An ontology-based framework for the automated analysis and interpretation of comic books' images

Clément Guérin, Christophe Rigaud, Karell Bertet, Arnaud Revel

An ontology-based framework for the automated analysis and interpretation of comic books' images

Clément Guérin, Christophe Rigaud, Karell Bertet and Arnaud Revel

clement.guerin@univ-lr.fr, christophe.rigaud@univ-lr.fr, karell.bertet@univ-lr.fr, arnaud.revel@univ-lr.fr
L3i – University of La Rochelle

Abstract

Since the beginning of the twenty-first century, the cultural industry has been through a massive and historical mutation induced by the rise of digital technologies. The comic books industry keeps looking for the right solution and has not yet produced anything as convincing as the music or movie have. A lot of energy has been spent to transfer printed material to digital supports so far. The specificities of those supports are not always exploited at the best of their capabilities, while they could potentially be used to create new reading conventions. In spite of the needs induced by the large amount of data created since the beginning of the comics history, content indexing has been left behind. It is indeed quite a challenge to index such a composition of textual and visual information. While a growing number of researchers are working on comic books' image analysis from a low-level point of view, only a few are tackling the issue of representing the content at a high semantic level. We propose in this article a framework to handle the content of a comic book, to support the automatic extraction of its visual components and to formalize the semantic of the domain's codes. We tested our framework over two applications: 1) the unsupervised content discovery of comic books' images, 2) its capabilities to handle complex layouts and to produce a respectful browsing experience to the digital comics reader.

Keywords: comic books, images, complex data, knowledge representation, ontologies, spatial reasoning

1. Introduction

Comic books are a unique art form which popularity grew increasingly from its beginning in the 19th century to the present day. Carrying an important legacy, the 9th art created its own narrative codes, mixing like no other art text and drawings to tell a story. Those very specific codes, evolving over decades, are intimately bounded to their physical support. That is probably one of the main reasons why the comic book's industry struggles to find a proper and efficient way on the digital market, like the movies, music and regular books industries did years ago. There are plenty of applications that would emerge from a smart digitization of comic books.

The enhancement of the small screen reading experience is an obvious example. Spotting panels, speech balloons and so on, is an essential feature to anyone who wants to provide a smooth and nice reading experience. Speech synthesis of written text lines in combination with the spotting of the talking characters would led to a very immersive experience that could help small children to apprehend comic books. Those pieces of text could be automatically translated in any required language if needed. The recognition of what is going on in a given panel, when and where the story takes place, what real or fictional characters are involved in it, could bring to the reader additional information that would help her of him to grasp the whole meaning of what she or he is reading. Some issues about old comic books preservation tend to rise as well, as the pieces produced at the early stages of the art's history are getting older every passing day. The world produced a large amount of heterogeneous comic books that now needs to be archived in some sort of database. Only information such as title, year and author are commonly used as intent for the queries, while an insightful content description would be very helpful to the archivists as well.

However, such use cases got stuck with a few scientific locks. How to extract the visual elements that compose a comic book, e.g. the panels, balloons, text and so on, is one of those locks. Then, this extracted content needs to be organized in a way that can show the original meaning of the author during the creation, in order to be enhanced, queried, adapted, with respect to that meaning. We started a wide study a few years ago to discuss these two issues. We described how we tackled the challenge of visual elements extraction in a previous publication [41]. The present article details the results of the latter, namely how we created a framework able to handle extracted comic books visual elements in a meaningful way.

We organized this article as follows. The first section presents several comic books formalisms from the literature. We present as well the state of the art on the use of formalized knowledge for image analysis, especially when it comes to discover what is the meaning of the extracted content. The second section introduces, through a complete and detailed presentation, our own proposition, including two ontologies, as a framework to handle the content of comic books' images. Then we discuss our conceptualization choices and test them over a public dataset. We also present some experimental results on two applications: a new panels ordering method and an iterative process to discover the content of a page. We discuss the results in the last section that concludes this article.

2. State of the art

This section introduces the different studies that came as a source of inspiration for our own work. We begin with presenting the major comic books' formalization endeavors from the last few years. We go through their pros and cons and highlight what is missing to meet the aim of our study.

Then we present a short survey of how semantics formalization showed up as an increasingly promising tool to enhance computer vision. We focused our analysis on how they can help narrowing the semantic gap and of what use they can be to supervise the image analysis process.

2.1. Multiple formalisms for comic books

If one wants to fully extract and analyze a comic books' content, it is necessary to know how that content is organized and the nature of the different relations that exist between its elements.

While a growing number of studies came up lately in linguistics [2] and psychology [12] communities, the computer science community, as we know of, conducted a very few works on the comic books language conceptualization. Besides the *Periodical Comics* web-schema proposal [49] and the *Grand Comics Database* platform [25], both focusing on the publishing metadata dimension, we could only find a very small amount of endeavors on the content formalization. Not all of them have been the subject of an academic publication.

In 2001, *ComicsML*, an XML formalism to describe the content of a comic books plate has been proposed [29]. The authors developed it in the perspective of becoming a standard for web-comics publishing which was barely emerging then. Indeed, web-comics, back then and still now, are often published as raw images, straight to the author's web page, with a varying frequency. Through its wish of standardization, the real ambition of ComicsML was to generate new usages, especially with the development of reading tools exploiting the language specificities. It provides means to describe a series of web-comics published online, from the bibliographical information to the kind of balloons drawn in the panels and the font used for the text lines. Its syntax is based on XML, it uses tags and implies a hierarchical structure of the described elements. The root node, `comic`, describes the series that the web-comics belongs to. Its children, `strip`, are the several issues of the series, i.e. the published images. These tags embed the bibliographical metadata about the author and the publication date. The `panel` tags, children of `strip`, describes the panels' content. It includes the kinds of balloons (thought, dialogue, etc.), the text typography (bold, italic, etc.) and the involved characters. The authors dropped the support of the language after version 0.3. A similar approach, but specifically adapted for manga, can be found in [35].

A second initiative, *CBML* (Comic Books Markup Language), came out a few years later with the work of Walsh [50]. Also based on an XML syntax, its philosophy is quite similar to *ComicsML* without being exclusively used for web-comics annotation. The main difference with *ComicsML* lies in the fact that *CBML* is based on the *"Text Encoding Initiative P5: Guidelines for Electronic Text Encoding and Interchange"* [46]. The *TEI* is a framework, proposed by the homonym consortium, to encode textual documents and illustrated documents especially. Its purpose is to make information retrieval in large encoded datasets easier. *CBML* extends the *TEI* vocabulary with some comic book's notions (panels, balloons, etc.), while reusing, as much as possible, the existing encoding.

Pastierovic proposed a third formalism, called *Advanced Comic Book Format* or *ACBF* [37]. It focuses on the encoding of digital comic books. Also XML-based, it aims to enhance the reading experience on computers. Its authors define it as the comic books archive file formats, such as CBR and CBZ, evolution, as it adds metadata on several elements. It includes the pages and

panels succession concept, as well as the distinction between the several kinds of balloons and the language in which the text is written. The text encoding is independent from the image's so it makes it easier to translate an album. Unlike the previous formalisms, *ACBF* also encodes the spatial location of panels and text areas in the image. It allows to catch mouse events during the reading and execute the right actions. Those three propositions can have useful applications in indexing, browsing and reading comic books. They provide powerful tools to encode a comics' content at different levels, depending on what one wants to use it for. However, it is quite clear that their authors thought of a manual annotation while they were designing them. Besides, their XML origin makes the inference of new information on the annotated elements difficult.

When we started the eBDtheque project[1] in 2011, there were no ontology of the comic books domain we could use to build our study on. However, an ontology sharing some common elements with our very own (detailed in Section 3.2) was published online [42]. The author focused on the design of an ontology to describe the content of a comic books rather than bibliographical information. The hierarchy of concepts includes the main components of a plate (represented by the concept of *Card*), namely panels (*Panel*), balloons (*Balloon*) and characters (*Character).* Five concepts specialize the balloon concept to represent different use cases, namely speech balloons, thoughts, exclamations, whispering and broadcasting (e.g. when a TV or a radio makes some sound). The *in-reply-to* object property represents the question and answer between two balloons, as well as the *linkedBalloon* property models the connection between two balloons. The lines of text are not strictly represented by a dedicated concept. When a balloon contains some text, an attribute on this balloon conceptualizes it. The *xxx-content* data property formalizes it, where *xxx* stands for the type of balloons (e.g. *exclamation-content*). Different branches of the concepts hierarchy, namely the *Effect Caption* and *Narrative Caption* concepts, respectively formalize onomatopoeia and narrative frames. They are subsumed by the more general *Caption* concept. The text content itself is once again, represented as a literal attribute of these concepts. The *Character* concept, imported from the OntoMedia ontology [23], formalizes the character notion. The ontology

[1] l3i.univ-larochelle.fr/eBDtheque

associates characters to balloons using the *source* and *directed-at* properties.

It links these elements to each other with constrained properties. Balloons, frames and onomatopoeia have to be associated with a panel, while a panel has to be in a page. The *panelOrder* and *balloonOrder* attributes represent the panels order in the page and the balloons order in a panel. The conceptualization contains several kinds of annotations as well. It is possible to specify the real location of the story with a concept from the RDF geographical vocabulary of Brickley [7]. One can add other pieces of information using the *Note* and *Link* concepts. The first one describes a panel or a balloon. The second one references a resource from the web with a URL. Even though it has not been developed in the perspective of being used in an image processing context, this ontology includes some conception choices well suited for that purpose. The reader will find some ideas shared by our ontology in Section 3.2.

2.2. Knowledge representation for image analysis

The use of knowledge representation to enhance the image analysis task came out a long time ago [33]. A survey presented in [14], regretted then the application-specific nature of the approaches and the lack of standardization. The field evolved a great deal since then, especially with the emergence of ontologies, providing a knowledge representation standard for the manual annotation, the interpretation and the analysis of images. The panorama described in [47] provides a nice overview of the different approaches used to disambiguate the task of images manual annotation. Based on thesaurus and taxonomies, different tools such as WordNet [32] and the SKOS vocabulary [31] are used to annotate heterogeneous sets of images [51, 28].

We will focus on what knowledge representation brought to the analysis and interpretation of images. While a very few works applied this kind of approach on comic books images, their generic nature gives us some perspectives to handle this task on comics.

2.2.1. Narrowing the semantic gap

The semantic gap challenge is often associated with the image analysis task though it is related to any data processing domain. The authors of Smeulders et al. [44] defined it as the lack of significance of what a computer can automatically extract from data (visual data in our case) compared to what this data means for a specific user in a given context. The atomic

components of an image are its pixels, carrying a color information. Extracting concrete objects, *a fortiori* abstract concepts, is not an easy job. Even more, the interpretation of an image's content is not unique nor absolute. It depends on the knowledge of the person who interprets it. Anyone can recognize a butterfly in a picture but only a few people can name the species of that butterfly.

The Content-Based Image Retrieval domain (*CBIR*) [18], i.e. image retrieval based on a given example, got stuck in the semantic gap issue. The pixel features alone are not enough to judge of the semantic proximity of two images. Some *a priori* knowledge, seems to be essential to bring some context into the task and make the image interpretation feasible [1]. One of the main identified means to narrow the semantic gap is the use of ontologies [27], along with machine learning, relevance feedback, the use of annotations from the web and semantic visual templates [8].

The community conducted many works on the formal representation of image's visual features. The authors of Mezaris et al. [30] proposed an image retrieval system built on a middle ontology (*Object Ontology*) that qualifies the visual properties of areas in the image with simple, human-friendly terms. The features of these areas (pixels value, position, dimensions, shape) are extracted and automatically matched with predefined labels. The objects to be found in the images are manually described using the concepts of the middle ontology. The authors used a relevance feedback mechanism to interact with the user and produce new instances from the annotated samples.

The work published in [24] focused on classifying art images, based on their nature (photography, painting, drawings, computer generated) and on the feelings they convey (joy, sadness, anger, etc.). The authors used an ontology to link the results of a pixel analysis (high gradients, saturation, luminance, etc.) to high-level semantic concepts. A user is then invited to query the system with a keyword to retrieve visually similar images.

