Extraction of Association Rules between Low-Level Descriptors and Semantic Descriptors in an Image Database

Edoardo Ardizzone[1], Thomas Daurel[2],
Umberto Maniscalco[1] and Christophe Rigotti[2]

(1) Dipartimento di Ingegneria Automatica e Informatica
Universit di Palermo
Laboratory of Computer Science and Artificial Intelligence (CSAI)
-oOo-
(2) Institut National des Sciences Appliquées (INSA) de Lyon
Laboratoire d'Ingénierie des Systèmes d'Information (LISI)

Abstract *In this paper we combine previous works done in the domain of image processing used to extract automatically low-level descriptors and previous works in the field of data mining used to find a kind of dependencies between variables called association rules.*

Using our own prototypes we extract low-level descriptors, and then extract association rules between these descriptors and semantic descriptors.

We run experiments on an image database containing 532 paintings. These experiments show that the combination of techniques proposed in this paper can be used to extract efficiently dependencies between low-level descriptors (color descriptors) and semantic descriptors (names of painters).

Keywords: data mining, image database, association rules, image descriptors

1 Introduction

The concept of association rule proposed by [1] has been investigated by the data mining community during the past years to find dependencies in large data sets (e.g., [3, 2, 12, 6, 7, 14, 15]). It has been shown that these rules can be used to extract useful dependencies between values of variables, for example in the analysis of customer behavior and the analysis of census data.

Another important effort has been made in the image processing community to extract descriptors of image features such as texture, color, motion, shape, and several prototypes and/or commercial systems have been realized (e.g., [8, 10, 11, 13]) leading for example to new developments in indexing, retrieval and classification of images.

In this paper we consider the extraction of dependencies in an image database between the image feature descriptors and high-level semantic descriptors by means of association rules.

In our case the database contains paintings, the image feature descriptors are color descriptors and the high-level semantic descriptors are the names of the painters. We process the images so that association rules extraction can be performed and then run several experiments. These experiments show that association rules describing dependencies between the color descriptors and the names of the painters can be found, with a good scalability wrt. the size of the image database. These association rules can be used as a complementary information to improve various processings in image databases (e.g., indexing, query optimization).

is_40	is_45	is_butcher	is_baristor	is_male	is_slim
1	0	1	0	1	1
1	0	1	0	0	1
0	1	0	0	0	1
1	0	0	1	1	0

Table 1: A binary matrix R.

This paper is organized as follows. Sections 2 and 3 present briefly respectively the notion of association rule and image feature descriptors. Then, we described more precisely the color descriptors and their encoding in a binary format allowing association rule extraction. Next, we present examples of rules extracted and we give experimental results showing the scalability of the extractions on this kind of data. Finally, we conclude with a summary in Section 5.

2 Association Rules

In this section, we present the concept of association rule introduced in [1].

Association rules are rules whose general pattern is $A \Rightarrow B$, where A and B are conjunctions of binary properties. The purpose of an association rule is to describe an association between the two phenomena A and B. Its intuitive meaning is the following: the conjunction of binary attributes B can be inferred from a primary conjunction A. An association rule is characterized by two main quality measures called confidence and support. For example, a rule $A \Rightarrow B$ with support 10 % and confidence 90 % means that "if we suppose A, then we can be 90 % sure that B is verified, and 10 % of the objects in the data set have both conjunctions A and B set to true".

Example. Let R be a binary matrix. Every line represents the description of an object, and every column is a binary property. An example of such a matrix is depicted in Table 1. In this example, every line represents a person and each person has six binary properties. If we consider the data in the matrix R, the rule is_40 & $is\_male \Rightarrow is\_butcher$ has a confidence of 50 % and a 25 % support.

An association rule is in fact an element of knowledge about the initial binary matrix. The obtained association rules have to be *good* enough, i.e. their confidence and support must be greater than fixed thresholds. It is important to note that the extraction process is exhaustive wrt. the required thresholds. In order to find these association rules, several techniques are available, most commonly based on the Apriori algorithm [3, 2]. In this paper, we use our own implementation of an algorithm called Dense-Miner and proposed by [5].

