Chapter 16

UNIFORM INDEXING FOR GEOSPATIAL DATA AND AUTHORIZATIONS

Vijayalakshmi Atluri and Pietro Mazzoleni

Abstract

High-resolution satellites are promising technologies to bolster global transparency by providing unprecedented access to accurate and timely information. A significant number in this satellite fleet are owned by private organizations, which can collect images of human activities and the environment, and make them commercially available around the globe. Uncontrolled dissemination of this information may have grave implications on national security and personal privacy, as some governments and private groups may exploit this information for aggressive purposes. Formal policies for prohibiting the release of imagery beyond a certain resolution, and notifying when an image crosses an international boundary or when such a request is made, are beginning to emerge. As such, there is a need for effective and efficient schemes for facilitating controlled dissemination of satellite imagery and the information products generated from it. Unlike conventional authorizations that can be implemented as access control lists, since authorizations on geospatial imagery involve spatial attributes, managing the authorization base and searching for authorizations based on the spatial extent require a spatial indexing structure. An access request thus requires a search on two index structures, one for the authorizations, and the other for the image database. In this paper, we propose an indexing structure, called *RMX-quadtrees*, that is capable of indexing multi-resolution satellite imagery and additionally allow representing authorizations on top of it. By employing a uniform index for both image database and authorization base, access requests can be processed more efficiently as only one index structure need to be traversed. We demonstrate how authorizations with privilege modes such as *view* and *zoom-in* can be implemented using the RMX-quadtrees.

Keywords: Access control, geospatial images, indexing

1. Introduction

High resolution satellites are promising technologies to bolster global transparency by providing unprecedented access to accurate and timely information

---

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: [10.1007/978-0-387-35697-6_26](https://doi.org/10.1007/978-0-387-35697-6_26)

E. Gudes et al. (eds.), *Research Directions in Data and Applications Security*

© IFIP International Federation for Information Processing 2003

on many important, perhaps controversial, developments. Due to the fact that a significant number in this satellite fleet are owned by private organizations, these can collect images of human activities and the environment and make them commercially available around the globe. There are now more than 15 commercial satellites (for example, IKONOS, ORBVUE, EROS and QUICK-BIRD [1–3]) that are privately owned and provide images with resolution of 1 meter or smaller. One-meter imagery enables the viewing of houses, automobiles and aircrafts, and will make it possible to create highly precise digital maps and three-dimensional fly-through scenes. Such images are available at prices as low as \$10 and one would expect improved quality at better prices in future. As low cost high resolution images become available, it is not unrealistic in near future to have information products such as that of Microsoft TerraServer [7] in which a data warehouse of image pyramids constructed from aerial, satellite, and topographic images of the earth (Some preliminary products can be found at: www.spaceimaging.com). With such data warehouses built with images taken at closer time intervals, one can attain near real-time surveillance of a desired region.

While, for the most part, the high resolution satellite imagery meets the wide ranging civil and commercial remote sensing exigencies such as environmental monitoring, map making, humanitarian and disaster relief operations, infrastructure planning, traffic control, crops and vegetation monitoring, emergency evacuation planning, it has implications at both national and domestic level. Without any political borders, a broad range of government agencies (including international), the news media, businesses and nongovernmental organizations will gain access to satellite images with the sharpness and quality previously available only from U.S., Russian, and French military satellites. Even individuals and small groups will be able to purchase images off the Internet. As one can imagine, not all the purchasers of this information have benign intentions. Some governments and private groups may exploit this information for aggressive purposes. This may have grave implications on national security and personal privacy.

The threat to privacy is compounded due to the fact that public entities, such as local governments and public utility companies, collect, use and disseminate a large amount of personal information. With the combination of this publicly available personal data pools and the high-resolution image data, it is not impossible to not only identify an individual or an organization by name and address, but can visually “see.” Potential threats include observation of military operations and movements by foreign countries, surveillance of outdoor activities by criminals, tracking of shipping volumes and patterns of a company by its competitors, harmful usage of satellite imagery by terrorists and narcotics criminal groups.