The authors of Petridis et al. [38] proposed a *Visual Descriptor Ontology* allowing to describe an actual object (a car, a tree, the sky, etc.) to a prototype of visual features defined in MPEG-7. Besides the image's areas annotation, the main purpose of the system is to answer complex queries, such as "the moments in a video where a tennis ball reaches a white line". The generic nature of the approach deserves to be noticed, defining new prototypes is all it takes to retrieve new objects.

The visual aspect of an object (color, shape, texture, etc.), is not the

only piece of information one can rely on to find it on an image. The elements of an image are often bound to some spatial constraints related to the document layout or, naturally, by the laws of physics. Let say we want to retrieve the signature in a digitized letter, it is probably better to look for it at the bottom of the document. In a similar way, the fact that the sky is usually on top of the sea eases the semantic annotation of two big blue areas automatically extracted from a beach picture. One can find examples of this approach in the *Object Ontology* of Mezaris et al. [30], where the "position" concept represents the position of an area in the image. Many real application studies included an *a priori* spatial information in their image analysis process [34, 22, 13, 10].

The ontology used in [13], to analyze images of dropped initials, formalizes the conditions for an extracted region to be identified as a letter. A SWRL rule translates the fact that a letter is the biggest area, with a limited amount of holes in its shape, centered in the image into a classification inference operation.

The study published in [22] introduces an ontology that conceptualizes different kind of spatial relations. Topological (overlapping, disjunction, inclusion, etc.), directional (on the left, below, etc.) and distance (close, far, etc.). The concepts are generic, therefore independent of any application context. The semantic of the concepts is expressed as fuzzy sets, the veracity of an atom not being expressed as a Boolean but as a value between 0 and 1. Fuzzy logic principles introduced into knowledge representation makes possible to model the doubt and vagueness that comes with a lot of real world applications. Another example of ontologies based on fuzzy logic, exclusively based on OWL-2 expressiveness, is detailed in [6]. However, the fuzzy semantic is context dependent as some ideas, such as *close to*, do not have the same meaning for a geographer analyzing a map or a physician looking at an x-ray.

More recently [4], ontologies contributed to the annotation of satellite images, using object based image analysis methods [5], especially on the problem of consensual labeling.

The work presented in [9] addresses the issue of recognizing the action depicted in an image. They consider the regions segmentation step as trivial. They compute, from the visual features of the corresponding area and from how the objects are placed relatively to each other, the probability of an

object to belong to a given class, from a controlled set of classes.

These many works show both a solid interest for the representation of spatial knowledge and the need for a standard. The Open Geospatial Consortium recently accepted the GeoSPARQL format [3] as a standard to represent and query geospatial data. It conceptualizes the shape of some spatial entities and their topological spatial relationships. Relations of direction and distance are not directly conceptualized but can theoretically be inferred from the coordinates of other elements. There are only a few actual implementations of the GeoSPARQL spatial reasoning standard for the time being, namely Oracle systems, Parliament and Strabon triple-stores.

The *Pellet Spatial* tool [45] extended the Pellet inference engine with some spatial reasoning capabilities. These capabilities are limited to the RCC-8 topological relations though [11]. It is able to check the consistency of a knowledge base and infer new topological relations from already existing annotations. It cannot infer relations from coordinates. Researchers recently presented in [10] a new extension for Pellet Spatial to make it able to infer cone-shaped directional relations. The work in [26] optimized RCC-8 relations composition, using concept lattices to reduce the potential combinatorial complexity.

2.2.2. Image analysis supervision

The different image interpretation systems presented in subsection 2.2.1 are based upon the fact that the preliminary regions extraction went smoothly. However, this step is far from being trivial and some studies determined how we can use ontologies before or during the image processing. A team used an ontology before the step of image processing to set up the system with operations adequately chosen for a given purpose. Some work developed this approach [40], where a domain expert, using a couple of ontologies, can state 1) the goals the image processing systems have to reach, 2) the physical (how it was generated), visual (what is its visual content) and semantic (what does it represent) features of the image.

A second example focused on a comic books automatic editing system, turning digitized pages to a mobile device fitted format [20]. Their system analyzes the image to extract panels and pieces of text to define the area carrying the greatest piece of information. The authors presents an ontology which can deduce what would be the best operations to apply on the image (resizing, cropping, etc.), based on the previously extracted knowledge.

The work described in [16, 17] presents a method of iterative labeling of area from satellite images. The labeling is progressively corrected based on the consistency between the annotations and a conceptualized domain knowledge. This is an approach we explored as well and that we detail in Section 4.2.

3. Description of our framework

This section introduces the ontology we propose, made of two smaller ontologies. We start by presenting a first ontology that formalizes the concepts related to image analysis, the different types of data, input, output and evaluations. Then, we explain the conceptualization process that led to create a second ontology on comic books domain knowledge. Finally, we show how we can use these ontologies together. We test our conceptualization choices over the eBDtheque ground truth dataset in the next section.

3.1. Image ontology

3.1.1. Image and regions of interest

The purpose of the first ontology was to formalize the very basic concepts of the image processing domain, from the extraction to the evaluation that is lacking in the state of the art. We put a special effort in making it as generic as possible, excluding any concepts that would confine it in a specific application domain. We intended it to be usable in any application based on image processing (e.g. document analysis, computer vision, etc.). It gathers the data produced by image analysis algorithms in a predefined structure. Then we use this structure to formalize the semantic of each of those elements. The data produced by those kinds of algorithms is often of a spatial nature. It can be regions of different shapes, lines or points taken from an image. We refer to these notions as *regions of interest*, or *ROI* in the remaining of the article. Those ROIs are defined by their coordinates in the image. Based on this macroscopic view of what is the image processing domain, we defined the first two concepts of our ontology: *Image* and *ROI*.

The *Image* concept stands for the notion of image as a digital entity, the raw material of the image processing task. It is the top concept from which the remaining of our ontology is derived. An image processing algorithm takes the atomic elements of an image, its pixels, as input. Those can either be analyzed one by one or be grouped in clusters but, in the end, the value

of each pixel matters in some way during the process and can influence the algorithm's output. Even if two images look the same to the human eye, the value of their pixels can be very different from each other. They can be of different definitions, in other words, be made of a different amount of pixels. Their resolution (the number of dots per inch) or the way they encode and compress the information, with or without loss, may vary as well. We had to add this information into the *Image* concept of our ontology as it can help to understand the output of an algorithm. The resolution is formalized as a data property named *hasResolution*. The definition is split in two attributes, *hasWidth* and *hasHeight*. Those three properties take integers as data type. One can use them to compute the original size of the digitized document, so they are sufficient to retrieve the initial conditions of the digitization process. The *hasFormat* and *isLossLess* data properties represent the storage format and the possible image compression. A string, such as "PNG" or "TIFF", specifies the format.

In the image processing domain, researchers manipulate images as two-dimensional matrices (sometimes three if the color channel is decomposed along its three components). We adopt in the remaining of this paper the global convention that puts the origin of the image in its top left corner. The *ROI* concept is a meronym of *Image*, i.e. a region of interest is a part of an image. This concept formalizes the notion of region of interest as the image analyst understands it. It is a limited area, included in the image, made of adjacent pixels with some specific visual features. Those features may correspond to some criteria sought for being representative of some visual objects. The coordinates of the regions are stored as character strings and expressed with the "Well-Known Text" format, shortened as "WKT". This format is an Open Geospatial Consortium endeavor [36], which is tightly working with ISO/TC 211, to make it become an international ISO standard. There are several reasons behind the choice of this format over others like SVG or GeoJSON. Firstly, it provides, as many other languages, a convenient way to express a large range of different geometrical shapes, including polygons and lines. Beyond those, it can be used to define polygons with holes and multi-polygons, i.e. disjoint polygons. Thus, it makes it possible to define a region of interest made of several disconnected areas as a single entity. Secondly, it comes with the advantage of being both explicit, a human can read and understand the nature of an object described with it, and concise, which can be of importance when one deals with a great volume of described data. A

keyword, placed at the beginning of the character string, describes the nature of the object, e.g. `POINT`, `LINESTRING` or `POLYGON`. Last but not least, WKT is open, well documented and supported by many API in a lot of different programming languages. It is also supported by GeoSPARQL.

The *hasWKT* attribute enriches the *ROI* concept. It takes string as values and describes both the type (polygon, line, dot, etc.) and the coordinates of ROI instances. Additional *hasArea*, *hasX* and *hasY* attributes can be used to specify the area and the center position of a region of interest.

The object property *hasROI* was created to link the *Image* and *ROI* concepts. The former being its domain, the latter its range. It represents the fact that an image can contain some regions of interest. However, a ROI is part of one and only one image. It is indeed created from the processing of a given image, as an area visually relevant for a specific application. Running the same algorithm on other images will produce different ROIs. Therefore, the *hasROI* relation is noted as being inverse functional to integrate that constraint. Figure 1 illustrates what we described to this point.

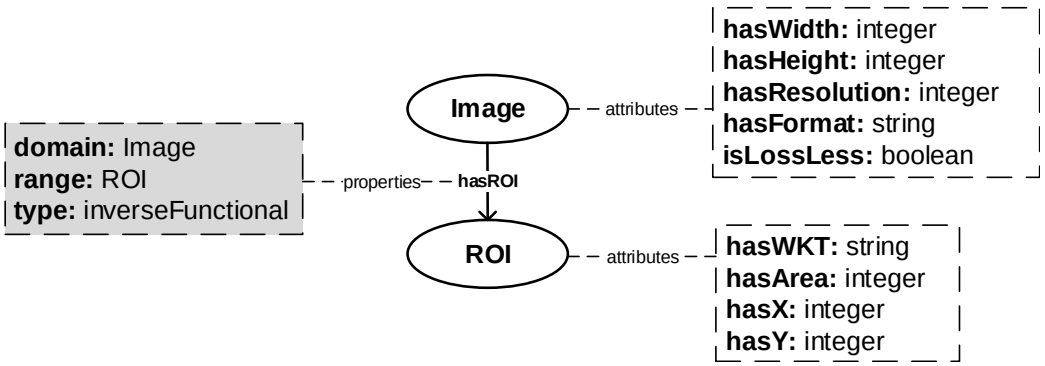

Figure 1: *Image* and *ROI* concepts from the Image Ontology, their properties and attributes.

3.1.2. Extractors

Image processing is a complex domain, especially when it comes to patterns recognition. The perfect algorithm that we can use to name any shape, any object in any image, whether it represents a natural scene, a digital document of anything else, does not exist. Depending on the nature of what the

analyst wants to retrieve in an image, he has to develop specific techniques that take into account the visual specificities of the objects. The focus will sometimes be put on the texture, sometimes on the color for instance. It all comes down to what the analyst is looking for, the robe of a zebra or the blue sky above the yellow beach. The shape, the size and the relative and absolute positions of these elements are some of the tools that can help him build the right algorithm for the right task.

One algorithm for one kind of visual objects. Nonetheless, it is important that the algorithm is robust and generic enough to run efficiently on different images. Therefore, the analyst has to develop as many algorithms as there is different kind of objects to detect. Those must produce accurate results on the majority of documents though. In the context of the analysis of administrative documents, such as bills or pay stubs, it would be interesting to detect logos, signatures or alphanumeric characters. If we come back to comic books, extracting panels, balloons, lines of text and characters (as a protagonist of the story) implies the same amount of algorithms. Each of them exploiting the specific features of the objects they are made to retrieve.

Panels take most of the image area. They are usually bordered by a black stroke and separated from each other with a white gutter. Their shape, often rectangular, can also be used as a discriminative feature for straight lines detection. The balloons contain text, traditionally written in a dark color on a light background. They are also delimited with a black contour but are also smaller than panels. Lines of text are made of small visual elements, the letters, that highly contrast with the background, so one can read them. Letters are aligned in a roughly regular way. Optical Character Recognition techniques can also help to filter false positives. Characters can be of any shape, realistic or not, caricature, human-like or not. They often show some level of regularity in their visual representation. They can be drawn in different sizes, positions and orientations.