3 The Image Descriptors

Image features such as texture, color, motion, shape, have been investigated during the past. Each one of them may be related to the whole image (global features) or to one or more image parts (local features). Local features are specially important if related to meaningful image parts, i.e. regions normally corresponding to objects present in the scene. When the image segmentation into regions is difficult or impossible, a possible alternative is offered by a subdivision of the image into regular areas, as more detailly described in the next section. In this case, each area has to be characterized in terms of the adopted features.

Several intrinsic features of images may be extracted in a (fully or partially) automatic way. This allowed the development of several prototypal and/or commercial systems [8, 10, 11, 13], where raw data are normally pre-analyzed in order to compute feature descriptors.

The aim of a feature extraction algorithm is to provide the value of one or more feature descriptors. For example, an object's shape may be described in terms of its boundary (in this case, a boundary descriptor such as a set of Fourier coefficients may be used) but also in terms of geometric properties like area, perimeter, aspect ratio, etc. The choice of the more appropriate descriptors is strongly depending on the application. For example, the color descriptors used in this paper is a quantized histogram in the HSV color space [9], with a finest quantization for the hue, that may be considered the perceptually more significant chromatic component [4]. In particular, 12 levels are retained for hue (H), only 3 each one for saturation (S) and intensity (V) values. In such a way a 108 bins histogram ($12 \times 3 \times 3 = 108$) is obtained, giving the color descriptor as a 108×1 vector. In other cases, other color descriptors are more convenient. For example, applications directly dealing with color image acquisition could take advantage from RGB descriptors.

4 Experiments

In this section we present the whole experimentation process. All our experiments have been run using an image database consisting of 532 paintings. Firstly, we describe how a binary matrix is built, and secondly how this matrix is processed in order to extract association rules of the form $A \Rightarrow B$, where A is a conjunction of values of color descriptors and B is the name of a painter.

4.1 Preprocessing Step

Performing an extraction of association rules requires to build a binary matrix from the image database. This first step is called "preprocessing".

4.1.1 Transforming Images into HSV Histograms

Each painting image is divided in five overlapping areas, as shown in figure 1. One HSV histogram is built for each area. There are twelve hue intervals, three saturation intervals, and three value intervals. For simplicity reasons we only take into account hue intervals in our experiments. (The reader not familiar with this encoding may regard hue intervals as intervals for blue, red, ...) Each histogram is thus described by twelve quantitative variables (one variable per hue interval) ranging from zero to one and representing the relative number of pixels of the corresponding hue. For example, if the first variable of histogram 2 has a value of 0.1 it means that 10 % of the pixels in area 2 have a hue in the first hue interval.

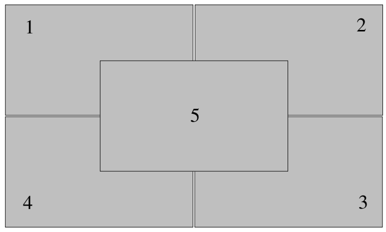

Figure 1: The five areas of a painting image.

4.1.2 Transforming HSV Histograms into a Binary Matrix

Now we build a binary matrix where each line describes one painting, and for each of these paintings we encode with binary variables the color descriptors and the name of the painter.

Painting images are actually described by five histograms, i.e. five groups of twelve quantitative variables, as defined in the previous section. These variables are not binary ones, since they are floating point numbers in the interval [0, 1]. However, they can be discretized, i.e. the definition domain can be partitioned. We need to know first the repartition of the values of these variables. Preliminary experiments showed us that most of the values were in the interval [0.05, 0.3]. Hence, this main interval has to be described by most of the partitions. Here are the partitions, called *frequency intervals*, we retain:

```
0 : for frequency interval ] 0, 1%]
1 :                        ] 1%, 5%]
2 :                        ] 5%, 10%]
3 :                        ] 10%, 15%]
(...)
6 :                        ] 25%, 30%]
7 :                        ] 30%, 50%]
8 :                        ] 50%, 70%]
9 :                        ] 70%, 100%]
```

Since we have ten possible values (the frequency interval number) we use ten binary variables to encode them. So, each painting is described using five histograms, each histograms has twelve variables, and each variable has ten possible values. Therefore, we used $5 \times 12 \times 10 = 600$ binary variables per painting to encode the color descriptors.