Policies for prohibiting the release of imagery beyond a certain resolution (e.g., U.S. Department of Commerce guidelines), notifying when an image crosses an international boundary, or when such a request is made, etc., are beginning to emerge [6, 11]. In fact, commercial organizations such as Space Imaging [4] agree that the image collection or distribution must be constrained where there was an immediate and substantial threat to national security or foreign policy commitment. For example, currently they enforce certain policies to control the dissemination of its imagery. It will not sell imagery to defined terrorist nations and violate UN or bilateral trade restrictions. Moreover, all of its affiliates and re-sellers must comply with applicable restrictions.

There is thus a great need for effective and efficient schemes for facilitating such controlled dissemination of satellite imagery and the information products generated from it. Recently, Chun and Atluri [9] have proposed an authorization model to provide access control for geospatial images based on their resolution, called GeoSpatial Authorization Model (GSAM). In GSAM, authorizations let one to specify that a subject is allowed to view a specific image or region with a certain resolution, or allowed to overlay a set of images with a specific resolution. To accomplish this, GSAM supports, in addition to *read*, *insert*, *delete* and *modify*, other privilege modes such as *view*, *zoom-in*, *overlay* and *identify* that can be defined based on the allowed resolution level for a given subject.

Although implementation of authorizations as access control list, capability list or access matrix is suitable for traditional data, it is not adequate for multimedia data. This is because, as some of the authorizations comprise of non-traditional attributes, such as spatial as in this case. Since authorizations on geospatial imagery involve spatial attributes, it is more efficient to use a spatial indexing structure for managing the authorization base and searching for authorizations based on the spatial extent. When a user issues a request to access an object(s), specified either as an area or an object identifier, in a specific privilege mode, the system must first verify whether such an authorization exists in the authorization base. Then it must retrieve the relevant objects from the image database. As such, serving an access request requires searching for authorizations from the authorization base, as well as searching for relevant images. That is, once it is verified that a subject can access an object, which is an image, its retrieval requires another index search. We argue in this paper that if we devise an indexing structure for images that is capable of additionally representing authorizations on top of the image index, processing of access requests can be done more efficiently as it is necessary to search only one index structure. In this paper, we propose such a structure, called *RMX-quadtree* that allows one to implement access control on top of a spatial index.

This paper is organized as follows. In Section 2, we present the authorization model for geospatial imagery and discuss how an authorization index

structure can be created. In Section 3, we present our proposed indexing structure, the RMX-quadtrees. In Section 4, we demonstrate how *RMX-quadtrees* can index both data and authorizations in a uniform way, and present our approach to conduct both searching for an image and evaluation of an authorization simultaneously. In Section 5, we provide some insight into our current and future research in this area.

2. Geospatial Authorizations and Indexing

2.1 Geospatial Authorization Model

In this section, we review and extend the GeoSpatial Authorization Model (GSAM), proposed earlier by Chun and Atluri [9]. For providing controlled access to geospatial images, GSAM uses publicly available user information, such as property ownership and voter registration records to determine the spatial extent that the user is allowed to access.

Objects in GSAM include geospatial *raster* images with multiple resolutions that represent a geographical region. We assume that each geospatial image object is associated with a unique identifier, *id*, and comprises of spatial features including *latitude(t)*, *longitude(g)*, *height(h)*, *width(w)*, *resolution(r)*, and a *link(l)*, or canonical landmarks like the name of the city or the street. In addition to the raster images, objects include digital *vector* data that store image as points, lines and polygons. Moreover, objects include *tabular* data linked to the images that includes information such as census, voter registration, land ownership, and land use data.

As specified in Section 1, in addition to the insert, delete and update *privilege modes*, GSAM supports privilege modes such as *view*, *zoom-in* and *identify* that are required to provide controlled access to multi-resolution imagery. The view privilege allows a user to view an image object covering a certain geographic area, the zoom-in privilege allows a user to view an image covering a certain geographic region at a specific higher resolution, the overlay privilege allows users to generate composite images, where a composite image is constructed by overlaying one image on top of another, and finally, the identify privilege allows the user to view the tabular data linked to an image.

We enhance the *subject* paradigm by associating them with a set *credentials*, and grouping them into classes (similar to *credential types* [5]). These classes can then be organized into a hierarchical structure in such a way that subjects belonging to a class inherit privileges from its super classes.