A region of interest produced by an image processing algorithm is a *proposition* of an area that could contain an instance of a sought kind of object. It may be true; it may be not. It all depends on the image characteristics, the visual complexity of the objects and, of course, the cleverness of the algorithm itself. It can produce some errors, that is why the extracted regions of interest remain *propositions* that need to be validated and cannot be taken

as the truth.

We call those kinds of segmentation systems *extractors* later in this paper. We included this notion to our ontology as the *Extractor* concept. Bound to *ROI*, it can express the tight link between an extractor and the regions it produces by its application on a specific image. The property *hasExtractor* between a ROI and an extractor is functional, as a region can be produced by one and only one extractor. The kind of visual elements that one designed the extractor to retrieve can be indicated with the attribute *hasROIType*. The possible values having to be defined accordingly to the application domain. Figure 2 illustrates the described additions.

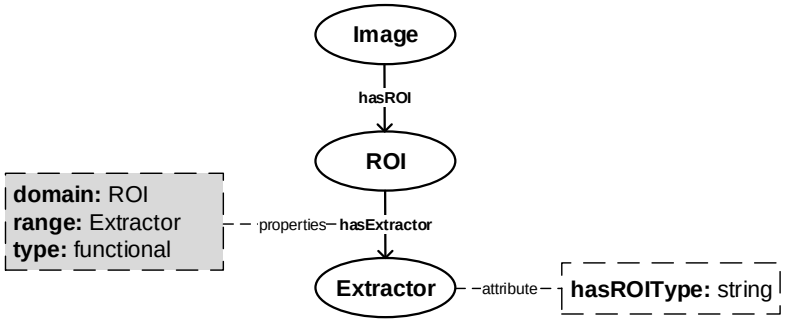

Figure 2: The ontology presented in Figure 1 enriched with the *Extractor* concept.

3.1.3. Ground truth

The process of objects spotting or recognition cannot go without evaluating its results. As we mentioned it earlier, those results are only *propositions*. They are the consequences of an image's visual features that went through the instructions of an extraction algorithm. The percentages of *false positives* (the propositions that do not match any element to extract) and *false negatives* (missed elements) among the results may vary depending on the domain, the image or the algorithm but are quite unlikely to be equal to zero. They are classically calculated through the measures of *precision* and *recall*. In order to be used in a real application context, the extraction algorithms have to be validated using these measures. When computed over a representative corpus of the application domain, they guarantee the robustness of the methods and their adequacy to the application domain.

In order to compute these measures, some information about the content of the corpus must be known. This information is the reference used to confirm the methods and cannot be questioned after one defined it. It is absolute, it defines what is true and all that is true in a closed-world assumption logic where everything that is not explicitly expressed does not exist. This piece of information is called a *ground truth*. A ground truth establishes the real composition of a document with a set of annotations that describes it completely and perfectly for a given application context. Indeed, the composition and the level of details of the ground truth depend on what one is trying to detect, extract or recognize in a set of documents. If the task is to identify the nature of a document (photography, written article, comic book page, etc.), a simple keyword would be enough as a ground truth. But if we want to analyze the very content of a document, how it is organized internally, we need to make a deeper annotation. For the analysis of a comic book page, it implies to spatially define the position of each element we want to extract, such as panels and characters. Semantic annotations can also be added to these elements. As it is meant to be used as a reference, a ground truth cannot be generated automatically. A human, who can be considered as an expert in the application domain, has to build it from scratch. Depending on the level of details, the building of such a dataset can become time-consuming. The expert's knowledge, formalized through his annotations, stands for the gold standard, the limit the algorithm needs to reach to pretend having a perfect behavior. Sometimes, the subjectivity of interpretation of the expert can influence the annotation. It is the case for the comic books domain as well for any other art form domain. That is why we do not speak of *the* ground truth but *a* ground truth for a corpus and an application. How to evaluate the quality of some crowd-sourced annotation is a challenge of its own. The question is being studied on the very case of comic books annotation [48].

In the context of an image processing application, a ground truth is made of spatial annotations, each one labeled with the class of objects they belong to. If we put aside how these spatial regions are produced, a ground truth seems to fit the definition of an extractor. The concept of ground truth is consequently formalized in our ontology as a hyponymy between the *Extractor* concept and two sub-concepts, *GroundTruth* and *ExtractorAuto*. The first one naturally stands for the ground truth notion while the second one represents the idea of extraction algorithm. These two concepts are funda-

mentally disjoint as their functions are very different. The regions of interest produced by a *ExtractorAuto* instance are unsupervised propositions that are meant to be used, in time, in a complete and functional productive system. We can question, use, modify or remove them. The ROIs provided by a ground truth are true by definition. Their only role is to validate at some point the propositions of an algorithm and, by that, the algorithm itself. We conceptualized the difference between those ROIs through by extending the *ROI* concept in two concepts: *ROIGT* (for Ground Truth) and *ROIAuto*. Those concepts, like the two *Extractor*'s sub-concepts are disjoint, i.e. a region is either a proposition or a reference. Similarly, an extractor cannot be automatic and be used as a ground truth at the same time. We defined the *ROIGT* concept, respectively *ROIAuto*, as being equivalent to the intersection of the individuals of *ROI* and the set of individuals having an instance of *GroundTruth*, respectively *ExtractorAuto*, as an extractor. Equations 1 and 2 show these equivalence relations, written in Manchester Syntax [21]. Figure 3 shows the latest additions to the model.

$$\text{ROIGT} \equiv \text{ROI } \textbf{and } \text{hasExtractor } \textbf{some } \text{GroundTruth} \quad (1)$$

$$\text{ROIAuto} \equiv \text{ROI } \textbf{and } \text{hasExtractor } \textbf{some } \text{ExtractorAuto} \quad (2)$$

3.1.4. Evaluation

Now that we conceptualized the idea of reference, we have to define the structure that evaluates the propositions with respect to this reference. In the image processing domain, it is quite bold to expect the algorithm to produce regions that we can identify and directly compare to their reference in the ground truth. The amount of extracted regions does not necessarily match the number of regions to extract, as the algorithm has to work without any *a priori* knowledge on the instances content. The only pieces of reliable information are the nature of the object in the segmented region (that is the purpose of the algorithm) and the position of this region in the image. Therefore, it is necessary to link each proposition to the references from the ground truth sharing a non-void spatial intersection with it. The closer the cardinality of their intersection (the amount of common pixels) is to the cardinality of their spatial union, the better is the proposition. We

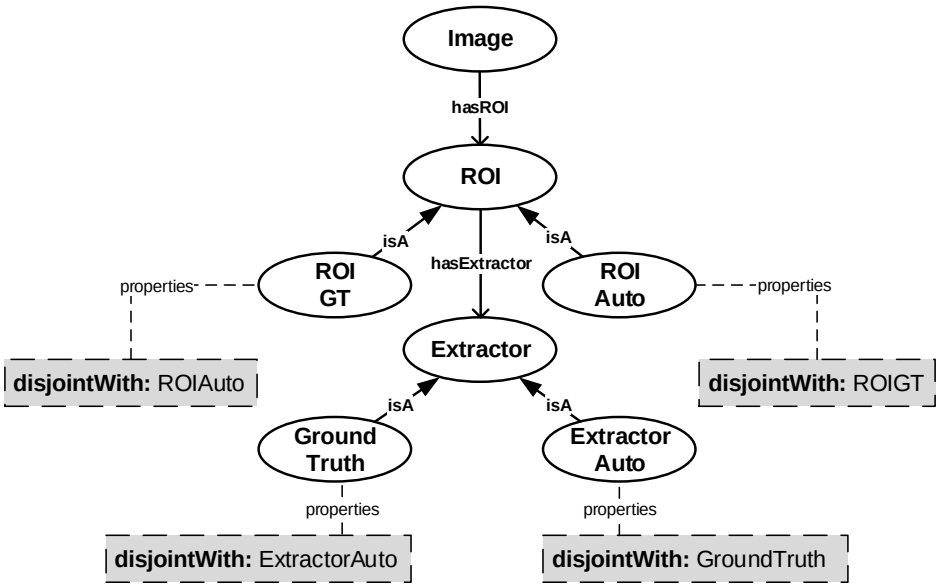

Figure 3: The ontology presented in Figure 2 extended with the notion of ground truth.

can estimate the proposition's quality with the precision and recall measures, the elements to detect being the pixels. We can combine those measures as a single F-measure to produce a general quality criterion. A proposition is considered validated (respectively rejected) if its F-measure value is above (respectively below) a predefined threshold s.

We formalized the link between a proposition and a reference through the concept of *Evaluation*. We enriched it with three decimal attributes: *hasRecall*, *hasPrecision* and *hasFScore*. The two relations *hasReference* and *hasCandidate* link an *Evaluation* instance to the elements of $ROIGT$ and $ROIAuto$. These relations are functional, each evaluation involving one reference and one proposition. One individual instance of *Evaluation* can be created for each spatial intersection between a reference and a proposition. The spatial relations used for this task are the disjunction (*disjoint*), the inclusion (*contains*) and the intersection (*overlaps*). The whole set of RCC-8 relations is compatible via the import of the spatial ontologies defined for

Pellet Spatial[2] and GeoSPARQL[3]. The equivalence between the *ROI* and *SpatialObject* concepts, formalized in GeoSPARQL, allowing the use of its own spatial relations. Recall, precision and F-measure are computed and inserted as attributes of the *Evaluation* instances.

We introduced the *Validation* and *Error* concepts as additional classifications of *Evaluation* instances. We defined them as equivalent to the individuals in *Evaluation* with a higher, respectively lower, *hasFScore* value than a predefined threshold s (see Equation 3 and 4). One can tune this rule based on the specificities of the application to use the recall, the precision, or even other metrics such as the one introduced by VOC [15]. Those concepts are disjoint, an evaluation being either valid or invalid.

$$\text{Validation} \equiv \text{Evaluation } \textbf{and } \text{hasFScore } \textbf{some } \text{double}[\geq s] \tag{3}$$

$$\text{Error} \equiv \text{Evaluation } \textbf{and } \text{hasFScore } \textbf{some } \text{double}[< s] \tag{4}$$

The choice to introduce a specific concept to conceptualize the evaluation and not a simple attribute to *ROI* is justified when you deal with multiple evaluations. Especially when a proposition has several intersections with different references or when more than one ground truth dataset is loaded into the A-Box. In that case, the use of attributes or properties is not enough to model the evaluation to a reference any more. Figure 4 illustrates the latest additions to the ontology.

3.2. Comics ontology

The ontology presented in Subsection 3.1 formalizes the basic concepts of the image processing domain and provides a way to organize and use input and output data in a formal structure. We present in this subsection our conceptualization of the comic books domain with its formal ontological implementation. We thought the following conceptualization with the idea of using it in an image processing application while making it as generic as possible.

