Journal of *Imaging*

MDPI

Article

Digital Comics Image Indexing Based on Deep Learning

Nhu-Van Nguyen *, Christophe Rigaud and Jean-Christophe Burie

Lab L3I, University of La Rochelle, 17000 La Rochelle, France; christophe.rigaud@univ-lr.fr (C.R.); jean-christophe.burie@univ-lr.fr (J.-C.B.)

* Correspondence: nhu-van.nguyen@univ-lr.fr

Received: 30 April 2018; Accepted: 27 June 2018; Published: 2 July 2018

Abstract: The digital comic book market is growing every year now, mixing digitized and digital-born comics. Digitized comics suffer from a limited automatic content understanding which restricts online content search and reading applications. This study shows how to combine state-of-the-art image analysis methods to encode and index images into an XML-like text file. Content description file can then be used to automatically split comic book images into sub-images corresponding to panels easily indexable with relevant information about their respective content. This allows advanced search in keywords said by specific comic characters, action and scene retrieval using natural language processing. We get down to panel, balloon, text, comic character and face detection using traditional approaches and breakthrough deep learning models, and also text recognition using LSTM model. Evaluations on a dataset composed of online library content are presented, and a new public dataset is also proposed.

Keywords: comics analysis; image indexing; deep learning; CNN; LSTM; handwritten text recognition

1. Introduction

Initially, comics were printed on paper books, but nowadays, born-digital comic books have become more and more popular. Physical museums such as the "Internationale de la Bande Dessinée et de l'image" (http://collections.citebd.org) in France and the Kyoto International Manga Museum (http://kyotomm.jp), online museums or archives such as The Digital Comic Museum (http://digitalcomicmuseum.com), Comic Book Plus (http://comicbookplus.com) and the Grand Comics Database (http://comics.org) have already digitized several thousands of comic albums and some are in the public domain. There are also online libraries selling digital and physical comic books such as DC Comics from Warner Bros (http://www.dccomics.com), comiXology (https://www.comixology.com), Sequencity (http://sequencity.com), Izneo (https://www.izneo.com) and Koomic (http://koomic.com). Other websites host a huge amount of comics and manga created by amateurs for free reading purpose, based on a Paid To Click business model, e.g., MangaFox (http://fanfox.net) and MangaHere (http://www.mangahere.cc).

Existing web image search engines such as Google, Yahoo, Bing and Qwant search images based on textual evidences from text around the images in web pages [1,2]. Although these approaches can find many relevant images, the retrieval precision is poor as they cannot confirm whether the retrieved images indeed contain the query elements. The result is that users often have to go through a long list to eventually find the desired images. This is a time-consuming process as the returned results almost always contain multiple topics which are mixed together. To improve web image retrieval performance, we have to fuse the evidences from textual and visual contents. Breakthrough techniques based on deep learning started to improve image retrieval relevancy only a few years ago [3,4]. We propose to adapt such techniques to specific images as comic strip images.

Regarding comics, there are extra difficulties related to the image layout for specifying a query and finding results at page and/or panel level. A panel being a subpart of a comic page, it may be more relevant to return to the user a panel region than the entire page, especially if the query is written or drawn inside a single panel (returning the full image would result in a lot of unwanted information for the user). Digital comic book analysis can then be investigated to help the creation or conversion of comics in an enhanced digital form and to improve user's reading and browsing experiences.

This field of research includes content segmentation [5], relation analysis [6], style analysis [7] and information retrieval [8]. While information retrieval research domain has a long history, content-based comics indexing and the creation of comics in a digital form are quite recent. Comic book contents (e.g., panels, balloons, texts, comic characters, etc.) and their relations (e.g., read before, said by, thought by, and addressed to) have to be automatically recognized to reconstruct the story and to keep the story coherent [6]. The more elements we are able to extract and recognize automatically, the better the story reconstruction quality and the result image relevances will be. The challenges that are facing researchers dealing with comic book images are related to the diversity of styles, layout, text, and actions from a mixture of representations of real and imaginary worlds. These challenges make the task of content learning and recognition much harder.

In parallel, deep learning approaches are becoming more and more popular by overpassing traditional methods in numerous domains using machine learning techniques [9–12]. In this work, we would like to highlight the advantages of combining traditional and deep learning based approaches and show how they can benefit from each other when applied to comic book image understanding (see Figure 1). We present how deep learning can be beneficial for comic book image analysis, by applying several Convolutional Neural Network (CNN) models to extract each element contained by these types of images (e.g., panel, balloon, text, and comic character). We have identified three kinds of task for extracting comic elements. The first one is to detect objects such as panels and comic characters by determining the bounding box of the objects. While a bounding box may be sufficient to represent a panel/character position, balloons require a more precise boundaries description according to their more rounded shape. This task can be performed using deep learning segmentation models. The last element we would like to deal with is the text which relates to the detection and recognition tasks. We will profit from deep learning recognition models to also recognize text. By detecting panel positions and contents, we provide a more precise description of comic book images that can be indexed and retrieved by web crawlers and image search engines.

As mentioned above, comic book images are composed of different elements such as panels, balloons, texts, comic characters and their relations (e.g., read before, said by, thought by, and addressed to). The existing methods have addressed these elements separately or analyzed some elements together for a deeper understanding. However, none of them processes the comic books as a complete system, from elements extraction, relation analysis to encoding and indexing. In this paper, we propose an indexing system for digital comic book images which makes a processing pipeline from global images analysis to precise content extraction. Then, the Comic Book Markup Language (CBML) is used to encode and index the visual and textual content of comic book images for content-based retrieval systems such as search engines. We have experimented existing methods (both traditional and deep learning approaches) for basic elements composing comic book images. We also propose some improvement techniques for some tasks such as balloon segmentation or text recognition.

In summary, the main contributions of this article are:

- An indexing system from global image analysis to precise content extraction and encoding.
- Deep learning adaptation for comic elements (panels, faces, and characters) for the detection task.
- Improvement techniques for balloon segmentation and text recognition.
- Comparison of traditional computer vision techniques versus deep learning approaches with intensive experimentation.
- A new public dataset with manual annotation of spatial coordinates for panels and comic characters' bodies and faces.

Figure 1. In the left part, an example of recognized elements and its connections with the corresponding description file that we propose to generate. Recognized elements are highlighted in the image by a colored contour: red for the panel, cyan for the balloons, orange for the characters, yellow for the faces and magenta for the links between balloons and faces. In the right part, an extract of the corresponding description file following the Comic Book Markup Language (CBML) and encoding positions and relations of all recognized elements.

This paper is organized as follows. First, we review previous works about comic book image analysis and related approaches using deep learning in Section 2. An overview of the proposed system is given in Section 3. Then, we detail several techniques to extract panels, comic characters and faces using deep learning networks in Section 4 and to segment speech balloons in Section 5. Section 6 focuses on text recognition. Our experiment protocol is described in Section 7. The proposed approaches are evaluated on existing public datasets and a newly proposed dataset in Section 8. Finally, conclusions are given in Section 9.

2. Related Work

To access digital comics in an accurate and user-friendly experience on all mediums such as smartphones, tablets, 3D books and computer's screens, it is necessary to extract and identify comic book elements [13]. Accordingly, the relations between these elements could be investigated further to assist the understanding of the digital form of comic books by a computer. This strategy will help the user to retrieve information very precisely in the image corpus.

Comic images are mixed-content documents that are processed differently depending on the type of elements being studied. The techniques involved can vary a lot depending on whether the focus is given on panels, balloons, texts or comic characters. We review each of these in the next sub-sections.

2.1. Panels

Panel extraction and ordering have originally been studied for panel to panel reading [14]. The demand has been continuously increasing in parallel with the evolution of screen quality and size of mobile devices such as smartphones and tablets. Readers usually want to have their favorite comics or mangas on the go, while carrying minimum weight. Printed comics need to be scanned and split to fit the screen size and avoid zooming and scrolling if the screen is different from the paper size they were originally designed for.

Several techniques have been developed to automatically extract panels [15], assuming that the size of the panels can be reduced to be comfortably read on mobile devices. Most of them are based on white line cutting with Hough transform [16,17], recursive X-Y cut [18] or density gradient [19]. These methods do not consider empty area [15] and border-free panels. These issues have been corrected by connected component labeling approaches, but these approaches are sensitive to regions

that sometimes connect several panels, which potentially increases the detection error rate [20,21]. Another approach based on morphological analysis and region growing can remove such connecting elements but at the same time create new holes in the panel border [22]. After region segmentation, heuristic filtering is often applied to classify panel regions according to their size ratio with the page size [22,23].

More recently, new methods have shown interesting results for mangas and European comics with different background colors. They are based on watershed [24], line segmentation using Canny operator and polygon detection [17], and region of interest detection [25] such as corners and line segments. Complex manga layouts have also been considered [26]. In this approach, the authors use a recursive binary splitting strategy to partition the panel block into disjoint panel regions to find the optimal panel position. Recent methods (e.g., [27,28]) incorporate three types of visual patterns extracted from the comic image at different levels, and a tree conditional random field framework is used to label each visual pattern by modeling its contextual dependencies. An attempt has been done to model the contextual relationships among the visual patterns, instead of using empirical rules [29].

