retrieval tasks are performed, certain metrics like *precision* and *recall* are calculated based on the relevance of the retrieval results to the images in the database and the ground truth; and (5) use those metrics to compare the effectiveness of different low-level features.

The feature distance measure  $D$  usually can only reflect the *visual* difference between images, therefore cannot represent the effectiveness of  $F$  in describing the high-level semantic concepts. For example, assume  $D$  is the most commonly used  $L_2$  distance metric (i.e., the *Euclidean* distance) and  $F$  is the *color histogram*. Then  $D$  can only reflect the difference of global color layout between images, which,

unfortunately, is usually not relevant to the semantic difference. Furthermore, correctly applying  $D$  to  $F$  is another issue. If two images with same semantic concepts have only *partially* similar  $F$ , applying  $D$  to all the feature elements of  $F$  may not reflect the semantic similarity.

In this paper, we tackle the *evaluation problem* by using the *decisive feature patterns (DFPs)* of the low-level features. Intuitively, a decisive feature pattern is a combination of low-level feature values that are unique and significant for describing a semantic concept.

Given a set of training sample images and a set of possible semantic concepts contained in the images, for each image, a certain low-level feature is extracted (e.g., a *color-based feature*  $F$ ). For a semantic concept, say "tiger", we call all the training sample images having tiger(s) as tiger's *positive samples*, and the remaining training sample images as its *negative samples*. If a combination of  $F$ 's feature values, e.g.  $V = \{(\text{color component "yellow"} = 0.4) \wedge (\text{color component "black"} = 0.2)\}$  is *frequent* in tiger's positive samples and *rare* in its negative samples,  $V$  is called *emerging* for tiger. If  $V$  is emerging for tiger *only*, then  $V$  is called *decisive* for tiger, i.e., an image containing  $V$  likely has a tiger. On the other hand, if  $V$  is emerging for both tiger and another concept, say "tree" (because tiger and tree may appear together in the same images),  $V$  is called *decisive* for tiger if  $V$  is still emerging for tiger in the subset of samples containing either tigers or trees, i.e.,  $V$  is frequent in tiger's positive samples and rare in the samples containing trees but not tigers.

The decisive feature patterns are related to the *emerging patterns (EPs)* [3]. In all previous studies of emerging patterns, each sample has only one class label. However, an image usually contains multiple semantic concepts, therefore it is very likely that one emerging pattern is emerging for multiple semantic concepts. As shown in [6], as the subset of emerging patterns which are unique and significant for a single concept, the decisive feature patterns are better than the emerging patterns in providing the associations between the semantic concepts and the low-level features, therefore can well address the *evaluation problem*.

We test our method extensively on real image data sets

selected from general-domain COREL images having a good variety of semantic concepts. That is, the selected semantic concepts, on the whole, do not have preference on any specific domain of low-level features, such as *color*, *texture* and *shape*. An evaluation study on three low-level features shows that the decisive feature patterns can properly characterize the low-level features' capabilities in describing the semantic concepts.

The remaining of the paper is organized as follows. In Section 2, we introduce the formal definition of the decisive feature patterns and briefly introduce how to mine the decisive feature patterns. We then discuss how decisive feature patterns characterize the associations between the semantic concepts and the low-level features in Section 3. In Section 4 we present the evaluation results and finally we conclude in Section 5.

## 2. DECISIVE FEATURE PATTERNS (DFP)

Given a set of  $n$  images  $Sample = \{S_1, \dots, S_n\}$ , let  $C = \{c_1, \dots, c_k\}$  be a set of  $k$  possible semantic concepts contained in the images. A low-level feature vector  $F$  is extracted for every image in  $Sample$ .

The feature values in  $F$  are in the form of "feature number = value", where the values are often real numbers. If we represent the decisive feature patterns by real numbers, it could be difficult to identify the noise from the meaningful feature values. Therefore we first discretize the feature values by the *entropy-based discretization method* [4], which utilizes the class information entropy of candidate partitions to select interval boundaries for discretization<sup>1</sup>. After discretization, each interval can be coded as an *item*, i.e., a discrete symbol. Thus, the low-level feature values of an image can be regarded as an itemset, i.e., a set of items.