3.2.1. Comics and plates

As seen before, a comic book is defined as a series of visual elements carrying a message. Those elements are spatially organized on a plane which

[2]URI: http://clarkparsia.com/pellet/spatial#
[3]URI: http://www.opengis.net/ont/geosparql#

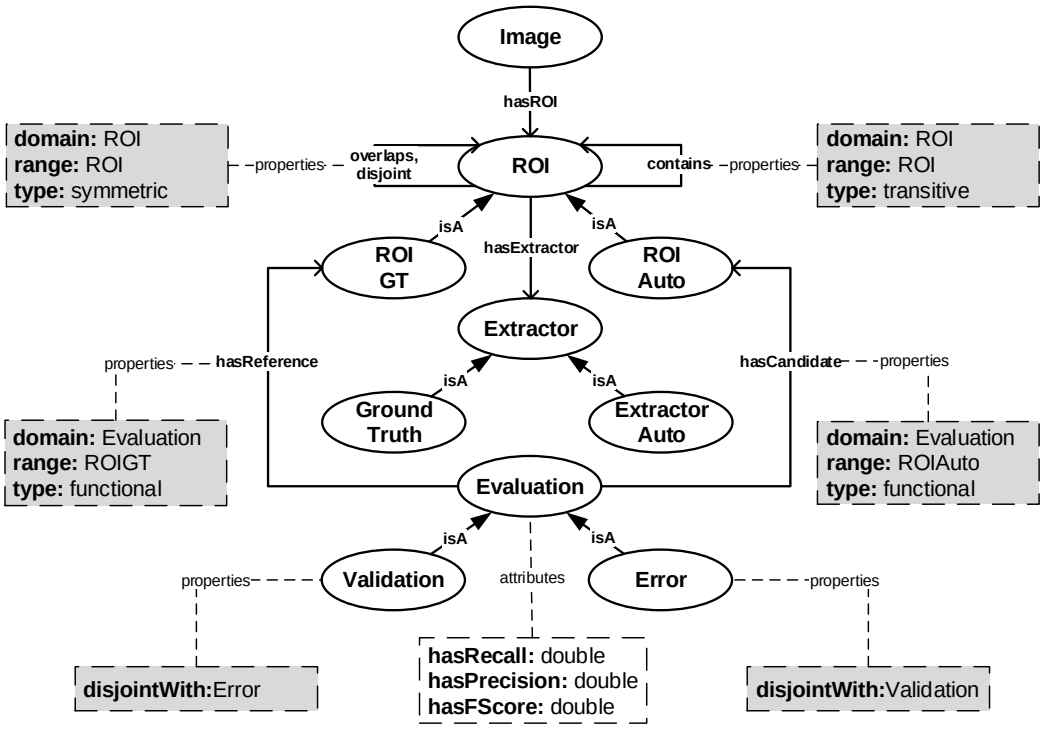

Figure 4: The ontology presented in Figure 3 enriched with the *Evaluation* concept and its two specializations, *Validation* and *Error*.

we will refer to as a *plate*. In their classic form, Franco-Belgian, Japanese and American comic books tell a story through a succession of plates drawn on a certain amount of pages. A plate can be drawn on one or two pages. Those pages are gathered in albums, a series contains several albums. Web-comics also have the notion of plate, which are of the form of a single digital image.

Thus, the two first concepts introduced to our ontology are *Comic* and *Plate*. We linked these concepts to each other with the *hasPlate* relation from *Comic* to *Plate*. It is inverse functional as a plate comes from one and only one comics. Bibliographical information about the instances of *Comic* is handled with a set of attributes. The title of the comics, its series, its authors, publication date or ISBN number are indicated thanks to the corresponding data properties. The reading orientation (from left to right or right to left) can also be indicated with the *right2Left* Boolean attribute.

We gave to the *Plate* concept two attributes: *hasNumber* and *onDou-*

blePage. They respectively represent the number of the plate in the comics and the fact that it is drawn, or not, on several pages. The *hasNextPlate* relation formalizes the succession of plates in a comic book. It is automatically instantiated, based on the *hasNumber* attribute, through Equation 5. Figure 5 shows these first notions.

$$\begin{aligned} &\text{Comic(?c), Plate(?p1), Plate(?p2), hasPlate(?c, ?p1), hasPlate(?c, ?p2),} \\ &\quad \text{hasNumber(?p1, ?n1), hasNumber(?p2, ?n2), subtract(1, ?n2, ?n1)} \\ &\qquad\qquad \rightarrow \text{hasNextPlate(?p1, ?p2)} \end{aligned} \tag{5}$$

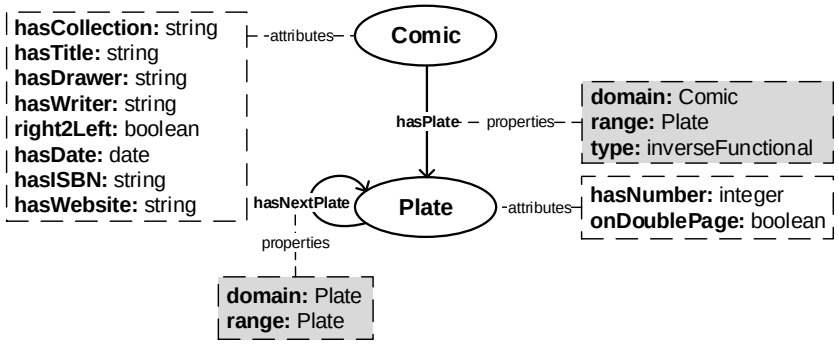

Figure 5: *Comic* and *Plate* concepts from the Comics Ontology, their properties and attributes.

3.2.2. The plate's content

As we know, a comic books' plate is the context where the panels, carrying the story, are drawn. Their relative position with each other and, sometimes, the reader sensibility defines the order of these panels in the reading flow. A closer look is given to the reading order matter in a Section 4.3. The panel's content is made of drawings representing essentially characters and balloons. We choose to focus on these two types of content because, 1) we developed algorithms to extract them in a parallel work [41], 2) they carry a great deal of the story. Considering other kinds of drawn elements (e.g. random objects, trees, etc.) is outside of the scope of this work, although it

will be easy to include them in the future. A similar approach to the one developed in [51], based on WordNet extraction would also be interesting. As for balloons, they mainly contain lines of text, representing the words of the characters and the story narration.

We gathered the panels under the *Panel* concept. It has an attribute, named *hasRank*, formalizing the rank of each instance of the concept in the reading sequence of the plate it belongs to. The *hasNextPanel* relation links each panel to its successor in the plate. Its instances are automatically generated through the SWRL rule expressed in Equation 6, with *Plate* and *Panel* as *Container* and *Element*. This relation is not functional because it is possible to have more than one possible reading path in a plate.

$$\begin{aligned} &\text{Container(?c), Extractor(?ex), Element(?e1), Element(?e2),} \\ &\qquad \text{hasElement(?c, ?e1), hasElement(?c, ?e2),} \\ &\qquad \text{hasExtractor(?e1, ?ex), hasExtractor(?e2, ?ex),} \\ &\text{hasRank(?e1, ?r1), hasRank(?e2, ?r2), subtract(1, ?r2, ?r1)} \\ &\qquad \rightarrow \text{hasNextElement(?e1, ?e2)} \end{aligned} \tag{6}$$

The *Balloon* concept represents the balloons, whether they are speech balloons, thoughts balloons or narrative frames (also called captions). Examples of such balloons can be found in Figures 6 and 7. Balloons are read in an order given by their position within their associated panel. A balloon may be associated to a panel even if it is not formally contained in it. It can be drawn outside or across multiple panels (see Figure 6). A panel is nonetheless a representation of a time limited moment of the story and its associated balloons are part of it. We defined the order of the balloons associated to a panel with the *hasRank* attribute and the *hasNextBalloon* property links them to each other (see Equation 6, with *Panel* and *Balloon* as *Container* and *Element*). Speech balloons present a small break on the regularity of their border, that we called a *tail*. It has an arrow-shape and points towards the character which "says" the words contained in the balloon. We introduced the *Tail* concept to represent that particularity. The *hasDirection* attribute indicates its direction. The *hasTail* relation links a balloon to a tail.

The *TextLine* concept represents lines of text. They are gathered inside balloons and are naturally read from top to bottom. Just like we did

Figure 6: Example of speech and narrative balloons. Some speech balloons are partially drawn outside their reference panel. Credits: T. McCall - Robin Hood And Company 31 - John Gordon Baker.

Figure 7: Example of thought balloon. Credits: M. Fox - Chilling Tales 17 Geo. - Youthful Magazines.

for panels and balloons, we used the *hasRank* attribute to set their rank in the balloon. The *hasNextTextLine* relation being defined by Equation 6, with *Balloon* and *TextLine* as *Container* and *Element*.. The actual textual content represented by the text line is transcripted through the *hasText* attribute.

Last, the *Character* concept formalizes the idea of characters. *Panel*, *Balloon*, *Tail*, *TextLine* and *Character* concepts are disjoint. An individual cannot belong to more than one of these class at a time. Figure 8 illustrates these additions to our ontology.

We introduced several different relations to our ontology to formalize the existing links between the different types of elements. A panel being related to a plate, the *hasPanel* relation links an instance of *Plate* to an instance of *Panel*. The domain and the range of this property make sure that only a plate and a panel can be its subject and object. It is also inverse functional, a panel being related to one and only one plate. We defined the *hasBalloon*, *hasCharacter* and *hasTail* properties similarly. The first two represent the belonging link between a panel and a balloon or a character. The last one links a balloon to an instance of *Tail*. The *hasTextLine* relation

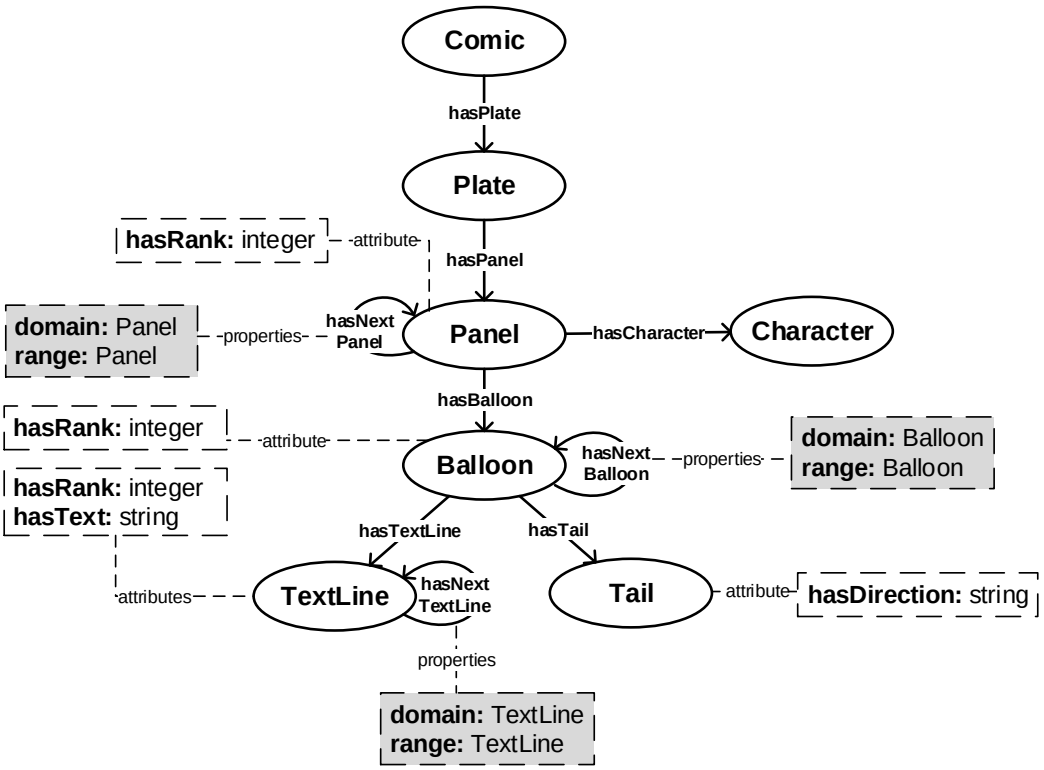

Figure 8: The ontology presented in Figure 5 extended with the *Panel*, *Balloon*, *TextLine*, *Tail* and *Character* concepts.

formalizes the belonging of a text line to a balloon. We made the choice of a very constrained relation, purposely rejecting a complete set of text outside balloons (onomatopoeia for instance). Subsection 4.1 presents the evaluation of the relevance of this constraint.

One can feel some sort of transitivity through the relations between a plate's elements. Text lines belong to balloons and balloons to panels. So does characters. Those panels are part of plates that, all together, make comics albums. The formalization of this transitivity offers a great advantage when it comes to information retrieval. A query on the comics presenting some characters saying some given words would be free of the notion of plates, panels and balloons. The *hasContent* transitive property formalizes this idea. It is a super property of *hasPlate*, *hasPanel*, *hasBalloon*, *hasTextLine* and *hasCharacter*. Figure 9 illustrates that point.

