GAT: A GRAPHICAL ANNOTATION TOOL FOR SEMANTIC REGIONS

Xavier Giro-i-Nieto, Neus Camps, Ferran Marques

Technical University of Catalonia (UPC)

Campus Nord UPC, Modul D5
Jordi Girona 1-3
08034 Barcelona
Catalonia / Spain
(+34) 93 401 16 27
{xavier.giro, ferran.marques} @ upc.edu
http://gps-tsc.upc.es/imatge/index.html

ABSTRACT

This article presents GAT, a Graphical Annotation Tool based on a region-based hierarchical representation of images. The proposed solution uses Partition Trees to navigate through the image segments which are automatically defined at different spatial scales. Moreover, the system focuses on the navigation through ontologies for a semantic annotation of objects and of the parts that compose them. The tool has been designed under usability criteria to minimize the user interaction by trying to predict the future selection of regions and semantic classes. The implementation uses MPEG-7/XML input and output data to allow interoperability with any type of Partition Tree. This tool is publicly available and its source code can be downloaded under a free software license.

KEYWORDS: annotation; region; ontology; navigation; semantic; hierarchical

1 Introduction

The large amount of new data acquired every day by multimedia systems has created new problems related to their analysis. The popularity of electronic acquisition devices, definition of standard coding formats and expansion of digital networks have significantly increased the amount of generated audiovisual data. The digital format used for the representation of these data offers promising opportunities for the automatic analysis of images and videos from a semantic point of view. While the intervention of a human expert has been traditionally required for this analysis, the tremendous growth in content volume has raised the interest in automatic solutions in order to avoid an analogous growth in the analysis costs.

These solutions often combine algorithms coming from the signal processing, pattern recognition and semantic reasoning fields that try to reproduce the signal interpretation that a human would produce. The design of these algorithms is commonly based on a ground truth that describes the expected system output to certain input data. Whether used for learning or evaluation, this ground truth is often generated by a human expert under the form of annotations. In the machine learning context, annotations are used to define training data sets to teach the system and test data sets to compare the system outputs with the expected results.

The research behind this paper is motivated by the interest from the TV production industry to develop techniques for the automatic annotation of the large amount of videos ingested every day to their content databases. The system requirements pursue the detection of semantic concepts present in video assets, to automatically generate metadata for indexing and retrieval. In the presented approach, videos are pre-processed to extract a set of key-frames that are assumed to be representative enough to describe the asset semantic content. For this reason, the basic work unit considered in this paper is a still image. This is a reasonable assumption as extensive research has been conducted on the problem of keyframe extraction (1) providing, as a result, a feasible scenario for keyframe-based video annotation. The system is not oriented to any specific object or event, so it must allow the user to define which semantic concepts are to be detected and must offer a tool for generating training data. Taking into account the previous comments, this paper presents a graphical interface capable of generating high-quality manual annotations of key-frames in an intuitive user environment.

1.1 Area of support

The semantics contained in an image can be annotated over different areas of support. The user may select different areas of support depending on two factors. Firstly, the nature of the semantic class, as some concepts may be represented by the whole set of pixels that conform the image, while others may be very specific to a certain group of pixels. For example, abstract semantic classes such as *country* or *sports event* are in many cases expressed by the whole image, while object classes like *car* or *football player* may be only depicted by a specific portion of the image.

The second factor that conditions the selection of the area of support is the final application of the annotation. It would be useless to annotate images at a precision higher than the one that will ever be required. For example, the high-level feature evaluation of TRECVID 2008 relies on manual annotations generated by IBM's EVA (2). This web-based interface generates positive and negative labels on video shots, depending if they contain a certain concept chosen from a TV archive. Labels are applied to the whole shot although some of the considered concepts may appear in the video temporary or in a specific spatiotemporal segment. On the other hand, the video object detection task in the PASCAL challenge (3) offers annotations of ground-truth images under the form of bounding boxes, polygons and masks.

The annotation of images is typically performed at two basic visual scales: global or local. In the global case the area of support is the full image, while local annotations mark a subset of the image pixels that depict a semantic object. The global-scale approach has been chosen in photo sharing websites like *Flickr* (4), where users index their uploaded images with textual tags. Local-scale solutions can be divided in two groups depending on the sought precision. A first family of techniques provides rough descriptions of the objects, giving approximate information about their location and shape, but without aiming at determining the exact pixels that represent the objects. Common solutions for rough local annotations are based on the drawing of points, lines, bounding boxes, ellipses or polygons over the object of interest. Examples of rough annotation at the local scale may be found in the person-tagging interface used by *Facebook* (5), where the user is prompted to click on the faces of people appearing in the pictures and their usernames. By doing so, a label identifying the person is associated to a predefined square around the face. Another example of rough annotation at a local scale is the *FourEyes* (6), an interface working on an arbitrary partition in blocks. In this case the interface assists the user annotation by expanding their local annotations to other blocks with similar perceptual features. Rough but more precise local annotations are generated by the social tagging effort of *LabelMe* (7), where polygons are defined by the user trying to adjust them as much as possible to the object contours. A second option for local annotations is the precise labeling of those pixels that represent the object, by defining the exact area of support associated to the object. Systems offering precise local annotations of any generic shape can be classified into region-based or contour-based approaches. Region-based annotations (8) (9) (10) let the user select among a set of segments from an automatically generated partition of the image, while contour-based solutions aim at generating a curve that adjusts to the pixels located at the border between object and background. Examples of contour-based annotation are proposed by (11) (12) (13) (14), where scribbles are painted by the user to mark the object.

In terms of user interaction, global annotations are less demanding than local ones, as well as local rough annotations are less demanding than precise ones. On the other hand, the more user interaction is required, the more data is collected and better descriptions are generated for the applications that will use the annotations. The cases of global and rough local annotations do not present nowadays important challenges in the research field, while current efforts focus on assisting the user into selecting local and precise segments in the image. This paper concentrates in this latter case.

Both region- and contour-based precise annotations provide similar accuracy, though the interaction can be simpler in the first case. In region-based annotation, the user is prompted to select among previously computed segments, while the contour-based case requires drawing a scribble that is later automatically adjusted to the object contour. The effort required to select a region is less demanding than the task of drawing a useful scribble, providing the first approach more opportunities to develop intuitive interfaces.

However, in both cases the assumption that one marker defined by the user on the image corresponds to only one semantic entity might be too restrictive in some scenarios. For example, let us consider an image showing a *painting* of a *person*. In this situation the pixels representing the *person* also belong to the *painting,* so their semantic interpretation depends on the spatial scale. Similar situations will occur every time a semantic instance is depicted by a segment located inside a larger segment that represents at the same time another semantic instance. In these cases the user should be offered all possible options and have the chance to easily select among the different spatial scales. This article proposes region-based hierarchical representations (15) as an intuitive framework to both represent the multiple spatial scales and to define a navigation path among them.

1.2 Ontology

Whatever method is used to select the area of support, an annotation assigns a semantic interpretation by annotating an instance of a semantic class. These classes are normally defined in a thesaurus or ontology that must be common to all annotations of a given domain. Thanks to a unified definition of semantics, a posterior analysis of the annotations can exploit the relations between instances of a same class as well as between instances of different classes. In the case of TV archives, these ontologies may include classes related to the domain (e.g. sports, news), events (e.g. goal, speech), locations (e.g. stadium, congress), people (e.g. player X, president Y) or objects (e.g. ball, flag). These classes are semantically linked in many cases and their relations can be exploited by knowledge-based systems, for example, to filter automatic analysis results or to assist the user during retrieval by expanding queries (16). For example, the open annotation tool of the VARS system (17) accesses an ontology of biological and geological concepts to assist users in a precise scientific tagging of submarine videos.