Given the sample set  $Sample$  and the semantic concept set  $C$ , after the discretization, let  $X = \{X_1, \dots, X_m\}$  be the set of all possible values in all features. Then, a sample  $S_i$  in  $Sample$  can be written as  $S_i = (S_i.Features, S_i.Concepts)$ , where  $S_i.Features \subseteq X$  and  $S_i.Concepts \subseteq C$  are the set of features and the set of semantic concepts in  $S_i$ , respectively.

For each concept  $c \in C$ , the sample set  $Sample$  can be divided into two complementary subsets: the *positive sample set*  $Sample_c^+$  contains the samples having concept  $c$ , and the *negative sample set*  $Sample_c^-$  contains the samples not having concept  $c$ .

A *feature pattern*  $P$  is a subset of feature values, i.e.,  $P \subseteq X$ . The *count* of  $P$  in a set of samples  $D$  is the number of images whose feature value set is a superset of  $P$ . That is,  $count_D(P) = |\{S | (S \in D) \wedge (P \subseteq S.Features)\}|$ . The *support* of  $P$  in  $D$  is defined as  $sup_D(P) = \frac{count_D(P)}{|D|}$ .

Given  $min\_sup$ , the *minimum support threshold* such that

( $0 < min\_sup \leq 1$ ), a pattern  $P$  is *frequent* for semantic concept  $c$  if  $sup_{Sample_c^+}(P) \geq min\_sup$ .

The *growth-rate*  $(P, c)$ , which is the ratio of  $P$ 's support in  $Sample_c^+$  against that in  $Sample_c^-$ , measures the difference of  $P$ 's frequencies in the positive and negative sample sets of  $c$ . That is,  $growth\_rate(P, c) = \frac{sup_{Sample_c^+}(P)}{sup_{Sample_c^-}(P)}$ . Here, we define  $\frac{m}{0} = \infty$  for any number  $m \neq 0$  and  $\frac{0}{0} = 0$ .

Given  $min\_sup$  and  $min\_growth$ , the *minimum growth-rate threshold* such that ( $min\_growth > 1$ ), a pattern  $P$  is *emerging* for a concept  $c$  if (1)  $P$  is frequent for  $c$ ; and (2)  $growth\_rate(P, c) \geq min\_growth$ .

A feature pattern  $P$  is *decisive* for  $c$  if among all the concepts in  $C$ ,  $P$  is only emerging for  $c$ . When  $P$  is emerging for multiple concepts in  $C$ , we need to further determine if it is decisive for  $c$ . Without loss of generality, suppose  $P$  is emerging for both concepts  $c_1$  and  $c_2$ . Let  $Sample_{c_2 \wedge c_1}$  be the set of samples containing  $c_2$  but not  $c_1$ ,  $P$ 's *decisive-rate* of  $c_1$  w.r.t.  $c_2$  is defined as:

$$decisive\_rate(P, c_1, c_2) = \frac{sup_{Sample_{c_1}^+}(P)}{sup_{Sample_{c_2 \wedge c_1}}(P)}. \quad (1)$$

The feature pattern  $P$  is *decisive* for  $c_1$  if  $decisive\_rate(P, c_1, c_2) \geq min\_growth$ . That is,  $P$  is also emerging for  $c_1$  in the subset of samples containing either concept  $c_1$  or  $c_2$ .

To mine the decisive feature patterns, heuristically, we can first mine the emerging patterns for each semantic concept  $c \in C$  separately, then we decide for each emerging pattern of  $c$ , whether it is decisive for  $c$ . If  $|C|$  is large, the above method can be very costly because we have to mine sample set  $|C|$  times and spend a large amount of fruitless resources finding those emerging but not decisive patterns. In [6], an efficient algorithm, *DFP-mine*, is developed to *directly* mine the decisive feature patterns for *all* the semantic concepts, by mining the sample set *only once*. Limited by space, the detailed description of *DFP-mine* is not introduced here.

## 3. ASSOCIATIONS BETWEEN SEMANTIC CONCEPTS AND LOW-LEVEL FEATURES

To characterize the association between a semantic concept  $c \in C$  and the low-level feature  $F$ , we build a *semantic rule base* for  $c$ . That is, we use the set of  $F$ 's decisive feature patterns for  $c$  to construct a *semantic classifier* to recognize  $c$  for images. In this way we can understand how well  $F$  captures the characteristics of the semantic concept  $c$ , by which we can address the *evaluation problem*.

