Mining Recurrent Items in Multimedia with Progressive Resolution Refinement*

Osmar R. Zaïane
Department of Computing Science
University of Alberta
Edmonton, Alberta, Canada
zaiane@cs.ualberta.ca

Jiawei Han, Hua Zhu
School of Computing Science
Simon Fraser University
Burnaby, British Columbia, Canada.
{han, hzhua}@cs.sfu.ca

Abstract

Despite the overwhelming amounts of multimedia data recently generated and the significance of such data, very few people have systematically investigated multimedia data mining. With our previous studies on content-based retrieval of visual artifacts, we study in this paper the methods for mining content-based associations with recurrent items and with spatial relationships from large visual data repositories. A progressive resolution refinement approach is proposed in which frequent item-sets at rough resolution levels are mined, and progressively, finer resolutions are mined only on the candidate frequent item-sets derived from mining rough resolution levels. Such a multi-resolution mining strategy substantially reduces the overall data mining cost without loss of the quality and completeness of the results.

1. Introduction.

Mining knowledge from large databases has been the focus of many recent studies and applications, however, most of these have emphasized corporate data typically in alphanumeric databases. Little research has been conducted on mining multimedia data. Nevertheless, a few interesting studies and successful applications involving multimedia data mining have been reported. For example, [12] describes the CONQUEST system that combines satellite data with geophysical data to discover patterns in global climate change. The SKICAT system [4] integrates techniques for image processing and data classification in order to identify "sky objects" captured in a very large satellite picture set.

Multimedia data has also been a focal point of our research, and we have integrated image processing with database mining techniques, and developed a multimedia data mining system prototype **MultiMediaMiner** [15, 14]. The system uses a data cube structure for mining characteristic, association, and classification rules. However, the system does not use image content to the extent we wanted. The presence of colours and textures in images was used, but not localization of these visual features, their spatial relationships, their motion in time (for video), etc.

In this paper we extend the concept of content-based multimedia association rules using feature localization, and introduce the concept of progressive refinement in the discovery of patterns in images from coarse to fine resolution. Two algorithms are developed for mining multimedia association rules with recurrent items and recurrent spatial relationships. Our major contribution in this paper is the development of a progressive multi-resolution refinement method for mining multimedia associations with recurrent objects and for mining spatial relationships between visual descriptors in large image collections.

The remainder of the paper is organized as follows. In Section 2, we introduce the concept of multimedia association rules. In sections 3 and 4, algorithms for mining multimedia multiset associations and associations with spatial relationships are presented. Also in Section 4, the progressive multi-resolution refinement approach is proposed. Section 5 briefly reports our performance study. Finally, we conduct a short discussion and conclude our study.

2. Multimedia association rules.

2.1. Feature localization in multimedia data.

Feature localization is a new concept of rough image segmentation introduced in [7]. Image segmentation is a process which segments an image into disjoint regions. A region consists of a set of pixels that share certain properties, e.g., similar colour, similar texture, etc. The traditional segmentation algorithms assume (1) regions are mostly $connected$; (2) regions are $disjoint$ ($R_i \cap R_j = \emptyset$, for $i \neq j$); and (3) segmentation is $complete$ in that any pixel will be assigned to some region, and the union of all

*Research is supported in part by the Natural Sciences and Engineering Research Council of Canada and the Canadian Networks of Centres of Excellence IRIS-3.

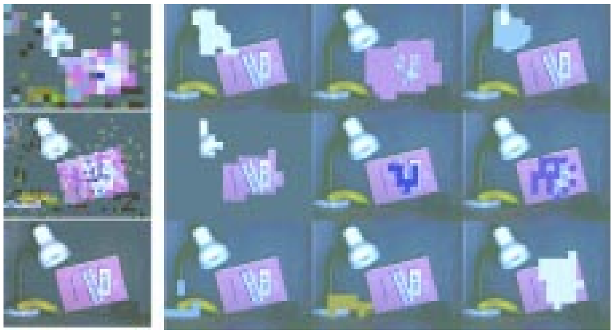

Figure 1. Example of colour localization for a multi-level resolution image.

regions is the entire image ($\cup_{k=1}^{m} R_k = I$). Segmentation does not give a good representation of image content. A more useful and attainable process is feature localization that identifies features by their locality and proximity.

As defined in [7, 13], a *locale* $\mathcal{L}_x$ is a local enclosure (or locality) of feature x. $\mathcal{L}_x$ has an envelope L_x which is a set of tiles to represent the locality of $\mathcal{L}_x$, and some geometric parameters: *mass* $M(\mathcal{L}_x)$ (i.e. the number of pixels in L_x that actually have feature x), *centroid* $C(\mathcal{L}_x)$ (i.e. the centroid of the pixels in L_x), *variance* $\sigma^2(\mathcal{L}_x)$ (i.e. the variance of the Cartesian distance from pixels in L_x to the centroid), and shape parameters for the locale, etc. A tile is a square area in an image. Its size is arbitrarily chosen as 16×16, but could be bigger or smaller. The tile is the building-unit for envelopes. A tile is 'red' if a sufficient number of pixels within the tile are red. It follows that a tile can be both 'red' and 'blue' if some of its pixels are red and some are blue. While a pixel is the unit for image segmentation, a tile is the unit for feature localization. Thus, feature localization is a kind of rough segmentation where overlap is possible and completeness is not necessary. The right columns of Figure 1 show an example of feature localization; each image illustrates a different locale for the same image. We also define minimum bounding circles around locales for topological relationship approximation.

In this study, we investigate efficient methods for mining associations in image databases after image segmentation by feature localization.

2.2. Multimedia associations with recurrent items.

Association rule mining has been studied extensively in data mining research community recently [1, 6, 11, 8, 10]. Many algorithms and approaches have been proposed for mining many types of association rules in large databases. However, typically, the databases relied upon are alphanumerical and often transaction-based. While some of the algorithms proposed can be applied to visual data, to a certain extent, after transforming the data into a form that can be processed, new algorithms should be better suited. Indeed, visual data has some peculiarities proper to images. For example, some visual features can occur multiple times in an image, and the repetition of the feature may carry more information than the existence of the feature itself.

