A Fine-Grained Image Access Control Model
Bechara Al Bouna, Richard Chbeir, Alban Gabillon, Patrick Capolsini

Abstract—Sharing and publishing images and photos have become the trend of nowadays (social networks, messengers, etc.). Providing appropriate techniques to preserve privacy and protect content of sensitive and private images is an essential need. In this paper, we present a novel security model for image content protection. In our model, we provide dynamic security rules based on first order logic to express constraints that can be applied to contextual information as well as low level features of images. We finally discuss a set of experiments and studies carried out to evaluate the proposed approach.

Index Terms—Multimedia Access control, Logic, Data protection.

1 INTRODUCTION

In the last decades, there has been a rapid evolution of Web technologies and mobile devices (PDAs, cell phones, etc.) enabling users to produce and easily share large amounts of data. Multimedia objects such as pictures, videos, and audio files operate across networks and are used mainly for personal purposes but also in a multitude of areas and information systems (medical, military, education, etc.). This massive use of multimedia objects has raised several problems including privacy and confidentiality preserving. In fact, providing safe browsing and publishing of multimedia objects’ contents particularly, masking out objects of interests is considered of great importance in several privacy scenarios (e.g., hiding the face of a popular person in a photo, hiding violent scenes in a TV show, etc.). This pushed the research community to extend and/or explore dedicated access control models for multimedia database [1] [2] [3] [4] [5] [6] [7]. While these models have proved effective in some contexts, their scalability is constrained by several drawbacks that can be summarized as follows:

• Lack of proper multimedia description model. Some of the proposed approaches such as [6] [7] adopt multimedia description models that are XML-Schema based which has been considered cumbersome for the Web. In fact, the wide range of vocabularies included in these multimedia models that should be processed by an access control mechanism limit direct machine processing of low level features and focus mainly on their semantic descriptions.

• Lack of proper decision-making under uncertainty. Multimedia objects processing leads eventually to uncertain decisions due to multimedia objects’ complex structure. For instance, computing a suitable similarity score between two distinct multimedia objects depends on 1) the choice of the similarity function, 2) the content of multimedia objects (e.g. person’s face, body, etc.) and 3) their semantic descriptions. This can be explicitly important when protecting multimedia objects resulting from similarity-based queries where uncertain decisions must be interpreted.

To point out these drawbacks, we provide the following scenario.

1.1 Motivating scenario

Let us consider, for example, the case of the company SOURA¹ provides, for her public image, a photo database accessible by all employees, visitors, students, and clients through terminals in the waiting rooms. Some of the images contained in the database are stored without textual description and they are categorized as follows:

• Social photos: includes all photos taken in the social gatherings of employees with their relatives

• Meeting photos: contains all photos captured during important official meetings and general assemblies of the company

• Promotional photos: gathers photos of heads and directors who have participated in promotional events (conferences, workshops, etc.).

To use and manage the photo database, a set of dedicated processing functions (face detection, blob detection, blur, etc.) are used along with a search engine allowing users to search for photos using different techniques (i.e., categories, queries by examples, keywords, colors, etc.).

For security reasons (privacy preserving, access controlling, surveillance, etc.), the company decided to install in most of the rooms inside the company building several surveillance cameras and adopted the following authorization rules that allow to specify the users (employees, trainees, visitors, etc.) who have the right to view the database content:

• Rule 1: All users must be authenticated using their identity and picture (trainees and employees via their photos in their personal profile form, and visitors via surveillance cameras capturing their faces at the entrance gate)

• Rule 2: All users and trainees to view social and meeting photos

• Rule 3: Deny a visitor, who identified himself as Bob when capturing his photo (Figure 1), to view social and meeting photos

• Rule 4: Deny users to view social and meeting photos when the visitor Bob is nearby

• Rule 5: Deny trainees and visitors to identify the existence of Mr. Dupont (head of the company) in all photo sets.

¹. works in photo editing, montage, and publishing
Consequently, the access control model to be adopted and used by the company must be able to:

- Define a user/subject via a photo/picture
- Define accessible/authorized photos using a textual and low-level criteria (e.g., so to detect Mr. Dupont since some photos are annotated while others are not)
- Manage uncertain-decision making that could rise due to low level multimedia processing
- Specify the context of a user.

To meet all these requirements, the need for a new solution capable to adequately controlling the access to image database becomes a must.

1.2 Contributions

In this paper, we propose a new model, called FOL-IBAC, for access controlling of Image data to cope with traditional access control models drawbacks. In FOL-IBAC, we use first order logic which has been adopted in several approaches [8] [9] [10], [11] and [12], to express complex policies and provide suitable rule specification formalism. Our contributions can be summarized as follows:

- We provide a simple data representation model to properly describe the content of images and their context. We also present a set of methods and functions to process images data at different levels.
- We propose a framework for specifying the security policy. In our model the security policy consists of contextual security rules which are based on constraints applying to subjects, objects and the environment, as it is done in the ABAC model and the ORBAC model.
- We provide a second framework for specifying access control mechanisms where several multimedia-based predicates are defined to handle content processing and uncertain decisions. Indeed, in image databases, access controls can be more complicated than simply granting or denying access to resources. For example, one may prefer to apply a blurring function on a sensitive object contained in a picture rather than denying access to the whole picture. In fact, this framework allows us to express the security mechanism that should be enforced whenever a prohibition is derived from the security policy and reduce as well possible uncertainty that comes across multimedia processing.