2.2. Balloons

Although speech balloons (or bubbles) are key elements in comics, they did not attract a lot of attention yet. However, they are the major link between graphic and textual elements and sometimes even part of the graphical style of the albums or series. They can have various shapes (e.g., oval, rectangular) and contours (e.g., smooth, wavy, spiky, or can even be partially or totally absent). Mainly speech balloons that are entirely surrounded by a black line (closed) have been studied, based on region detection and filtering rules [23,30]. In his thesis, Obispo [31] details several approaches to detect speech balloons mixing image processing, expert system, and machine learning. Liu et al. [32] proposed a clump splitting based speech balloon localization method which can detect both closed and unclosed speech balloons. In addition, Liu et al. [33] and Rigaud et al. [34] proposed approaches making use of text contained in speech balloons for detecting speech balloon contours at pixel level. At such level of precision, it is also possible to localize and find the direction of the tail. We proposed a method for this task in 2015 [5] and also to associate speech balloons and comic characters [35].

2.3. Text

Text extraction and text recognition have attracted a lot of attention in complex image analysis domains such as real scenes, checks, maps, floor plans and engineering drawings [36]. Few contributions are related to comics, probably because it is a niche. In early studies from the 2000s, a top-down approach which starts from speech balloon detection (white blobs) followed by mathematical morphology operations was proposed by Arai et al. [23]. Layout analysis was proposed by Yamada et al. [14]. First, bottom-up approaches based on binary segmentation, connected component extraction and labeling were proposed by Ponsard et al. [24]. Su et al. used Sliding Concentric Windows for text/graphic separation and then mathematical morphology and an SVM classifier to classify text from non-text components [37]. An adaptive binarisation process based on minimum connected component thresholding followed by a text/graphic separation based on contrast ratio and text line grouping was proposed by our team in 2013 [38]. In the same year, Li et al. proposed an unsupervised speech text localization for comics based on the training of a Bayesian classifier on aligned connected components and then detecting the rest of the text using the classifier for text/non-text separation [39]. More recently, we compared the performances of pre-trained OCR and segmentation-free approaches on a small sample of speech text from comic books written in Latin script [40].

2.4. Comic Characters

Among primary tasks of comics analysis, comic characters (protagonists) detection is one of the most challenging tasks because their appearance can change a lot from one comic book to another and even from one panel to another. Fortunately, they follow few conventions widely adopted by

comic book authors to avoid confusing the reader [41,42]. However, the authors are entirely free in the drawing of their comic characters. Comic characters detection is also different from human detection even through many comics are reproductions of human life situations. Comic characters are hand drawn and, therefore, there are much more variants regarding deformations, shapes, and appearances than real life humans [43–45]. Hence, human detection based methods cannot directly be applied to comics. This difficulty is one of the reasons why there are few works on comic character detection, including its variants such as comic character face detection or comic character retrieval.

In [46,47], the authors proved that Viola–Jones detection framework [48] is sufficient for detecting faces in mangas (Japanese comics). However, in [49], the authors showed that prior techniques for face detection and face recognition for real people's faces (including [48]) can hardly be applied to colored comics characters because comic character faces considerably differ from real people faces in respect of organ positions, sizes and color shades. They proposed another face detection method using skin color regions and edges. In [50], the authors profited from color attributes to boost object detection task, especially for comic character detection.

Another approach using graph theory was proposed by Ho et al. [51]. The authors detected the comic characters by representing each panel as an attributed adjacency graph in which color regions are used as nodes. The approach consists in finding redundant color structures to localize automatically the most frequent color group apparitions and label them as main characters. With a similar idea, the work in [52] uses SIFT descriptor with redundant information classification to also find the most repeated elements.

Some other works focus on character retrieval [53–55]. The work in [53] shows good results for character retrieval using local feature extraction and the approximate nearest neighbors (ANN) search. In [54], authors used Frequent Subgraph Mining (FSM) techniques for comic image browsing using query-by-example (QBE) model. In [55], authors proposed a manga-specific image-describing framework. It consists of efficient margin labeling, edge orientation histogram, feature description, and approximate nearest-neighbor search using product quantization. Their system retrieves comic characters using sketch-based query model.

2.5. Deep Learning Approaches for Comic Book Image Analysis

Deep learning models (or deep neural networks) are composed of multiple layers of nonlinear processing units which extract or transform features from the input data. The learning process may be supervised or unsupervised, with the layers forming a hierarchy from low-level to high-level feature representations [56]. Due to the recent trends of deep learning, there are many papers with major improvements in different domains such as speech recognition [57], natural language processing [58], computer vision [11], etc. We are interested in deep neural networks for computer vision and specifically for comic book image analysis.

Neural networks models are well developed and have significant achievements in most computer vision tasks such as image classification [3,10,59,60], object detection [10,11,61–63], object segmentation [12,64–66] and text recognition [67–71]. These achievements open new avenues to many domains which require the image analysis tasks. To our knowledge, starting from 2017, there are few works on comics analysis that have used deep learning [7,72,73].

2.6. Encoding and Indexing

To our knowledge, there has been little work on the conceptualization of comic language. In addition to the web schema proposal Periodical Comics [74] and the platform Grand Comics Database [75] which are both focused on the bibliographic organization of the work, only a few initiatives describing their content have been developed, but not systematically as academic publications. In 2001 an XML formalization was proposed, called ComicsML, allowing to describe the content of a board [76]. It has been developed in the hope of becoming a framework for the publication of then nascent webcomics. Indeed, the publication of a webcomic is often done directly on the web

page of the author and the frequency of update can be very heterogeneous depending on the series. The ambition of ComicsML was therefore to become a formalism that, if adopted by a large number of authors, would allow emerging new uses of reading webcomics, especially through tools exploiting the specificities of the language. ComicsML allows describing a series of webcomics published on the web, from its most general bibliographic aspects to the structure of the page: layout, type of bubbles or the form of the text used in a box. The syntax, based on tags from the XML language, implies a hierarchical approach in the description of elements composing the image. An approach sharing a number of common points was observed by Morozumi et al. [77] (applied to manga).

A second initiative, called Comic Book Markup Language (CBML), was launched a few years later by John A. Walsh [78]. In addition, based on XML, it is quite similar in its philosophy to the proposal of [76], although it is not reserved for Text Encoding Initiative P5: Guidelines for Electronic Text Encoding and Interchange [79] (TEI). The TEI is a framework, proposed by the consortium of the same name, for encoding illustrated textual documents, to facilitate the search for information in large databases of encoded works. The CBML, therefore, extends the vocabulary of the TEI to integrate the specific comics concepts (e.g., panel, balloon, who, etc.), while reusing, as much as possible, the existing encoding.

The different proposals find their respective applications in indexing, searching, and reading comics. They allow encoding the content of a comic book image explicitly at different degrees of granularity, depending on the task for which they are intended.

3. System Overview

In this paper, we propose an indexing system for digital comic book images which combines traditional methods and deep learning approaches. The system's objective is to make a processing pipeline of comic book page images, from global images analysis to precise content extraction. The latter is used to index the visual and textual content of comic book images for content-based retrieval systems such as search engines.

In the proposed architecture (Figure 2), the entire system consists of offline and online procedures that exchange information. In the offline procedure (blue blocks), we analyze and extract information from comic book images to produce a content description text file for each image. All analysis computations in this procedure can be processed offline over all comic books already present in the digital comics database, or on-demand to any new comic book image. The information we are referring to includes panels, balloons, text, faces, characters and their relations such as the reading order of panels/balloons and the association between speech balloons and comic characters. The results from the offline procedure are stored in a normalized markup language encoding the spatial coordinates in pixels of each element, text and also the relationships between those elements. The obtained description files are encoded using CBML which can later be, for instance, automatically parsed to produce sub-images of panels with associated textual description stored in text alternative image tags such as *alt* or *longdesc* on the web.

The online part (grey blocks) consists of engines such as search, inter-books navigation or intra-book navigation, based on the corresponding description files from the offline procedure. The user, in the online procedure, enters a query word and then the proposed system searches for the keyword in the recognized text or keywords of all panels to give the user the list of relevant panels/pages. The user can also navigate through the comic books by visualizing the panels, the balloons and the characters.

The online procedure can adopt state-of-the-art approaches coming from the information retrieval field which has a long history. However, the offline procedure which analyzes comic book images and extracts information demands some specific techniques from image processing, computer vision, and machine learning to deal with the specificities of comic book images which started to be extensively researched only few years ago. To achieve this goal, we investigated traditional approaches in computer vision and also recent approaches related to deep learning for computer vision. Our proposed

indexation/analysis system for comic book images performs seven major tasks, as illustrated in Figure 2.