Previous definitions and usage of association rules have limitations vis-à-vis mining association rules from an image and video collection. An image, for instance, can indeed be modelled by a transaction with items being the visual features in the image, and image ID being the transaction ID. However, items in the antecedent of the rule repeating in the consequent can be an interesting factor in image analysis applications. Moreover, recurrent objects in images are very common, and important as argued above. In addition, one may be interested in finding associations with a coarse-to-fine search strategy. In other words, association rules can first be found at a low resolution, then progressively confirmed at higher resolutions. Thus, we can rapidly approximate multimedia association rules at a coarse level, then eliminate false positives by verifying them at a higher resolution. Moreover, the approximation of a locale by a minimum bounding circle can speed up the discovery of association rules at a high conceptual level for spatial topological concepts such as closeness, overlap or containment. The precision of the rules discovered are improved by eliminating the minimum bounding circle and using the locale envelope with higher image resolutions. The main advantages of this approach is that: (1) the extraction of locale features can be conducted at multiple (often reduced) resolutions to save processing time; (2) locale intrinsic features can be defined at appropriate resolutions to avoid too much detail/noise or insufficient detail. Dominant colours are well preserved at a low resolution, but some texture information can be lost when the resolution is too low. The coarse-to-fine search strategy is important for large image and video databases even when the features are extracted and analyzed at pre-processing time.

The previous studies of association mining assume that the items are unique in $\mathcal{I}$, hence the definition of *support*. For example, with the well-known *Apriori* algorithm [1], duplicates are never considered when the k-item candidate sets C_k are formed. In multimedia mining, we would like to mine rules such as "2 $blue\ circles \Rightarrow high\ texture\ density$". This means that the sole existence of blue does not necessarily imply the consequent. That is, two occurrences have to co-exist in the image for the rule to be valid. In addition, the definition of *strong rules* based on *large support* is quite inadequate in some imaging applications. Features appearing very frequently (i.e., having a large support) in some medical images, for example, can be normal phenomenon, and uninteresting to users. A low support, on the other hand, could generate item-sets with extremely rare items. While these rare items could either be just meaningless noise or sought for rare phenomenon (in medicine applications for example), they fall in the realm of outlier analysis and are out of the scope of this study. We believe that a range for an acceptable support should be introduced for such applications. Hence the definition of the *sufficiently strong association rule* (Def. 2.3).

The above discussion leads to the introduction of two notions of *support*. Traditionally, the support is the percentage of transactions that contain an item or verify a condition. It measures how interesting and frequent an item or predicate is in a data set. In our model, images are represented by transactions, but identical objects can repeat in an image, therefore, the second notion of *support* reflects a count of objects rather than a count of transactions (or images). It is the user's choice to select an appropriate definition of support, depending upon the application. This new notion of support is called **object-based support**, which is distinguished from the "traditional" **transaction-based support**. Also, an association rule that allows items to repeat in the rule is called **association rule with recurrent items**.

Definition 2.1 *A* **multimedia association rule with recurrent items** *is a rule that associates visual object features in images and video frames, and is of the form:*
$\alpha P_1 \wedge \beta P_2 \wedge \ldots \wedge \gamma P_n \rightarrow \delta Q_1 \wedge \lambda Q_2 \wedge \ldots \wedge \mu Q_m \ (c\%)$ *where $c\%$ is the confidence of the rule, one or more predicates $P_i, i \in [1..n]$ and $Q_j, j \in [1..m]$ are predicates instantiated to topological, visual, kinematics, or other descriptors of images, and $\alpha, \beta, \gamma, \delta, \lambda$ and μ are integers quantifying the occurrence of the object feature or item. αP is true if and only if P has α occurrences.* □

The predicates P_i and Q_j in the rules are not just topological, visual or kinematics descriptors, but can also be other descriptors such as picture size, video duration, or just related keywords. In a medical imaging system, for example, the physician's diagnosis attached to the image can be extremely beneficial in an association rule.

Definition 2.2 *The* **support** *of a predicate P in a set of images $\mathcal{D}$ denoted by $\sigma(P/\mathcal{D})$ is the percentage of objects in all images in $\mathcal{D}$ that verify P at a given conceptual level. The* **confidence** *of a multimedia association rule $P \rightarrow Q$ is the ratio $\sigma(P \wedge Q/\mathcal{D})$ versus $\sigma(P/\mathcal{D})$, which is the probability that Q is verified by objects in images in $\mathcal{D}$ that verify P at the same conceptual level. Such support is called* **object-based support** *in contrast to* **transaction-based support**. □

Definition 2.3 *A pattern p is* **sufficiently frequent** *in a set $\mathcal{D}$ at a level ℓ if the support of p is no less than its corresponding minimum support threshold $\sigma\prime$, and no more than its corresponding maximum support threshold $\Sigma\prime$.* □

Definition 2.4 *A multimedia association rule $P \rightarrow Q$ in a set of images $\mathcal{D}$ is* **sufficiently strong** *in $\mathcal{D}$ if P and Q are sufficiently frequent (P and $Q \in [\sigma\prime..\Sigma\prime]$) and the confidence of $P \rightarrow Q$ is greater than $\varphi\prime$.* □

Note that the *strength* of a rule and the values of $\sigma\prime$ and $\Sigma\prime$ depend upon the concept level in which the predicates are applied. All attributes such as colour, texture, motion direction, etc., are defined on concept hierarchies. Depending on the concept level selected by the user, and the resolution level of the images, $\sigma\prime$ and $\Sigma\prime$ can be higher or lower.

Given an image I as a transaction and locales $\mathcal{L}_i$ (or objects) as the items in the image I, we envision two types of multimedia association rules: association rules based only on atomic visual features that we call **content-based multimedia association rules with recurrent visual descriptors**, and association rules with spatial relationships that we call **multimedia association rules with recurrent spatial relationships**. What we call atomic features are descriptors such as colour, texture, etc. They are attributes of an object defined along concept hierarchies. Association rules based on atomic visual features are similar to multi-dimensional, multi-level association rules, emphasizing the presence of values of some attributes at given concept levels.

The second type of multimedia association rules uses the topological relationships between locales (*v-next-to* for vertical closeness, *h-next-to* for horizontal closeness, overlap, and include). Each predicate P describes the relationship between two objects O_a and O_b, such as *Overlap(O_a, O_b)*, each object being multi-dimensional. Binary predicates involve a join of more than one relation. Moreover, spatial predicates on the same object values can be recurrent.