An object in the *authorization* specification can be a single image, a set of images, or a region. Authorizations in GSAM allow one to specify that a subject is allowed to view a specific region with a certain resolution, or allowed to overlay a set of images with a specific resolution. In the following, we formally present authorizations on geospatial data. These authorizations are

similar to those in GSAM, but we extend them with another attribute, called *condition*, which is an expression over subject and object attributes. We allow both negative and positive authorizations. Let $S = \{s_1, s_2, \dots\}$ denote a set of subjects, $C = \{c_1, c_2, \dots\}$ a set of credentials, $O = \{o_1, o_2, \dots\}$ a set of objects, and $M = \{view, zoom-in, overlay, identify, insert, delete, update\}$ a finite set of privilege modes. Formally,

Definition 1 (Authorization) *An authorization a is a tuple: $\langle sub, obj, pr, cond, +/- \rangle$ where sub is: (i) a subject $s \in S$, or (ii) a set of credentials $c \in C$; obj is (i) an object $o \in O$, (ii) a region represented as a rectangle with longitude, latitude, height, width, or (iii) a set of object ids; pr is (i) a single privilege mode $m \in M$ or (ii) a set of privilege modes $\{m_1, m_2, \dots\} \subseteq M$; and $cond$ is a logical expression, and $+/-$ indicate it is allow/deny.*

2.2 Indexing Geospatial Authorizations

Typically authorizations are implemented either as access control lists or capabilities. As can be seen above, authorization specifications on geospatial data include spatial attributes (such as objects represented as regions), and privilege modes containing geospatial operations (such as zoom-in). Therefore, to efficiently identify the authorizations relevant to an access request, an index structure is necessary. Before we introducing such structure, the following helpful observations can be made.

(1) Within a single level of resolution, the number of authorizations associated with a region is directly proportional to the spatial extent of the area. In other words, if a subject has the privilege to access to a region, he will also be allowed to access all the regions included in it. If a subject does not have the privilege to an entire region, he may possess an authorization associated with a smaller region. Starting from this assumption, it is possible to organize the authorizations into a hierarchy based on the area they cover.

(2) The above concept can be also applied to subject classes as they inherit the privileges of their super-classes. (3) The last observation applies to the permissions on a region through different levels of resolution. It is reasonable to assume that if a subject is allowed to view a region with a certain level of resolution; he is always allowed to view the same area with a lower resolution. In other words, the zoom-out operation is always allowed.

Based on these assumptions, we can organize the authorizations into a hierarchical structure, where authorizations in the higher levels are associated with lower resolution images covering larger spatial regions, and subjects of more general classes. At lower levels of this structure we will find authorizations associated with higher resolution images covering smaller regions, and subjects belonging to more specific classes. However, such an index structure may re-

sult in inefficient organization (unbalanced trees) when authorizations are not uniformly distributed over the region.

3. RMX-Quadtree

In this section, we present our proposed indexing structure called *RMX-quadtree*, which is an extension of the well-known MX-quadtree that not only is capable of storing geospatial data organized through different levels of resolution for efficient retrieval, but also allows overlaying of authorizations on top of this index structure. In Section 3.1, we introduce the structural details of the RMX-quadtree. In Section 3.2, we discuss the need for enlarging a region in the RMX-quadtree in order to process insert, delete and search operations that are relevant to process access requests.

3.1 RMX-Quadtree Construction

As a first attempt to build a resolution based indexing structure, in this paper we impose some restrictions on the nature of the images. Later, we intend to relax these assumptions and extend our work to more general cases. We assume that (1) the spatial region represented by each image is a square, (2) images with the same resolution are non-overlapping, (3) images with the same level of resolution have the same dimensions, and (4) higher resolution images imply smaller spatial extent. In addition, we ignore the temporal aspects such as the time at which the image is valid, etc.