The same language based on first-order logic is used to specify the security policy and the access control mechanisms.

The organization of the remainder of this paper is as follows. In Section 2, we review existing access control models for image and multimedia database, and we discuss their drawbacks. In Section 3, we present several definitions needed to fully understand our approach detailed in Section 4 where we also discuss our security model. Section 5 is dedicated to present the set of experiments conducted to validate our approach. The last section concludes this work and pinpoints several future directions.

2 RELATED WORK

Several studies [1] [4] [3] [27] [35] were dedicated and proposed in the literature to protect multimedia objects of different types. In the following, we will give a brief overview of these approaches and discuss their drawbacks.

Models proposed by Bertino et al. [4] and Thuraisingham et al. [5] [36] create an abstraction between the content of the video and their description. This can actually facilitate the specification of permissions based on hierarchical concepts. Thus, access control is actually implemented on the description of objects instead of their content. However, specifying multimedia objects content using expressions based on the concepts and/or objects’ identifiers is not always sufficient especially when multimedia objects lack description and annotation. In addition, classification techniques are still inefficient to reduce the gap between the semantic description of multimedia objects and their content. In other words, these techniques often require a preprocessing phase which can be less effective with few trained multimedia objects.

In [1], Nabil R. Adam et al. propose an access control model dedicated to the protection of digital libraries’ content. Although the proposed model handles fine granularities (including digital libraries content and links to object at different levels), the ability to specify flexible permissions remains limited due to the abstraction that the authors wished to create through their model descriptions of electronic libraries which is limited to concepts and objects specified by their IDs. However, the structure of multimedia objects can not be limited to the model description as multimedia objects are characterized by attributes describing their semantics (annotation, keywords, concepts, etc.), content (URI, Blobs, etc.), and relations of different types (semantic, spatial, etc.) which may be the target of an authorization or restriction.

In [37], the authors present an access control model for geo-spatial data. They specify objects using expressions that take into consideration raw data (images, satellite image, vector, etc.), regions, object resolution, timestamps, and data type(s). Regardless of permission definition flexibility which is based on low level attributes and description, the approach holds some limitations related to: 1) the complexity of defining permissions based on attributes with missing semantics such as coordinates and region types (rectangle, circle, etc.), and 2) the lack of tolerance for error due to uncertain values (geo-spatial coordinates, longitude, latitude, etc.) that may be directly related to the used equipments.

In the following, we present our Image Access Control Model that cope with previously discussed drawbacks.

3 DEFINITIONS AND TERMINOLOGIES

In this section, we will present several definitions needed to fully understand the proposed approach.

3.1 Data Representation

Below, we will explain our proposal for representing images. The proposed definitions are built on relational-object paradigm is order to be able to consider both relational, object-oriented DBMS, as well as XML-Based DBMS.
3.1.1 Image

An image is an object that contains embedded attributes with descriptions that illustrate and give important information about the captured scene. More formally, we represent an image in the following way:

$$IM :< iid, O, meta, desc, f, SOs, rel >$$

where:

- **iid** is a unique identifier associated to an image
- **O** is a reference to the raw data of the object (or the file) which can be stored as **BLOB** or **URI**
- **meta** is the set of technical descriptions and metadata embedded in the image that takes the form of data about data. Here, metadata can be defined as information not related to the semantic content of the image (such as author, location, date/time, etc.). For instance, these values can be extracted from the following attributes: exif:gpsLatitude, and exif:gpsLongitude.
- **desc** is the set of textual descriptions, keywords or annotations provided by the main user as an image and/or album/folder\(^2\) caption(s). Usually, an image description gives some relevant information about the image in terms of places, objects, people, etc. In this study, the description of an image is represented using the Dublin Core (DC) standard\(^3\)
- **f** is the set of features describing the physical and spatial content of an image such as color, texture, etc. The physical features (e.g., colors) can be described via several descriptors (such as color distribution, histograms, dominant color(s), etc.) where each descriptor is obtained by assembling a set of values with respect to a color space model. Similarly, spatial features (e.g., shape) can have several descriptors such as MBR (Minimum Bounding Rectangle), bounding circle, surface, volume, etc.
- **SOs** is a set of salient objects contained in the image which will be detailed in the next section
- **rel** is a set of triplets where each contains two sets of objects (either image or salient object) as well as their relationship(s) (semantic, spatial, and spatio-temporal).