The use of ontologies in a system drives to the question of how they should be created and which relations should be considered to link the semantic classes. Solutions can range from a manual definition by an expert to an automatic generation by data mining algorithms. Whatever option is chosen, the semantic annotation of images provides valuable data which commonly refer to two types of relations. Firstly, the concepts appearing in an image annotation are linked by a co-occurrence relation, as they are present in the same document (18). Secondly, local annotations provide geometric and topographic data (eg. relative position or size between instances) which can be measured and used to generate models of the interaction between semantic entities (19).

Given the region-based nature of the proposed approach, this work also considers a third type of semantic relations to describe the parts of an object. The *part* relation links a semantic class with those other classes that compose it. This relation provides valuable data for image analysis algorithms that may use the annotation to build models of the object parts and their relations, sometimes easier to create than a model for the whole composite object.

The definition of ontologies in the multimedia field has received the attention of several researchers in the last years. A first family of non-semantic languages has been developed in the framework of the MPEG-7 (20) and MPEG-21 (21) standards. These initiatives have mainly focused in a structured description of multimedia content and offer tools for the description of both low-level perceptual features as well as high-level semantic concepts. VideoAnnEx (22) and SVAS (23) are tools that use the first of these two standards; however, these two languages miss the formality required by Semantic Web technologies, normally based on RDF or OWL languages (24). For this reason, a second group of initiatives have developed formal semantic ontologies to solve this limitation, mapping the multimedia descriptors and schemes proposed in MPEG-7 to semantic languages (25). Once the multimedia related concepts are formally described in an multimedia ontology, they can be complemented with another domain-specific ontology. By doing so, the concepts common to any multimedia content are referred to the multimedia ontology while those particular to the application are defined in the domain-specific one. This approach was proposed in (26) to combine four different ontologies, two for multimedia concepts (structure and perceptual descriptors), and two for the domains of interest (athletics event and geographic information). An example of a region-based annotation tool combining multimedia and domain-specific ontologies is M-OntoMat-Annotizer (10), which incorporates a plug-in to enrich semantic descriptions with low-level visual features automatically extracted from the image segments. This tool has evolved into KAT (27), an annotation tool designed as a framework for external plug-ins to generate and manage annotations based on the multimedia COMM ontology (28).

sent work, the generated annotations are encoded in an MPEG-7/XML description, which provides tools to describe the sets of regions associated to each instance as well as the *part* relations between different instances.

This paper presents a tool for the manual annotation of semantic objects and their parts using a region-based hierarchical representation of the images. The proposed solution expands the approaches in (9) (10) with an intuitive navigation through the image partitions at different scales. A second contribution focuses in defining an integrated annotation cycle for the objects and their parts, also by means of a friendly navigation through the concepts defined in the ontology. **Figure 1** shows a screenshot of the tool in which two instances of the semantic class *Car* have been annotated.

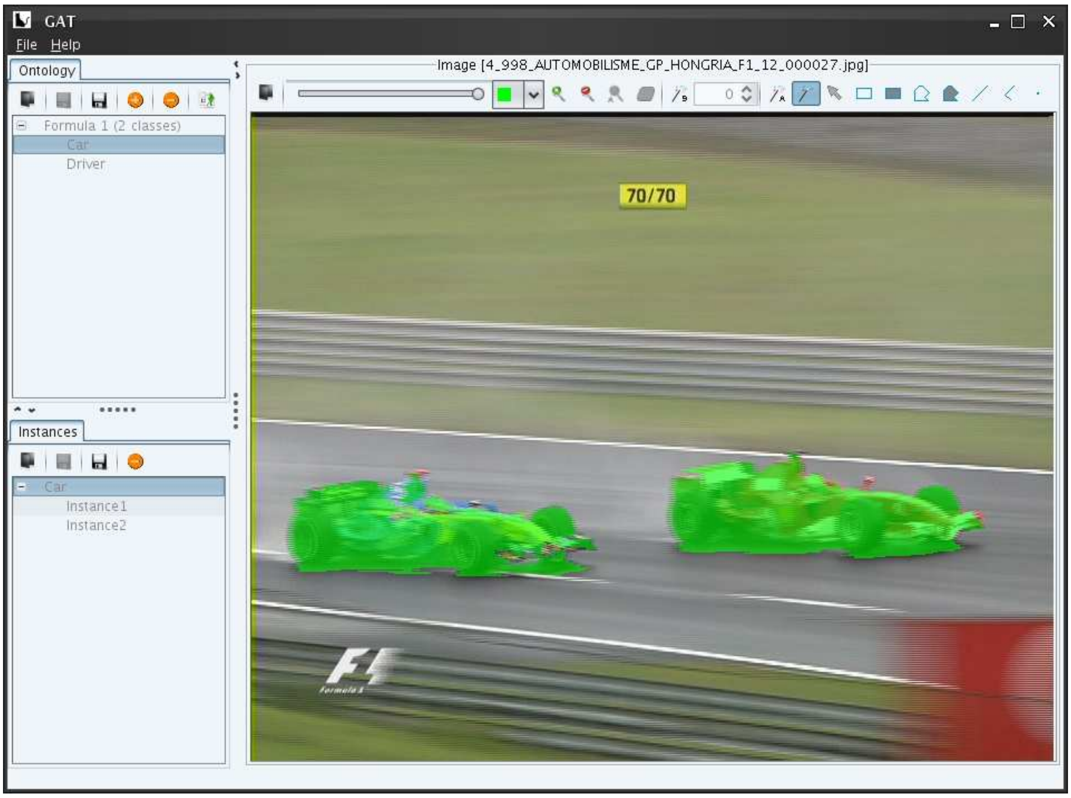

Figure 1: Screenshot of the annotation tool

The paper is structured as follows. The usefulness of a part-based annotation both at the visual and semantic levels is discussed in Section 2. Section 3 describes the user interaction with a graphical user interface to navigate through region hierarchies and semantic hierarchies. Section 4 describes the input and output data to show the interoperability of the presented interface. Finally Section 5 presents the conclusions and current work.

2 Annotation of parts

2.1 Image Partition Trees

The goal of the presented tool is to help the user into the selection of regions of the image which represent, individually or collectively, an instance of a certain semantic class. The main contribution is the proposal of an intuitive navigation through a region-based hierarchical representation of images.

The available regions for selection are automatically defined after a segmentation process that generates an initial partition of the image. These regions, by themselves or combined with others, must be precise enough to represent the semantic entities contained in the image. That is, single regions or combinations of them should define contours that adjust to the semantic segmentation that the user has in mind. It is reasonable to take the assumption that state-of-the-art segmentation algorithms can fulfill this requirement (27).

As previously explained, semantic objects can be present at any scale in the image and, moreover, it may be too restrictive to assume that the regions defined by the initial partition will correctly match the area of support of the semantic objects. For these reasons, a multi-scale representation of the image is automatically generated by combining the regions in the first segmentation. Starting from the initial partition, an algorithm based on the perceptual characteristics of the regions iteratively merges sets of regions to define new and larger ones. Several criteria can be used to determine the fusion sequence; like color, texture, connectivity or combinations of them (28). As a result, a data structure represented by a tree graph is generated, where each node in the graph corresponds to a combination of fused regions. The leaves of the tree correspond to the regions in the initial partition, while the root of the tree represents the whole image. This structure is called a Partition Tree (PT) because it encodes multiple partitions of one image at different scales.

Figure 2 shows an example of a hierarchical decomposition of an image into seven different regions using the algorithm described in (28). Note that such a simple PT is presented here for illustration purposes. Actually, annotators commonly work with PTs defined over lower scale initial partitions, typically containing around 200-300 regions, leading to more complex PTs as the ones presented in **Figure 3**.