As in the MX-quadtree, the region is split into a grid of size $(2^k \times 2^k)$ where k (the level of detail or granularity) is fixed in advance. The nodes in the RMX-quadtree represent geospatial regions as opposed to the nodes in MX-quadtree that represent points. We represent regions by considering an image as a tuple $\langle center, side \rangle$, where *center* and *side* represent the center and the edge of the image, respectively. Given an image i , we use l_i to represent its resolution level and p_i to represent its side. Based on our third assumption stated above, p_i is a function of l_i . The intuitive idea behind the RMX-quadtree is as follows. Assume we group the images with the same resolution, and conceptually construct a MX-quadtree for each group of images separately. The RMX-quadtree is then constructed by superimposing these independent MX-quadtrees into one by joining them at the appropriate levels. We elaborate this process below. The height (k) of each of these trees can be computed as follows. Given a single level of resolution l_i representing non-overlapping images of side p_i , it is easy to demonstrate that the maximum number of images representing the region is $num_i = (\frac{n}{p_i} + 1)^2$ where n is the side of the square representing the entire spatial region of interest. Since the lower levels of the tree are generated by splitting the node into 4 equal-size regions, we need to

have the number of partitions (k_i) at least equal to the 4th root of the total number of images. Formally:

Definition 2 (Level of granularity k_i) *Given the level of resolution l_i , k_i represents the minimum level of granularity to guarantee that the MX-quadtrees contains at most one image per leaf. $k_i = \lceil \sqrt[4]{num_i} \rceil = \lceil \sqrt{(\frac{n}{p_i} + 1)} \rceil$*

Figure 1. Storing images in the RMX-quadtrees.

The assumption that k_i generates an MX-quadtrees, where each leaf is associated with at most one image of size p_i , is feasible because images are represented using their center (but not the region covered by the image). The same assumption could not be met if upper-right or lower-left corner were used instead of the center. For example, from Figure 1, we can see that by using the upper-right corner of the images maps two objects (A and B) into the same node (the most north-west leaf). Instead, if we use the center of the image, A, B and C will fall into different leaves as desired. One can formally prove that the same is true by comparing the minimum distance between two images and the side of each leaf. Using k , one can construct a tree for each set of images with the same level of resolution. Given that an MX-quadtrees sequentially splits the region into four equal-size areas independent of the inserted data, it is possible to combine all the MX-quadtrees for each level of resolution into a single RMX-quadtrees. The resulting tree will have a height equal to the k_i of tree for the set of images with the highest level of resolution and images stored not only in the leaves but as well as in the intermediate nodes.

3.2 Region Enlargement

Representing regions as points in RMX-quadtrees does not change the procedures for inserting and deleting images used for MX-quadtrees. The only difference is that in RMX-quadtrees the process has to traverse the tree only until

the level corresponding to the k_i associated with the image, and then insert or delete it from the database. However, transformation of regions to points makes *searching* for an image, and therefore *evaluating an access* request, more complex. The image requested by a user's query covering a specific region with a certain level of resolution may (i) *exactly* match an image, (ii) *contained* in an image, or (iii) *span* several images stored in the database. We refer the first two cases as an *exact query* as a specific image needs to be retrieved. Since the third case requires a number of images, we consider this as a special case of a *range-query*. An additional complication due to the transformation of region to points is that the region covered by an image stored in a node might not correspond to the area represented by the node (unlike in MX-quadtrees where the data stored in the leaves are points in the region represented by the node). We explain this problem with the following example.

Example 1 Assume the black square (in Figure 2) represents the region the user wants to retrieve. It is possible that this black square is either part of image A (gray region) located in the NW subtree of the root, or part of image B (gray region) stored in the NE subtree of the root. Since we do not know in advance where the images are stored, at a certain point in the search process, it is necessary to traverse different subtrees.

Figure 2. The region of interest may belong to different subtrees.

As a result, the cost for searching increases. The reason for this ambiguity is because we construct the index based on points rather than regions. However, if we can compute in advance the maximum possible region covered by the node in a subtree, we can restrict the search between multiple subtrees. Essentially, we adopt the following strategy to minimize the cost of search. We enlarge the bounds of the area represented by a node, and choose a subtree path only if the enlarged area overlaps the region we want to retrieve. This operation depends

upon the side of the images we are searching; the bigger the side, the wider the “possible area” covered by images in the nodes. We formally define the enlarged area as follows.