3. Frequent item-sets with recurrent items.

Mining a large collection of multimedia artifacts can be very costly. The idea of progressive resolution refinement, presented in Section 4.1, is to mine the artifacts at different resolution levels and reduce the search space at each level. The knowledge extraction at each resolution level is similar but the result at each resolution level is used to filter out unnecessary features and images to reduce the data collection for the next levels.

We will discuss in the following subsection the algorithm for enumerating sufficiently frequent item-sets with recurrent items at a given resolution level.

3.1. A naïve approach.

To illustrate our algorithms and test their performance, we have generated synthetic images with random locales and random features. Tables 1 and 2 are filled with the visual feature descriptors of the images, and with the spatial relationships between objects in the images.

Image ID	Object ID	Colour	Texture	Mass	Shape	Motion	...
I_1	$O_{(1,1)}$	Colour$_1$	Texture$_1$	Size$_1$	Shape$_1$	Direction$_1$	...
I_1	$O_{(1,2)}$	...	...	...	...	...	...
...							
I_2	$O_{(2,1)}$	...	...	...	...	...	...
...							
I_n	$O_{(n,\alpha)}$	...					

Table 1. Relation with Visual Atomic Features.

Image ID	Object ID	V-Next-to	H-Next-to	Overlap	Include
I_1	$O_{(1,1)}$	$\{O_{(1,3)}, O_{(1,5)}\}$	$\{O_{(1,2)}\}\}$	$\{O_{(1,7)}\}$	$\{\}$
I_1	$O_{(1,2)}$	$\{...\}$	$\{...\}$	$\{...\}$	$\{...\}$
I_n	$O_{(n,\alpha)}$	...			

Table 2. Relation with Spatial Relationships.

If the Apriori algorithm [1] is to be used to discover frequent item-sets in such data sets as the image collections, it would miss all item-sets with recurrent items. A naïve way to find all frequent item-sets with recurrent items would be to first find all frequent 1-item-sets, check how often they might re-occur in an image (maximum occurrence), and then, for each k-item-set, combine these frequent 1-item-sets in sets of k elements where elements can be repeated up to their respective maximum occurrence possible. The calculation of the support would filter out the infrequent ones.

This naïve algorithm, which guarantees to find all frequent item-sets with recurrent items, could be improved by replacing F_1 as the starting set for enumerating candidates of all k-item-sets by a set composed of F_1 and all item-sets with single items twinned to their maximum capacity, such as $\{x_\alpha\}, \{x_\alpha, x_\alpha\}, \{x_\alpha, x_\alpha, x_\alpha\}$, etc., where the number of x_α is smaller or equal to $M[x_\alpha]$.

In the next sub-section we present our algorithm **MaxOccur**, a more efficient algorithm for discovering multimedia association rules with recurrent items.

3.2. MaxOccur algorithm.

A method for enumerating sufficiently strong multimedia association rules that are based on recurrent atomic visual features is presented in this section. We will give an abstract example and then present the algorithm. To simplify our discussion, we will use a one-dimension, one-level problem where images are transactions of objects and the same objects can repeat. While objects are multidimensional, in this discussion we will treat them as items with only one dimension and no concept hierarchy. The algorithm can be extrapolated to the multi-level association rules discovery algorithm and the multi-dimensional issue can also be solved by using a data cube.

Example 3.1 Let us consider the images represented in Table 3(top) by a set of transactions $\mathcal{D}_1$. Each image is a set of objects that can repeat. At this point, we ignore the descriptors of the objects for simplicity. To determine the support of each object, a first scan of the database is done and each time a distinct object appears, its counter is incremented. At the same time, a second counter keeps track of the maximum appearances of the same object in an image. Table 3(bottom) shows the result of the counting. C_1 contains all unique objects with their support and M contains the maximum number of times a given object occurs in an image. For simplicity, the support is expressed in an absolute value. Let the minimum support $\sigma\prime$ be 2 and the maximum support $\Sigma\prime$ be 5. Frequent k item-sets can be found using $\sigma\prime$ by filtering the non-frequent k-1 item-sets. However, pruning with $\Sigma\prime$ to find sufficiently frequent iten-sets should be left to the end of the process since too frequent k-1 item-sets may end-up frequent enough at k level. Table 4(top) shows F_1, the list of frequent 1 item-sets. Notice that O_2 and O_4 were not eliminated even if they appear too often in the data set ($\sigma(O_2/\mathcal{D}_1) > \Sigma\prime$ and $\sigma(O_4/\mathcal{D}_1) > \Sigma\prime$). Given F_1, we can filter out from $\mathcal{D}_1$ all irrelevant objects, and all transactions that do not contain frequent objects present in F_1. Table 4(bottom) shows $\mathcal{D}_2$, the image transactions with only the interesting objects. The generation of the candidate 2 item-sets is done by joining F_1 with itself to create all possible pairs with frequent objects. It is similar to the *apriori-gen* in [1] except that the information stored in M, regarding replication of objects in images, is used to generate new pairs of the same objects that occur in a transaction more than once. The 2 item-sets $\{O_2, O_2\}$ and $\{O_4, O_4\}$ in Table 5(left) are produced that way. The candidate 3 item-set list C_3 is produced by joining F_2 elements and eliminating 3 item-sets that contain 2 item-sets not recognized as frequent (i.e. not in F_2). The counters in M are also used to generate item-sets such as $\{O_2, O_2, O_2\}$ in Table 6(left). After filtering the infrequent 3 item-sets, F_3 is produced. The candidate 4 item-sets is produced the same way by joining the frequent 3-item sets and pruning the unnecessary ones. For instance $\{O_2, O_2, O_3, O_4\}$ and $\{O_2, O_3, O_4, O_4\}$ are eliminated since, respectively, $\{O_2, O_2, O_3\}$ and $\{O_3, O_4, O_4\}$ are not in F_3. Finally, since no 5 item-set can be induced, the result is all F_i without their item-sets that have a support higher than the maximum support $\Sigma\prime$. Given the sufficiently frequent item-sets, sufficiently strong association rules could be found by generating all rules from a k-item-set of the form "(k-p) item-set $\rightarrow$ p item-set" with 0 < k < p and such that the confidence of the rule is higher than a given confidence threshold. With a confidence threshold set to 100%, only these rules are induced:
(1) $\{O_4, O_4\} \rightarrow \{O_2, O_2\}[100\%]$; (2) $\{O_2, O_4, O_4\} \rightarrow \{O_2\}[100\%]$; (3) $\{O_3, O_4\} \rightarrow \{O_2\}[100\%]$; (4) $\{O_3\} \rightarrow \{O_2, O_4\}[100\%]$; (5) $\{O_2, O_2\} \rightarrow \{O_4\}[100\%]$; (6) $\{O_4, O_4\} \rightarrow \{O_2\}[100\%]$; (7) $\{O_3\} \rightarrow \{O_2\}[100\%]$; (8) $\{O_3\} \rightarrow \{O_4\}[100\%]$.