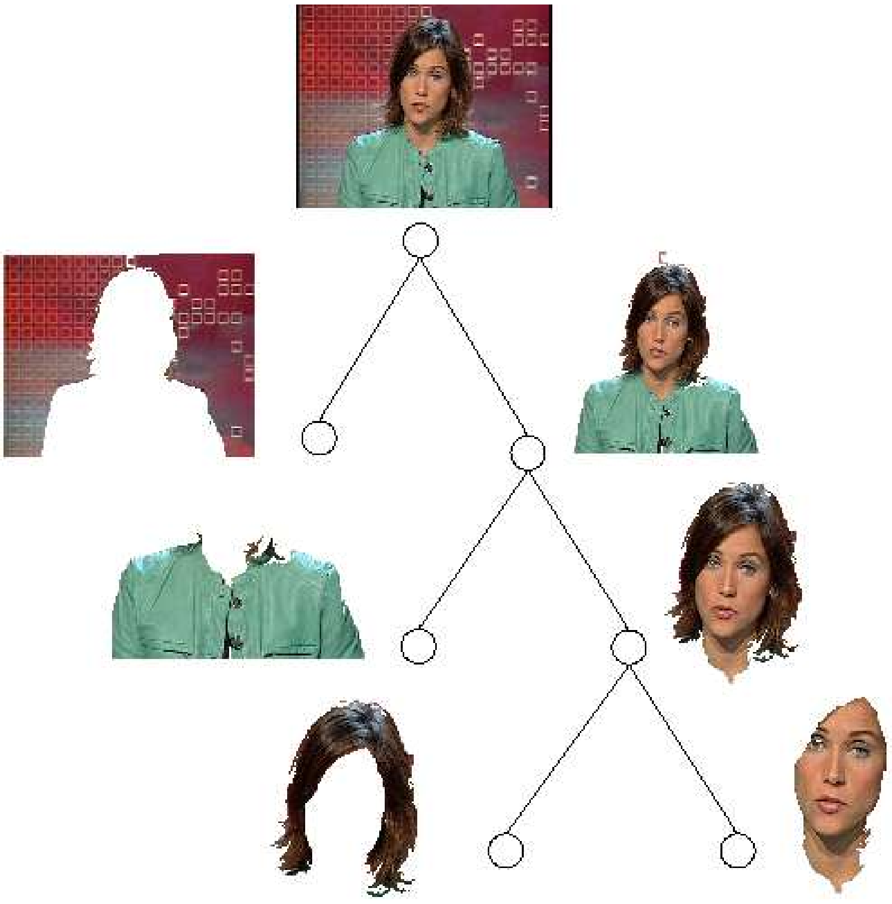

Figure 2: Hierarchical region-based decomposition with a Partition Tree (PT)

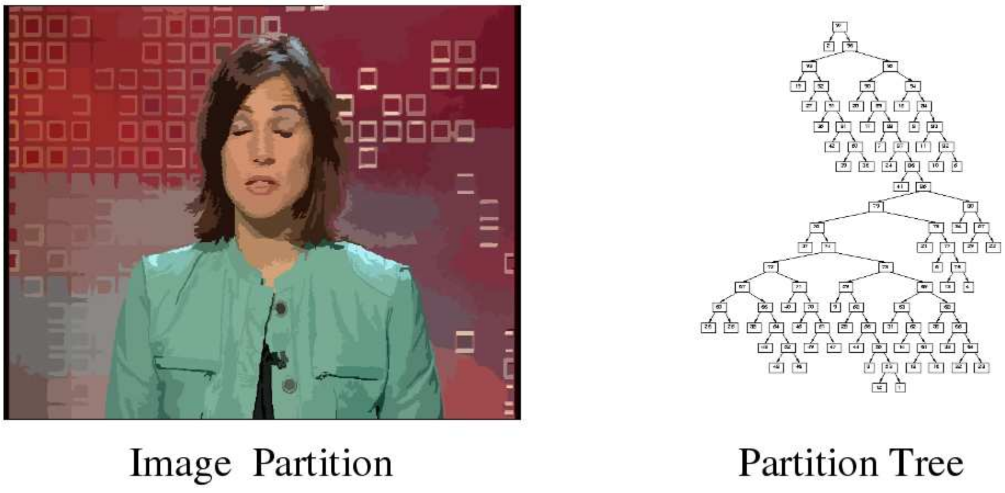

Figure 3: Examples of an Image Partition and a Partition Tree

Different solutions have been proposed for the creation of PTs, like the quad-tree (29), min-max trees (30) or tree of shapes (31) (32). In the presented examples and current implementation the *Binary Partition Tree* (33) has been chosen, although the proposed navigation and selection strategies are applicable to any type of PT. A Binary Partition Tree is a specific case of PT where merges are restricted to two neighboring nodes.

2.2 Part-based ontologies

In addition to the assisted navigation through hierarchies of regions, the presented work focuses on assisting the annotation of the parts composing semantic objects. Semantic objects and their parts are defined in an ontology accessed by the user to select which classes are being annotated. The semantic relation *part* is established during the annotation process to provide inter-class knowledge to those systems that can exploit this type of semantic relations.

Regions representing the semantic parts of an object belong at the same time to the regions representing the complete object. Annotating the two sets of regions separately may present ambiguous interpretations in some cases, as an identical annotation may be generated from a case in which one object in the foreground is occluding another larger object in the background. In order to distinguish between the two cases, the *part* relation must be explicitly indicated by the user during the annotation.

The definition (or not) of the semantic parts distinguishes between two types of annotations from the semantic point of view: atomic and composite. In case of an atomic annotation, the selected regions instantiate a single semantic class. In turn, composite annotations instantiate a class and also describe its semantic parts. As an example, the hierarchy of regions in Figure 2 could be used to generate an atomic annotation of a *TV anchor* or a more complex composite annotation of the semantic parts that compose it: *head* and *body*. Composite annotations are only possible when the semantic parts can be selected separately; that is, when they are represented by different sets of regions in the PT.

There is no limitation in the levels of semantic decomposition supported; so an instance of a semantic class which is a part of a higher level class can also be composed by instances of other sub-classes. In these cases, the manual annotation defines as well a semantic tree structure that represents the visual composition of an object from the semantic point of view. In the example of Figure 2, this concept is illustrated by the semantic part *head*, which is a part of *TV anchor* and, at the same time, is decomposed into *face* and *hair*.

3 User interaction

The previous sections have presented the required data structures to assist the user in the manual annotation of regions as instances of object classes and their decomposition in semantic parts. This section provides implementation details of an interface and navigation systems that exploit the presented concepts.

3.1 Graphical User Interface

The graphical user interface has been designed and implemented to offer the user an intuitive environment for the semantic annotation of regions. The presented interface is implemented by software that can be run on a standard workstation with a mouse and a keyboard for user input and a display to visualize results. The tool window is divided in three panels, one for each basic element of the annotation process: the ontology, the image and the annotated instances. The ontology panel is located on the upper left part of the window, the instance panel below it and the image panel occupying the central and right areas of the window, as shown in Figure 4.