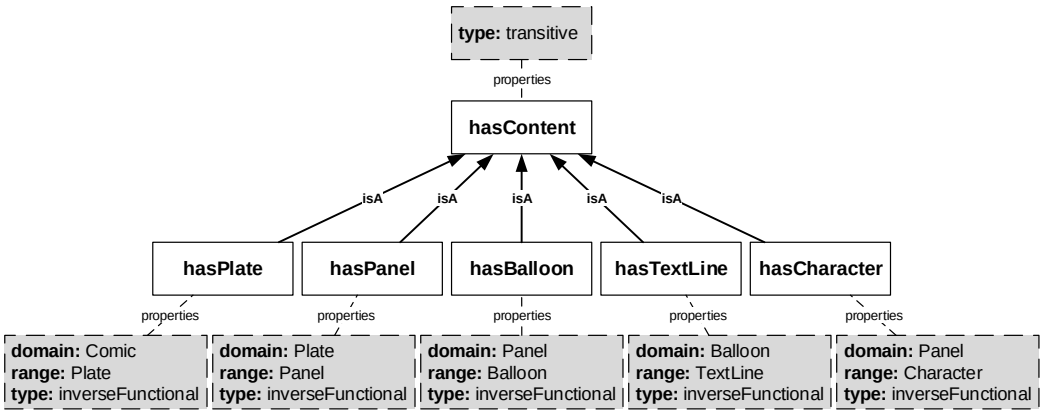

Figure 9: The transitivity of the object properties of our ontology.

3.2.3. Concepts specialization

We can refine the semantic of some of the presented concepts. Balloons especially can be classified into two sub-concepts. On one side, we can find the balloons related to some characters, whether their words are thought or spoken. On the other side, we found the narration frame, told by what would be the narrator. The main difference between those two types is the presence, or the absence, of a tail on the balloon's border.

The *NarrativeBalloon* and *SpeechBalloon* concepts were introduced. The former stands for all kinds of balloons making the story moving forward (narrative frame, dialog, etc.). The latter formalizes the concept of balloons being part of the global speech, whether it is thought or spoken. They are defined by the fact that they are linked to, at least, one line of text. *SpeechBalloon* instances also have to be related to an instance of *Tail* as expressed in Equations 7 and 8.

$$\text{NarrativeBalloon} \equiv \text{hasTextLine } \textbf{some } \text{TextLine} \tag{7}$$

$$\text{SpeechBalloon} \equiv \text{NarrativeBalloon } \textbf{and } \text{hasTail } \textbf{some } \text{Tail} \tag{8}$$

Consequently, the semantic of the *TextLine* concept can be refined as well. Some of its instances are part or the global narration while others are part

of the speech. We introduced the *SpeechTextLine* and *NarrativeTextLine* concepts to formalize these notions. They are defined with Equations 9 and 10. The *isLineOf* relation is the inverse of *hasTextLine*.

$$\text{SpeechTextLine} \equiv \text{TextLine } \textbf{and } \text{isLineOf } \textbf{some } \text{SpeechBalloon} \quad (9)$$

$$\text{NarrativeTextLine} \equiv \text{TextLine } \textbf{and } \text{isLineOf } \textbf{some } \text{NarrativeBalloon} \quad (10)$$

The location of a speech balloon's character is usually in the same panel. The *says* relation expresses the relation between a speech balloon and a character. It takes *Character* as domain and *SpeechBalloon* as range. The *Speaker* concept formalizes the idea of a character saying some speech balloon at some point of the story. It is defined by Equation 11.

$$\text{Speaker} \equiv \text{Character } \textbf{and } \text{says } \textbf{some } \text{SpeechBalloon} \quad (11)$$

Figure 10 shows the introduced subsumption relations for *Balloon*, *TextLine* and *Character*.

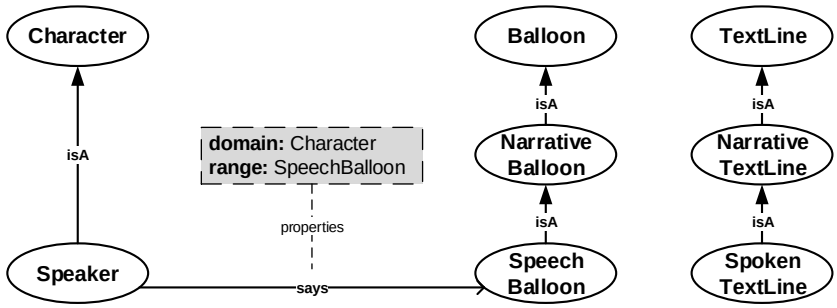

Figure 10: Specialization of the *Character*, *Balloon* and *TextLine* concepts.

3.3. Binding the ontologies

The image processing ontology, $\mathcal{O}_{ip}$ hereafter, and the comic books ontology, $\mathcal{O}_{cb}$, detailed in Sections 3.1 and 3.2 are entirely independent from each other. They must be connected in order to be used together in a context of comic books' images analysis. We considered that a digitized comic book's plate is the raw piece of data provided to the processing algorithms. Therefore, we stated an equivalence between the *Image* concept of $\mathcal{O}_{ip}$ and the *Plate* concept of $\mathcal{O}_{cb}$ (Equation 12).

$$\text{Image}_{ip} \equiv \text{Plate}_{cb} \tag{12}$$

As we put it in Subsection 3.1.2, the purpose of an image processing algorithm is to extract a specific kind of visual elements from an image. In our context, the explicit purposes of our different algorithms were the extraction of panels, balloons, balloon tails, lines of text and characters. We used this information to link several distinct subsets of ROI_{ip} individuals with the corresponding elements of the comic books ontology, as formalized in the following equivalence relation, where *Element* must be replaced by the desired kind of element:

$\text{Element}_{cb} \equiv \text{ROI}_{ip}$ **and** hasExtractor **some** (hasROIType **value** "Element")

The concepts from $\mathcal{O}_{cb}$ are automatically instantiated with the individuals provided by their corresponding extractors in $\mathcal{O}_{ip}$. The *hasROIType* attribute being functional, the uniqueness of the link between a ROI_{ip} instance and a $\mathcal{O}_{cb}$ concept is guaranteed. Consequently, the properties of the ROI_{ip} concept are added to $\mathcal{O}_{cb}$ concepts so their instances can be manipulated both as comic books elements, having their role and place in the comic book's structure, and as image items having a spatial position in the original image. Figure 11 illustrates these interactions. The ontologies and a small A-Box are available online[4] and can be manipulated through any ontology editor, such as Protege.

4. Experiments

We present in this section the experiments that used the framework we presented in Section 3.1 or were made possible by it.

We start by evaluating our conceptualization choices with regards to an annotated dataset that we used to populate the A-Box of our ontologies. Then, we present a first experiment on the progressive discovery of a comic books' page's content. We go through the protocol very briefly as we fully described it in a previous publication [41]. The second presented experiment defines a proper way to browse a digitized comic books' content, that is respectful of the original author's intentions.

[4]https://git.univ-lr.fr/cguerin/articles/tree/master/20150601_IS/code/ontologies

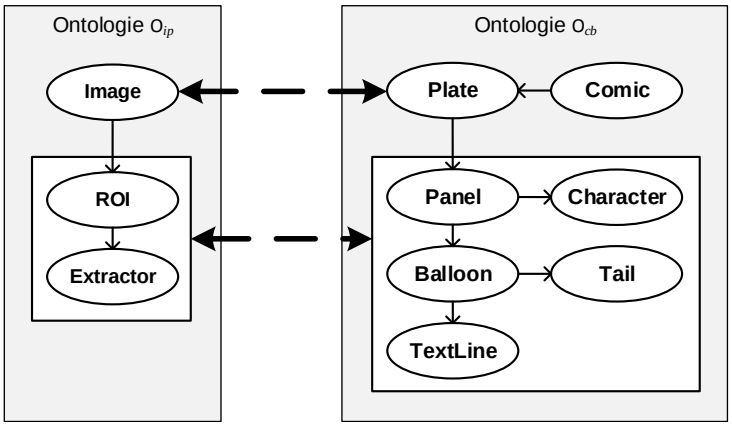

Figure 11: Interaction between the image and comic books ontologies.

4.1. Evaluation of our conceptualization

Beyond the concept hierarchy and its relations, our conceptualization formulates the strong assumption that: 1) a panel *belongs* to one and only one plate, 2) a balloon *belongs* to one and only one panel, 3) an instance of character *belongs* to one and only one panel, 4) a line of text *belongs* to one and only one balloon. These constraints represent the long time established conventions between authors and their readers. We expect them to be true for a wide majority of comics. They are usually respected by comic books drawers to make sure the reading remains straightforward and accessible.

The position of visual elements must reflect their belonging to a scene depicted in a panel or their association with a speech balloon. If it is quite common to find balloons or characters that partially step outside their reference frame, this phenomenon usually remains limited though. So even if a spatial inclusion of an element to another is matter of debate, their semantic belonging is formal and unquestionable. The chosen term *belongs* reflects the situation where a graphical element overlaps another element, (*a fortiori* when it includes the other), on a *significant* proportion of its area. When it intersects several other elements, we only keep the largest intersection. This way, we consider that a line of text belongs to the balloon that includes it, not to the panel that includes this balloon.

Confronted to the lack of reliable information on what could be a *significant* intersection of comic books' elements, we estimated its value using the eBDtheque dataset [19]. This dataset is made of a hundred images from

various comic books styles. For each element of the dataset, we determined what would be the type of its direct container for a given value of intersection. Figure 12 shows, for each concept and intersection value, the amount of instances with a valid container, regarding the domain and range constraints of our conceptualization. We independently processed each type of element as there is no reason to assume that they would share the same optimal value.

When we select the values that give the best results for each type of element, we get a conceptualization that is valid on 99.6% of panels, 87.4% of balloons, 81.6% of text lines and 94.9% of characters from the eBDtheque dataset. The reasons behind the two lowest figures, balloons and text lines, are quite understandable. As stated in the ground truth protocol, panels without borders were annotated as close as possible from their drawn content, ignoring balloons. Those balloons end up being not included in any panel. Similarly, the 18.4% of text lines that are outside of any balloon are explained by the presence of a few web-comics with bibliographical annotations on the page. The dataset also includes some comics that are older than the invention of speech balloons [43]. The authors positioned the text below each panel then, i.e. outside of any balloon.

The drop that we see for an overlapping value of 100% is caused by an understandable side effect. Indeed, each element having to be entirely contained into a bigger one, a single outside pixel makes it belong to the bigger one. The effect does not affect panels that need to belong to the page, the biggest element of all.

4.2. Iterative comics page understanding

We proposed a complete and scalable protocol based on a looping process between a set of image processing algorithms and our ontologies. In the remaining of this section, we refer to the image analysis process as *Low Level System* (LLS) and to our framework as *High Level System* (HLS).

4.2.1. Global description

The high level system is made of the two presented ontologies, $\mathcal{O}_{ip}$ and $\mathcal{O}_{cb}$. Once populated with the output data from the LLS, they compose our knowledge base. An inference engine process it to produce logical conclusions depending on the data (the A-Box) and its consistency with the formalized domain (the T-Box). Those conclusions might be the validation, rejection or creation of elements that are fed back to the LLS.

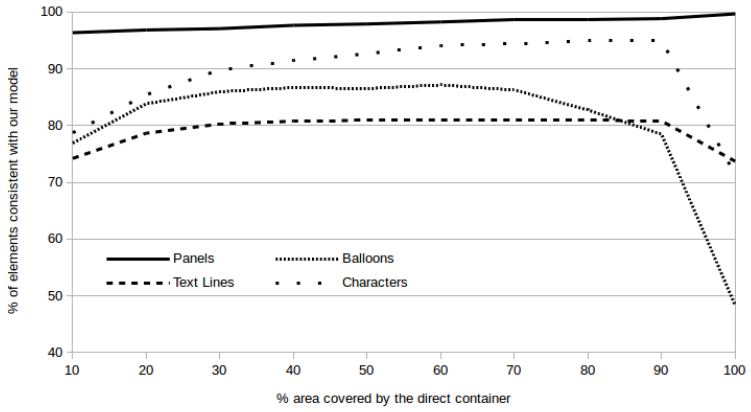

Figure 12: Percentage of panels, balloons, lines of text and characters from the eBDtheque dataset that are valid with respect to our conceptualization, for a given overlapping value with their direct container.