A simple scan of these rules can count replicated objects and produce the following rules:

2 $O_4 \rightarrow 2\ O_2$ [100%], $O_2 \wedge 2\ O_4 \rightarrow O_2$ [100%], $O_3 \wedge O_4 \rightarrow O_2$ [100%], $O_3 \rightarrow O_2 \wedge O_4$ [100%], 2 $O_2 \rightarrow O_4$ [100%], 2 $O_4 \rightarrow O_2$ [100%], $O_3 \rightarrow O_2$ [100%], and $O_3 \rightarrow O_4$ [100%]. Notice that the rule "$O_4 \rightarrow O_2$" is not confident enough, while "2 $O_4 \rightarrow 2\ O_2$" or "2 $O_4 \rightarrow O_2$" are 100% reliable. This would not have been true had the support been based on the number of images rather than on the number of objects.

Image ID	Objects
I_1	$\{O_2, O_2, O_2, O_4, O_5\}$
I_2	$\{O_2, O_2, O_4, O_4\}$
I_3	$\{O_2, O_3, O_4\}$
I_4	$\{O_6, O_7\}$
I_5	$\{O_1, O_2, O_2, O_3, O_4, O_4\}$

Object	Support	Max. Occurrence
$\{O_1\}$	1	1
$\{O_2\}$	8	3
$\{O_3\}$	2	1
$\{O_4\}$	6	2
$\{O_5\}$	1	1
$\{O_6\}$	1	1
$\{O_7\}$	1	1

Table 3. Top: Image transaction table $\mathcal{D}_1$. Bottom: C_1 and M tables.

Object	Support	Max. Occurrence
$\{O_2\}$	8	3
$\{O_3\}$	2	1
$\{O_4\}$	6	2

Image ID	Sufficiently Frequent Objects
I_1	$\{O_2, O_2, O_2, O_4\}$
I_2	$\{O_2, O_2, O_4, O_4\}$
I_3	$\{O_2, O_3, O_4\}$
I_4	$\{O_2, O_2, O_3, O_4, O_4\}$

Table 4. Top: F_1 and M tables. Bottom: Filtered image transaction table $\mathcal{D}_2$.

2 item-sets	Support
$\{O_2, O_3\}$	2
$\{O_2, O_4\}$	6
$\{O_3, O_4\}$	2
$\{O_2, O_2\}$	3
$\{O_4, O_4\}$	2

2 item-sets	Support
$\{O_2, O_3\}$	2
$\{O_2, O_4\}$	6
$\{O_3, O_4\}$	2
$\{O_2, O_2\}$	3
$\{O_4, O_4\}$	2

Table 5. Candidate 2 item-sets C_2 and sufficiently frequent 2 item-sets F_2.

The above example and discussion proceed to the following algorithm for mining content-based multimedia association rules. Note that the supports used in the example are absolute values for the sake of simplicity. Support for a k-item-set should be $\frac{\text{Count k-item-set in } \mathcal{D}_k}{\sum_{\forall \text{ transaction } t} \binom{|t|}{k}}$, where $\binom{|t|}{k}$ are

3 item-sets	Support
$\{O_2, O_3, O_4\}$	2
$\{O_2, O_2, O_3\}$	1
$\{O_2, O_2, O_4\}$	3
$\{O_2, O_4, O_4\}$	2
$\{O_3, O_4, O_4\}$	1
$\{O_2, O_2, O_2\}$	1

3 item-sets	Support
$\{O_2, O_3, O_4\}$	2
$\{O_2, O_2, O_4\}$	3
$\{O_2, O_4, O_4\}$	2

Table 6. Candidate 3 item-sets C_3 and sufficiently frequent 3 item-sets F_3.

4 item-sets	Support
$\{O_2, O_2, O_4, O_4\}$	2

4 item-sets	Support
$\{O_2, O_2, O_4, O_4\}$	2

Table 7. Candidate 4 item-sets C_4 and sufficiently frequent 4 item-sets F_4.

k-combinations of objects in transaction t without redundancy of unique objects.

Algorithm 3.1 (**MaxOccur**) Find sufficiently frequent item-sets for enumerating content-based multimedia association rules in image collections.

Input: (i) $\mathcal{D}_1$ a set of transactions representing images, with items being the visual and non-visual descriptors of the images; (ii) a set of concept hierarchies for each attribute; (iii) the minimum and maximum support thresholds $\sigma\prime$ and $\Sigma\prime$ for each conceptual level.

Output: Sufficiently frequent item-sets with repetitions.

Method: The pseudo-code for generating sufficiently frequent item-sets is as follows:

begin
(1) $C_1 \leftarrow \{$Candidate 1 item-sets and their $support\}$
(2) $F_1 \leftarrow \{Sufficiently\ frequent$ 1 item-sets and their $support\}$
(3) $M \leftarrow \{Maximum\ occurrence$ in an image of $frequent$ 1 item-sets$\}$
(4) Count # of k-item-sets (total[$1..k$])
(5) for ($i \leftarrow 2$; $F_{i-1} \neq \emptyset$; $i \leftarrow i+1$) $do\{$
(6) $C_i \leftarrow (F_{i-1} \bowtie F_{i-1}) \cup$
$\{y \oplus X \mid X \in F_{i-1} \wedge y \in F_1 \wedge Count(y, X) < (M[y]-1)\}$
(7) $C_i \leftarrow C_i - \{c \mid (i-1)$ item-set of $c \notin F_{i-1}\}$
(8) $\mathcal{D}_i \leftarrow$ *FilterTable*($\mathcal{D}_{i-1}$, F_{i-1})
(9) foreach image I in $\mathcal{D}_i$ do {
(10) foreach c in C_i do {
(11) $c.support \leftarrow c.support$ + *Count*(c, I)
(12) }
(13) }
(14) $F_i \leftarrow \{c \in C_i \mid \frac{c.support}{total\ i\ itemset} > \sigma\prime\}$
(15) }
(16) Result $\leftarrow \bigcup_i \{c \in F_i \mid i > 1 \wedge c.support < \Sigma\prime\}$
end

Line 1, 2, 3 and 4 are done in the same initial scan. M contains the maximum number of times an object appears in the same image. This counter is used later to generate potential k-item-sets. The total number of k-item-sets is used for the calculation of the item-set support in line 14.