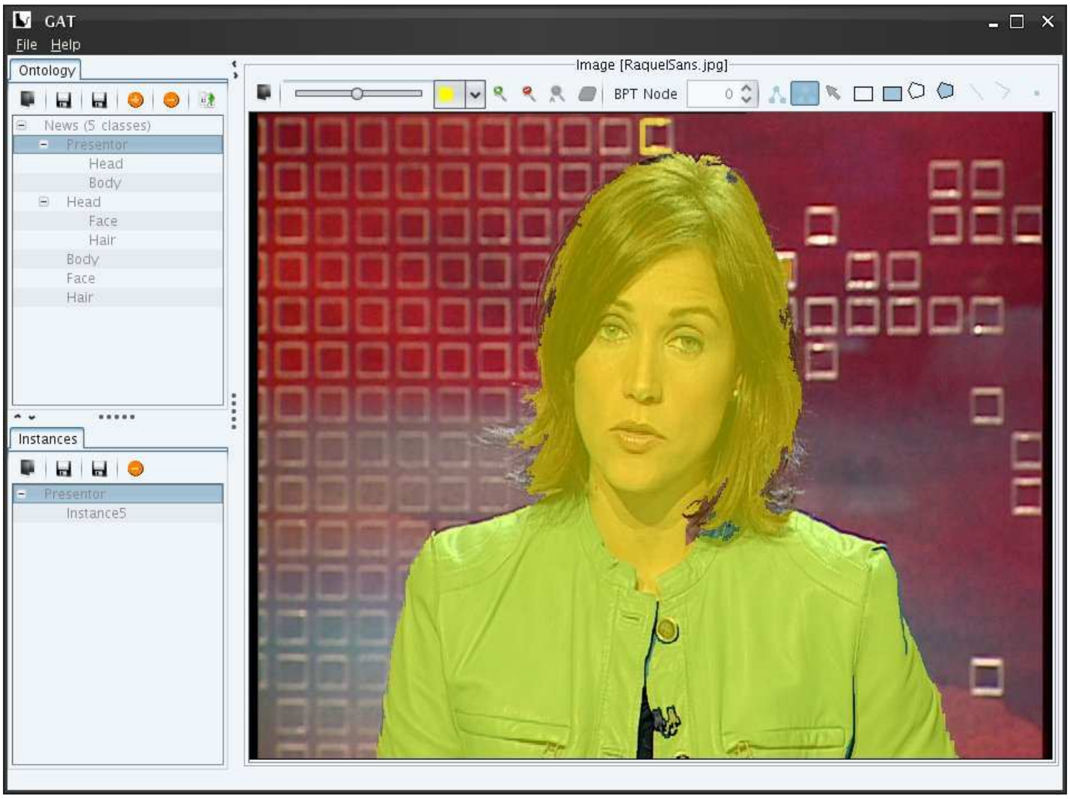

Figure 4: Screenshot of the graphical user interface

The ontology panel shows the semantic classes defined in the ontology, sorted from the most to the less recently used by the user. Those classes that have ever been used for the annotation of composite instances can be expanded to explore which semantic classes have been associated to the parts. The user can manually edit the ontology with a toolbar located at the upper part of the panel, to add, remove or rename semantic classes and their parts, as well as loading (saving) an ontology from (to) a file.

The image panel is used to determine the area of support of the semantic instances contained in an image. During region-based annotation, user selections are shown by painting a transparency mask over selected region, as shown in Figure 4. Although this paper is mainly focused in the annotation of regions, the presented tool also offers solutions for the global annotation of images as well as for the local annotation of points, lines, rectangles and polygons, whether empty or filled, as shown in Figure 5. In case of filled markers, the visualization of the selected areas is achieved through transparencies, while one-dimensional markers such as point, lines and contours are shown by opaque lines. The type of annotation can be selected on the icons at the toolbar located at the upper part of the image panel. Together with these icons, the toolbar also includes a button to select and open an image or PT file from a file, another button to clear the selected area of support, a slide bar to control the transparency of the mask layer as well as a selection palette to choose its color (see Figure 5).

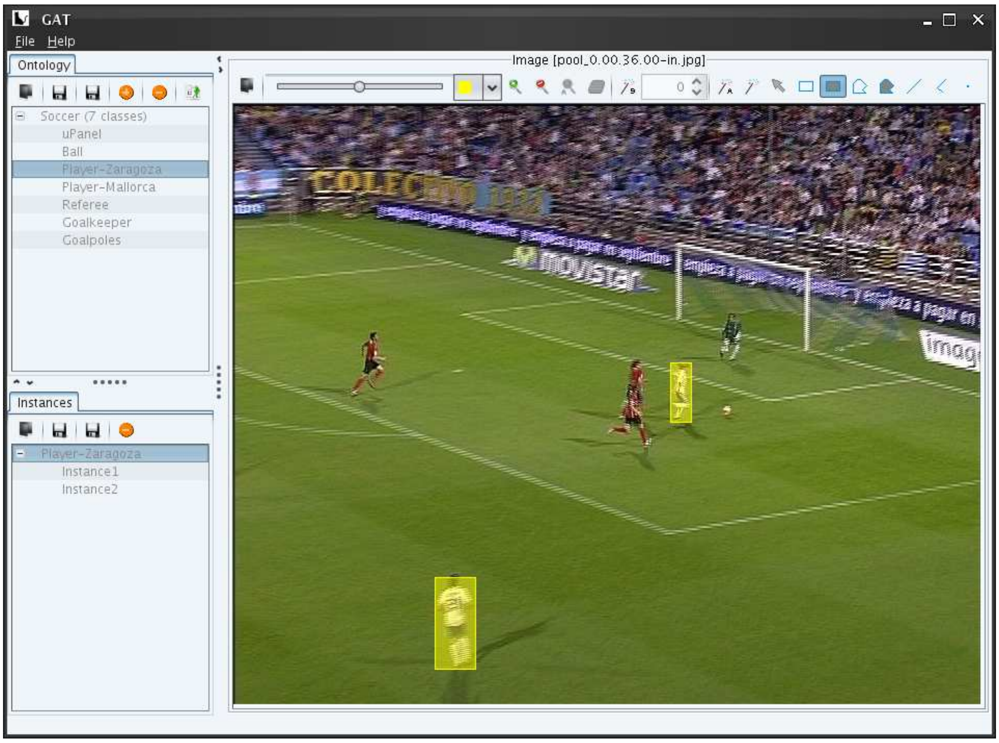

Figure 5: Annotation of filled rectangles

The instance panel shows previously annotated instances and their hierarchical structure according to the *part* relation. Whenever a new instance is created, a new entry is added to this panel. By selecting any of the instance entries, the image panel shows the areas of support related to them. The panel also includes a toolbar with buttons for loading (saving) the complete annotation to (from) a file, and another button for deleting instances or any of their parts.

3.2 Navigation

The region-based annotation of semantic parts requires both navigating through the image regions and through the semantic classes defined in the ontology. The presented tool implements different techniques to perform these operations in an efficient way, trying to minimize the interaction by predicting the user choices when selecting a visual region as well as when selecting the classes to be annotated.

3.2.1 Hierarchical region navigation

The region navigation system provides the user with an intuitive interface for selecting the Partition Tree (PT) nodes for its annotation. It uses the links between parent and children nodes to define navigation paths through the region hierarchy. The presented interface takes the input user commands from the mouse interaction and shows the selected regions on the image panel.

There are two basic scenarios for the selection of regions from a PT. Firstly, the simple case in which the selected region corresponds to a PT node; secondly, a most complex situation that requires the composite selection of more than one PT node. The two cases are presented separately, being the composite an extension of the single one.

The selection of a single PT node starts with a user inspection of the image and identification of the instance to annotate. The first interaction step consists on placing the mouse cursor on a pixel included in the area of support of the semantic object to be annotated. With this action, the user is implicitly selecting one branch from the PT, as every pixel in the image corresponds to one, and only one, branch in the PT.

After this first user interaction, the system focuses on the selected PT branch and automatically selects one of its nodes. The choice is based on a previous computation of the merging cost at every PT node, a calculation that measures the heterogeneity of the two fused nodes. The intuition behind this value is that during the PT construction some merges are more representative than others and that, in most of the cases, the most representative merging corresponds to the most meaningful region from a semantic point of view. By automatically choosing a PT node, the system expects to predict the user behavior. This type of approaches has been implemented by different authors in various region-based image representation contexts (28) (34) (35).

Once a PT node on the branch is selected, the region is highlighted on the screen so that the user can see it. Note that all this process is instantaneous and transparent to the user so that when the mouse is moved over the image, the perceived effect is the automatic selection of the representative region at the current cursor location. The selection can be validated with just a left-click on the mouse.