Definition 3 (p_i -enlarged region of N) *Given a node N and image of side p_i associated with a level or resolution l_i , the p_i -enlarged region of N , denoted by $Enl_{p_i}(N)$, is the spatial region obtained by extending N 's bounds by $\frac{p_i}{2}$.*

4. A Uniform Index Structure for Geospatial Data and Authorization

In this section, we show how authorizations can be layered on top of the RMX-quadtrees. Before we proceed with the details, it is important to note that a single authorization specification may apply for several images, as well as one image may have several authorizations applicable to it. To directly map the authorizations onto the RMX quadtree, we layer all the authorizations related to each level of resolution l_i within the nodes at the corresponding level of detail k_i . This way, while searching for images, one has to first reach the level containing images, then evaluate the authorizations associated with the area represented by that node, and finally retrieve the image if the user is authorized. If an authorization refers to an area represented by several images, it needs to be placed in each node overlapping that area. Although this solution is better (in terms of number of authorizations to evaluate) than simply using an access control list, it can be further improved. Instead of layering the authorizations applicable only to a node at a certain level k , we also store them at all its ancestors' nodes. The following definition formalizes the condition that must be satisfied to store an authorization at a node.

Definition 4 (Spreading rule) *Given the level of resolution l_i , and side p_i of the image(s) i stored at node N of the RMX-Quadtree, an authorization $a = \langle \text{sub}, \text{obj}, \text{pr}, \text{cond}, +/ - \rangle$ is stored in N , if the enlarged-region $Enl_{p_i}(N)$ is totally included in $\text{region}(a.\text{obj})$ and $l_{a.\text{obj}} = l_i$.*

In other words, an authorization is stored into a node N when it applies to all the possible images in N 's subtrees. Spreading rule is based on the concept introduced in Section 2 that if a subject is allowed to access an area, he is also allowed to access all the regions at the same level of resolution within it. In the RMX-Quadtree, the area covered by a node N with a given level of resolution represents the enlarged area generated according to Definition 3.

Example 2 *Figure 3 shows a simple example where the authorization-area (gray rectangle) covers the entire enlarged-area (dotted square) of the NW child of the root. In this case, according to our spreading rule, the authorizations applied to all the images stored in its subtrees can be directly stored in the NW node (gray colored).*

Figure 3. Spreading rule in the RMX-quadtrees.

The spreading rule process has to be blocked when it reaches the level of detail the authorization refers to. Going deeper in the tree will not do any good as the process of retrieving images stops when it reaches the level containing the images of that resolution. The solution above is suitable for cases where one authorization is applicable to one or more images. When an authorization is applicable to only part of an image, there are no nodes in the tree that satisfy the spreading rule. However, we still do not want these authorizations to be stored at levels deeper than the one containing images, as lower levels refer to images of higher resolution. To address this issue, we store all such authorizations in a separate spatial structure, called AUTH-quadtrees, which can be linked to this node (similar to the MX-CIF quadtree [10]). The following spread-rule algorithm details these steps.

Spread_rule

```

Input: RMX-quadtrees, N = Root of the tree, a = ( sub,obj,pr,cond,+/- )
      p = side of the image (level of resolution) of obj
//var x: number of levels traversed (initial value is 0)
//var T: enlarged area covered by Node
compute k, level of granularity for the given p;
Procedure: Spread_rule(N, a, p, x) {
  T.XLB = N.XLB - p/2; T.XUB = N.XUB + p/2;
  T.YLB = N.YLB - p/2; T.YUB = N.XLB + p/2;
  if (T) ∩ region(a.obj) = ∅ then Halt; //(Case 1)
  if (T) ⊆ region(a.obj) then //Store the auth in the node (not leaf)
    { Store a at N of RMX-quadtrees; //(Case2) Halt; }
  if (k = x) then //search has reached the leaves of the tree (Case 3)
    { if N.image ∩ region(a.obj) then store a in AUTH-quadtrees; Halt; }
  x = x+1 //Further Traversal is required (Case 4)
  Spread_rule (N.NE, a, p, x); Spread_rule (N.NW, a, p, x);
  Spread_rule (N.SE, a, p, x); Spread_rule (N.SW, a, p, x);
}

```

The algorithm traverses the tree and for each node compares the authorization object with the node's enlarged-area (T). One of the following 4 cases is possible: (1) T does not overlap the authorization object's region. Then the process stops, and the authorization is not saved at that node. (2) T is included in that of the object. The spreading rule is verified and the rule is stored in the node. Hence, the process ends because the authorization already applies to all the subtrees of the node. (3) The process has reached the level of resolution of the authorization object. Hence, the authorizations will be stored in

the AUTH-quadtrees linked to the node. (4) All the previous conditions are not satisfied meaning that the region of the authorization object overlaps the area covered by the node, but the spreading rule is not verified. Then the process will be repeated with all the four subtrees of the node until one of the above three cases is reached.