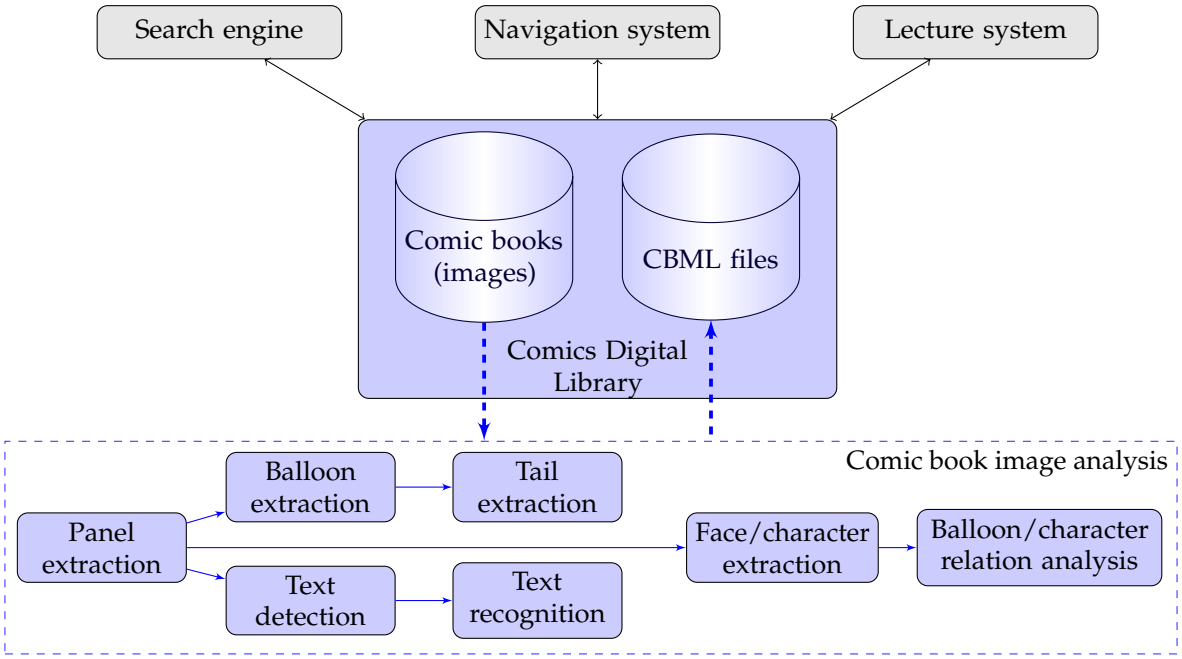


Figure 2. Illustration of the proposed architecture. Offline and online procedures are represented as blue and grey blocks, respectively.

Strategy Used to Build the Proposed System

The offline part of Figure 2 shows the complete indexing/analysis pipeline. In the first step, comic book images are processed and then panels are extracted. Panel extraction can be considered as an object detection process. In this work, we have compared the algorithms using image processing techniques with the state-of-the-art object detection methods using deep learning models which are trained on available datasets. In the next step, we have studied text detection/recognition, character/face detection, and balloons segmentation from detected panels in the first step. We use a state-of-the-art algorithm to detect (localize) texts in panels and then an incremental model based on a recurrent neural network with Long Short Term Memory (LSTM) is proposed to recognize texts. We also compare the traditional algorithms of balloon segmentation with some state-of-the-art segmentation deep learning models. Then we propose a simple technique to boost the performance of balloon segmentation by combining the two approaches. To detect comic characters and faces, we have performed empirical experimentation using state-of-the-art object detection models. Finally, the results from previous steps are used to extract the tail of balloon and to analyze the relationship between balloons and characters. In the last stage, all extracted information is stored in a description file (CBML) to support the online procedure (e.g., search engines). As all the best object detection deep learning models require a significant amount of labeled data for training, we introduce a new comic book image dataset including panels, characters, and faces positions.

4. Proposed Approach for Face, Character and Panel Detection

In this section, we propose a general deep learning based approach for extracting faces, comic characters, and panels. We suggest realizing the detection of the three elements separately. There are two reasons that the separated detection would be better. Firstly, the nature of face, character, and panel are different regarding shapes and drawing details. While faces are often small and homogeneous, characters are very diverse regarding their shape, size, texture, and colors. Panels are most of the

time similar to a rectangle and often clearly separated from the page background by a black stroke. Secondly, characters appear inside panels and faces appear inside characters (by considering a strict topological relation). All of them can be detected using object detection techniques.

We study the state-of-the-art deep architectures for object detection such as SSD, Faster R-CNN, and YOLOv2. In our work, we have applied the YOLOv2 model for face, comic character, and panel detection due to its state-of-the-art performance and because it is the fastest one in comparison to the other models. We have tried to apply the model to the comic images by using two techniques: anchor boxes learning and representation learning. In the next sub-sections, we present the YOLOv2 model and the two techniques.

4.1. YOLO Model

YOLOv2 [11] is the improved version of the model YOLOv1 [80] of the same author. While YOLOv1 predicts the coordinates of bounding boxes directly using fully connected layers on top of the convolutional feature extractor, YOLOv2 predicts bounding boxes using hand-picked priors as introduced in Faster-RCNN [61]. Instead of predicting offsets of bounding boxes as Faster-RCNN, the YOLOv2 model predicts location coordinates relative to the location of the grid cells as YOLOv1. Besides, the YOLOv1 model uses fully connected layers together with convolutional layers, while YOLOv2 uses only convolutional layers.

The generic model YOLOv2 is proved to have good results for generic datasets [11]. However, for the task of object detection in comic book images, there are some shortcomings related to the domain knowledge and the limited labeled data. We have customized the YOLOv2 model to take into account the characteristics of the comic domain: anchor boxes learning and representation learning as detailed below.

4.2. Anchor Boxes Learning for Comic Images

YOLOv2 model uses anchor boxes, a technique presented in Faster R-CNN [61]. Instead of predicting coordinates directly, Faster R-CNN predicts bounding boxes using hand-picked priors [61]. Hand-picked priors are size pre-defined anchor boxes. The network starts the prediction from the anchor boxes and learns to adjust the coordinates to move and resize the anchor boxes to the right position. However, anchor boxes in Faster R-CNN are chosen for general use (not database specific), and anchor boxes in YOLOv2 are learned from VOC [81] and COCO [82] datasets. These anchor boxes have not been designed for face, panel and character detection in comic book images. It makes the network harder to predict good detections because the size and aspect ratio (width/height) of faces, panels or characters in comic book images are different from general objects in VOC [81] or COCO [82] datasets. For example, if the anchor boxes are chosen very big, it is difficult for the network to detect small objects; especially faces which have the width often smaller than the height. Thus, we should choose anchor boxes appropriately. Using the same idea in YOLOv2, we find the right anchor boxes for a comic book dataset by using k-means clustering on the bounding boxes of the ground-truth. We apply this clustering to find one set of anchor boxes for each element: face, character, and panel on the DCM772 dataset (see Section 7.1.3). In Figure 3, we show the anchor boxes for characters on the DCM772 dataset. In our experiments, we have used $k = 5$ in the k-means clustering algorithm to learn five anchor boxes. The five anchor boxes represent the whole objects bounding boxes in the dataset. We use the average Intersection Over Union (IOU) metric to evaluate how well the anchor boxes represent the objects. The best average IOU of 100% is obtained if and only if any bounding box has the same size as one of the five anchor boxes. The average IOU is computed by the following formula:

$$averageIOU = \sum_{bi \in B_{gt}} \frac{area(a_i \cap b_i)}{area(a_i \cup b_i)} \quad (1)$$

where B_{gt} represents the set of all bounding boxes in the ground-truth and a_i is the closest anchor box for bi.

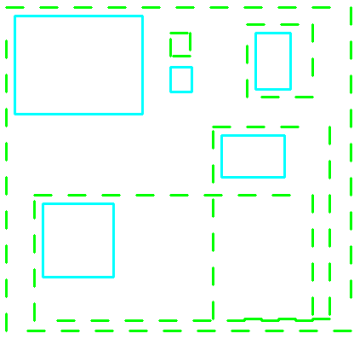


Figure 3. An example of learned anchor boxes: the anchor boxes for characters on DCM772 dataset (blue solid) and the original anchor boxes (green dashed) of YOLOv2 for objects on VOC dataset [81]. We can see that the boxes are very different in the two cases. Note that the spatial organization in this figure has no particular meaning, it is only for enhancing the visualization. The average IOU (see Equation (1)) of the ground-truth to closest prior of new anchor boxes and the original anchor boxes in YOLOv2 are, respectively, 67.62% and 51.13%, which shows that new anchor boxes provide a better representation of characters.

4.3. Representation Learning

One of the problems we have when working on the comic page analysis is that the number of labeled data is limited. It is hard to have significant labeled training data because the cost in human resources is very high. This limitation usually leads to the lower performance of the detector or classifier even if we have a robust learning algorithm/model. In a convolutional neural network such as YOLOv1, the first set of convolutional layers aims at learning the representations of the dataset images for a target task, then the last fully connected layer set is a classifier or detector. If the number of training data is limited, convolutional layers cannot learn good representations for the target task.

Some techniques can tackle this problem, for example, "unsupervised representation learning" and "transfer representation learning", which can learn good representations from unlabeled data. The unsupervised representation learning profits from massive amounts of unlabeled data to find the relation between input data and output by using models such as Stacked Autoencoders [83], Deep Belief Network [84] and Deep Boltzmann Machine [85]. In the transfer representation learning, a simpler supervised or unsupervised task with more raw data is used to find good representations of the data, then these representations are fine-tuned to deal with the more difficult task using fewer labeled data. While unsupervised representation learning has been largely abandoned except in the field of natural language processing because of its unstable advantage [9], transfer representation learning for convolutional networks is popular [86]. In this section, we present an approach for transfer learning with comic book image datasets.

In transfer learning, we have to perform two different tasks (T_1 and T_2) with the assumption that many of the factors explaining the variations in T_1 are relevant to the variations that need to be captured for learning in T_2 [9]. If there are significantly more data in the first task (sampled from T_1), then that may help to learn representations which are useful to generalize quickly from only a few examples drawn from T_2. The assumption is assured with the fact that many visual categories share low-level notions of edges and visual shapes, the effects of geometric changes, changes in lighting, etc.