In line 6 and 7, the candidate item-sets are generated by joining (i-1) frequent item-sets and the use of M to generate repetitive objects ($M[y] > 1$). The pruning process (line 7) eliminates infrequent item-sets based on the *apriori* property.

Line 8 filters the transactions in $\mathcal{D}$ to minimize the data set scanning time.

In line 14, only the frequent item-sets that are higher than the minimum support $\sigma\prime$ are kept. It is only at the end of the loop (line 16) that maximum support $\Sigma\prime$ is used to eliminate item-sets that appear too frequently.

The calculation of the support for one item-set is based on the occurrence of the item-set in the images. Line 11 cumulates this count. A particular precaution has to be taken when counting appearances of k-item-set in an image, especially that objects and features can be repeated. A simple k-permutation ($\mathcal{C}_n^k = \frac{n!}{n!(n-k)!}$ where $n = \mid t \mid$) can lead to miscalculations. The correct calculation of the repetitions of these item-sets in the transaction requires caution in order not to calculate occurrences more than necessary [13].

4. Mining multimedia association rules with spatial relationships.

While the previously presented content-based multimedia association rules exclusively use visual atomic features such as in Table 1, multimedia association rules with spatial relationships in addition use the extended relation with spatial predicates such as in Table 2. A method for mining multimedia association rules with spatial relationships is introduced in this section. The method uses MaxOccur after minimizing predicates. Since spatial predicates (next-to, overlap, etc.) have two arguments, the strategy is to find frequent one and two-item-sets, combine the spatial predicates with only these frequent item-sets and consider the result as the candidate 1-item-sets of the multimedia association rules with spatial relationship. MaxOccur is then used to find the k-item-sets of frequent spatial predicates. This strategy is based on the following property: **for a spatial predicate $P(X,Y)$ to be sufficiently frequent, X and Y have to be sufficiently frequent, and the 2-item-set $\{X,Y\}$ has to be sufficiently frequent**. This can be done at any conceptual level, starting from the highest concept in the hierarchy to the lowest ones. The naïve method would be to combine all pairs of object attributes at a given conceptual level and join them with all spatial predicates to derive potential 1-item-sets. This, however, would generate a very large number of candidates and even candidates that do not exist in the data set. Our modus operandi is to lessen the candidate set to the minimum before computing the frequent spatial predicate k-item-sets. To simplify the discussion, we will analyze an abstract example with one conceptual level and one dimension (shape) as follows:

Example 4.1 Considering the three images in Figure 2 with one dimensional objects, we would like to find association rules involving the spatial relationships between the objects in the images. For simplicity, we are only considering the dimension shape at a given conceptual level, but the same can be applied for other dimensions such as colour, texture, etc. with related concept hierarchies. Finding sufficiently strong association rules with spatial relationships essentially consists of finding the sufficiently frequent conjunctions of spatial predicates. To do so, given the transaction-based minimum support threshold $\sigma\prime = 3$, a first scan of the image set reveals only three frequent items: ◯, △ and ▭, each occurring in the three images and appearing at maximum twice in an image. Considering only these three frequent items, a second scan of the data set reveals the frequent pairs of items. The first table in Tables 8 indicates the support of each of these pairs. Only three of them are frequent enough and are coupled with the spatial predicates. Notice that if we added a wildcard $*$ to the frequent items with a de facto support equal to $\sigma\prime$, we could combine it with the frequent pairs of items, and thus generate association rules with wildcard attributes. Since we only have four spatial predicates (H-next-to, V-next-to, overlap, and include), this gives us up to 12 possibilities. However, a scan of the data set would reveal that only 7 combinations are possible, and at the same time would also compute their support and maximum occurrence in an image. The second table in Tables 8 shows the result of this scan, which is the set of frequent 1-item-set found in the first step of the MaxOccur. MaxOccur can then be used to discover the following frequent k-item-sets: Overlap(◯, △), H-Next-to(◯, △); Overlap(◯, △), H-Next-to(◯,▭); H-Next-to(◯, △), H-Next-to(◯,▭); Overlap(◯, △), H-Next-to(◯, △), H-Next-to(◯,▭), and all the derived association rules such as: H-Next-to(◯,▭) $\wedge$ H-Next-to(△,▭) $\rightarrow$ Overlap(◯, △) [100%]

Pairs of Objects	Frequency
{◯, ◯}	1
{◯, △}	3
{◯,▭}	3
{△, △}	2
{△,▭}	3
{▭,▭}	1

1-item-set	Frequency	Max Occurrence
Overlap(◯, △)	3	2
H-Next-to(◯, △)	1	1
H-Next-to(◯,▭)	3	2
H-Next-to(△,▭)	3	2
H-Next-to(▭,▭)	1	1
V-Next-to(◯, △)	1	1
V-Next-to(△,▭)	2	1

Table 8. Frequent pairs of objects and Frequent spatial predicates.

The above example and discussion proceed to the following algorithm for mining multimedia association rules with spatial relationships.

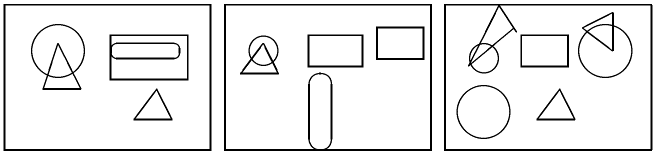

Figure 2. Examples of images with objects.

Algorithm 4.1 (**MM-Spatial**) Find sufficiently frequent item-sets for enumerating multimedia association rules with spatial relationships in image collections.