3.1.2 Salient Object

A salient object represents an object of interest in the image such as a person (face), a logo, a building, etc. Formally, a salient object is defined as:

$$SO :< soid, loc, f, desc >$$

where:

- **soid** is a unique identifier associated to the salient object
- **loc** is the region containing the salient object in the image. The salient object is not assigned a fixed width and height. Its coordinates determine the location and the size of the salient object region. We represent a salient object as a region in the image using the Image vocabulary\(^4\)
- **f** describes the physical and visual content of a salient object (such as color histogram(s), color distribution, texture histogram, etc.)
- **desc** is the set of textual tags and annotations assigned to this salient object.

2. When an image belongs to an album or a folder
3. DC is an XML/RDF based syntax used to describe textual as well as audiovisual documents (see http://dublincore.org/)
4. http://www.bnowack.de/w3photo/pages/image_vocabs

3.1.3 Relations

Images and salient objects can share several types of relations between them.

Spatial relations as described in [13] may exist either between two salient objects or a salient object and its image. Using our representation model, they can be automatically computed using the location and images and salient objects identifiers.

Furthermore, semantic relations are of high importance and can be represented as well. In several domains, users need to describe and search images using familiar terms like invading, attacking, shifting, etc. In this manner, the user will have access to a comprehensive and intelligent description of image and its related data.

We note that some of the semantic relations can be described using spatial relations and derived according to domain application requirements. For instance, to define the relation MarriedTo, one can use the following rule:

$$\forall a_1 \forall a_2 \ (\text{MarriedTo}(a_1, a_2) \rightarrow \text{co} - \text{occur}(a_1, a_2))$$

That is, if in a given situation, the semantic relation MarriedTo is not explicitly defined between two distinct objects, the co-occur relation holds.

3.2 Data Methods and Predicates

Each attribute in our representation model can be associated to a set of simple functions/methods/predicates. Table 2 shows a sample non-exhaustive list of predicates. In addition, similarity and derived functions/predicates can be defined as well as detailed in the following subsections.

3.2.1 Distance and Similarity

Since in our representation model, we consider two kinds of attributes, comparing images and/or salient objects comes down to comparing atomic values \(A\) (e.g., tag, caption etc.), multimedia values \(M\) (e.g., low-level features), or both.

On one hand, when images come to play, comparing two atomic attribute values using standard operators (<, >, Like, \(\leq\), \(\geq\), etc.) can be inefficient and inaccurate. More particularly, when multimedia objects (images and/or salient objects) are associated with textual or spatial attributes (e.g., genre, place, etc.), the use of such operators becomes inappropriate since they do not take into account the related semantics (for example, comparing the strings Paris to France is not relevant).

On the other hand, similarity between two multimedia values in \(M\) can be computed using various distance measures defined on feature spaces (color, texture, etc.) [25] which eventually gives rise to uncertain decisions depending on the various scores returned.

In our approach, we take into account these important features of atomic and multimedia attributes by assuming that, given an attribute \(A\) in \(U = A \cup M\), either atomic or multimedia, several distances can be adopted. Consequently, the similarity of two values is defined according to a function that aggregates the results of related distances which provide reduced uncertainty and more precise results as we will show in section 5. Let \(A\) be an attribute in \(U\) (atomic or multimedia) over which \(n\) distance functions \(d_{A}^{1}, \ldots, d_{A}^{n}\) are used. We associate \(A\) with a global distance function, denoted by \(\omega_{A}\), such that for all \(a_1\) and \(a_2\) in its domain \(\text{dom}(A)\):
\[ \omega_A(a_1, a_2) = g_A(d_A^1(a_1, a_2), \ldots, d_A^n(a_1, a_2)) \]  

(3)

where \( g_A \) is an aggregation function used to aggregate a relevant score.

We note that if \( n = 1 \), then \( g_A \) is the identity function and so \( \omega_A(a_1, a_2) = d_A^1(a_1, a_2) \), for all \( a_1 \) and \( a_2 \) in \( \text{dom}(A) \).

In what follows, given an attribute \( A \) in \( U \) and a real number \( \epsilon \), when writing expressions such as \( \omega_A(a_1, a_2) \leq \epsilon \), \( \epsilon \) refers to the radius of the range operator or to the number of \( k \) neighbors to be returned, depending on the similarity operator being considered.