In our context, we have a massive amount of unlabeled data and a tiny amount of labeled data for the task of face, character and panel detection in comic book pages. If our learning model can exploit this unlabeled data effectively, then we might be able to achieve better performance for the detection task. Fortunately, all comic images are organized by albums/authors. Thanks to this, we can formalize a classification task to classify any comic image corresponding to an album or an author. We have learned the representations via a deep neural network based on the architecture Darknet-19

used in YOLOv2. The learned weights are then fed into YOLOv2 to train the face, panel and character detection tasks.

5. Speech Balloon Segmentation

While a bounding box may be sufficient to represent a panel/character position, speech balloons that are fundamental elements in comic books, require a more precise boundary description according to their more rounded shape. Among existing models in deep neural networks, the segmentation model is the closest related task to the balloon extraction. In this section, we investigate the state-of-the-art deep learning segmentation models to extract balloons in comic book images and discuss the pros and cons at the end of this section.

In our works, we have experimented with the balloon segmentation using the segmentation model DeepLabv2 [65]. In this work, similar to [87], the authors combine advantages of a CNN and a Conditional Random Fields (CRF). The authors have introduced two techniques to improve the segmentation performance: atrous (or dilated) convolutions and multiscale processing. The atrous convolutions help to achieve a wide receptive field without coarsening spatial dimensions. The multiscale processing helps robustly to capture objects as well as image context at multiple scales. In the end, the structured prediction is done by fully connected CRF.

The details about the experimentation of deep learning segmentation models trained on eBDtheque dataset are discussed in Section 7.

5.1. Refinement/Cross-Validation with Other Balloon Approach

In Section 7, we have observed that traditional approaches [31–34] detect many false positive examples. However, although the deep learning model can localize very well the balloons, it cannot identify precisely the boundaries of the balloons as the traditional approaches. Unlike other detection tasks where we only need to detect a rectangle (a bounding box), balloon segmentation requires an algorithm able to detect the boundaries correctly. To boost the performance of balloon extraction, we propose to combine the deep learning approach and a traditional algorithm [34].

We will use the results of the deep learning segmentation model to remove the FP (False Positive) examples of [34]. This simple technique can also benefit the high boundaries precision of the traditional algorithm and the detected open balloons of the deep learning model to give the best final results (see detail evaluations in Section 7).

Let D^a and D^b denote the regions of the balloons detected by [34] and by the deep learning model, respectively. The final regions D^f will be computed by the following algorithm 1.

$$[H]IOU(a,b) = \frac{area(a \cap b)}{area(a \cup b)} \tag{2}$$

where $a \cap b$ denotes the intersection of the the two regions and $a \cup b$ their union.

5.2. Tail Detection and Association to Characters

As presented in Section 2.2, there are few papers in the literature on this topic and already published methods give good results on most balloon types (with a tail that extends from the balloon background). We propose to use our previously published methods for detecting the tail position along the edge of the speech balloons, which determines its direction and associates it with a speaker (comic character) [5,35].

In [5], tail detection is performed on the analysis of convexity defects of the convex hull of the speech balloon. Once we find the tail position, we compute the orientation and direction of the last part of the tail which is directed towards the speaking character. The association with the speaker is established using our method [35] which performed well on 93.32% of the tails from the eBDtheque dataset [88]. In this approach, we build a geometric graph in a Euclidean plane within the panel where

vertices are spatial positions of tail and comic character body (or faces if detected) centroids. Edges are straight-lines segments (associations). We formulated an optimization problem where we search for the best pairs (2-tuples) of tail and face or body corresponding to associations in the story. The main optimization criterion is the Euclidean distance between each entry of the 2-tuples.

Algorithm 1 Balloons refinement

Input: *two sets of detected balloons* D^a *and* D^b *and the threshold* α *(in our experiment* $\alpha = 0.5$*)*
Output: D^f *final set of detected balloons*
Note: *the* IOU *(Intersection Over Union) is computed by Equation (2) below.*
Begin
$D^f = [\,]$
Step 1: keep detections from [34] if there are detections by the deep learning model at the same positions
For d_i^a in D^a
    For d_j^b in D^b
        If $IOU(d_i^a, d_j^b) \geq \alpha$
            $D^f \leftarrow d_i^a$

Step 2: keep detections from the deep learning model if there are no detections by [34] at the same positions
For d_j^b in D^b
    For d_i^a in D^a
        If $IOU(d_i^a, d_j^b) < \alpha$
            $D^f \leftarrow d_j^b$
End

6. Text Recognition

In this section, we extend a text recognition approach that we have proposed in the first attempt on a small subset [40]. This approach has shown promising preliminary text recognition results on comics material and it is adaptive to handwriting styles, which is a strong advantage in our case compared to other approaches from the literature (see Section 2.3). This approach is based on an automatic neural network training for specific handwritten style text recognition.

Similar to Section 4, in our context, we have many unlabeled data from the text line extraction algorithm and few or no labeled data for the task of text recognition. In [40], we designed a method which is able to automatically learn a writing style given several unannotated images from the same scriptwriter (the person who writes text into speech balloons). We call this approach segmentation-free in the sense of fully automatic pseudo ground truth generation and training (no manual image crop or annotation is needed). The approach consists of three steps after having extracted text lines from comic book images using any text extraction algorithm (Section 6.1). First, we apply one or several standard OCR systems (pre-trained OCR) to recognize all extracted text lines images (Section 6.2). Then, we check the quality of each recognized text line using a "lexicality" measure (Section 6.3). Note that this measure works best with a lexicon corresponding to the language of the analyzed text. In general, the overall quality of the recognized text is quite low at this stage because comic book writing styles are quite different from the generic fonts that are generally used to train such standard OCR.

Finally, the recognized text lines are used as input for training a second OCR system from scratch. This last step produces a style-specific recognition system which does not require manual segmentation for training (see Section 6.4). This second OCR system is used to recognized (again) all the extracted text lines from the related album (taking the pre-trained OCR output as pseudo ground truth for the subsequent training of style-specific OCR) (see Figure 4). This new model is specially trained for a

specific writing style, and it can improve overall text recognition if the pre-trained OCR outputs feed it with enough training samples (see Section 7).

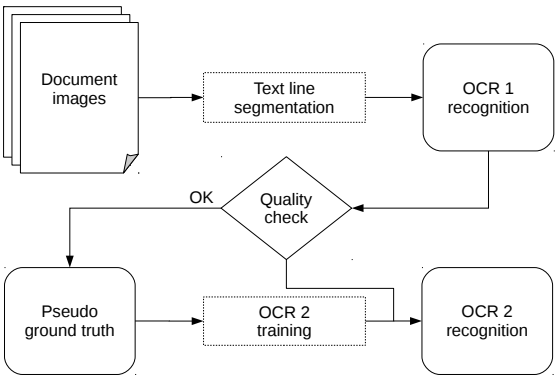


Figure 4. The complete pipeline of the segmentation-free OCR system. The first row represents traditional text recognition sequence. In the middle block, text lines are quality checked, and the good ones (text line images and associated transcriptions) are used as pseudo ground truth for training the OCR 2. Final OCR output is produced by OCR 2 using its automatically trained model.

6.1. Text Localization

At this stage, any speech text extractor can be used, but text recognition performance highly relies on its performance. In this paper, we use a state of the art algorithm that reaches 75.8% recall and 76.2% precision for text line localization on eBDtheque dataset [38].

6.2. Pre-Trained OCR Systems

The proposed approach requires at least one pre-trained OCR system able to recognize some text lines with good accuracy in order to feed the learning stage of a second OCR system which is expected to recognize much more text lines than the first OCR. Tesseract and FineReader are the two most popular OCR systems presently available. They both come with pre-trained data for several languages. Tesseract is considered one of the most accurate free open-source OCR engines [89]. It is open-source software that can be easily integrated into research experiments, which is the main reason we chose to use it. FineReader OCR engine is a well-known commercial OCR system as well. We do not use it in this experiment, but it can be added easily as an additional pre-trained OCR.

In this approach, the number of pre-trained OCR is not limited. Several pre-trained OCR systems can be used, and their best results can be combined for each text line. Pre-trained OCR output will feed the second OCR system with as much correct text lines as possible in order to improve training and recognition quality by this last OCR output.

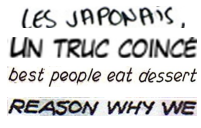
6.3. The Lexicality Measure

As presented, we need to check the quality of each recognized text line using a "lexicality" measure. We use a readily observable quantity that correlates well with true accuracy, giving us a ranking scale which allows selecting the transcription given by the best performing model automatically. For this purpose, we chose to use the *mean token lexicality L*, a distance measure of OCR tokens (words) which can be calculated for any OCR engine [70]. The original paper also mentions the *mean character confidence C* but because it is OCR specific we do not use it. The lexicality measure calculates, for each OCR token, the minimum edit distance (Levenshtein distance) to its most probable lexical equivalent from a lexicon of the corresponding language. The sum of these Levenshtein distances over all tokens is, therefore, a (statistical) measure for the OCR errors of the text, and the lexicality defined as $L = (1 - \textit{mean Levenshtein distance per character})$ is a measure for accuracy. Problems with this

measure arise from lexical gaps (mostly proper names) and very garbled tokens (e.g., short text lines such as sequences of single letters, or too long because of merged tokens with unrecognized whitespace). These issues are not restrictive in our case because we do not need such garbled tokens to build a good pseudo ground truth; they could even bias it if they are not so frequent.