To encode the names of the painters we must also used binary variables. In the database used in the experiments they are 532 paintings from 20 different painters. So, we simply used 20 additional binary variables to represent the names of the painters.

4.2 Results

We present some of the results of our experiments showing that association rules between color descriptors and the names of the painters can be found, and that these rules can be extracted with a good scalability wrt. the size of the database.

4.2.1 Typical Example

In this experiment we run the association rule extractor to find all rules having a confidence at least equal to 75% and a support of more than 0.5%. If we consider only the rules having in their right hand side the name of the painter "Waterhouse" we obtain:

```
h1_2:2 h2_10:3 h3_10:5 => Waterhouse  75%
h1_2:2 h3_2:3 h4_11:2 => Waterhouse 75%
h1_2:2 h3_2:3 h4_11:2 h2_3:1 => Waterhouse 75%
h1_2:2 h3_2:3 h4_11:2 h4_6:1 => Waterhouse 75%
h1_2:2 h3_2:3 h4_11:2 h2_3:1 h4_6:1 => Waterhouse 75%
h2_1:7 h2_2:4 h5_1:8 => Waterhouse  75%
h1_2:2 h2_11:1 h4_6:1 h4_11:2 => Waterhouse  75%
h1_2:2 h1_6:1 h3_3:1 h4_11:2 => Waterhouse  75%
h1_2:2 h1_6:1 h3_3:1 h4_11:2 h2_6:1 => Waterhouse  75%
h2_11:1 h3_2:3 h4_6:1 h4_11:2 => Waterhouse  100%
h2_11:1 h3_2:3 h4_6:1 h4_11:2 h1_2:2 => Waterhouse  100%
h2_11:1 h3_2:3 h4_6:1 h4_11:2 h2_3:1 => Waterhouse  100%
h2_11:1 h3_2:3 h4_6:1 h4_11:2 h1_2:2 h2_3:1 => Waterhouse  100%
```

where hi_j:k means that in histogram i the pixels of hue in hue interval number j have a frequency in frequency interval k.

For example, the first rule means that if

1. the pixels in area 1 having a hue value in the hue interval 2, appear with a frequency in the frequency interval number 2, i.e] 5%, 10%] and
2. the pixels in area 2 having a hue value in the hue interval 10, appear with a frequency in the frequency interval number 3, i.e] 10%, 15%] and
3. the pixels in area 3 having a hue value in the hue interval 10, appear with a frequency in the frequency interval number 5, i.e] 20%, 25%]

then the name of the painter is "Waterhouse" with a rule confidence of 75%.

4.2.2 High-Confidence Rules

In the previous experiment some rules with a confidence of 100% have been obtained. These rules may be particularly useful for example in query optimization. It should be noticed that such high-confidence rules are also found for other painters and if we consider "Monet" we have among others:

```
h2_2:1 h3_4:1 h3_12:2 => Monet 100%
h2_2:5 h5_2:6 h5_5:1 => Monet 100%
```

4.2.3 Scalability

We present in this section experiments showing that the extraction of association rules as proposed in this paper can be done even if the binary matrix encoding the image database has a large number of lines (i.e., there is a lot of images in the database).

These experiments have been performed by concatenating the original data file to itself several times. It should be notice that this process does not fully reflect the real growth of an image database, but remains an interesting way to test the scalability especially wrt. the database scans. So, we concatenated the original file to itself 9 times, 19 times and 99 times. We obtained three additional files having respectively a size of $532*10, 532*20$ and $532*100$ image descriptions. Then the same extraction was performed using these three files and the original one, and we measured the total elapsed CPU times on a 500 MHz PIII, with 128 MB of

RAM, running Linux. Figure 2 shows that the extraction time grows linearly wrt. the number of images.