Its purpose is to handle progressively the output of the LLS, to evaluate the corresponding data and extract meaning from it, based on its formalized semantics. It begins with visual elements that we identified as the easiest ones (e.g. panels, balloons and text lines) to progressively extract more complex elements, such as characters. The two systems interact until all extractions are consistent with the ontological models. The loop presented in Figure 13 illustrates this interaction.

During the first step, the LLS provides the HLS with hypothesis of image regions labeled as panels, balloons, etc. During the second step, the HLS evaluates these hypothesis, validates those which are consistent with the model and puts away those which are not. New information is produced during the third step, based on the validated image regions and the modeled knowledge.

We do not go through the presentation in full details of the processing loop in this article. We recall the qualitative results in the following subsection. For more information, we invite the interested reader to refer to our previous paper [41].

4.2.2. Results

We evaluated the system on the eBDtheque dataset [19]. As detailed in Section 4.1, our model is consistent with 99.6% of panels, 87.4% of balloons,

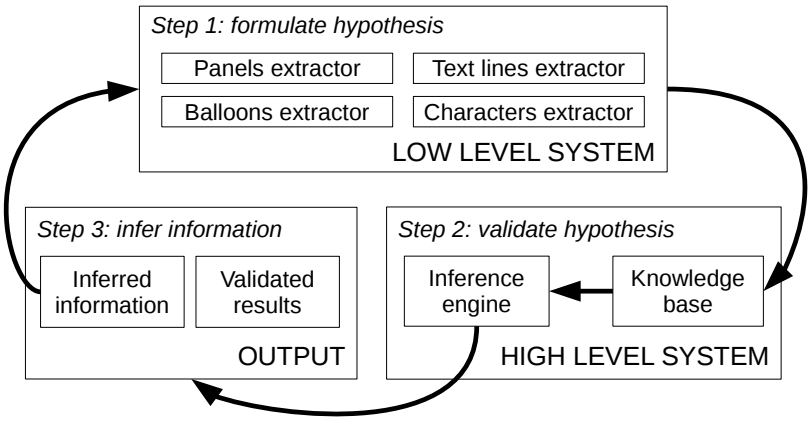

Figure 13: The interaction loop between low and high level systems.

81.6% of text lines and 94.9% of characters. So we can take those figures as the top values we aim to reach. The A-Box of $\mathcal{O}_{cb}$ was populated with the elements from the eBDtheque ground truth dataset and the elements from the image processing algorithms. For each spatial intersection between i_t, an automatically extracted element of type t, and i'_t an element from the ground truth of the same type, we created a new instance of *Evaluation* and linked it to them with the properties *hasCandidate(e,*i_t*)* and *hasReference(e,*i'_t*)*. We added to each of these instances a *hasVOCScore* attribute, representing the measure introduced by Everingham et al. [15]. This measure provides a quality indicator q, based on the spatial intersection and union of the two compared sets. It is formalized in Equation 13 where ri_t and ri'_t respectively stand for i_t and i'_t ROI's set of pixels.

$$q_{ii'} = \frac{|ri_t \cap ri'_t|}{|ri_t \cup ri'_t|} \tag{13}$$

Based on this measure, a reference i' validates a region i if the value of $q_{ii'}$ is greater than or equal to $0,5$. The rule of Equation 3 is adequately modified into Equation 14 to classify the instances of *Evaluation* into the *Validation* or *Error* concepts.

$$\text{Validation} \equiv \text{Evaluation } \textbf{and } \text{hasVOCScore } \textbf{some } \text{double}[\geq 0{,}5] \quad (14)$$

$$\text{Error} \equiv \text{Evaluation } \textbf{and } \text{hasVOCScore } \textbf{some } \text{double}[< 0{,}5] \quad (15)$$

The overall quality of the page analysis is given through the classical measures of recall R, precision P and F-Measure F, given in Equations 16.

$$R = \frac{\text{nbValidatedElements}}{\text{nbGTElements}} \quad P = \frac{\text{nbValidatedElements}}{\text{nbAutoElements}} \quad F = \frac{2RP}{R+P} \quad (16)$$

The values of "nbValidatedElements", "nbGTElements" and "nbAutoElements" are provided by the SPARQL queries detailed in Listing 1, where `<TYPE>` needs to be changed with the class of elements one wants to evaluate.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX bd: <http://l3i.univ-larochelle.fr/eBDtheque/domain#>
PREFIX img: <http://l3i.univ-larochelle.fr/eBDtheque/image#>

SELECT (count(DISTINCT ?element) as ?nbValidatedElements)
WHERE {
  ?element rdf:type bd:<TYPE>.
  ?validation rdf:type img:Validation.
  ?validation img:hasReference ?element
}

SELECT (count(DISTINCT ?element) as ?nbAutoElements)
WHERE {
  ?element rdf:type bd:<TYPE>.
  ?extractor rdf:type img:AutoExtractor.
  ?element img:hasExtractor ?extractor.
}