To pass the quality check step, a text line should be composed of at least two words and have a lexicality $L = 1.0$. This means that it is composed only of words that are part of the lexicon (see Section 7.2.4). Examples of lexicality measure are given in Table 1. In the first row, the image where it is written "LES JAPONAIS" has been mistakenly transcribed with the nine following characters "Les mmmsc", (ignoring spaces).The first three characters composing the word "Les" is part of the French lexicon (Levenshtein distance equal to 0) but the second word is not a French word. The "closest" word from the lexicon is "immiscé" which is at a Levenshtein distance of 3 from "mmmsc" (number of characters that are different or at a different place). The lexicality $L = 1 - (0 + 3/9) = 0.67$.

Table 1. Examples of lexicality measure. Image credits: eBDtheque dataset [88] and public domain.

Image	OCR Output	Lex.
	Les mmmsc,	67%
	UN TRUC COINCÉ	91%
	best" people eat dessert	95%
	REASON WHY WE	100%

6.4. Segmentation-Free Training

As introduced in Section 1, handwritten texts are very challenging for OCR systems. They require many annotated data of each font to train their recognition model to be able to recognize text from similar fonts accurately. In fact, it is not feasible to annotate all scriptwriter styles as they are continuously trying to be different from others (new authors are making comics everyday). Instead of annotating a massive amount of handwritten styles and trying to build a generic handwritten OCR system, we propose to automatically train a specific OCR for each writing style (single scriptwriter). This approach has the advantage of minimizing confusion between visual similarities (e.g., letter "i" from a given scriptwriter may be similar to letter "l" from another scriptwriter). The idea is to use, for each writing style, correct pre-trained OCR outputs to train OCRopus algorithm and then recognize all text from the same writing style using the newly trained model.

This approach removes image annotation time (groundtruthing) but may introduce false negative and false positive text lines. False negatives are not important in our study because they will not decrease the quality of the ground truth; they just ignore some text lines from the story. However, false positives may bias the ground truth, so they must be ignored. To detect and ignore false positives, we filter pre-trained OCR output using a lexicality measure as mentioned in the previous section.

7. Evaluation Protocol

In this section, we present the proposed dataset and other available datasets used in the first part of the experiments. In the second part, we describe the data selection and the evaluation measure used for each step: panel detection, face and character detection, balloon segmentation and text recognition.

7.1. Datasets

We compare classical and deep learning approaches on several annotated datasets that provide all or partial region locations. The first dataset is the public *eBDtheque* dataset. The second dataset that we call *Fahad18* has been used and obtained from the authors of [50]. Finally, we propose a new annotated dataset that we call *DCM772* and a raw dataset called *DCM_raw*. The community can easily extend the two new proposed datasets because it is based on public domain American comic book images.

7.1.1. eBDtheque

This dataset is the last version of the public eBDtheque dataset [88] which includes comic character locations. This dataset is composed of one hundred comic book images containing 850 panels, 1550 comics characters, 1092 balloons and 4691 text lines in total. Note that not all characters were annotated in this dataset: only the ones speaking at least one speech balloon in the album are annotated with the coordinates of their horizontal bounding boxes. This selection aimed to retain only the main characters who had a direct influence on the story and ignore the secondary characters. Certain parts of the characters (e.g., hand and foot) were ignored in certain postures to maximize the area occupied by the characters and minimize the background information in their bounding box. More description can be found in the original paper [88].

7.1.2. Fahad18

The Fahad18 dataset is a collection of 586 images of 18 favorite cartoon characters obtained from Google Images by the authors [50]. There are 18 cartoon characters in this dataset: Bart, Homer, Marge, and Lisa (The Simpsons); Fred and Barney (the Flintstones); and Tom, Jerry, Sylvester, Tweety, Bugs, Daffy, Scooby, Shaggy, Roadrunner, Coyote, Donald Duck and Mickey Mouse. In the whole dataset, the number of images for each character is ranging from 28 (Marge) to 85 (Tom). It is necessary to note that an image may contain more than one character. See more description in the original paper [50].

7.1.3. DCM772

For our experiments, we built a dataset called *DCM772* which is composed of 772 images from 27 golden age comic books. We freely collected them from the free public domain collection of comic books: Digital Comic Museum (DCM) (http://digitalcomicmuseum.com). We selected one album per publisher to get as many different styles as possible. We made ground-truth bounding boxes of all panels, all characters (body + faces), small or big, human-like or animal-like, speaking or not. Images, annotations, and ground-truth tool are freely available here (https://git.univ-lr.fr/crigau02/dcm_dataset) (final version only, please send pull request) to allow interested people to reproduce results or extend the dataset. The two image lists for the training set and the testing set are also publicly available.

To create this ground-truth, we have identified four different types of characters that we classified into: human-like, object-like, animal-like and extra (supporting role characters). Human-like are characters that look like humans, such as Spiderman, Batman, etc. Object-like characters are the ones that are similar to objects such as Sponge Bob, Cars, etc. Animal-like could be Garfield, the Pink panther, etc. The extras are characters from any of the classes mentioned earlier but not easily distinguishable or in the shadow. Note that faces have been annotated (when visible) only for human-like class. An example of each class is given in Figure 5. Panels and faces have been annotated with horizontal bounding boxes. Faces are defined as eyebrows, eyes, nose, mouth and chin and ears if visible, similar to other common datasets from the domain [90] (see Figure 6).

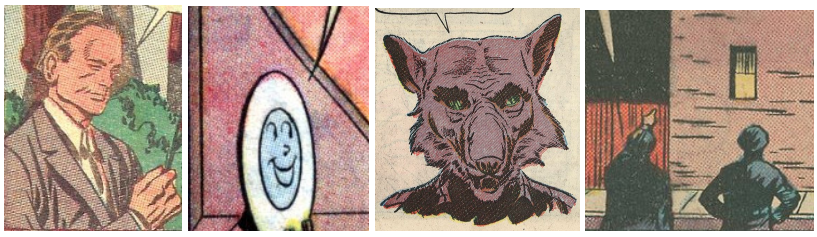


Figure 5. Examples of each annotated character class in the dataset DCM772. From left to right: human-like, object-like, animal-like and extra.

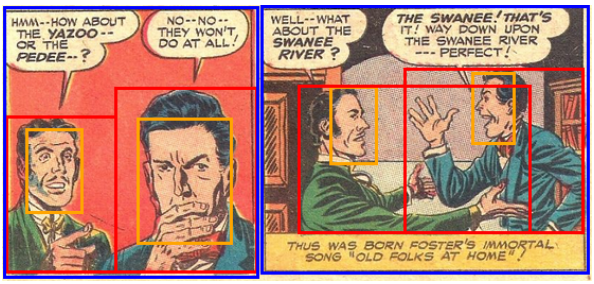


Figure 6. Examples of annotated bounding boxes for panel (blue), character (red) and face (yellow). Better viewed in color.

7.1.4. DCM_raw Dataset

In Section 4.3, we have proposed to do the representation learning on a raw dataset to boost the performance of the detection task. In our experiments, we would like to learn the representations on images similar to the DCM772 dataset. To do so, we have collected 70 other comic books from the Digital Comic Museum which are different from the comic books in DCM772. This dataset contains 3188 raw comic images and is divided into two sets: training set of 2870 comic images and testing set of 318 comic images. The comic books list and its training and testing sets are freely available in the same Git repository as DCM772 dataset (see Section 7.1.3).

7.2. Evaluation Measure

This section presents different evaluation measures we used to evaluate the performance of the proposed methods for face, character, panel and text analysis.

7.2.1. Character and Face Detection

To evaluate detection performance, we follow the Pascal VOC evaluation criteria [81], a well-known benchmark for object detection in computer vision. We report the interpolated average precision (AP%). For the detection task, we need to judge if a detected bounding box is true or false, compared to its corresponding ground-truth bounding box overlap.

According to Everingham et al. [81], a correct detection should have the overlap ratio $ao > 50\%$, between the predicted bounding box B_p and ground-truth bounding box B_{gt}. See Formula (3).

$$ao = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})} \tag{3}$$

where $B_p \cap B_{gt}$ denotes the intersection of the predicted region and ground-truth bounding boxes and $B_p \cup B_{gt}$ their union.

Firstly, we compare the proposed detection model with the method in [50] on the Fahad18 dataset. We follow the same setting as described in the original paper [50]. This dataset is divided into two sets. A training set of 304 comic images and a testing set of 182 comic images. To evaluate detection performance, the authors also follow the PASCAL VOC evaluation criteria [81].

Secondly, we test the proposed model on the proposed dataset DCM772 (see Section 7.1.3). We trained the face detector and the character detector for comic "panels" on the DCM772 dataset. The dataset is divided randomly into three sets: a train set containing 3500 panels; a valid set containing 406 panels; and a test set containing 433 panels. We also trained the face detector and the character detector for comic "pages" on the DCM772 dataset. The dataset is divided randomly into three sets: a training set containing 650 pages; 50 pages for validation set; and a testing set containing 72 pages.