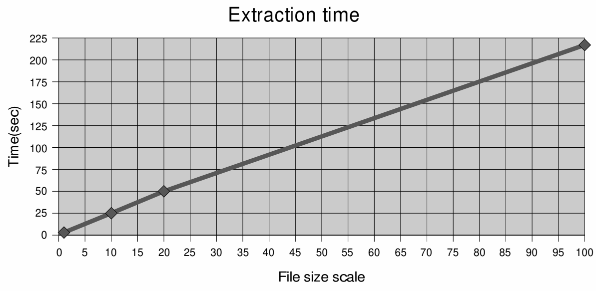

Figure 2: Evolution of rule extraction time vs. scaling factor.

5 Conclusion

We considered previous works done in the domain of image processing to extract low-level descriptors and previous works in the field of data mining to find dependencies in the form of association rules between variables in large data sets.

Using our own prototypes we extracted low-level descriptors, encoded these descriptors as binary variables and then generated association rules between the low-level descriptors and semantic descriptors.

We ran experiments on an image database containing 532 paintings. These experiments showed that the combination of techniques proposed in this paper can be used to extract efficiently dependencies between low-level descriptors (color descriptors) and semantic descriptors (names of painters).

References

[1] R. Agrawal, T. Imielinski, and A. N. Swami. Mining association rules between sets of items in large databases. In *Proc. of the 1993 ACM SIGMOD International Conference on Management of Data (SIGMOD'93)*, pages 207–216, Washingtion, D.C., Sept. 1993.

[2] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. I. Verkamo. Fast discovery of association rules. In *Advances in Knowledge Discovery and Data Mining*, pages 307–328. AAAI Press, 1996.

[3] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *Proc. of the 20th International Conference on Very Large Data Bases (VLDB'94)*, pages 487 – 499, Santiago de Chile, Chile, Sept. 1994.

[4] E. Ardizzone and M. La Cascia. Automatic video database indexing and retrieval. *Multimedia Tools and Applications*, 4(1):29–56, 1997.

[5] R. J. Bayardo, R. Agrawal, and D. Gunopulos. Constraint-based rule mining in large, dense databases. In *Proceedings of the 15th International Conference on Data Engineering (ICDE'99)*, pages 188–197, Mar. 1999.

[6] J. Boulicaut, A. Bykowski, and C. Rigotti. Approximation of frequency queries by mean of free-sets. In *Proc. of the 4th European Conf. on Principles and Practice of Knowledge Discovery in Databases (PKDD'00)*, volume 1910 of *LNAI*, pages 75–85, Lyon, France, Sept. 2000. Springer-Verlag.

[7] S. Brin, R. Motwani, J. D. Ullman, and S. Tsur. Dynamic itemset counting and implication rules for market basket data. In *Proc. of the 1997 ACM SIGMOD International Conference on Management of Data (SIGMOD'97)*, pages 255 – 264, Tucson, Arizona, June 1997.

[8] M. Flickner et al. Query by image and video content: the QBIC system. *IEEE Computer*, 1995.

[9] R. Gonzalez and R. Woods. *Digital Image Processing.* Addison-Wesley, 1992.

[10] P. Kelly, T. Cannon, and D. Hush. Query by image example: the comparison algorithm for navigating digital image databases CANDID approach. In *Proc. of SPIE Storage and Retrieval for Image and Video Databases III*, pages 238–248, 1995.

[11] M. La Cascia and E. Ardizzone. Jacob: Just a content-based query system for video databases. In *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing, ICASSP-96*, 1996.

[12] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Efficient mining of association rules using closed itemset lattices. *Information Systems*, 24(1):25–46, 1999.

[13] A. Pentland, R. Picard, and S. Sclaroff. Photobook: Tools for content-based manipulation of image databases. In *Proc. of SPIE Storage and Retrieval for Image and Video Databases II*, pages 37–50, 1994.

[14] A. Savasere, E. Omiecinski, and S. Navathe. An efficient algorithm for mining association rules in large databases. In *Proc. of the 21th International Conference on Very Large Data Bases (VLDB'95)*, pages 432 – 444, Zürich, Switzerland, Sept. 1995.

[15] H. Toivonen. Sampling large databases for association rules. In *Proc. of the 22th International Conference on Very Large Data Bases (VLDB'96)*, pages 134 – 145, Mumbay, India, Sept. 1996.