Let \( A \) be an attribute in \( U \) (atomic or multimedia) and \( \epsilon \) a positive real number. For all \( a_1 \) and \( a_2 \) in \( \text{dom}(A) \), \( a_1 \) and \( a_2 \) are said to be similar within \( \epsilon \), denoted as \( \sim_{\epsilon}(a_1, a_2) \):

\[ \forall a_1, \forall a_2, \forall \epsilon, \sim_{\epsilon}(a_1, a_2, \epsilon) \equiv \omega_A(a_1, a_2) \leq \epsilon \]  

(4)

3.2.2 Derived Methods

In addition to the similarity method(s) and simple functions/predicates to be assigned to each component of our data model, several derived methods and predicates can be defined as well. To illustrate this, let’s consider the following two examples. The next definition says that any picture taken on 31st of December 2010 and associated to a textual keyword party should be described by “New year’s eve 2010”

\[ \forall o \ (\text{TypeO}(o, \text{Image}) \land \text{Keyword}(o, \text{Party}) \land \text{Date}(o, 31/12/10) \rightarrow \text{Description}(o, \text{New Year 10})) \]

The function \( \text{Belong} \) can be defined as follows:

\[ \forall o_1, \forall o_2, \text{ and } \forall o_3 \]

\[ \text{Contain}(o_1, o_2) \lor (\text{Contain}(o_1, o_3) \land \sim_{\epsilon}(o_2, o_3, \epsilon)) \rightarrow \text{Belong}(o_1, o_2, \epsilon) \]

This means that one object \( o_2 \) is contained in \( o_1 \) or is similar to one of its salient objects \( o_3 \).

4 Security Model

The security model defined in this section is based on first-order logic and on the model provided in [10] and [11]. As already mentioned in the introduction, it allows us to express the security policy and the access control mechanisms that should be enforced whenever prohibitions are derived from the security policy. First, we define predicates of our language i.e., predicates applying to subject, objects and actions. We also specify predicates related to the environment since we aim at specifying dynamic contextual security policies. Second, we define the authorization rules and show how to solve the conflicts that may arise between positive and negative authorization rules. Finally, we define rules specifying access control mechanisms which should be enforced.

4.1 Subject

A subject is a user or a process run on behalf of a user. The predicate \( \text{Subject}(s) \) reads “\( s \) is a subject”. We assume here that each subject has associated atomic and complex attributes \( A = A \cup \mathcal{M} \) to describe its identity, role, age, location, photo and so on. Each attribute can be associated to a predicate in our language. Table 1 shows a sample list of predicates representing some subject attributes. Of course, this is not an exhaustive list and can be extended depending on the application. Instances of some predicates (e.g., identity/2, age/2, etc.) have to be specified one by one, whereas instances of other predicates (e.g., role/2) may be partially or totally specified by logical conditions. To illustrate this, let’s consider the following examples. The first example says that any subject which belongs to the University of French Polynesia (UPF) should be considered as trusted.

\[ \forall s (\text{Organization}(s, \text{UPF}) \rightarrow \text{TrustLevel}(s, \text{Trusted})) \]

The second example says any subject whose identity is unknown (i.e., any unregistered user) should belong to role Guest:

\[ \forall s (\neg \exists i \text{ Identity}(s, i) \rightarrow \text{Role}(s, \text{Guest})) \]

The third definition indicates that every guest whose photo is similar to (Figure 1) must not be trusted:

\[ \forall s (\text{Role}(s, \text{Guest}) \land \sim(s, \text{Suspect}.jpg, 0.7) \rightarrow \text{TrustLevel}(s, \text{Untrusted})) \]

Note that, some predicates should respect some integrity constraints. We did not include these integrity constraints into Table 1, since they may depend on the application. For example, the following integrity constraint requires all subjects to provide a photo so to be authenticated:

\[ \forall s (\text{Subject}(s) \rightarrow \exists p (\text{Identity}(s, i) \land \text{PictureOf}(s, p))) \]

The following integrity constraint requires users to have only one identity:

\[ \forall s \forall i_1 \forall i_2 (\text{Identity}(s, i_1) \land \text{Identity}(s, i_2) \rightarrow i_1 = i_2) \]

Note that, predicates for which a similar constraint applies represent a mono-valued attribute whereas predicate for which there is no similar constraint represent a multi-valued attribute.

<table>
<thead>
<tr>
<th>Subject Predicates</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identity/2</td>
<td>identity ( s ) has an identity ( I )</td>
</tr>
<tr>
<td>Role/2</td>
<td>role ( s ) plays role ( D )</td>
</tr>
<tr>
<td>Username/2</td>
<td>username ( s ) is Bob</td>
</tr>
<tr>
<td>Password/2</td>
<td>pass ( s ) is a pass</td>
</tr>
<tr>
<td>Age/2</td>
<td>age ( s ) has 18</td>
</tr>
<tr>
<td>TrustLevel/2</td>
<td>trust level ( s ) has a trusted level</td>
</tr>
<tr>
<td>Organization/2</td>
<td>organization ( s ) is affiliated to UPF organization</td>
</tr>
<tr>
<td>PictureOf/2</td>
<td>picture of ( s ) is ( \text{pic1.jpg} )</td>
</tr>
<tr>
<td>Sim/3</td>
<td>similarity ( s ) has a predefined set of textual attributes similar to the word Bob within a threshold 0.7</td>
</tr>
</tbody>
</table>

TABLE 1: Sample subject predicates
Table 2: Sample object predicates

<table>
<thead>
<tr>
<th>Action Predicates</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Belong/3</td>
<td>Belong(o_1, o_2, 0.7): reads “o_1 is similar to at least one of the objects contained in o_2” within a threshold 0.7.</td>
</tr>
</tbody>
</table>