Input: (i) $\mathcal{D}_1$ a set of image descriptors with spatial relationships being the visual and non visual descriptors of the images; (ii) a set of concept hierarchies for each attribute; (iii) the minimum and maximum support thresholds $\sigma\prime$ and $\Sigma\prime$ for each conceptual level.

Output: Sufficiently frequent spatial predicate conjunctions.

Method. The pseudo-code for generating sufficiently frequent item-sets with spatial relationships is as follows:

begin
(1) $P_1 \leftarrow \{$Frequent atomic items$\}$
(2) $P_2 \leftarrow \{$Frequent pairs in $P_1 \times P_1\}$
(3) $C_1 \leftarrow \{P_2 \times \{$Spatial predicate set$\}$ and their $support\}$
(4) $F_1 \leftarrow \{Frequent$ 1 item-sets from $C_1\}$
(5) line 3 to line 16 of **MaxOccur**
end

In the process of discovering multimedia association rules with recurrent spatial relationships, we have assumed the existence of enumerated spatial relationships such as in Table 2. These relationships are simply processed by comparing the centroid of each locale as well as the radius of the locale's shape approximated to a circle (minimum bounding circle). The centroids and the radii of locales are sufficient to rapidly and efficiently give a good approximation of spatial relationships between objects in an image such as closeness, overlap and inclusion. There exist other methods for determining more precise spatial relationships. However, these methods to be effective can be computationally costly. The coarse-to-fine strategy of the PRR algorithm simplifies the process by de facto eliminating in each round the images and objects not leading to interesting rules. Ideally, we would preprocess once the detailed spatial relationships at a fine granularity and lower granularity, and have a table such as Table 2 provided to the mining module. If this computation is not preprocessed before the discovery of association rules, another step could be added to the loop of PRR (Algorithm 4.2) to determine rough spatial relationships at the current resolution level and discover association rules with these approximate spatial relationships; then, the next rounds would refine the spatial relationships for only the frequent item-sets discovered. Notice that removing the minimum bounding circles at any resolution level like in Figure 3, assists in removing false positives from enumerated frequent spatial relationships.

4.1. A progressive refinement methodology.

For effective and efficient discovery of patterns in multimedia databases, we chose a multi-resolution strategy by first finding patterns at a low (i.e. rough) resolution and persevering the search at a higher (i.e. finer) resolution with only the data selected in lower resolutions. This assumes the preservation of the patterns to be discovered in coarse resolutions. The basic idea of progressive refinement is to quickly approximate patterns at a coarse level, then eliminate false positives by verifying them at a higher resolution. The refinement, however, has to be done carefully without inadvertently eliminating false negatives. For instance, by knowing how visual features are preserved in coarse resolutions, some visual features can be tested at low resolution such as colours, others like edge density could be tested at an intermediate level, while fine texture should only be tested at a high resolution. Spatial relationships are not completely preserved. The topological characteristics are not fully retained, making the topological features change from one resolution level to the other. We discuss later the preservation and the potential changes of topological features when the image resolution is altered or improved. The refinement of the image resolution can be done in many ways. We distinguish three different refinements: (i) a cleansing at the pixel level (raster refinement). The pixel based is the traditional definition of resolution for images as in the left column of Figure 1. This refinement has many resolution levels; (ii) an approximation with minimum bounding circles. This refinement has only two resolution levels: one that approximates locales to their minimum bounding circles, and one that excludes the minimum bounding circles; and (iii) a zooming by changing the size of local tiles (tile shrinking). This refinement has five or more levels, with tile sizes $32 \times 32, 16 \times 16, 8 \times 8, 4 \times 4$, and 2×2.

The following is the general algorithm of the progressive resolution refinement for multimedia data mining.

Algorithm 4.2 (**PRR**) Progressive Resolution Refinement for Mining Multimedia Association Rules.

Input: (i) $\mathcal{D}$ a set of transactions representing images at different resolution levels, with items being the visual and non visual descriptors of the images; (ii) a set of concept hierarchies for each attribute; (iii) the minimum and maximum support thresholds $\sigma\prime$ and $\Sigma\prime$ for each conceptual level; (iv) the maximum number of resolution level available.

Output: Sufficiently frequent item-sets with recurrent items at different resolution levels R_i.

Method. The progressive resolution refinement mining of multimedia association rules proceeds as follows:

begin
(1) $i \leftarrow 0$ /* Lowest resolution level */
(2) $\mathcal{D}_0 \leftarrow \mathcal{D}$
(3) while ($i < maximum\ resolution\ level$) do { /* Coarse to fine discovery */
(4) $\quad R_i \leftarrow \{r \mid r$ is a sufficiently frequent item-set at resolution level i $(in \mathcal{D}_i)\}$
(5) $\quad i \leftarrow i + 1$ /* Move to higher resolution level */
(6) $\quad \mathcal{D}_i \leftarrow Filter(\mathcal{D}_{i-1}, R_{i-1})$
(7) }
end

The algorithm is a loop with two considerable steps: (a) finding frequent item-sets at a given resolution level; (b) reducing the size of the data set by filtering out images and infrequent objects to prepare the next round at a higher resolution. The move from one level to another does not have to be one at a time (Line 5). It is sometimes desirable to skip some resolution levels and jump to a higher one. Note that depending upon the application and the user's needs, it is not always necessary to do all the resolution levels and iterate to the highest resolution (Line 3). Line 4 calls the algorithm for enumerating frequent item-sets with recurrent items at a given resolution level. This can either be for frequent visual features or for frequent spatial relationships. We will discuss in the coming subsections the discovery procedure for these two types of association rules. $Filter(\mathcal{D}_{i-1}, R_{i-1})$ in line 6 removes images that do not contain the frequent item-sets discovered at the resolution level $i - 1$ and filters out the infrequent objects in the remaining images. This reduces the set of images and visual features to be processed at higher resolution. The filtering, however, does not consider the re-occurrence of items since the low resolution can affect the numbering of visual features. Figure 3, for example, shows one blue locale at a coarse level that becomes clearly two distinct blue locales at a finer resolution. This shows that only the presence and absence of a feature should be considered in the filtering process, and not the frequency of appearance of the features in the image. Figure 3 also illustrates an example depicting the relativity of some spatial relationships, like overlap, based on the resolution used for defining locales. While two locales may appear overlapping because their minimum bounding circles intersect, considered at the locale envelope level, they do not. Moreover, reducing the size of the tile's edge form 16×16, as in our experiments, progressively down to pixel by pixel, another level of coarse-to-fine refinement can be performed.