If the proposed region does not depict correctly the semantic object desired by the user, the selection can be modified with the mouse wheel. This is typically the case when the system proposes an object at a certain spatial scale but the user wants to analyze the image at a different scale (e.g.: the system selects the *head* while the user is willing to annotate the *face* or the complete *TV anchor*). By rotating the mouse wheel, the user can navigate through the PT branch, moving upwards or downwards in the branch at every wheel rotation. The sense of the rotation on the wheel determines whether next selection corresponds to the parent or child node. If it is the child node, this one is selected as the one containing the pixel where the mouse pointer is currently placed. The extreme situations correspond to the PT root, where the whole image is selected, and a PT leaf, where a region at the initial partition is shown.

Figure 6 shows an example in which the system automatically selects a region corresponding to an instance of semantic class *head* and the user can manually modify the selection by moving the mouse wheel and choosing other nodes in the same branch. Moving upwards would select an area representing the whole *TV anchor* while moving downwards would focus on the *face*.

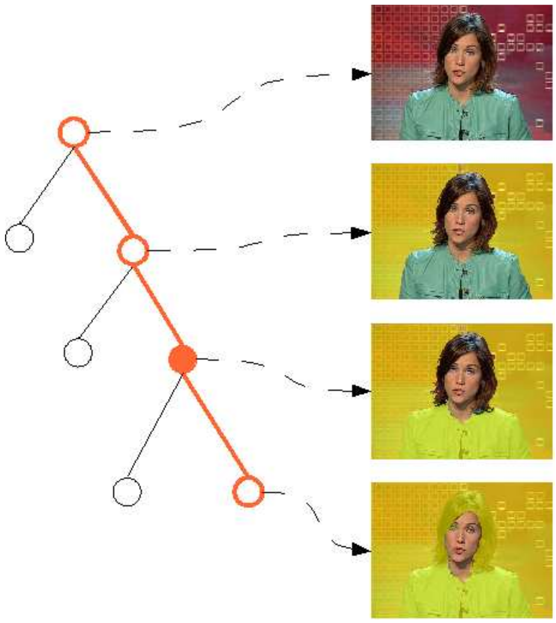

Figure 6: Navigation through the BPT with the mouse wheel

In many cases, focusing on a single PT node is not enough for determining the regions to be annotated. There are basically two situations that require choosing nodes from different branches. Firstly, when using a PT whose fusion sequence does not correctly characterize the semantic contents, in such a way that connected objects have been split into different PT branches. Secondly, those cases where the semantic contents to be annotated are represented by non-connected regions. In these cases, since the algorithm that generates the PT forces its nodes to be formed by connected regions, the object has to be represented by more than one PT nodes. In both situations, the single selection solution must be extended to the composite selection.

The process of multiple selection starts with the definition of a first component as described in the single selection case. Once the first component is validated with a left-click, the user must move the mouse cursor to a pixel located in the next segment to select. Once the cursor leaves the area of support of the first component, it is implicitly leaving the branch of the initially selected region and is changing to a new branch. When this happens, a similar recommendation mechanism to the one used for the single selection is applied, automatically choosing a new PT node in the new branch. However, the new choice is no longer based on a pre-computed recommendation but on the spatial scale of the previous selected component. This is estimated according to the size in pixels of the previous region (area). In the new branch, the recommended node is the one with the most similar area among those nodes in the new branch whose area is smaller than the reference one. By taking a smaller area, it is ensured that the new selection will never correspond to an ancestor of the previous component, a choice that would have been made by the user in the first selection had he/she been interested.

The cursor can be moved all around the image with no limitation, always activating new regions whenever a branch is left behind. Node recommendations will be based on the area of the initial selected component as it is reasonable to consider that after making a selection at a certain spatial scale, subsequent selections are likely to have a similar size. The reference area can be manually modified by the user at any time by rotating the mouse wheel, which will change the considered PT node the same way as explained in the atomic selection case. If this happens, the area of the new selected region is to be used as reference in future node recommendations.

Once the second component is selected with a left click, the user can decide to continue the selection of more components by repeating the described mechanism.

3.2.2 Semantic navigation

The second main contribution of this work aims at the assisted navigation through the semantic classes defined in an ontology. The annotation of an instance requires the selection not only of an area of support on the image but also of the concept which is to be annotated among the ones defined in the ontology. The proposed approach defines an annotation cycle for objects and their parts that tries to minimize the user interaction and, by doing so, speeding up the complete annotation process. The proposal considers the interaction only through the mouse in order to keep the same interaction methodology as in the selection of the area of support.

The basic annotation cycle of an instance starts by determining its associated semantic class. The ontology panel displays a semantic class tree whose root node represents all classes in the ontology and is labeled with a textual identifier of the used ontology. The root has as many children nodes as semantic classes in the ontology, and each of these nodes is labeled with a textual name associated to the semantic class. The semantic classes that can be decomposed into other classes are represented by expandable nodes. The semantic relation between the parent and child nodes corresponds to *partOf*, that is, the parent node can be decomposed in the semantic parts represented by its children.

The class tree shows first the most recently annotated classes by the user to try to minimize the interaction. A semantic class can be chosen whether with a left-click or with the mouse wheel. In case that the requested class is not present in the ontology, the user can define a new one by selecting the root of the class tree and clicking on the addition icon at the toolbar of the ontology panel. The system will prompt the user to introduce a textual identifier to the new class and will create a new node to the class tree. Only in this operation, as well as any other related to text input or edition, the user interacts through the keyboard instead of the mouse. Every time a new node in the semantic class tree is selected, the instance panel is refreshed to display the previous annotations of the current class. A tree structure is also employed in the instances panel, with a root node labeled with the name of the class and its children corresponding to the created instances. A left-click on any of these leaves displays the associated area of support on the image panel, while a left-click on the root shows all instances of the current class. The visualization of previously annotated instances can help the user to review his/her work and avoid repeating a previous annotation.

Once the selection of a semantic class is made, a right click creates a new node of the selected class in the instance tree. The class tree is also updated by placing the node of the selected class just after the root of the tree in order to make it more accessible in case of further annotations. At that moment the user must decide whether to initiate an atomic or a composite annotation.

An atomic annotation is started by locating the cursor over the image panel and selecting an area of support, following the previously described mechanisms (see Section 3.2.1). Whether one or multiple areas are selected, they are not associated to the new instance until the process is finished with a last right click. If no areas are selected, the instance is still created, but not associated to a specific area but to the whole image. In this way, it is possible to generate global annotations with this region-based. The complete workflow of an atomic annotation on a PT is shown in Figure 7.