Table 3: Sample action predicates

<table>
<thead>
<tr>
<th>Action Predicates</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TypeA/2</td>
<td>TypeA(a, view): reads “a is an action”</td>
</tr>
<tr>
<td>ISA/2</td>
<td>ISA(o_1, o_2): reads “o_1 is hierarchically linked to o_2”</td>
</tr>
</tbody>
</table>

4.2 Object

In our model, an object is mainly an image represented using our data representation previously detailed to describe its owner, content and features (colors, textures, etc.), semantic (keywords), as well as its relationships with other objects. As for subjects, some integrity constraints can be associated to object predicates depending on the application at hand. For instance, the following integrity constraint requires objects to have only one owner:

$$\forall a, \forall s_1, \forall s_2 \ (Owner(o, s_1) \land Owner(o, s_2) \rightarrow s_1 = s_2)$$

4.3 Action

An action corresponds to a composition of (sequential or parallel) a number of operations to visualize, create, edit, delete images. Each action is associated to a set of atomic attributes allowing to define its type, category and various parameters attached to the action. Predicate Action(a) reads “a is an action” and each attribute corresponds to a predicate. Table 3 shows a list of sample predicates representing some action attributes. Potentially, instances of some action predicates may be partially or totally defined by logical conditions and should also respect some integrity constraints. These integrity constraints may depend on the application. For example, the following constraint says that all actions should be typed.

$$\forall a \ (Action(a) \rightarrow \exists TypeA(a, t))$$

The second following domain constraint says that any action should be of type Read or Edit.

$$\forall a, t (TypeA(a, t) \rightarrow t = Read \lor t = Edit)$$

The third following constraint says that the type of an action is unique.

$$\forall a, t_1, t_2 (TypeA(a, t_1) \land TypeA(a, t_2) \rightarrow t_1 = t_2)$$

4.4 Environment Signature

In order to enforce access control mechanisms and to consider the context information sensed from multimedia devices and sensors, we include in our language the notion of environment signature so to represent user environment properties. Each signature is: 1) composed of atomic and complex attributes A in U = A ∪ M such as user location, user clothes, temperature, etc., and 2) associated to some predicates. For instance, InArea(s, pic, jpg, 0.7) reads “subject s is in the area described by the picture pic jpg”, i.e., $\exists a_1 PictureOfl(a_1, s) \land Belong(a_1, pic, jpg, 0.7)$.

4.5 Security Rule

Security rules specify how (and in which context) subjects can execute actions on objects. Our model includes permissions (positive rules) and prohibitions (negative rules). We define a positive authorization rule as a logical rule having the following form:

$$\forall s, \forall a, \forall o (Condition \rightarrow Permit(s, a, o))$$

where Condition is a Boolean formula and Permit(s, a, o) reads “s is permitted to execute action a on object o”.

Similarly, we define a negative authorization rule as a logical rule having the following form:

$$\forall s, \forall a, \forall o (Condition \rightarrow Deny(s, a, o))$$

where Deny(s, a, o) reads “s is forbidden to execute action a on object o”.

Authorizations propagate to sub-objects i.e., the following entailments should be considered as part of the security policy:

**E1:**

$$\forall s, \forall a, \forall o_1, \forall o_2 (Permit(s, a, o_1) \land Contain(o_1, o_2) \rightarrow Permit(s, a, o_2))$$

**E2:**

$$\forall s, \forall a, \forall o_1, \forall o_2 (Deny(s, a, o_1) \land Belong(o_2, o_1, e) \rightarrow Deny(s, a, o_2))$$

Authorizations propagate to sub-actions i.e. the following entailments should be considered as part of the security policy:

**E3:**

$$\forall s, \forall a, \forall o, \forall a_1, \forall o_1 (Permit(s, a, o) \land Contain(a_1, a_2) \rightarrow Permit(s, a_2, o))$$

**E4:**

$$\forall s, \forall a, \forall o, \forall a_1, \forall o_1 (Deny(s, a, o) \land Contain(a_1, a_2) \rightarrow Deny(s, a_2, o))$$

The default policy of our model is closed. This means that, given a subject s requesting to execute action a on object o, if neither Permit(s, a, o) nor Deny(s, a, o) can be derived from the security policy then the subject s should be denied to execute action a on object o.

Let us consider the following example of security policy which consists of the aforementioned propagation rules and the three following rules:

**Rule 1:** This statement grants to people who are on PicUsers.jpg the right to see pictures from the New Year’s Eve party:
∀s∀a∀ο0∀ο2∀ο3 (PictureOf(ο1, s) ∧ Belong(ο2, PicUsers.jpg, 0.7) ∧ Sim(ο1, ο2, 0.8) ∧ TypeA(a, Read) ∧ Album(ο3, NewYear10) → Permit(s, a, ο3))

The first line is for authenticating subject s. Subject s should exhibit a picture of himself (ο1) which looks like somebody (ο2) appearing on PicUsers.jpg. The second line says that action a should be a read action. The third line says that security object ο3 should belong to the NewYear10 album.