In [2, 3] Max Egenhofer presents a formal derivation for eight spatial relationships namely **disjoint, inside, contains, equals, meets, covered by, covers,** and **overlap**. The relationships are formulated for areas based on intersections of the boundary of an area A denoted ∂A, the interior of the area denoted A° , and the exterior of the area denoted A^-. In our study, we use the eight relationships as described by Egenhofer, but we use only boundary (∂A) and interior(A°) to define them since the boundary and interior suffice to distinguish between the different spatial relationships in our case.

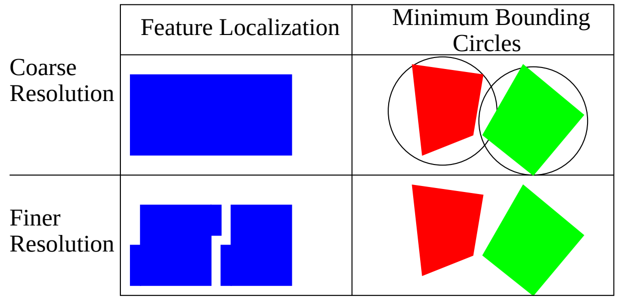

Figure 3. Relativity of visual feature concepts at different resolution levels.

Spatial relationships between locales are non deterministic from one resolution level to a finer resolution level. In other words, a given topological configuration between two areas can become a different topological configuration at a higher resolution level. Fortunately, the possible changes in topological configurations from one resolution level to the other are limited. In [13], we demonstrate the restrictions in these changes which allow a reasonable and effective filtering procedure for the progressive resolution refinement process.

Figure 4 shows the possible topology change from one resolution level to the other for the resolution refinement with exclusion of minimum bounding circles. Formal demonstrations and examples for the topology changes in both cases of resolution refinement with exclusion of minimum bounding circles and of resolution refinement with resizing of locale tiles can be found in [13].

5. Performance.

We have generated sets of synthetic images, each image with up to 15 objects. The different sized image sets produced were intended to demonstrate the scalability of the algorithms and compare their performance. Since the algorithm for mining multimedia association rules with recurrent spatial relationships uses the **MaxOccur** algorithm after two extra scans of the data set, we will only show in this section the performance of **MaxOccur**. It is obvious that the scalability of both algorithms are related. We implemented the Apriori algorithm [1] and two versions of the MaxOccur algorithm, as well as the naïve algorithm presented earlier, in ANSI C on a Pentium PC 166Mhz with

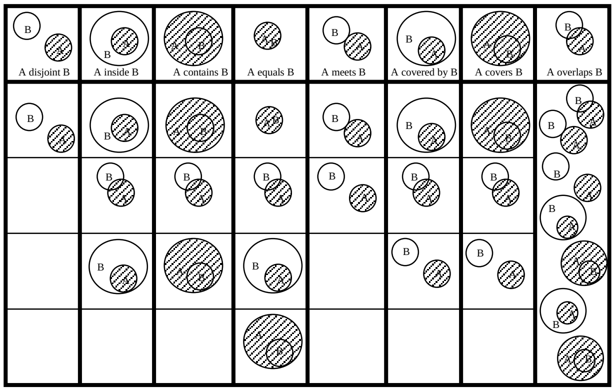

Figure 4. Topology and resolution increase with minimum bounding circles.

64Mb of main memory. Since the Apriori algorithm uses the number of transactions as support, and we wanted to compare our algorithm with Apriori, we have implemented MaxOccur and the naïve with transaction based support (MaxOccur1). The second version of MaxOccur (MaxOccur2) used the object-based support as presented in Algorithm 3.1. Table 9 shows the average execution times for the four algorithms with different image set sizes and $\sigma\prime = 0.05$ for Apriori, "Naïve" and MaxOccur1, and 0.0035 for MaxOccur2. The results are graphically illustrated in Figure 5. Clearly, MaxOccur scales well with both versions treating one thousand images in 1.3 seconds, on average, regardless of the size of the data set. The running time for filtering the frequent item-sets with $\sigma\prime$, the maximum support threshold (line 16 of Algorithm 3.1), is negligible since it is done in main memory once the frequent item-sets are determined. Moreover, the calculation of the total number of items (line 4 of Algorithm 3.1) is done during the first scan of the data set and has limited repercussion on the algorithms' execution time. The major difference between Apriori and MaxOccur is in ascertaining the candidate item-sets and counting their repeated occurrences in the images. Obviously, MaxOccur discovers more frequent item-sets. The naïve algorithm also finds the same frequent item-sets but is visibly capable of less performance in execution time. The left graphic in Figure 6 shows the average number of frequent item-sets discovered with the three algorithms: Apriori found on average 109 different frequent k-item-sets, while MaxOccur1 and Naïve found 148 on the same data sets, and MaxOccur2 found 145 on average. The discrepancy between MaxOccur1 and MaxOccur2 is basically due to the different definition of support. The price we pay in performance loss with MaxOccur is gained by more frequent item-sets and thus, more potentially useful association rules with recurrent items discovered.

# of images	Apriori	Naïve	MaxOccur1	MaxOccur2
10K	6.43	70.91	13.62	13.68
25K	15.66	176.69	32.35	34.11
50K	30.54	359.38	66.07	67.44
75K	44.93	514.33	97.27	101.23
100K	60.75	716.01	130.12	137.81

Table 9. Average execution times in seconds with different number of images.

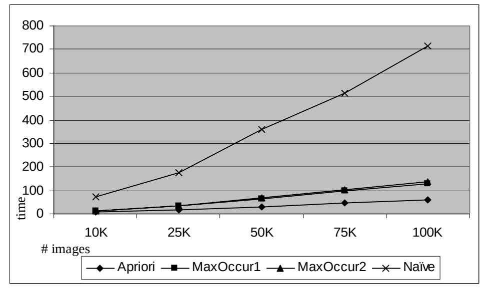

Figure 5. Scale up of the algorithms.