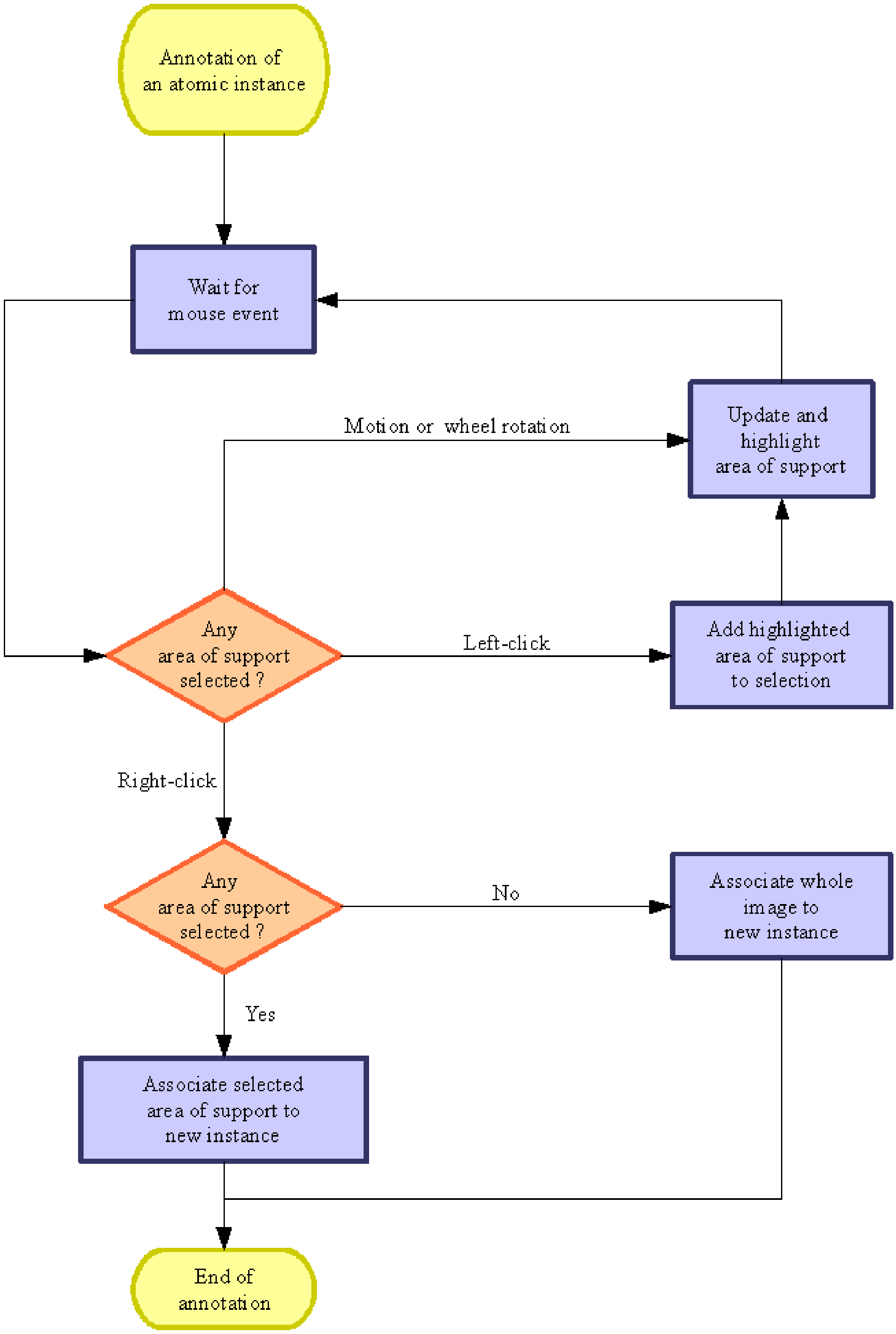

Figure 7: Annotation of an atomic instance on a PT

On the other hand, a composite annotation requires specifying the individual semantic parts that form the new instance. The composite annotation is indicated by right clicking on the tree node that represents an instance. This action adds a child to the instance node and the user is prompted to select the semantic class that is to be associated to the new part. The selection of the class is again performed on the ontology panel, showing first those classes that have been in previous occasions used as parts of the current new instance.

After selecting the semantic class of the part on the ontology panel, a second right-click launches the selection of the part’s area of support. The user must choose between two paths. A first option is to define a new atomic instance for the part by moving the cursor on the image panel and following the steps previously presented for the annotation of atomic instances. A second solution is to choose among previously annotated instances of the part class, an option which is considered by the interface while the cursor is over the ontology or instance panels. The association is achieved by choosing among a set of nodes in the instance tree which are added as children to the new part being annotated.

Whether by selecting a new area of support or referring to a previously annotated instance, the process ends with a final right click. If more parts are to be added, the process can be repeated until the composite annotation is completed.

Notice that there are cases when the union of regions associated to the semantic parts may not represent the complete instance. These situations are also considered in this tool, as when an entry is selected in the instance panel, the associated regions can be edited on the image panel. Regions can be added or removed with the selection mechanisms of PT nodes previously described and, in the case of composite annotations, new regions can complete the area of support of the instance. However, in this case it is not possible to remove from the image panel the regions associated to the instance parts; this action requires the explicit deletion of a part from the instance.

4 Input and output data

The presented tool uses two main sources of information: visual data associated to the still image and a semantic ontology from where classes are selected. The annotation tool generates an output file describing which regions of the input image depict instances of semantic classes defined in the ontology. Moreover, the same interface includes instruments for the creation and edition of the semantic ontology. User interaction is mainly acquired from the mouse as the keyboard is only used for the naming of new semantic classes or output files. Figure 8 shows the different types of data inputs to the interface.

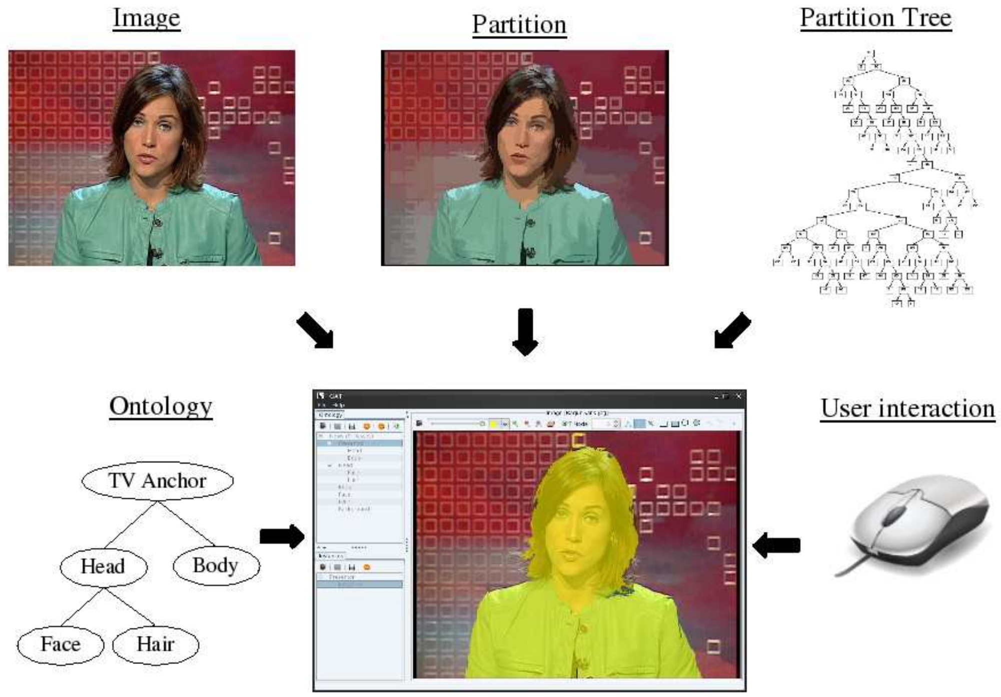

Figure 8: Input data

Three different types of input data related to the visual content are necessary:

- The still image itself; that is, the actual pixels in any standard format (JPG, PNG...).
- An initial partition of the still image previously generated through a segmentation process. This partition is represented by another image whose pixel values correspond to region labels which, in turn, correspond to PT leaves.
- The Partition Tree defining an additional set of regions formed as combinations of the regions in the initial partition.

The access to the three sources of visual data is achieved through a single MPEG-7/XML file that contains the parent-child relations among the nodes in the PT as well as the references to the files with the input image and initial partition. The standardization of the input data format allows the interoperability of this tool, as any software capable of generating an image partition and a partition tree produces valid inputs.

The concepts that can be annotated are uniquely identified and structured in an ontology of semantic classes. These semantic classes can be linked by a *part* relation, which is learnt after the annotation of a composite instance or can also be manually defined on the ontology panel. Each semantic class is characterized by a textual label defined by the user and a numeric ID which is automatically assigned by the ontology editor when created. The current implementation of the system uses MPEG-7/XML format for the definition of the ontology and can be loaded form an external file or created/edited to be later saved as an output data.