**Rule 2:** This statement denies to Bob the right to see pictures from the New Year’s Eve party.

∀s∀a∀ο0∀ο2∀ο3∀ο4 ∀t (PictureOf(ο1, s) ∧ Sim(ο1, Bob.jpg, 0.8) ∧ TypeA(a, Read) ∧ Album(ο2, NewYear10) → Deny(s, a, ο3))

The first line is for authenticating Bob. Subject s should exhibit a picture of himself (ο1) which looks like a reference picture of Bob (Bob.jpg). The second line says that action a should be a read action. The third line says that security object ο3 should belong to the NewYear10 album.

**Rule 3:** This statement denies to people who were on PicUsers.jpg the right to see Bob on the party images.

∀s∀a∀ο0∀ο2∀ο3∀ο4 ∀t (PictureOf(ο1, s) ∧ Belong(ο2, PicUsers.jpg) ∧ Sim(ο1, ο2, 0.8) ∧ TypeA(a, Read) ∧ Album(ο3, NewYear10) ∧ Belong(ο4, ο3, 0.7) ∧ Sim(ο4, Bob.jpg, 0.8) → Deny(s, a, ο4))

The first line is for authenticating subject s. Subject s should exhibit a picture of himself (ο1) which looks like somebody (ο2) appearing on PicUsers.jpg. The second line says that action a should be a read action. The third line says that security object ο3 belongs to a picture (ο3) from the NewYear10 album. The fourth line says that ο4 should look like a reference picture of Bob (Bob.jpg).

### 4.6 Conflicts

A positive rule may conflict with a negative rule. In the security policy specified in the previous section, we may have the following two conflicts:

- If Bob appears on PicUsers.jpg and attempts to see pictures from the NewYear10 album then Rule 1 conflicts with Rule 2.
- If somebody who is on PicUsers.jpg attempts to see a New Year’s Eve picture where Bob appears then Rule 3 conflicts with propagation rule EI (Rule 1 says that people who are on PicUsers.jpg have the right to see pictures from the New Year’s Eve party; EI grants to these people the right to see all sub-objects (including Bob) belonging to these pictures, but Rule 3 denies to these people the right to see Bob).

There exist various strategies for solving such conflicts such as the most specific takes precedence, the negative rule takes precedence, the first match, or a policy based on priority levels which would be assigned to rules.

We prefer discarding the most specific takes precedence strategy. Indeed, it is sometimes impossible to decide on which rule is more specific than the other (for instance, consider the case where the first rule has a more specific subject whereas the second rule has a more specific object). The negative rule takes precedence strategy is acceptable but only if we assume that the global policy can always be expressed by means of positive rules and that negative rules are used only for specifying exceptions to the global policy. Unfortunately, there are some policies which are easier to express by means of negative rules specifying the global policy and positive rules specifying exceptions. The first match strategy can in some scenarios be adopted but it requires being very careful when ordering the rules since the risk of having rules which never apply is very high.

For all these reasons, our model requires that the security administrator assigns priority levels to rules. If given an access request, a positive rule conflicts with a negative rule then:

- If the two rules have the same priority then the “first match” strategy is used i.e., the rule which is first read applies
- otherwise, the one with the highest priority applies.

Propagation rules E1 to E4 are also assigned with a priority level i.e., authorizations propagate downward to sub-objects (or sub-actions) unless overridden by higher priority rules.

Considering these principles, the two potential conflicts which may arise between Rule 1 & Rule 2 and E1 & Rule 3 can be solved as follows:

- by assigning a priority level to Rule 1 and Rule 2 such that the priority level of Rule 2 is higher than the priority level of Rule 1
- by assigning a priority level to Rule 3 which is higher than the priority level of propagation rule E1.

Note that, in a model enabling contextual security rules, it is not always easy to predict the conflicts which may arise between rules and consequently the security administrator may find difficult to assign priority levels to rules. The Or-BAC model [26] includes an algorithm for a priori detecting potential conflicts between security rules. Once these potential conflicts have been identified, the security administrator may specify extra information saying that some of these potential conflicts will never occur and therefore should be ignored. For example, if the security administrator indicates that Bob is not on PicUsers.jpg then, according to the Or-BAC model, there is no need to assign priority levels to Rule 1 and Rule 2 since these two rules will never conflict.

Now, if Bob is on PicUsers.jpg, then priority levels should be assigned to Rule 1 and Rule 2. We believe we can easily adapt the Or-BAC potential conflicts detection algorithm to our model although this remains work to be done.