6. Discussion and conclusion.

We have introduced in this paper multimedia association rules based on image content and spatial relationships between visual features in images using coarse to fine resolution approach and we have demonstrated the preservation and changes in topological features during resolution refinement. We have put forth a Progressive Resolution Refinement approach for mining visual media at different resolution levels, and have presented two algorithms for the discovery of content-based multimedia association rules. These rules would be meaningful only in a homogeneous image collection; a collection of semantically similar images or received from the same source channel.

Many improvements could still be added to the multimedia mining process to speed up the discovery or to refine or generalize the discovered results.

- One major enhancement in the performance of the multimedia association rule discovery algorithms is the addition of some restrictions on the rules to be discovered. Such restrictions could be given in a **meta-rule** form. Meta-rule guided mining consists of dis-

# of images	$\sigma\prime = 0.25$	0.20	0.15	0.10	0.05
10K	1.43	2.20	2.70	5.06	13.51
25K	2.80	4.78	6.31	11.20	32.35
50K	6.27	9.28	11.59	22.74	66.07
75K	8.24	13.57	17.69	33.94	97.27
100K	11.32	17.63	23.13	46.74	130.12

Table 10. Average execution time in seconds of MaxOccur with different thresholds.

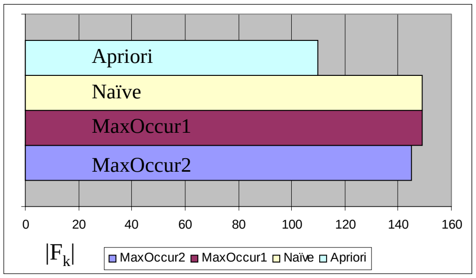

Figure 6. Frequent item-sets found by the different algorithms.

covering rules that not only are frequent and confident, but also comply with the meta-rule template. For example, with a meta-rule such as "**H-Next-to(**X, Y**)** $\wedge$ **Colour(**x**, red)** $\wedge$ **Overlap(**Y, Z**)** $\rightarrow$ P**(**Y, Z**)**" one need only to find frequent 3-item-sets of the form $\{$H-Next-to(red, Y), Overlap($Y, *$), $P(Y, *)\}$ where Y is an attribute value and P a visual descriptor or spatial relationship predicate. Obviously, such a filter would greatly reduce the complexity of the search problem. A method for exploiting meta-rules for mining multi-level association rules is given in [5].

- We have approximated an object in an image to a locale which is an area with a consistent visual feature such as colour. Objects in images and videos are obviously more complex. In a recent paper [9], regions and their signatures are used as objects in a similarity retrieval system. A computationally efficient way to identify distinct objects in images is however still to be proposed. Automatically identifying real objects and using spatial relationships between real objects would reduce the number of rules discovered and make them more significant for some multimedia applications.

- Object recognition (or identification) in image processing and computer vision is a very active research field. Accurately identifying an object in a video, for example, as being an object in itself, is a very difficult task. We believe that data mining techniques can help in this perspective. Multimedia association rules with spatial relationships using the motion vector of locales as a conditional filter, can be used to discover whether locales moving together in a video sequence are part of the same object with a high confidence.

- There are many application domains where multimedia association rules could be applied and should be tested, such as global weather analysis and weather forecast, medical imaging, solar surface activity understanding, etc. We are investigating the application with Magnetic Resonance Imaging (MRI) to discover associations between lesioned structures in the brain or between lesions and pathological characteristics.

Further development and experiments with mining multimedia data will be reported in the future.

References

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proc. VLDB*, pages 487–499, 1994.

[2] M. J. Egenhofer. *Spatial Query Languages*. PhD thesis, University of Maine, 1989.

[3] M. J. Egenhofer and J. Sharma. Topological relations between regions in r^2 and z^2. In *Advances in Spatial Databases (SSD'93)*, Singapore, 1993.

[4] U. M. Fayyad, S. G. Djorgovski, and N. Weir. Automating the analysis and cataloging of sky surveys. In U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, *Advances in Knowledge Discovery and Data Mining*, pages 471–493. AAAI/MIT Press, 1996.

[5] Y. Fu and J. Han. Meta-rule-guided mining of association rules in relational databases. In *Proc. 1st Int. Workshop Integration of Knowledge Discovery with Deductive and Object-Oriented Databases*, pages 39–46, Singapore, Dec. 1995.

[6] J. Han and Y. Fu. Discovery of multiple-level association rules from large databases. In *Proc. VLDB*, pages 420–431, 1995.

[7] Z.-N. Li, O. R. Zaïane, and Z. Tauber. Illumination invariance and object model in content-based image and video retrieval. *Journal of Visual Communication and Image Representation*, 10(3):219–244, September 1999.

[8] R. Miller and Y. Yang. Association rules over interval data. In *Proc. ACM-SIGMOD*, pages 452–461, Tucson, 1997.

[9] A. Natsev, R. Rastogi, and K. Shim. WALRUS: A similarity retrieval algorithm for image databases. In *Proc. ACM-SIGMOD*, pages 395–406, Philadelphia, 1999.

[10] R. Ng, L. V. S. Lakshmanan, J. Han, and A. Pang. Exploratory mining and pruning optimizations of constrained associations rules. In *Proc. ACM-SIGMOD*, Seattle, 1998.

[11] R. Srikant and R. Agrawal. Mining quantitative association rules in large relational tables. In *Proc. ACM-SIGMOD*, pages 1–12, Montreal, 1996.

[12] P. Stolorz, H. Nakamura, E. Mesrobian, R. Muntz, E. Shek, J. Santos, J. Yi, K. Ng, S. Chien, C. Mechoso, and J. Farrara. Fast spatio-temporal data mining of large geophysical datasets. In *Proc. Int. Conf. on KDD*, pages 300–305, 1995.

[13] O. R. Zaïane. *Resource and Knowledge Discovery from the Internet and Multimedia Repositories*. PhD thesis, School of Computing Science, Simon Fraser University, March 1999.

[14] O. R. Zaïane, J. Han, Z.-N. Li, J. Y. Chiang, and S. Chee. MultiMediaMiner: A system prototype for multimedia data mining. In *Proc. ACM-SIGMOD*, Seattle, 1998.

[15] O. R. Zaïane, J. Han, Z.-N. Li, and J. Hou. Mining multimedia data. In *CASCON'98: Meeting of Minds*, Toronto, 1998.