SELECT (count(DISTINCT ?element) as ?nbGTElements)
WHERE {
  ?element rdf:type bd:<TYPE>.
  ?extractor rdf:type img:GroundTruthExtractor.
  ?element img:hasExtractor ?extractor.
}
```

Listing 1: Query that returns the number of elements of a given type from ground truth that validated an extracted element.

We evaluated the results of the extraction in two ways. As our system is based upon the idea of iterative content discovery, the results of the first iterations have an impact on the detection of elements that come later in the process. Thus, we evaluated each type of element independently, by taking all the other elements from the ground truth. The results showed an anticipated drop of recall values as, for the time being, only deletion operations came out from our framework. This drop is well compensated by an increase in the precision from +0.48% for the panels, to 22.62% for the text lines. The average increase of F-Measure value on all types of elements is 2.9%.

In a second time, we evaluated how our system performs on a complete unsupervised extraction. While the global results are significantly lower than before, which we expected as the error spread to higher iterations, the gain in the overall F-Measure brought by the HLS is comparable to the independent evaluation, with a value of 3%. Detailed and fully discussed results are available in [41].

4.3. Reading order

The reading order is a simple but important piece of information that we can deduct from the spatial organization of the elements. We focus on this section on panels but we could extend out reasoning process to balloons and text lines.

We saw that comic book's elements share a tight spatial relationship with each other. The panels' order is defined in the page reference, balloons' order is from the containing panel and so on. Traditionally, a panel has a rectangular shape. A black stroke (and often a white gutter) allows the reader to visually separate it from other panels. Usually, the author puts a special attention to make his drawings intelligible by making each panel clearly identifiable. An example of different panels separation is given on Figure 21. Still, panels are not always well separated and organized in a simple waffle structure. Japanese manga and modern western comics provide many counterexamples where panels overlap, partially or totally, on each other. The order relation between such couple of panels are not as simple as when they are rigorously disjoint. This spatial positioning is not suffered by the author and actually reflects the way he wants to tell his story. For instance, Figure 21 shows the inclusion of the two lower panels in a bigger one. The produced effect is a feeling of simultaneity in the depicted scenes. The biggest panel making a kind of wider background for the smallest ones

that are more focused on the action. The reader is encouraged to let his eyes wander back and forth between the three panels to appreciate the scene as a whole.

4.3.1. Spatial relations

The subsumption of $\mathcal{O}_{cb}$'s *Panel* concept by $\mathcal{O}_{ip}$'s *ROI* concept makes possible to attach spatial coordinates to each panel of a page through the *hasWKT* attribute. We can formalize the spatial relations existing between the instances of *Panel* sharing a belonging to the same instance of *Plate*. Let C_i and C_j be two panels, they can be tied by two kinds of spatial relations. The first kind is a topological spatial relation, as described by RCC-8 [11] (see Figure 14).

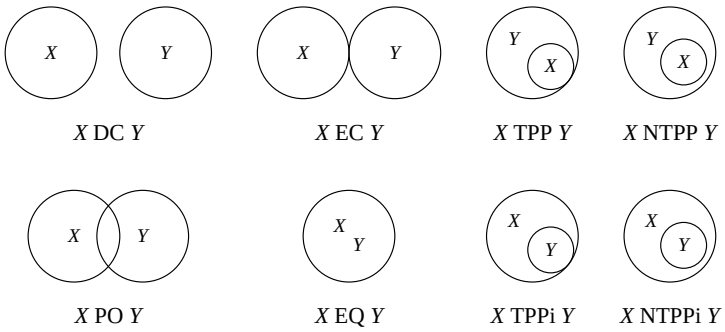

Figure 14: The eight relations of RCC-8.

We grouped these eight relations in three use cases that can happen on a comics' page. Let A_i and A_j respectively be the inside of C_i and C_j.

- $|A_i \cap A_j| = 0$, i.e. C_i and C_j have no intersection whatsoever.
  This case corresponds to disjunction (DC) and externally connected (EC). The panels can be seen as separable and we called the relation SP.

- $0 < |A_i \cap A_j| < |A_i|, |A_j|$, i.e. C_i and C_j share a part of the page area, without one being included in the other.
  This is simply the partially overlapping situation (PO).

- $A_i \cap A_j = A_i$ or $A_i \cap A_j = A_j$, i.e. C_i is included into C_j or vice versa. This matches the proper part relations, with and without tangential property (TPP, NTPP, TPPi, NTPPi). A panel is either containing or contained by the other one. We called the relation CO.

The second kind of spatial relations is the Cone Shaped Direction (CSD). A panel C_i can be up (North), right (East), down (South) or left (West) of a panel C_j. As human readers, we naturally arrange the panels horizontally, then vertically. Introducing transitional direction (like North-East) would not lead to better performances. The relative position of C_i and C_j is set in different ways, depending on their topological relationship. Let us say we want to evaluate the relation $South(C_i, C_j)$, i.e. C_i down C_j. If the couple of panels is separable, every single point of C_j must have a higher y value of any point of C_i for the relation to be true (see Figure 15a and Equation 17). Similarly, for the relation $East(C_i, C_j)$ to be true, the x value of C_j must exceed those of C_i (see Figure 15b).

$$South(C_i, C_j) \Leftarrow SP(C_i, C_j),\ (\forall(y_{i_i}, y_{j_i}),\ y_{j_i} > y_{i_i}) \tag{17}$$

In the case of two panels overlapping each other, we reduce them at their center of gravity and compare their position according to a four divisions circle. The $South(C_i, C_j)$ is validated if the position of C_j's center is inside the cone defined between $-\pi/4$ and $-3\pi/4$, on a circle centered on C_i's center (see Figure 15c and Equation 18). We compute the $East$ relation in the same way, with an angle between $-\pi/4$ and $\pi/4$. From Equations 18 and 17, we can deduce that $South(C_i, C_j)$ and $East(C_i, C_j)$ have inverse functions and, consequently, that $North(C_i, C_j) \Leftrightarrow South(C_j, C_i)$ et $West(C_i, C_j) \Leftrightarrow East(C_j, C_i)$. Last, when we have $CO(C_i, C_j)$, i.e. C_i contains or is contained by C_j, we consider the relation as being undecidable. This indecision provides the reader with the freedom to interpret the authors positioning choices for his story, as it would on a printed comic book.

$$South(C_i, C_j) \Leftarrow PO(C_i, C_j),\ (y_{j_c} > y_{i_c}),\ (y_{j_c} - y_{i_c} > |x_{j_c} - x_{i_c}|) \tag{18}$$

4.3.2. Panels ordering

The topological and directional spatial relations being defined, we can now state the equation to order a set of panels. Let the $Next(C_i, C_j)$ relation

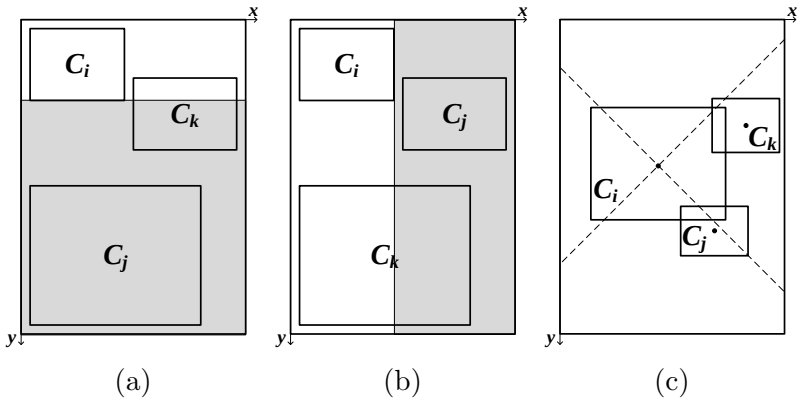

Figure 15: (a) C_i, C_j and C_k are separable, C_j is south of C_i but C_k is not. (b) Similarly, C_j is east of C_i but C_k is not. (c) C_i, overlaps C_j and C_k, C_j is south of C_i but C_k is east.

express the fact that a panel C_j must be read later than a panel C_i Not necessarily *right after* C_i, just later. This relation is true when C_j is below or at the same level but on the right of C_i, as formalized in Equations 19 and 20.

$$Next(C_i, C_j) \Leftarrow South(C_i, C_j) \tag{19}$$

$$Next(C_i, C_j) \Leftarrow \neg South(C_i, C_j),\ \neg South(C_j, C_i),\ East(C_i, C_j) \tag{20}$$

Note that Equation 20 is only valid for the western reading point of view, from left to right. If we want it to work with manga, which are read from right to left, we must switch the parameters of the *East* predicate like this:

$$Next(C_i, C_j) \Leftarrow \neg South(C_i, C_j),\ \neg South(C_j, C_i),\ East(C_j, C_i) \tag{21}$$

The flowchart in Figure 16 illustrates our method to order a couple of panels, i.e. to produce a $Next$ relation. A panel C_i has a $Next$ relation with a set of panels S_i. We compute the reading order with a merge sort algorithm, applied on a page's set of panels. Algorithm 1 describes the comparison criterion of two elements from this set, stating if one should be put before or after the other. It can return three values for each couple of panels (C_i, C_j), depending on (S_i, S_j). If $C_j \in S_i$ then C_i is put before C_j, else if $C_i \in S_j$ then C_i is ordered after C_j, else C_i and C_j are given the same rank.

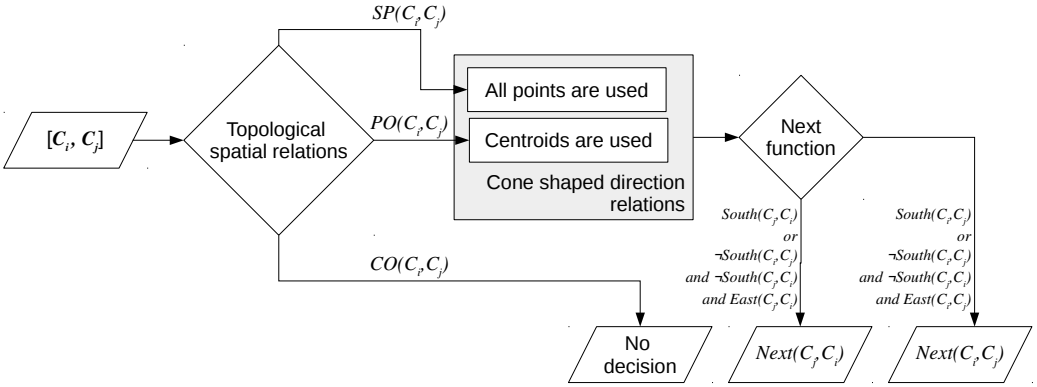

Figure 16: Flowchart of our ordering algorithm. A couple of panels is passed as input. Their topological and directional spatial relationships are computed to produce a Next relation through the Next function defined by Equations 19 and 20.

Please note that, unlike the reading order, the `compare` function is not transitive. Indeed, Figure 17 shows a situation where, according to Equation 20, the horizontally hatched panels must be read before the squared panel. Still according to Equation 20, the latter must be browsed before the vertically hatched panels. However, according to Equation 19, the horizontally hatched panels must also be read *after* the vertically hatched ones. Those must paradoxically be read after *and* before the squared panel.

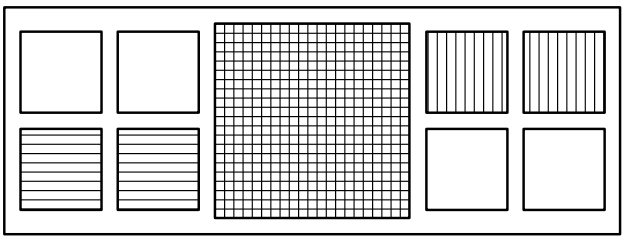

Figure 17: Special case of inconsistent reading order.

The merge sort algorithm being stable, the final ordering of the ambiguous cases depends of their initial position in the input set. However, we can easily detect those rare situations by checking if, for every panel C_i of rank i, and for every panel C_l of rank l, with $l > i$, $C_l \in S_i$ is true. This measure returns the cases where two panels can be read in any order. We can filter those cases

Name: `compare`
Data: Two panels C_i and C_j and their respective followers sets S_i and S_j.
Result: An integer between -1 and 1 stating the position of C_i regarding C_j.
begin
  if $CO(C_i, C_j)$ **then** return 0 ;
  else if $C_j \in S_i$ **then** return -1 ;
  else return 1 ;
end

Algorithm 1: Comparison function of two panels for their sequential arrangement.

out because their comparison with the 1 function returns 0. Once we detected the inconsistency, we can ask for a human intervention to manually define the right reading order. Experimentations, detailed in subsection 4.3.4, show that these cases are unusual and show up on a very limited number of pages. It deserves to be noted though that a crooked digitized image can produce a similar behavior. Two panels, that are disjoint in the straight image, might intersect over the height axis of the crooked image. Therefore we introduced a tolerance threshold to avoid unfortunate miscalculation.

4.3.3. Integration into the ontology

The sort method we used provides a partial reading order (see Figure 19b). It can be extended into a total order with a topological sort, or be inserted as it into the A-Box of our model. The rank i, given to each panel C, is formalized through the assertion of the *hasRank(C, i)* axiom into the ontology. In the case of an indecision in the order of a couple of panels, they are given the same rank value. A large panel has as many *hasRank* instances as there are smaller panels inside it (see Figure 18).

We added a new rule (see Equation 22) to the $\mathcal{O}_{bd}$ ontology to formalize the succession relation between panels.

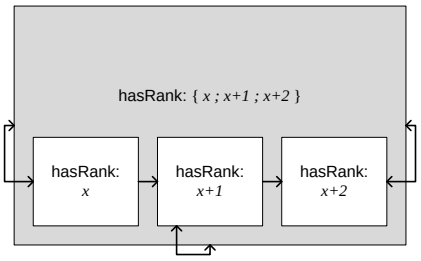

Figure 18: Output relations in the case of multiple inclusions of several panels (white) in a bigger one (gray). The arrows represent the *hasNextPanel* relations.

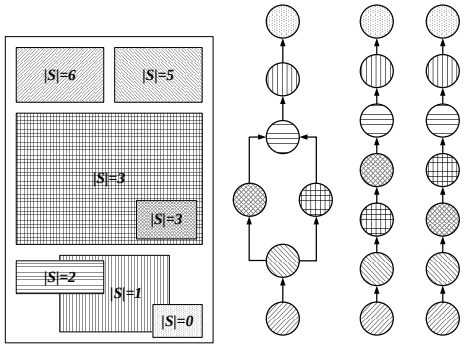

Figure 19: (a) A page example. S is the set of panels coming after a given panel, according to Equation 19. (b) The partial order obtained by a merge sort algorithm, according to the comparison method 1. (c) Both total order that can be generated from the partial order with a topological sort.

$$\begin{aligned} &\text{Plate(?p), Extractor(?e), Panel(?p1), Panel(?p2), hasPanel(?p, ?p1),} \\ &\text{hasPanel(?p, ?p2), hasExtractor(?p1, ?e), hasExtractor(?p2, ?e),} \\ &\text{hasRank(?p1, ?r1), hasRank(?p2, ?r2), equal(?r1, ?r2)} \\ &\rightarrow \text{hasNextPanel(?p1, ?p2)} \end{aligned} \tag{22}$$

4.3.4. Evaluation

We evaluated how our method performs on the eBDtheque dataset. Let $r_i = r(C_i)$ and $r'_i = r'(C_i)$ the respective functions giving to a panel C_i a rank r_i from the ground truth and r'_i from our ordering method. We computed the value of Equation 23 for each couple of panels $\{C_i, C_j\}$ with $r'_i = r'_j + 1$.

$$r(C_i) - r(C_j) = r'(C_i) - r'(C_j) = 1 \tag{23}$$

If C_i is right next C_j (or the other way around) by our ordering method but not according to the ground truth, the equality is not true and neither is

the calculated relation. This measure is more accurate than the direct comparison of r_i and r'_i or each panel C_i. Indeed, it evaluates the relative position of panels from each other and does not penalize the whole set for one shift at the beginning of the sequence. The eBDtheque dataset is made of 100 pages, containing 849 panels. Therefore, 749 relations have to be evaluated. The results of this evaluation are given in Table 1, along with the results of our implementation[5] of a state of the art method [39]. Both algorithms not being strictly transitive, the given results are the best we obtained out of ten tries (results may be lower up to 2%).

Dataset	**# of rel.**	**Results**	**Ponsard [39]**
Full Dataset (FD)	749	94,65%	87.85%
FD - Inclusions (I)	746	95,03%	87.85%
FD - I - Doubles pages (DP)	703	97,58%	91.47%
FD - I - DP - Extreme Page (EP)	581	99%	94.32%

Table 1: Results of our panels ordering method compared to Ponsard's [39].

The presented results on the full dataset are relatively good with an accuracy over 94%. It outperforms the state of the art in every tested configuration. We explain the reasons for the 6% errors below.

Firstly, the annotated ground truth has only total reading orders and does not handle the simultaneity of included panels. Our algorithm giving the same rank value to those panels, Equation 23 is not verified in those cases. Nevertheless, the reading order asserted in the ground truth dataset is indeed one of the possible topological sorts of our own reading order. This causes 0.5% of the errors. Secondly, five images of the dataset represent two distinct pages that are not meant to be read as one. The left page having to be completely read before the right page. We designed our algorithm to handle single pages, or actual double pages. It is irrelevant to test it over such images. By getting rid of those five images, the results rise up over 97%. The third and last reason lies in the fact that one special page (see Figure 20) has a lot of panels with an unusual layout. It also illustrates the weakness of our method on the layout described in Figure 17. We can find it in several places inside the image.