### 4.7 Protection Mechanism

Let us consider Rule 3 and assume that the priority level of Rule 3 is higher than the priority level of E1 (i.e., Rule 3 applies). Rule 3 denies to people who are on PicUsers.jpg the right to see Bob on the New Year’s Eve party images although

5. The first match strategy is particularly used in firewalls where security rules are read sequentially. The first rule which matches the packet applies whether it is a positive or a negative rule.
these people have the right to see these pictures. Therefore, we may wonder what should happen if a subject who is on PicUsers.jpg requests viewing a picture from NewYear10 where Bob appears. Should the subject be denied to see the whole picture? Should Bob be hidden in the picture which is shown to the subject? Should the object representing Bob be blurred? etc.

To answer this question, we extend our security framework in order to enforce the access control mechanism when a given subject is forbidden to execute a given action on a given object. The idea is to hide and/or disseminate an information in order to enforce the access control mechanism when a given request (s,a,o) is rejected.

Several techniques are provided in the literature to protect textual and image data by encrypting them [27], adding some noise [28], blurring a face [28], etc.

Consequently, a negative rule is extended as follows:

$$\forall s\forall a\forall o \ (\text{Condition} \rightarrow \text{Deny}(s,a,o) \land \text{Protect}(o,m))$$

where Protect(o,m) reads “o should be protected with a mechanism m” and should respect the following integrity constraint:

$$\forall o\forall m \ (\text{Protect}(o,m) \rightarrow m = \text{Reject} \lor m = \text{Blur} \lor m = \text{Avatar} \lor m = \text{Downgrade} \lor ...)$$

If the conclusion does not include the Protect predicate then a default protection mechanism applies. For instance, if the default protection mechanism is Reject i.e., if Deny(s,a,o) is derived from the security policy then request (s,a,o) is rejected. To illustrate this, we consider the following two rules which consists of applying protection functions on a sensitive object Bob.

The first rule consists of blurring the object, while the second one allows substituting Bob’s face by an avatar (Figure 2).

**Rule 1**: $\forall s\forall a\forall o \ (\text{Type}(o_1, \text{Image}) \land \text{Description}(o_2, \text{Bob}) \land \text{Belong}(o_2, o_1, 0.7) \land \text{IsA}(a, \text{Read}) \rightarrow \text{Deny}(s,a,o_2) \land \text{Protect}(o_2, \text{Blur}))$

**Rule 2**: $\forall s\forall a\forall o \ (\text{Type}(o_1, \text{Image}) \land \text{Description}(o_2, \text{Bob}) \land \text{Belong}(o_2, o_1, 0.7) \land \text{IsA}(a, \text{Read}) \rightarrow \text{Deny}(s,a,o_2) \land \text{Protect}(o_2, \text{Avatar}))$

An even more restrictive option would be to deny access to pictures containing sensitive objects. However, this cannot be done using protection mechanisms. If pictures containing sensitive objects should not be displayed, then it means that such pictures should not be authorized at all. This should be expressed at the security policy level as follows:

$$\forall s\forall a\forall o \ (\text{Type}(o_1, \text{Image}) \land \text{Belong}(o_2, o_1, 0.7) \land \text{IsA}(a, \text{Read}) \land \text{Deny}(s,a,o_2) \rightarrow \text{Deny}(s,a,o_1))$$

This rule propagates authorizations upward in the image aggregation functions. If stated, it should have a higher priority than Rule 1.

It is important to note that protection mechanisms can also be extended to target some attributes of images and salient objects instead of main objects’ raw data. However, we do not consider this issue in this study since this process is application dependent and depends on how the protection mechanism is implemented. For instance, if one intends to protect the annotation attribute of the image rather than the image itself, (s)he could develop/use a protection function which encrypts the textual annotation of an image.

## 5 Experiments

In order to validate our approach, we conducted two sets of experiments using our prototype described in [29]. In the first one, we evaluated decision making and studied the effects of false negatives and positives when using multimedia and textual predicates in the policy enforcement. In the second one, we evaluated the utility of using FOL in multimedia access control. In the following, we detail each of the two experiments and discuss obtained results.

### 5.1 Decision Making Evaluation

In order to assess and evaluate decisions made when similarity predicates are used, we carried out various tests as shown in the following subsections. To do that, we used a computer with 2.2 GHz and 4GB RAM, a webcam with 1.3 mega pixels in video mode and more than 5 mega pixels in image mode. We also used a set of 100 images with more than 200 tagged persons (a tag represents an object of interest in a Minimum Bounding Rectangle).

#### 5.1.1 Test 1: Evaluation of false positives and false negatives when using a textual similarity predicate

In this test, we used 100 images randomly tagged with 4 tags per image. We prefixed 6 tags representing the tags that should be protected. We used in this case, a predicate of the form Sim(o_1, o_2, \beta) to search for all tags whose textual description is similar to the textual description of supplied multimedia object o_1. The similarity score is computed using three different aggregation functions (Average, BayesianNetwork^6, and Max). After varying the similarity threshold from 0 to 1, we obtained the result shown in Figure 3.