Combined Image Search and Authorization Evaluation: As a result of the *spread_rule* process, a uniform indexing structure now stores both geospatial images and authorizations. Now, we describe our *search_evaluate* algorithm for processing an access request, which simultaneously searches for the image(s) that satisfy the request, identifies the applicable authorizations, and evaluates them.

We assume the access request $ar = \langle sub, obj, l_{obj}, pr \rangle$. From this ar , the *search_evaluate* process computes the level of granularity (k) based on the resolution, the side of the image (p) for being able to enlarge the node-area, the geospatial region the user wishes to retrieve ($region(obj)$). Then the process traverses the RMX-quadtrees starting from its root (N). The *search_evaluate* process encounters one of the following three cases: (1) Stops evaluating the authorizations. This occurs when N contains an authorization that allows the user to retrieve a wider area. That is, the user, in fact, is allowed access to all the images in N 's subtree, and therefore, no more authorizations need to be evaluated. But, the search process continues by traversing the tree for reaching the level containing the relevant image(s). (2) Keep traversing the tree and check for more authorizations until reaching the node on level k , and (3) Halt the process completely. This occurs when a negative authorization is applicable at a non-leaf node N .

Since this search process checks for the authorizations early on in the tree, the number of authorizations to be verified is significantly low. In this paper, we do not deal with the issue of conflict resolution [5, 8]. The output of the *search_evaluate* procedure will be all the images overlapping the area requested by the user.

5. Conclusions and Future Work

In this paper, we argue that commercially available high-resolution satellite imagery has benign as well as malicious implications. We present an authorization model for geospatial image databases for facilitating controlled dissemination of satellite imagery and the information products generated from it. For providing effective and efficient means to serve access requests, we propose a uniform indexing structure, called *RMX-quadtrees*, which is capable of indexing multi-resolution satellite imagery and additionally allow representing authorizations on top of it. Although implementation of authorizations as access control list, capability list or access matrix is suitable for traditional

data, it is not adequate for multimedia data, as some of the authorizations comprise of non-traditional attributes, such as spatial as in this case. The situation is similar for temporal databases, moving object databases, etc.

In the future, we intend to explore developing uniform indexing schemes for such data. In addition, we will enhance the RMX-quadtrees to capture the temporal attributes of the geospatial data and authorizations. Furthermore, we intend to conduct a performance evaluation to demonstrate that our uniform indexing scheme indeed has significant impact on the response time.

References

- [1] <http://www.satellus.se>.
- [2] http://www.orbimage.com/corp/orbimage_system/ov3/.
- [3] <http://www.digitalglobe.com>.
- [4] <http://www.spaceimaging.com>.
- [5] N. Adam, V. Atluri, E. Bertino and E. Ferrari, A content-based authorization model for digital libraries, *IEEE Transactions on Knowledge and Data Engineering*, vol. 14(2), pp. 296-315, 2002.
- [6] J. Baker, K. O'Connell and R. Williamson, *Commercial Observation Satellites*, Rand Edition, 2001.
- [7] T. Barclay, D. Slutz and J. Gray, Terraserver: A spatial data warehouse, in W. Chen, J. Naughton and P. Bernstein (eds.), *Proceedings of the ACM SIGMOD Conference*, vol. 29, pp. 307-318, 2000.
- [8] E. Bertino, P. Samarati and S. Jajodia, Authorizations in relational database management systems, *Proceedings of the First ACM Conference on Computer and Communications Security*, 1993.
- [9] S. Chun and V. Atluri, Protecting privacy from continuous high-resolution satellite surveillance, *Proceedings of the Fourteenth IFIP WG 11.3 Conference on Database and Applications Security*, pp. 399-420, 2000.
- [10] H. Samet, *The Design and Analysis of Spatial Data Structures*, Addison-Wesley, 1990.
- [11] R. Wright, Private eyes, *The New York Times Magazine*, 1999.