We have experimented on the detection model with several settings: with/without anchors learned, with/without a pre-weights (see Sections 4.2 and 4.3). We have used two different pre-weights, the standard pre-weights trained from PASCAL VOC dataset for classification task and the pre-weights

from the DCM_raw dataset. To learn the pre-weights for face, character detection tasks, we have trained the Darknet-19 model with raw data from the DCM_raw dataset for a classification task. The classifier is trained from 2870 images of the DCM_raw training set with 200 epochs, and we achieve the result of Top-1 is 96.07% and Top-5 100% for the classification task. Top-1 score means the classifier's top prediction (highest confidence) is correct. Top-5 score means the correct class is in the classifier's Top 5 predictions. We hope that the representations learned from many non-labeled data (DCM_raw dataset) can help the detection tasks with limited labeled data from the DCM772 dataset.

7.2.2. Panel Detection

The panel detection task is different from character detection task. There are not many inaccuracies in bounding boxes in the ground-truth data. That is the reason we do not follow the evaluation measure [81] as described in the previous section. We compute the precision/recall of the panel detection by using the IOU metric of the detection boxes with the ground-truth boxes. Note that multiple detections of the same panel in an image were considered as false detections.

We train the panel detector for comic pages on the DCM772 dataset. The DCM772 dataset is divided randomly into three sets as in the case of face/character detector for comic pages mentioned in the previous subsection. The trained detection model is used for detecting panels on the DCM772 testing set and the eBDtheque dataset. We do not train a detection model using eBDtheque because the number of examples in eBDtheque is relatively small (100 images with 850 panels in total). We evaluate the proposed model on the eBDtheque and the DCM772 datasets to compare with the works in [6,27].

7.2.3. Balloon Segmentation

The balloon segmentation classifies each pixel to either balloon class or "background" class. To evaluate the balloon segmentation performance, we follow the Pascal VOC evaluation criteria [81] for the segmentation task:

$$seg.\,accuracy = \frac{true\ pos.}{true\ pos. + false\ pos. + false\ neg.} \tag{4}$$

where $true\ pos$, $false\ pos$, $false\ neg$, are the result of pixel classification. Pixels marked as "background" in the ground truth are excluded from this measure. We can note that this measure gives the same result as the IOU metric used in panel detection (Section 7.2.2). We also provide the F1 score for balloon segmentation. We have experimented with the balloon segmentation on the eBDtheque dataset. This is the only dataset which contains ground-truth of balloon positions to our knowledge. We divided this dataset into two sets: a training set of 90 pages and a testing set of 10 pages.

7.2.4. Text Recognition

Concerning the evaluation of text recognition, we rely on standard metrics of speech recognition such as Character Error Rate (CER) and Word Error Rate (WER) for determining the recognition accuracy of the segmentation free OCR output [91]. Note that we measure the quality of the generated pseudo ground-truth only by counting the number of validated text lines because, in our experiment, these data are automatically generated (no manual ground-truth available).

Pseudo Ground-Truth Generation

To generate the pseudo ground-truth for the segmentation-free OCR, we used only one pre-trained OCR (Tesseract version 3.04) to ease the comparison. However, several OCR systems can be used in parallel, and only the best output should be used as pseudo ground-truth (see details in Section 6).

Lexicon

We selected complete lexicons for each language containing a list of flexed forms for measuring the *lexicality* of each text line. For French, we used the "Dicollecte lexicon" version 6.1 (used in LibreOffice and Firefox). For English, we used the "Dallas lexicon" from the SIL International Linguistics Department which contains inflected forms, such as plural nouns and the -s, -ed and -ing forms of verbs. The lexicons contain about 500,000 and 110,000 entries, respectively.

Dataset Selection

For the evaluation of text recognition performance, we selected 11 albums from the eBDtheque dataset and 20 from DCM772 dataset according to the following criteria.

From eBDtheque dataset, we selected page images from albums for which we could find other page images of the same album digitized under the same conditions. We ended up with a selection of 53 of 100 available images (11 of 20 albums) from the eBDtheque dataset. From the selected albums, eight were in French and three in English.

From the DCM772 dataset, we selected all albums with more than 10 page images, and we annotated two pages per album for the testing set (5 albums did not meet this criterion). We ended up with 20 albums where we removed front, back and advertising pages because these pages often use typewritten fonts that are really different from speech text handwritten writing styles and therefore could bias the learning stage of the handwritten speech text. All albums from this dataset were uppercase golden age American comics handwritten in English. We identified them as Album 12 to 32. We manually annotated (rectangular text line clipping and transcription) the two first pages of each album and trained our algorithm on the rest of the unannotated images for each album.

In total, we ended up with 95 and 1122 page images for testing and training sets, respectively. An example of text lines from some albums from both datasets is given in Table 2. The exhaustive list of selected pages, page numbers, generated pseudo ground-truth and learned models is available in the dataset Git repository (see Section 7.1.3).

Table 2. Image examples for some albums from eBDtheque and DCM772 datasets. OCR transcriptions are written between double quotes below text line images for the two OCR systems (Tesseract and OCRopus), with corresponding Character Error Rate (CER). OCRopus transcriptions are given for the best trained model between 10,000 and 50,000 iterations. Image credits: eBDtheque dataset [88] and public domain.

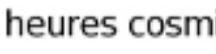
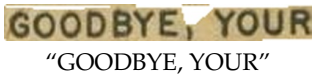
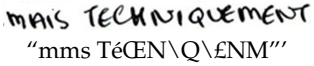
OCR/im.	Image/Transcription	CER (%)	OCR/im.	Image/Transcription	CER
Album 2			Album 12		
Tess.	"heures cusmuques,"	11.76	Tess.	"AND FLOWERS ./"	15.38
OCRop.	"neures cosmlques."	17.64	OCRop.	"AND FLOWERS /"	7.69
Album 4			Album 16		
Tess.	"TDWRRD THE GRTE."'	31.25	Tess.	"TIME TO FIND"	0.00
OCRop.	"TOWTRD THE CUT !"	25.00	OCRop.	"TIME TO FIND"	0.00
Album 5			Album 20		
Tess.	"GOODBYE, YOUR"	0.00	Tess.	"HAVE TO GEM/ND HE"	23.53
OCRop.	"GOOOBYTE. YOUR"	27.08	OCRop.	"HAVE TO GROED ME"	11.76
Album 7			Album 24		
Tess.	"UN NOEIL OUI DIT"	6.25	Tess.	"To THEIR PLOTS!"	6.67
OCRop.	"UN NOEIL QUI DIT"	0.00	OCRop.	"To THEIR PLOTS!"	6.67

Table 2. *Cont.*

OCR/im.	Image/Transcription	CER (%)	OCR/im.	Image/Transcription	CER
Album 8			Album 28		
Tess.	"coLLÈ6-Œ DE LA"	35.71	Tess.	"SAY SUCH A THING?"	0.00
OCRop.	"bULEUE D E"	57.14	OCRop.	"SAY SUCHATHING?"	11.76
Album 9			Album 32		
Tess.	"mms TéŒN\Q\£NM"'	72.22	Tess.	"HALF STARV: D. "E's"	33.33
OCRop.	"asEEa L m Nw"	88.89	OCRop.	"HALF STARVGD. HES"	16.67

8. Results

In this section, we present and analyze the results we have obtained for character, panel, face detection, balloon segmentation and text recognition. In addition, we detail the output indexing format.

8.1. Character Detection

In this section, we prove the effectiveness of the deep learning approach by finding answers to the two following questions: (1) Does it work? (2) Does it generalize well? We have compared the proposed model with the work in [50] on the Fahad18 dataset. Then, we have experimented with the approach on two other datasets: eBDtheque and DCM772.

8.1.1. Does the CNN Detection Model Work?

Character Detection on Fahad18 Dataset

In [50], the authors used color attributes as an explicit color representation for object detection. They proved that their method is most effective for comic characters in which color plays a pivotal role. To compare with the work in [50], we used strictly the same setting as described in Section 7.2.1 to train and evaluate our model on this dataset. Table 3 shows results on the dataset of our model and [50]. The Fahad18 dataset contains 586 images of 18 classes. The AP% for 18 classes are shown with the mean AP% over all classes in the last column. Note that the proposed approach outperforms the method presented in [50] with the detection and recognition of 14/18 classes and it gives a significant improvement for the mean AP about 18.1%. The proposed approach shows higher performance than [50]. However, there are 4/18 classes where the method in [50] gives better results than our approach: Bart, Marge, Lisa, and Barney. It is interesting to note that 3/4 of these classes come from the same comic book: The Simpsons. This lower AP% may originate from the fact that all characters of The Simpsons have a small number of training examples. Table 4 shows the numbers of examples in the training set for the 18 classes. We observed that these four classes are in the last five in terms of training instances (the five classes which have the least instances in the training set are: Bart, Homer, Marge, Lisa, and Barney). This is a potential indicator that the deep learning approach may be more sensitive to the number of class instances in the training set than other approaches.

Table 3. The mAP results on Fahad18 dataset.

Method	Bart	Homer	Marge	Lisa	Fred	Barney	Tom	Jerry	Sylve	Tweety
[50]	**72.3**	40.4	**43.4**	**89.8**	72.8	**55.1**	32.8	52.3	32.9	51.4
CNN model	63.6	**60.6**	41.7	65.6	**78.5**	54.5	**84.5**	**59.1**	**54.5**	**56.1**
Buggs	**Daffy**	**Scooby**	**Shaggy**	**Runner**	**Coyote**	**Donald**	**Micky**	**MeanAP**		
22.2	35.6	19.8	25.2	21.9	10.0	27.9	45.3	41.7		
54.5	**60.3**	**65.4**	**61.4**	**60.5**	**63.1**	**42.4**	**59.3**	**59.8**		

Table 4. Number of examples in training set for 18 classes.