Please note that, on one hand, the rates of false positives (objects of interest which should not be targeted by the predicate) are located between 0% and 100% of the error rate axis, while, on the other hand, the false negatives (objects of interest which should be targeted but they are not) are identified when the error rates are less than 0%. This shows in reality the tradeoff in this test between privacy and availability of information. Please note also:

- MAX gives the highest (with respect to the other functions) error rate when similarity threshold \leq 0.8,
- AVG returns a false positive error rate lower than the others when similarity threshold \leq 0.5. Nevertheless, it returns a relatively high false negative error rate (with

6. We invite the reader to consult [34] for more details regarding aggregation functions
respect to the other functions) when similarity threshold $> 0.55$.
- BN reaches, when similarity threshold $= 0.8$, an error rate of 0. In addition, when threshold $\in [0.5, 0.8]$, the false negative error rate is the lowest.

5.1.2 Test 2: Evaluation of false positives and false negatives when using a multimedia similarity predicate

The objective of this test was to evaluate the error rate related to the protection of multimedia objects using their raw content and without referring to their textual descriptions. Here, we used Oracle Intermedia similarity function to protect all tags similar to an image specified by its URI. The used predicate was of the form of $\text{Sim}(o_1, \text{shape1.jpg}, \beta)$. Moreover, we tagged several shapes in the photos to specify the tags to be protected. More precisely, we specified 9 tags to protect in order to estimate the corresponding error rate which appears in Figure 4. The graph above shows that the similarity function used has an error rate close to 0 when threshold $\in [0.5, 0.8]$. However, we noticed that the false negative error rate is relatively too high when threshold $> 0.8$, and it decreases quickly until reaching -38%.

5.1.3 Discussion

According to the tests conducted in the first experiment, protecting multimedia objects using textual and multimedia predicates is appropriate to a certain extent. In fact, it is crucial to take into consideration several aspects in order to obtain satisfactory results:

- The choice of similarity functions must be done according to the application domain and the types of images to be protected,
- The profile of the authorization manager cannot be only security-based anymore and must include some background in image processing,
- The value of the similarity threshold must be chosen carefully and its tuning remains a complex task.

5.2 Utility Evaluation

As mentioned earlier, the goal of this experiment was to evaluate the utility of using our approach for access control. To do so, we designed an online questionnaire that has been answered by 124 graduate students. 67 students were from Computer Science (CS) department while the others were from Telecommunication’s (TE). The first section of the questionnaire contained the same scenario defined in the motivation section as a working example for which the students had to write several security rules using FOL. A set of sample predicates were supplied to help students write their own rules and avoid potential errors since all chosen students were discovering FOL for the first time. The rest of the questionnaire was divided into three main sections as we present in the following.

5.2.1 Evaluating Security Rule Expressiveness

This was computed using two measures: relevance and usefulness of using FOL to express security rules. The graphs displayed in Figure 5 show students’ feedback. One can notice that the majority of students agreed on the fact that FOL is both useful and relevant to express security rules (although students in TE were less convinced about FOL relevance obviously due to their background).

5.2.2 Evaluation the Difficulty of Expressing Security Rules

The graph given in Figure 6 shows the feedback of students regarding the difficulty of using FOL to express security rules. Most of the students agreed on the fact that expressing security rules is a moderate task, with a slight difference between CS and TE students who think that FOL might complicate the use of security rules.

5.2.3 Evaluating the Utility of using Multimedia Predicates

The aim here was to evaluating multimedia predicate utility when handling multimedia objects security. The results of the test are presented in Figure 7. A relevant number of CS students agree on the fact that multimedia predicates are useful and relevant to include in security rules. However, TE students were less convinced.

5.3 Accuracy of the Study

Unlike common experiments including computational tests and computer analysis, human-based experiments are often subjective and involve considerable amount of ambiguity. Hence, the results presented here, regardless of their modesty, remain valid even for large margin of errors. In fact, the goal of these studies was limited to demonstrate the usability of our approach and its ability to express security rules to address at the same time the low-level features and textual descriptions of multimedia objects. To deal with complexity of rule specification, adopting a user friendly GUI for users in order to specify their requirements easily can be considered in future work.

6 Conclusion

In this paper, we presented the FOL-IBAC, a fine-grained access control model for images described based on their semantic description as well as their low level features. We used an FOL-based language to define the model’s related information and specify security rules addressing images content. We also provided a new mechanism to deal with uncertain decision-making at policy enforcement time which has been tested in a set of experiments with textual and multimedia predicates.

In a future work, we intend to extend the proposed model to make it suitable for videos taking into consideration the complex aspects of video segments. We are also exploring some studies to automate the value of the similarity threshold so to ease the work of the end user.

ACKNOWLEDGMENTS

This study is partly funded by the CEDRE research collaboration program, project AO 2011, entitled: Easy Search and Partitioning of Visual Multimedia Data Repositories, jointly funded by the French CNRS (National Center for Scientific Research) and the Lebanese CNRS. It has also benefited from the efforts carried out by Mr. Jean Khayrallah from the Antonine University (Lebanon) during his graduation project.

REFERENCES