The output annotation is also expressed in MPEG-7/XML data format, describing which visual parts in the image (whether global, region or point-based) depict instances of the semantic classes defined in the ontology. Apart from the visual representation of the instances, the generated annotation includes semantic information between objects and their parts in case that the user has generated it during annotation. By doing this, composite instances are not only characterized by their perceptual characteristics but also by the semantics of their parts. Apart from the MPEG-7/XML output data format, the tool can also generate images containing the segmented objects. The area of support of these images is reconstructed by considering the PT leaves below the PT nodes associated to each instance. The locations of the pixels at the PT leaves are defined by the initial partition while their values are coded by the input image. Figure 9 summarizes the three types of output data that can be generated by the annotation tool.

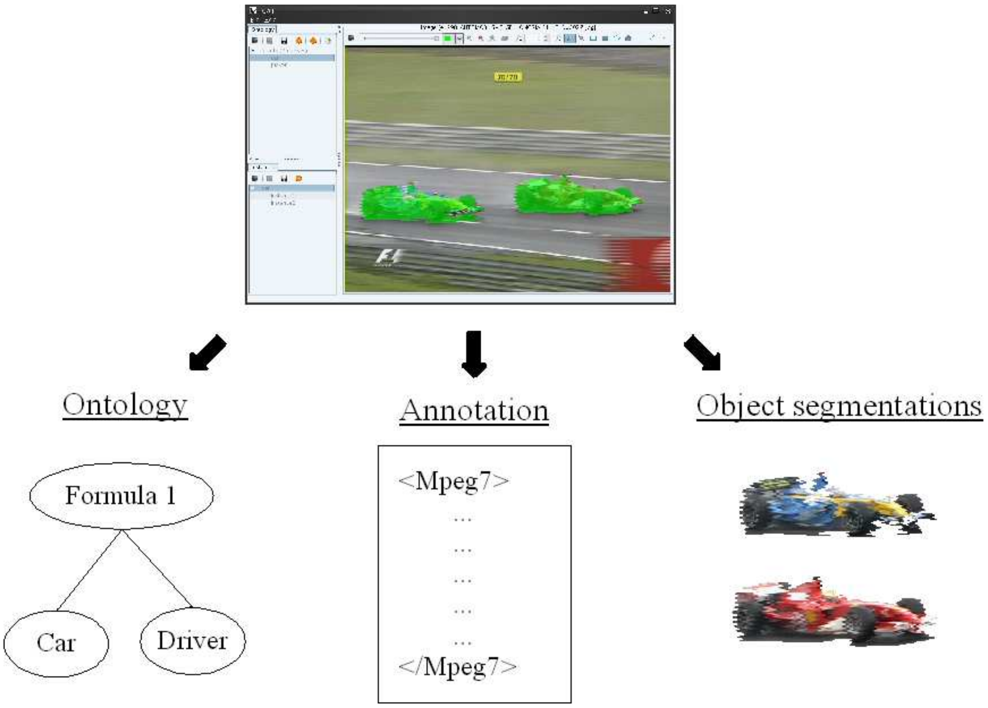

Figure 9: Output data

Table 1 shows an example of an annotation of a *TV anchor* as composed of two semantic parts, *head* and *body*. The MPEG-7/XML document has two parts. The first one instantiates the regions associated to each of the three annotated instances included in a *<Description>* tag of the *ContentEntity* type. The second part of the document describes the semantic contents of the annotation between the *<Description>* tags of the *SemanticDescription* type. The three semantic entities are related to the *StillRegion* elements through the semantic relation *depiction*, while the relations between the semantic entity *TV anchor* and its parts *head* and *body* are established by the semantic relation *part*.

```
<?xml version="1.0" encoding="UTF-8"?>
<Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
 <Description xsi:type="urn:ContentEntityType"
 xmlns:urn="urn:mpeg:mpeg7:schema:2001">
    <MultimediaContent xsi:type="urn:ImageType">
     <Image>
       <SpatialDecomposition>
         <StillRegion id="SegmentInstance1">
     <SemanticRef idref="Class2"/>
     <SpatialDecomposition>
      <StillRegionRef idref="Region1"/>
      <StillRegionRef idref="Region2"/>
     </SpatialDecomposition>
    </StillRegion>
    <StillRegion id="SegmentInstance2">
     <SemanticRef idref="Class3"/>
     <SpatialDecomposition>
      <StillRegionRef idref="Region3"/>
     </SpatialDecomposition>
    </StillRegion>
    <StillRegion id="SegmentInstance3">
     <SemanticRef idref="Class1"/>
     <SpatialDecomposition>
      <StillRegionRef idref="Region4"/>
     </SpatialDecomposition>
    </StillRegion>
   </SpatialDecomposition>
  </Image>
 </MultimediaContent>
</Description>
<Description xsi:type="urn:SemanticDescriptionType">
 <Semantics>
  <SemanticBase id="Instance1">
   <Label>
    <Name>Head</Name>
   </Label>
   <Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:depiction"
   target="SegmentInstance1"/>
  </SemanticBase>
  <SemanticBase id="Instance2">
   <Label>
    <Name>Body</Name>
```

```
        </Label>
        <Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:depiction"
        target="SegmentInstance2"/>
      </SemanticBase>
      <SemanticBase id="Instance3">
        <Label>
          <Name>TV Anchor</Name>
        </Label>
        <Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:depiction"
        target="SegmentInstance3"/>
        <Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:part"
        target="Instance1 Instance2"/>
      </SemanticBase>
    </Semantics>
  </Description>
</Mpeg7>
```

Table 1: Example of MPEG-7/XML annotation

5 Conclusions

The presented tool offers a solution for generating region-based annotations of images, describing their semantic contents and to relate them to an ontology. The interface uses image processing algorithms to assist the user in the selection of regions through a pre-computed hierarchical structure. This structure allows an intuitive navigation at different spatial scales. Furthermore, the tool integrates a mechanism to annotate the semantic relation between objects and the parts that compose them.

These annotations generate high-quality data for the training of systems that try to automate the learning of models for automatic image analysis. In the annotations not only semantic classes are instantiated by precise local regions, but these semantic classes are also expressed as combinations of simpler classes, providing training data for the automatic creation of ontologies.

The navigation workflow has been designed to minimize the user interaction while providing as many visual data as possible to guide the annotation process.

All major actions can be executed through mouse interaction in order to simplify and speed up the selection of regions from the image and of semantic classes from the ontology.

The presented tool can be used on any type of hierarchical region-based representation as long as it is coded following the MPEG-7/XML standard. The presented software is publicly available from a public website[1], as an online service and as an open source project released under the GPL license. Readers are referred to the same website for video recordings demonstrating the tool usage. Future efforts will concentrate on the introduction of ontology languages such as RDF and OWL for the manipulation of semantic data (ontologies and annotation). Moreover, coming versions of the tool will also include solutions for the massive annotation of large amount of images at the global level.

[1] http://gps-tsc.upc.es/imatge/i3media/gat/

Acknowledgements

This work was partially founded by the Catalan Broadcasting Corporation (CCMA) and Mediapro S.L. through the Spanish project CENIT-2007-1012 i3media, by TEC2007-66858/TCM PROVEC project of the Spanish Government and by a grant from the Commissioner for Universities and Research of the Innovation, Universities and Industry Department of the Catalan Government.

Copyright warnings

References

6 Bibliography

1. *Video keyframe extraction and filtering: a keyframe is not a keyframe to everyone.* **Dimitrova N, McGee T, Elenbass H.** Las Vegas : ACM New York USA, 1997. Proceedings of the sixth international conference on Information and knowledge management . pp. 113-120.