	Bart	**Homer**	**Marge**	**Lisa**	**Fred**	**Barney**	**Tom**	**Jerry**	**Sylve**
Frequency in train set	**19**	**17**	**14**	**16**	30	**15**	43	42	28
	Tweety	**Buggs**	**Daffy**	**Scooby**	**Shaggy**	**Runner**	**Coyote**	**Donald**	**Micky**
Frequency in train set	21	62	23	31	26	21	26	25	20

Character Detection from Pages on DCM772 Dataset

In Table 5, we present a summary of results of the character detection on comic pages. We can see that the detection results are generally better than the character detection on the Fahad18 dataset (see Table 4). Compared to the model trained from scratch (57.32% mAP), the anchor boxes learned from the DCM772 dataset give slightly better results (58.24% mAP) while using the pre-weights trained from Pascal VOC dataset or pre-weights trained from DCM_raw dataset does not provide any benefits (57.20% mAP). The images from Pascal VOC dataset are not similar to comic pages but similar to panels of comic pages regarding size, style and structure. That why the pre-weights does not work well for comic pages. Besides, the DCM_raw pre-weights are learned from the classification task. This task is too simple for the neural network to find good presentations of the DCM_raw dataset which are expected to be useful for the detection task. We have obtained a result of classification task of 96.07% for Top-1 measure and 100% for Top-5 measure. Another task such as a panel detection task (close to the character detection) should learn better representations, which are useful for character detection. We can do this by first applying an existing panel detection algorithm and keep panels with high confidence as pseudo ground-truth.

Table 5. A summary of results for character detection from pages on the DCM772 testing set. The model is trained on the DCM772 pages training set.

pre-weights from Pascal VOC	no	yes	no	no
pre-weights from DCM_raw	no	no	yes	yes
Anchor boxes learned from DCM772	no	no	no	yes
mAP (%)	57.32	57.17	57.20	58.24

Character Detection from Panels on DCM772 Dataset

We trained a character detector using a train set containing 3500 panels and a valid set containing 406 panels. The detector is tested on a testing set containing 433 panels. In Table 6, we present a summary of results of the character detection on comic panels. In contrast with the character detection on pages, we can train a better detector for characters on panels. The important reason for this better performance is that the size of characters is relatively bigger on panels than on pages, since the input image is resized to match with the input layer of the network. Compared to the model trained from scratch (65.34% mAp), we can see that the model is significantly better if it is trained with pre-weights from Pascal VOC dataset (74.38% mAp) and the learned anchors (76.76% mAp). However, we have poor results when using pre-weights trained from the classification task on the DCM_raw dataset. We have discovered that the classification task learned representations for the DCM_raw dataset from "pages", which have no relation with good representations for "panels" in this case. This is why the detector trained from the pre-weights has that low mAP.

We have evaluated the CNN detection model on the Fahad18 dataset and the DCM772 dataset. Based on the results, we can conclude that the CNN model works well for character detection task in comics. In this section, we had a small improvement by learning the anchor boxes from the target dataset. The transfer representation learning does not help the task in our experiments.

Table 6. A summary of results for character detection from panels on the DCM772 testing set. The model is trained on the DCM772 panels training set.

pre-weights from Pascal VOC	no	yes	no	yes
pre-weights from DCM_raw	no	no	yes	no
Anchor boxes learned from DCM772	no	no	no	yes
mAP (%)	65.34	74.38	53.26	76.76

8.1.2. Does CNN Detection Model Generalize Well?

In the comic image analysis domain, we can have different datasets with very different drawing styles, color schemes and structures. That is why we would like to know if the CNN detection model can learn robust representations of comic characters, faces, and panels. We can use the model trained from one dataset to detect objects in a different dataset. We have tested our learned detection model from the DCM772 dataset on the eBDtheque dataset and the testing set of the Fahad18 dataset (see Table 7). For the Fahad18 testing set, we have achieved the mAP of 47.34% which is not as good as the model trained on Fahad18 train set (59.8%), but it is still better than the original method in [50] (41.7%). While both datasets have different styles of images with DCM772, the eBDtheque images are more similar to DCM772 than the Fahad18 as the Fahad18 images come from cartoons and images in eBDtheque are from comics such as DCM772. That is why we have achieved a better accuracy for the eBDtheque dataset (58.36% mAP).

Table 7. The model trained on the DCM772 dataset generalizes well on other datasets such as Fahad18 or eBDtheque.

Dataset	mAP (%)
eBDtheque	58.36
Fahad18	47.34

8.2. Face Detection

In this section, we present the results of face detection on comic panels and compare with the character detection on panels. Then we, once again, answer the two following questions: (1) Does it work? (2) Does it generalize well?

8.2.1. Does CNN Detection Model Work?

In Table 8, we can see that we do not have better results with learned anchor boxes for face detection. We understand that, because the sizes of faces are not as various as those of characters, the learned anchor boxes do not help as much as in the case of characters. Finally, both face and character detections give similar accuracy which shows that the CNN detection model works well on face detection task.

Table 8. Comparison between face and character detections from panels on the DCM772 dataset with two settings: with and without learned anchor boxes.

Anchor Boxes	Face Detection	Character Detection
Not learned	74.75%	74.38%
Learned	74.94%	76.76%

8.2.2. Does CNN Detection Model Generalize Well?

For the generalizability, because ground-truth of faces is not provided in the eBDtheque dataset, we present only some visual results of face detection on the eBDtheque dataset by using the detection model trained from the DCM772 dataset in Figure 7. We can see that some particular character's faces

are not detected (e.g., eye-ball characters) because these character's faces are totally different from faces in DCM772. Otherwise, the model works well for other character's faces.

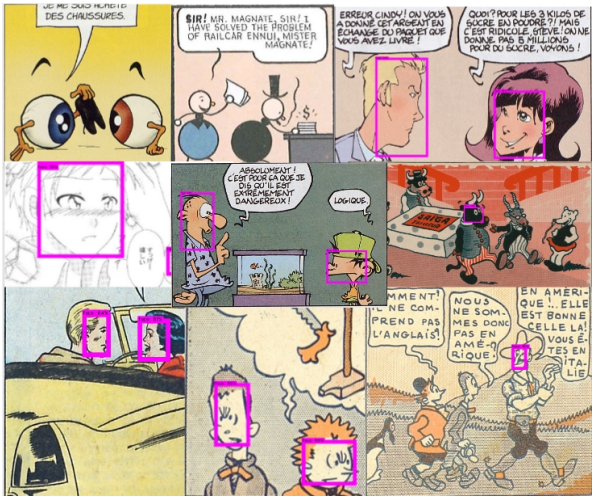


Figure 7. Face detection on eBDtheque using model trained on DCM772.

8.3. Panel Detection

Traditional approaches have worked well on the task of comic panel detection such as the methods proposed in [6,28,29]. To compare with existing works, we have trained the panel detector on the DCM772 dataset, with the anchors learned from this dataset. Firstly, we compare the trained detection model with the traditional connected component based approach [6] on the test set of DCM772. Then, the trained detection model is used to detect panels on the hundred pages from the eBDtheque dataset, which have been tested in [28,29].

In Table 9, we can see that the CNN model is better in both precision and recall for the DCM772 dataset. In the case of the eBDtheque dataset, the CNN model does not give the best result because it is trained from images in the DCM772 dataset and we reused the threshold = 0.8 which is validated using the validation set of the DCM772 dataset. During our experiments, we have observed two important remarks.

Table 9. Panel detection: A comparison between the CNN detection model with [6] on DCM772 and eBDtheque datasets. The CNN model trained on the DCM772 training set is used to detect panel for both datasets. Using trained CNN model, we keep only detections with a confidence score bigger than 80%. This threshold is selected using the validation set of DCM772 (P = Precision/R = Recall).

Method	DCM772 (%)	eBDtheque (%)
CNN model	P/R = 84.75/86.62	P/R = 83.44/58.96
[6]	P/R = 75.92/85.72	P/R = 78.86/77.59

8.3.1. The Instability of Detected Bounding Boxes

Firstly, the CNN detection modeldetects well the panels, but the detected bounding boxes are not very close to the ground-truth regions (see Figure 8). If we follow the evaluation measure of other works in panel detection presented in [28,29], many detected regions are considered as wrong detections because the detected bounding boxes are not close enough to the boxes in ground-truth, which lead to a lower precision/recall of the result.

Table 10 shows the comparison of panel detection on the eBDtheque dataset for the CNN model and two other traditional methods [27,29] under the strict evaluation measure called Jaccard index. The Jaccard index used in [27,29] considers that a correct detection should have the overlap ratio $ao > 90\%$,

between the predicted bounding box B_p and ground-truth bounding box B_{gt} (see Formula (3)). With this strict evaluation measure, we can see that the CNN model give poor results compared to the traditional approaches. This is the main reason we propose to use the algorithm in [6] for the panels detection step.