For simplicity, we denote  $c$ 's positive and negative sample sets as  $S_+$  and  $S_-$ , respectively. We also denote  $F$ 's decisive feature patterns for  $c$  as  $DFP(c)$ . Assume  $p$  is one of the DFPs in  $DFP(c)$ , the predicting power of  $p$  in correctly recognizing  $c$  lies in two factors: (1)  $p$ 's differentiating power between  $S_+$  and  $S_-$  and (2)  $p$ 's coverage of  $S_+$  for

<sup>1</sup>Limited by space, the detailed description of the method is omitted.

c. We use the *importance* of  $p$  to characterize its predicting power for  $c$ . From the definition of *support*, we know that  $p$ 's coverage of  $S_+$  is represented by  $sup_{S_+}(p)$ . To represent  $p$ 's differentiating power, we adopt the *general probability that a certain sample  $S$  belongs to  $S_+$ , given that  $S$  contains  $p$* . Denoting such a probability as  $\mathcal{P}(S \in S_+ | p \subseteq S)$ , we can calculate it as:

$$\mathcal{P}(S \in S_+ | p \subseteq S) = \frac{growth\_rate(p, c) * |S_+|}{growth\_rate(p, c) * |S_+| + |S_-|}. \quad (2)$$

The *importance* of the DFP  $p$  w.r.t. concept  $c$ , denoted as  $Imp(p, c)$ , is defined as the product of  $p$ 's differentiating power and its coverage for  $c$ , i.e.,

$$Imp(p, c) = \mathcal{P}(S \in S_+ | p \subseteq S) * sup_{S_+}(p). \quad (3)$$

In this way, the  $Imp(p, c)$  is proportional to both the differentiating power and the coverage, therefore can characterize  $p$ 's predicting power for  $c$ .

The predicting power of a single DFP is not enough to characterize the association between  $c$  and the low-level feature  $F$ . Therefore we combine the predicting power of all the DFPs in  $DFP(c)$ . For each of the sample  $S \in S_+$ , we first find out the complete subset of  $DFP(c)$  which is also contained in  $S$ , then sum the *importance* of the DFPs in the subset as the **association** of  $S$  and  $c$ , which is defined as:

$$association(S, c) = \sum_{p \subseteq S, p \in DFP(c)} Imp(p, c), \quad (4)$$

where  $p$  is a DFP contained in both  $S$  and  $DFP(c)$ .

The set of **normalized** association values for all the samples in  $S_+$  provides a sample space for us to empirically estimate the *probability density function* [1] for the association values. Given a *confidence coefficient* [1]  $\alpha$  (e.g., 90%), we can calculate the *confidence interval* [1]  $(\theta_c, 1]$  such that the probability of an association value to fall into  $(\theta_c, 1]$  is no less than  $\alpha$ , i.e.,

$$\mathcal{P}(association(S, c) \in (\theta_c, 1]) \geq \alpha, \quad (5)$$

where  $association(S, c)$  is the **normalized** association value. For later use, we also sort the association values for all the training samples in  $S_+$ , and denote the result as  $asso_{sorted}(c)$ .

Finally the  $DFP(c)$ , the endpoint  $\theta_c$  and  $asso_{sorted}(c)$  are stored as the semantic rule base for  $c$ . Similarly, we can build the rule bases for all the semantic concepts in  $C$ .

To recognize the semantic concepts for a test image  $S_N$ , the low-level feature  $F$  is first extracted and discretized for  $S_N$ . For each concept  $c \in C$ , if  $association(S_N, c) \geq \theta_c$ ,  $S_N$  is considered to contain  $c$ . If  $|C|$  is relatively large, to facilitate the efficient searching, for each concept  $c \in C$  we calculate the *relative rank* of  $association(S_N, c)$  in the  $asso_{sorted}(c)$  and *sort* the ranking results for all concepts in  $C$ . Only the top ranked concepts are considered as candidates to be further recognized.  $association(S_N, c)$  can be compared to  $\theta_c$  to finally decide if  $S_N$  contains those concepts.