2. *A web-based system for collaborative annotation of large image and video collections: an evaluation and user study.* **Volkmer T, Smith JR, Nastev A.** Singapore : s.n., 2005. Proceedings of the 13th annual ACM international conference on Multimedia. pp. 892 - 901.

3. **Everingham M, Van Gool L, Williams CKI, Winn J, Zisserman A.** The PASCAL Visual Object Classes Challenge 2008 Results. [Online] http://www.pascal-network.org/challenges/VOC/voc2008/workshop/index.html.

4. *Flickr.* [Online] http://www.flickr.com.

5. Facebook. *Facebook.* [Online] http://www.facebook.com.

6. *Interactive Learning using a "Society of Models".* **Minka TP, Picard RW.** 4, 1997, Pattern Recognition, Vol. 30.

7. *LabelMe: a database and web-based tool for image annotation.* **Russell BC, Torralba A, Murphy KP, Freeman WT.** 1-3, May 2008, International Journal of Computer Vision, Vol. 77, pp. 157-173.

8. *A video generation tool allowing friendly user interaction.* **Marcotegui B, Correia P, Marques F, Mech R, Rosa R, Wollborn M, Zanoguera F.** Kobe, Japan : s.n., 1999. Proceedings of the ICIP 99, IEEE International Conference on Image Processing.

9. *Partition-based image representation as basis for user-assisted segmentation.* **Marques F, Marcotegui B, Zanoguera F, Correia P, Mech R, Wollborn M.** Vancouver : s.n., 2000. Proc. International Conference on Image Processing (ICIP). Vol. 1, pp. 312-315.

10. *M-OntoMat-Annotizer: Image Annotation. Linking Ontologies and Multimedia Low-Level Features.* **Petridis K, Anastasopoulos D, Saathoff C, Timmermann N, Kompatsiaris I, Staab S.** Bournemouth, U.K : s.n., 2006. Proc. of 10th International Conference on Knowledge-Based & Intelligent Information & Engineering Systems (KES 2006).

11. *GrabCut: Interactive Foreground Extraction using Iterated Graph Cuts.* **Rother C, Kolmogorov V, Blake A.** 2004, ACM Transactions on Graphics, pp. 309-314.

12. *An interactive image segmentation scheme.* **Kruse S, Bardella X, Schweitzer F, Valero M.** Portland : s.n., 1998. Proc. of Picture Coding Symposium. pp. 169-173.

13. *Distancecut: Interactive Segmentation and Matting of Images and Videos.* **Xue B, Sapiro G.** San Antonio, USA : s.n., 2007. Proc. of the IEEE International Conference on Image Processing (ICIP). Vol. 2, pp. II -249-II -252.

14. *Semiautomatic segmentation and tracking of semantic video objects.* **Gu C, Lee MC.** 5, 1998, IEEE Trans. on Circuits and Systems for Video Technology, Vol. 8, pp. 572-584.

15. *Region-Based Representation of Image and Video: Segmentation Tools for Multimedia Services.* **Marques F, Salembier P.** 8, 1999, IEEE Trans. on Circuits and Systems for Video Technology, Vol. 9, pp. 1147–1167.

16. *Context-Sensitive Semantic Query Expansion.* **Akrivas G, Wallace M, Andreou G, Stamou G, Kollias S.** Geelong, Australia : s.n., 2002. Proc. of IEEE International Conference on Artificial Intelligence Systems (ICAIS). p. 109.

17. **Monterey Bay Aquarium Research Institute.** *Video Annotation and Reference System.* [Online] http://vars.sourceforge.net.

18. *Recognizing high-level audio-visual concepts using context.* **Naphade MR, Huang TS.** Thessaloniki, Greece : IEEE, 2001. Proceedings International Conference on Image Processing (ICIP). Vol. 3, pp. 46-49.

19. *Semantic Image Analysis Using a Learning Approach and Spatial Context.* **Papadopoulos GT, Mezaris V, Dasiopoulou S, Kompatsiaris I.** Athens, Greece : Springer Berlin / Heidelberg, 2006. Vol. 4306/2006, pp. 199-211.

20. **Manjunath BS, Salembier P, Sikora T.** *Introduction to MPEG 7: Multimedia Content Description Language.* s.l. : Wiley, 2002.

21. **Burnett IS, Pereira F, Van de Walle R, Koenen R.** *The MPEG-21 Book.* s.l. : Wiley, 2006.

22. *Visual annotation tool for multimedia content description.* **Smith JR, Lugeon B.** Boston, MA, USA : Proc. SPIE, 2000. Vol. 4210. DOI:10.1117/12.403831.

23. **Rehatschek H, Bailer W, Neuschmied H, Ober S, Bischof H.** *A Tool Supporting Annotation and Analysis of Videos.* Vienna : Reconfigurations. Interdisciplinary Perspectives on Religion in a Post-Secular Society, 2007. pp. 253-268.

24. **Troncy R, Van Ossenbruggen J, Pan JZ, Stamou G.** *Image Annotation on the Semantic Web.* s.l. : W3C Incubator Group, 2007.

25. *Semantic annotation of images and videos for multimedia analaysis.* **Bloehdorn S, Petridis K, Saathoff C, Simou N, Tzouvaras V, Avrithis Y, Handscuh S, Kompatsiaris I, Staab S, Strinzis MG.** Heraklion, Greece : s.n., 2005. Proc. 2nd European Semantic Web Conference.

26. **Dasiopoulou S, Tzouvaras V, Kompatsiaris I, Strinzis MG.** Capturing MPEG-7 Semantics. *Metadata and Semantics.* s.l. : Springer US, 2008, pp. 113-122.

27. *Image sequence analysis for emerging interactive multimedia services-the european cost 211 framework.* **Alatan A, Onural L, Wollborn M, Mech R, Tuncel E, Sikora T.** 7, 1998, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 8, pp. 802–813.

28. *Binary Partition Trees for Object Detection.* **Vilaplana V, Marques F, Salembier P.** IEEE Transactions on Image Processing, 2008, Vol. 17, pp. 2201-2216. 11.

29. *Image segmentation by texture using pyramid node linking.* **Rosenfeld A, Pietikainen M.** 12, Dec 1981, IEEE Trans. Systems, Machines and Cybernetics, Vols. SMC-11, pp. 822-825.

30. *Anti-extensive Connected Operators for Image and Sequence Processing.* **Salembier P, Oliveras A, Garrido L.** 4, 1998, IEEE Trans. on Image Processing, Vol. 7, pp. 555-570.

31. *The tree of shapes of an image.* **Ballester C, Caselles V, Monasse P.** 2003, ESAIM: COCV, Vol. 9, pp. 1-18.

32. *Fast computation of a contrast-invariant image representation.* **Monasse P, Guichard F.** 9, May 2000, IEEE Trans. on Image Processing, Vol. 5, pp. 860-872.

33. *Binary partition tree as an efficient representation for image processing, segmentation and information retrieval.* **Garrido L, Salembier P.** 2000, IEEE Trans. on Image Processing, pp. 561–576.

34. *Object-Based Evaluation of Hierarchical Region-Based Representations Based on Information Theory Statistical Measures.* **Calderero F, Marques F.** London : s.n., 2008. Proceedings CBMI 2008 (International Sixth International Workshop on Content-Based Multimedia Indexing).

35. **O'Connor NE, Adamek T.** An automatic stopping criterion for meaningful region-based image segmentation. [book auth.] Michela Spagnuolo, Bianca Falcidieno, Ebroul Izquierdo, Noel E. O'Connor, Evaggelos Spyrou José M. Martínez. *Semantic Multimedia.* Lecture Notes in Computer Science. Genoa, Italy : Springer Berlin / Heidelberg, 2007, Vol. 4816/2007, pp. 15-27.

7