[5]Source code: https://git.univ-lr.fr/cguerin/articles/tree/master/20150601_IS/code

Figure 20: Unusual layout. Credits: C. Ware - Acme Novelty Library - New York Pantheon.

Figure 21: The two top panels are separated by a white gutter, while the two lower panels are only separated by a black stroke. Credits: O. Jolivet - L'Affaire Pradi - Clair de Lune.

Ponsard's method [39] shares some similarities with ours, as it handles disjoint and overlapping panels differently. The two methods get different in the way they order included panels. Ours is more flexible by allowing the reader to go back and forth between the included panels and then go on with his reading. Every action can be achieved by following the *hasNextPanel* property of our ontology. Ponsard's is more consistent with the way the ground truth was annotated, that is why removing included panels does not show an increase of the results in our experiments. It also suffers from the same weakness than ours regarding the layout of Figure 17.

5. Discussion and conclusion

This article developed and detailed a conceptualization that allows semantic analysis and information deduction from automatically extracted comic books visual elements. It is made of two independent ontologies, one formalizing the image processing domain, the other the comic books domain. When we use these two ontologies together, it allows us, on one hand, to handle the

extracted visual elements as comic books entities, ruled by the established codes of the art of comics and, on the other hand, to process the content of a comic book as simple image areas which physical properties can be evaluated.

The image ontology we proposed in this paper does not redefine any standard. We conceptualized it so it remains compatible with the existing ones though. It has the advantage of being quite simple, with a few concepts, and of being independent from any specific application domain. While it is quite efficient for our own study context, it can easily be used to any other domain, such as documents or natural scenes analysis.

The conceptualization choices that led to the proposed comic books ontology were driven by the idea of an image analysis use case. While the layout conventions and vocabulary are widely shared through the worldwide literature, comic books remain a form of art. We expect from artistic pieces to break the established conventions in order to make the art evolve and experiment new things. Thus, the worldwide comic books legacy presents a very heterogeneous face. There are no formal rules that can be true for all instances of comics, beyond the idea of "sequential drawings". Such a conceptualization would be too flexible to be of any practical use. We evaluated the relevance of our choices on an annotated dataset, resulting in a solid consistency of our model over 90% of panels, balloons, lines of text and characters.

We presented the application on our framework in two different contexts. We first used it as an image analysis support to progressively understand the content of a comic book's page. It not only enhanced the extraction results on the simplest elements, such as panels and balloons, but it also made possible to tackle the issue of detecting more complex elements, such as characters. Characters are particularly challenging to spot automatically as their shape are not consistent from one panel to another. While we focused our attention on increasing the precision value of the extraction results, we are thinking of different possibilities to limit the recall drop. One idea would be to increase the number of loops between the low and high level systems with, for instance, a warning on empty balloons to spot potentially missed text lines is promising.

The flexibility of our model is also illustrated through a new way to browse the elements of a page, that is respectful to the original author's intentions. We are thinking of a deeper integration of spatial reasoning into our model. A

standard such as GeoSPARQL in the image ontology would make possible the creation of logical rules based on the position of the elements. It would allow to classify elements as new concept instances, such as voice-over balloons, which are usually displayed in corner of panels for instance.

There is plenty left to do before being able to claim that the content of a comic book can be fully described automatically. New extractors should be developed to inject more pieces of information into our model. More than increasing the amount of encoded elements, they would allow further interaction and reasoning possibilities over their intrinsic properties. It would also be very interesting to adapt methods from natural scenes understanding to comic book images, based on what makes their specificity. Their shape, size and position, as well as how their content is drawn, the color that were used and so on, are pieces of information that carry some strong semantic information. Inspiration would probably come from the increasing interest on comic books given by psychology and linguistic communities. A lot of work is to be done in other domains as well, such as text analysis. The characters recognition itself is a challenge considering the common irregularity of the written text lines. Spotting and recognition of onomatopoeia is even harder. The analysis of panels' unusual layout or the spotting of twin pages are some other examples of what would be interesting to tackle in a near future.

6. Acknowledgements

The authors would like to thank Jean-Christophe Burie and Jean-Marc Ogier for their help with the low-level processing. This work was supported by a Doctorate scholarship of the University of La Rochelle, the Region Poitou-Charentes (France) and the Future Investment Program iiBD (French government). We are grateful to all authors and publishers of comics and manga from the eBDtheque dataset for having allowed us to use and share their works.

[1] Bannour, H. and Hudelot, C. (2011). Towards ontologies for image interpretation and annotation. In *Content-Based Multimedia Indexing (CBMI)*, pages 211–216. IEEE.

[2] Bateman, J. A. and Wildfeuer, J. (2014). A multimodal discourse theory of visual narrative. *Journal of Pragmatics*, 74:180–208.

[3] Battle, R. and Kolas, D. (2012). Enabling the Geospatial Semantic Web with Parliament and GeoSPARQL. *Semantic Web*, 3(4):355–370.

[4] Belgiu, M., Hofer, B., and Hofmann, P. (2014). Coupling formalized knowledge bases with object-based image analysis. *Remote Sensing Letters*, 5(6):530–538.

[5] Blaschke, T. (2010). Object based image analysis for remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65(1):2–16.

[6] Bobillo, F. and Straccia, U. (2011). Fuzzy Ontology Representation using OWL 2. *International Journal of Approximate Reasoning*, 52(7):1073–1094.

[7] Brickley, D. (2006). Basic Geo (WGS84 lat/long) Vocabulary.

[8] Chang, S.-F., Chen, W., and Sundaram, H. (1998). Semantic Visual Templates: Linking Visual Features to Semantics. In *International Conference on Image Processing*, pages 531–535.

[9] Chen, N., Zhou, Q.-Y., and Prasanna, V. K. (2012). Understanding web images by object relation network. *International conference on World Wide Web*, pages 291–300.

[10] Christodoulou, G., Petrakis, E. G., and Batsakis, S. (2012). Qualitative Spatial Reasoning Using Topological and Directional Information in OWL. In *International Conference on Tools with Artificial Intelligence*, pages 596–602. Ieee.

[11] Cohn, A. G., Bennett, B., Gooday, J., and Gotts, N. M. (1997). *Representing and Reasoning with Qualitative Spatial Relations About Regions*, pages 97–134. Springer Netherlands, Dordrecht.

[12] Cohn, N., Paczynski, M., Holcomb, P., Jackendoff, R., and Kuperberg, G. (2011). Comics on the Brain : Structure and Meaning in Sequential Image Comprehension. *Psychophysiology*, 48.

[13] Coustaty, M., Bouju, A., Bertet, K., and Louis, G. (2011). Using ontologies to reduce the semantic gap between historians and image processing algorithms. In *International Conference on Document Analysis and Recognition (ICDAR)*, pages 156–160. IEEE.

[14] Crevier, D. and Lepage, R. (1997). Knowledge-Based Image Understanding Systems: A Survey. *Computer Vision and Image Understanding*, 67(2):161–185.

[15] Everingham, M., Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. (2009). The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88(2):303–338.

[16] Forestier, G., Puissant, A., Wemmert, C., and Gançarski, P. (2012). Knowledge-based region labeling for remote sensing image interpretation. *Computers, Environment and Urban Systems*, 36(5):470–480.

[17] Forestier, G., Wemmert, C., and Puissant, A. (2013). Coastal Image Interpretation Using Background Knowledge and Semantics. *Computers & Geosciences*, 54(4):88–96.

[18] Gudivada, V. N. and Raghavan, V. V. (1995). Content-Based Image Retrieval Systems. *Computer*, 28(9):18–22.

[19] Guérin, C., Rigaud, C., Mercier, A., Ammar-Boudjelal, F., Bertet, K., Bouju, A., Burie, J.-C., Louis, G., Ogier, J.-M., and Revel, A. (2013). eBDtheque: A Representative Database of Comics. In *2013 12th International Conference on Document Analysis and Recognition*, pages 1145–1149. IEEE.

[20] Han, E., Yang, J., Yang, H., and Jung, K. (2007). *Automatic Mobile Content Conversion Using Semantic Image Analysis*, pages 298–307. Springer Berlin Heidelberg, Berlin, Heidelberg.

[21] Horridge, M., Drummond, N., Goodwin, J., Rector, A., Stevens, R., and Wang, H. H. (2006). The Manchester OWL Syntax. In *OWL: Experiences and Directions (OWLED)*.

[22] Hudelot, C., Atif, J., and Bloch, I. (2008). Fuzzy spatial relation ontology for image interpretation. *Fuzzy Sets and Systems*, 159(15):1929–1951.

[23] Jewell, M. O., Lawrence, F., and Tufeld, M. M. (2005). OntoMedia: An Ontology for the Representation of Heterogeneous Media. In *SIGIR Workshop on Multimedia Information Retrieval*.

[24] Jiang, S., Huang, T., and Gao, W. (2004). An Ontology-based Approach to Retrieve Digitized Art Images. In *IEEE/WIC/ACM International Conference on Web Intelligence*, pages 131–137.

[25] Klein, B. and Stroup, T. (1994). Grand Comics Database.

[26] Le Ber, F. and Napoli, A. (2002). Design and comparison of lattices of topological relations based on Galois lattice theory. In *International Conference on Principles of Knowledge Representation and Reasoning*.

[27] Liu, Y., Zhang, D., Lu, G., and Ma, W.-Y. (2007). A survey of content-based image retrieval with high-level semantics. *Pattern Recognition*, 40(1):262–282.

[28] Marszalek, M. and Schmid, C. (2007). Semantic Hierarchies for Visual Object Recognition. In *CVPR*.

[29] McIntosh, J. (2011). ComicsML http://jmac.org/projects/comics_ml/.

[30] Mezaris, V., Kompatsiaris, I., and Strintzis, M. G. (2003). An Ontology Approach to Object-Based Image Retrieval. *ICIP*, pages 511–514.

[31] Miles, A., Brickley, D., Matthews, B., and Wilson, M. (2005). SKOS Core: Simple Knowledge Organisation for the Web. In *International Conference on Dublin Core and Metadata Applications*, number 5, pages 3–10.

[32] Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K. J. (1990). Introduction to WordNet: An On-line Lexical Database. *International Journal of Lexicography*, 3(4):235–244.

[33] Minsky, M. (1975). A framework for representing knowledge. In *The Psychology of Computer Vision*. McGraw-Hill.

[34] Miron, A. D., Gensel, J., Villanova-Oliver, M., and Martin, H. (2007). Relations spatiales qualitatives dans les ontologies géographiques avec ONTOAST. *SAGEO*.

[35] Morozumi, A., Nomura, S., Nagamori, M., and Sugimoto, S. (2009). Metadata Framework for Manga: A Multi-paradign Metadata Description Framework for Digital Comics. In *International Conference on Dublin Core and Metadata Applications*, pages 61–70.

[36] Open GIS Consortium Inc. (2011). OpenGIS Implementation Standard for Geographic informations - Simple feature access - Part 1: Common Architecture.

[37] Pastierovic, R. (2011). Advanced Comic Book Format. https://launchpad.net/acbf.

[38] Petridis, K., Anastasopoulos, D., Saathoff, C., Timmermann, N., Kompatsiaris, Y., and Staab, S. (2006). M-OntoMat-Annotizer : Image Annotation Linking Ontologies and Multimedia Low-Level Features. In *KES*, pages 633–640.

[39] Ponsard, C., Ramdoyal, R., and Dziamski, D. (2012). An OCR-Enabled Digital Comic Books Viewer. *Computers Helping People with Special Needs*, 7382:471–478.

[40] Renouf, A., Clouard, R., and Revenu, M. (2007). How to formulate image processing applications? In *International Conference on Computer Vision Systems*.

[41] Rigaud, C., Guérin, C., Karatzas, D., Burie, J.-C., and Ogier, J.-M. (2015). Knowledge-driven understanding of images in comic books. *International Journal on Document Analysis and Recognition (IJDAR)*, 18(3):199–221.

[42] Rissen, P. (2012). *A Comics Ontology*. http://www.r4isstatic.com/231.

[43] Saint-Ogan, A. (1934). *Prosper et le monstre marin*. Hachette Editions.

[44] Smeulders, A. W., Worring, M., Santini, S., Gupta, A., and Jain, R. (2000). Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380.

[45] Stocker, M. and Sirin, E. (2009). PelletSpatial : A Hybrid RCC-8 and RDF / OWL Reasoning and Query Engine. In *Web Ontology Language: Experience and Directions (OWLED)*, volume 2009. Springer Verlag.

[46] Text Encoding Initiative Consortium (2014). Text Encoding Initiative.

[47] Tousch, A.-M., Herbin, S., and Audibert, J.-Y. (2012). Semantic hierarchies for image annotation: a survey. *Pattern Recognition*, 45(1):333–345.

[48] Tufis, M. and Ganascia, J.-G. (2016). Adding semantics to comics using a crowdsourcing approach. working paper or preprint.

[49] W3C (2012). Periodical Comics.

[50] Walsh, J. A. (2012). Comic Book Markup Language : An Introduction and Rationale. *Digital Humanities Quarterly (DHQ)*, 6(1):1–50.

[51] Zinger, S., Millet, C., Mathieu, B., Grefenstette, G., Hède, P., and Moëllic, P.-A. (2005). Extracting an Ontology of Portrayable Objects from WordNet. In *MUSCLE/ImageCLEF workshop on Image and Video retrieval evaluation*, pages 17–23.