## 4. EVALUATION RESULTS

The 6004 manually labeled images in the test dataset (named *CORELI*) are randomly selected from an image collection with 31,438 COREL images. *CORELI* covers 39 different semantic concepts, including the concepts for the semantic objects (such as atom, cat, dog, lion, mushroom, owl, rose, swimming girl and so on) and the concepts for the semantic topics (such as city, office, fitness, kitchen, skiing, museum furniture and so on). Certain images in the dataset have multiple semantic concepts. The full set of semantic concepts can be found in Table 1. As stated before, in general, the semantic concepts set  $C$  of *CORELI* does not have any preference on any specific domain of low-level features. In our experiments, for each semantic concept  $c \in C$ , we use half of the images containing  $c$  as the training set and the other half as the testing set.

To test the effectiveness of the semantic rule base for the semantic concept  $c$ , we use the *recognition accuracy* of  $c$ , defined as the number of correctly recognized test images with  $c$ , over the total number of test images with  $c$ .

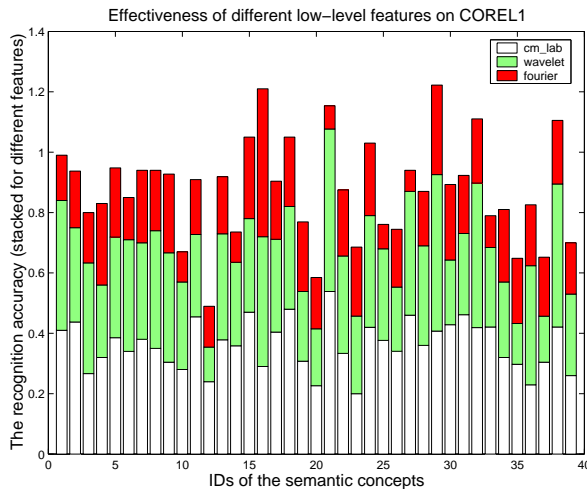
The selection of the low-level features (to be evaluated) is based on their availability, i.e., we do not have preference on any particular features. In our experiments, we evaluate the following low-level features: **(1) Color** is represented by a 9-bin color moment in  $L^*a^*b$  color space [5]. **(2) Texture** is represented by a 10-bin wavelet-based feature [5]. **(3) Shape** is represented by a 32-bin Fourier shape descriptor [2]. For each of the low-level feature to be evaluated (denoted as *cm\_lab*, *wavelet* and *fourier*, respectively), we mine the decisive feature patterns, build the semantic rule base, and calculate the *recognition accuracy* for each semantic concept in  $C$ .

To evaluate low-level features' *general effectiveness* to the semantic concepts, for each low-level feature we compute the *averaged recognition accuracy (ARA)* of all the semantic concepts in *CORELI*. Table 2 shows the results of ARA for different low-level features, under different combinations of *min\_sup* (row) and *min\_growth* (column). The highest ARA for *cm\_lab* is 0.3611 when *min\_sup* = 0.10 and *min\_growth* = 2.75. The highest ARA for *wavelet* is 0.3180 when *min\_sup* = 0.10 and *min\_growth* = 3.75. The highest ARA for *fourier* is 0.1977 when *min\_sup* = 0.175 and *min\_growth* = 2.00. Note that for each low-level feature, there are fluctuations of ARAs for different *min\_sup* and *min\_growth*. This happens because of the difference in the number and quality of the decisive feature patterns for different *min\_sup* and *min\_growth*. Detailed explanation is omitted here due to space limit.

From the results of the highest ARAs shown in Table 2, we can conclude that the low-level feature *cm\_lab* is generally better than *wavelet* (about 5% more accurate), and both of *cm\_lab* and *wavelet* outperform *fourier* (more than

ID	<i>c</i>	ID	<i>c</i>	ID	<i>c</i>	ID	<i>c</i>	ID	<i>c</i>
1	<i>atom</i>	2	<i>barnie(bear toy)</i>	3	<i>beach</i>	4	<i>bobby(doll)</i>	5	<i>butterfly</i>
6	<i>cat</i>	7	<i>office</i>	8	<i>china</i>	9	<i>city</i>	10	<i>colored eggs</i>
11	<i>crab</i>	12	<i>deer</i>	13	<i>desert</i>	14	<i>dog</i>	15	<i>duck(porcelain)</i>
16	<i>eagle</i>	17	<i>fish</i>	18	<i>fitness</i>	19	<i>flying bird</i>	20	<i>hipo</i>
21	<i>ice</i>	22	<i>kicthen</i>	23	<i>lion</i>	24	<i>molecule</i>	25	<i>mountain</i>
26	<i>mushroom</i>	27	<i>owl</i>	28	<i>rose</i>	29	<i>shark</i>	30	<i>sign(road sign)</i>
31	<i>skiing</i>	32	<i>sunset</i>	33	<i>swim girl</i>	34	<i>museum furniture</i>	35	<i>turtle</i>
36	<i>wave</i>	37	<i>waving</i>	38	<i>whale</i>	39	<i>wildcat</i>		

**Table 1.** The semantic concepts of dataset *COREL1*



**Fig. 1.** Effectiveness of different low-level features.

10% more accurate). Considering the large variety of the concepts in *COREL1*, these accuracy numbers clearly show the effectiveness disparity among the low-level features we compared, in describing the high-level semantic concepts. Figure 1 also shows the effectiveness for each *individual* semantic concept, when the ARAs of different low-level features achieve highest values. The x-axis represents the IDs of semantic concepts. The y-axis shows the recognition accuracy (stacked for different features). For each semantic concept  $c \in C$ , the figure shows how well low-level features represent  $c$  and which feature is most effective for  $c$ .

## 5. CONCLUSION

In this paper, the effectiveness of the low-level features is evaluated by the decisive feature patterns, which are the combinations of low-level feature values that are unique and significant for describing the semantic concepts. An evaluation study on three low-level features shows that our method can tackle the evaluation problem well. That is, the decisive feature patterns can properly characterize the low-level features' capabilities in describing the semantic concepts.

	2.00	2.25	2.50	2.75	3.00
0.100	0.3158	0.3152	0.3420	<b>0.3611</b>	0.2964
0.125	0.2523	0.2254	0.2810	0.2758	0.2145
0.150	0.1988	0.2167	0.2446	0.2067	0.1856
0.175	0.1408	0.1712	0.1648	0.1563	0.1275

(a) color moment in  $L^*a^*b$  space

	3.00	3.25	3.50	3.75	4.00
0.100	0.2907	0.3111	0.3137	<b>0.3180</b>	0.3096
0.125	0.2531	0.2868	0.2873	0.2591	0.2611
0.150	0.2485	0.2612	0.2630	0.2486	0.2468
0.175	0.2043	0.2293	0.2310	0.2163	0.2074

(b) wavelet-based texture feature

	1.75	2.00	2.25	2.50	2.75
0.125	0.1856	0.1749	0.1657	0.1487	0.1245
0.150	0.1934	0.1874	0.1678	0.1504	0.1321
0.175	0.1945	<b>0.1977</b>	0.1648	0.1621	0.1279

(c) fourier shape descriptor

**Table 2.** The averaged recognition accuracy (ARA)

## 6. REFERENCES

- [1] S. Pillai A. Papoulis. Probability, random variables, and stochastic process. In *McGraw-Hill Higher Education*, 2002.
- [2] S. Brandt, J. Laaksonen, and E. Oja. Statistical shape features in content-based image retrieval. In *Proc. ICPR*, 2000.
- [3] G. Dong and J. Li. Efficient mining of emerging patterns: discovering trends and differences. In *ACM KDD*, 1999.
- [4] U. Fayyad and K. Irani. Multi-interval discretization of continuous-valued attributes for classification learning. In *Proc. 13th IJCAI*, pp 1022–1027, 1993.
- [5] W. Y. Ma and H. Zhang. Handbook of multimedia computing. In *chapter of Content-based Image Indexing and Retrieval*, pp 19–20. London: Routledge, 1998.
- [6] W. Wang and A. Zhang. Extracting semantic concepts from images: A decisive feature pattern mining approach. In *CS-Technical Report 2003-11, SUNY at Buffalo*, 2003.