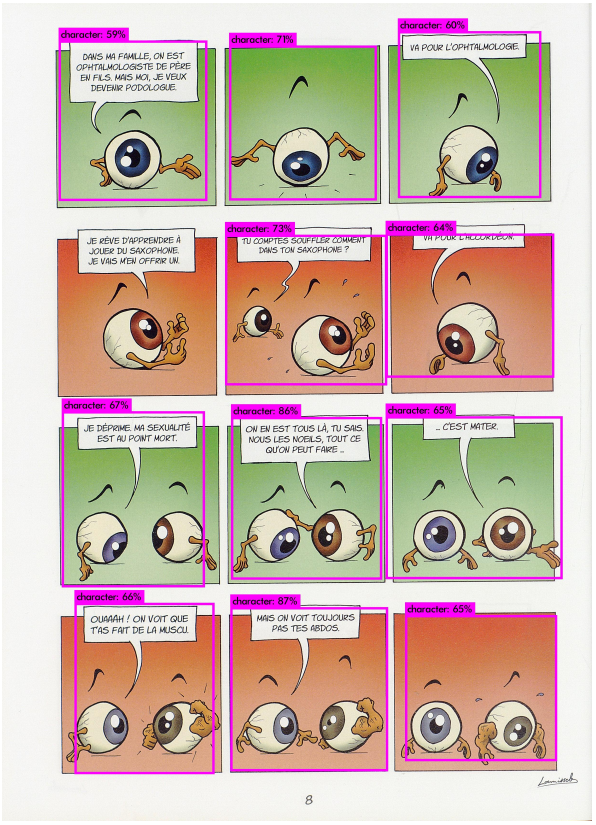


Figure 8. Result of panel detection on the eBDtheque dataset using the CNN model trained from the DCM772 dataset: The detected bounding boxes do not always fit the panel borders.

Table 10. Panel detection evaluation on the eBDtheque dataset using Jaccard index; results for [27] are taken from the paper.

Method	DCM772 (%)	eBDtheque (%)
CNN model	P/R = 70/61	P/R = 68/44
[6]	P/R = 79/78	P/R = 79/78
[27]	–	P/R = 84/70

8.3.2. Detections on Difficult Pages

An interesting remark is that the deep learning model can detect some kinds of delicate panels present in the eBDtheque dataset for which existing methods might fail (see Figures 9 and 10).

The CNN detection model provides good results on a dataset which has ground-truth data to train the model. However, the detection results for other different style datasets are not as good as the existing works which generalize better.

Figure 9. Panel detection on difficult pages. The first row shows detections from the CNN model. The second row shows detections from the algorithm in [6]. The last column is ground-truth regions from the eBDtheque dataset (defined as smallest graphics regions ignoring outgoing elements).

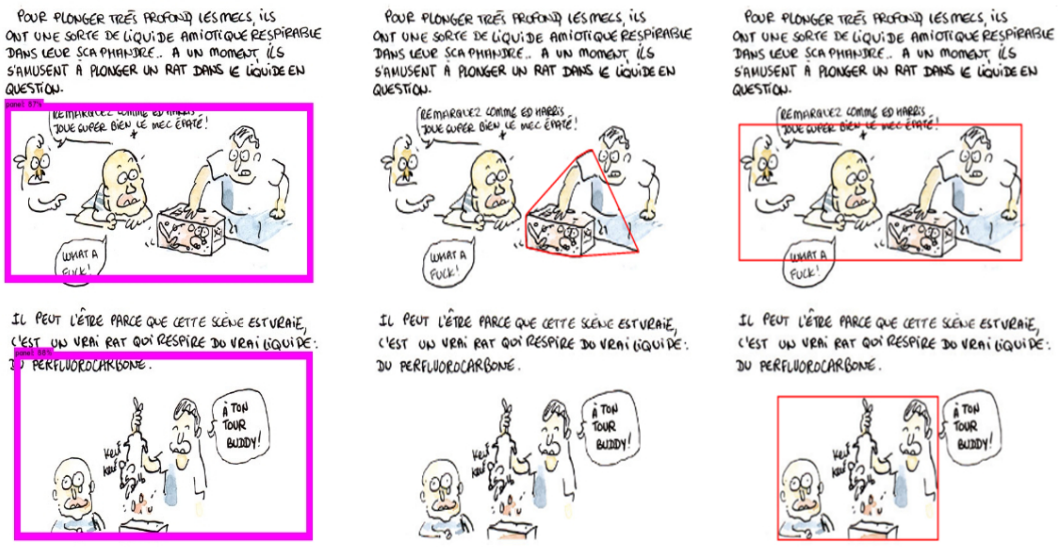


Figure 10. Panel detection on difficult pages (eBDtheque). The first column shows panel detections from the CNN model. The second column is detections from the algorithm in [6]. The last column is for ground-truth regions. Detections are represented by pink or red polygons.

8.4. Balloon Segmentation

For experimenting with the balloon segmentation task, we have trained two segmentation models, DeepLabv2 [65] and CRFasRNN [87] on the eBDtheque dataset. We can see the examples of balloon segmentation of DeepLabv2 model on the eBDtheque testing set in Figure 11. In Table 11, we observe that the segmentation accuracy of the CNN model (DeepLabv2) is better than the algorithm of [34]. The main reason for the low accuracy of [34] is that it detects many false positive examples, especially in non-color comics because white regions may contain aligned black graphics similar to balloon content.

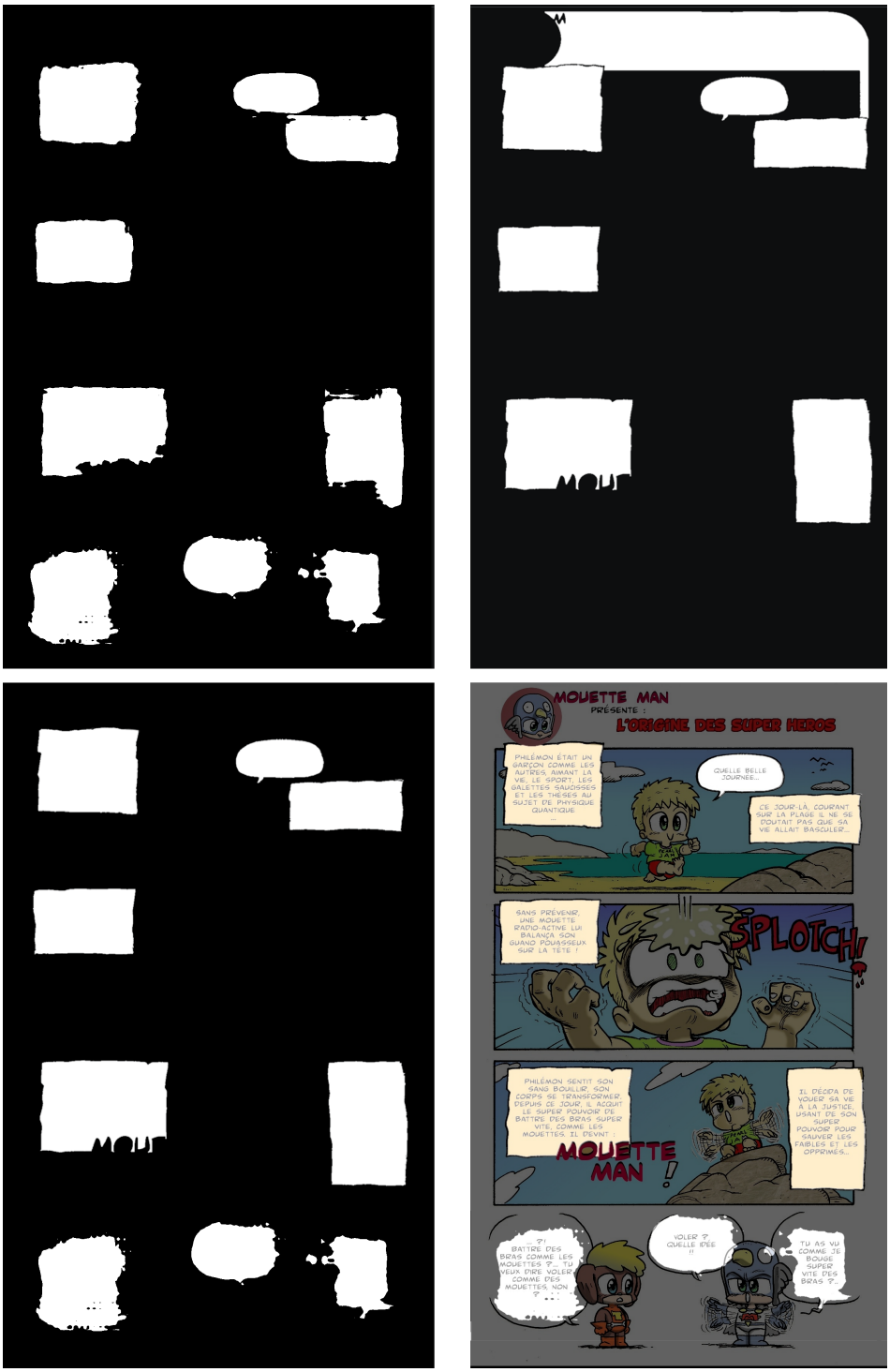


Figure 11. Examples of balloon segmentation from: DeepLabv2 (**top-left**); [34] (**top-right**); mixed (**bottom-left**); and over the original image (**bottom-right**). The white pixels represent the pixels labeled as part of a balloon. In the mixed (**bottom-left**) result, we can see that CNN model indicates better the true positives examples while the traditional method gives the shape of the balloons.

Table 11. The accuracies of balloon segmentation on comic pages, on the eBDtheque testing set.

Method	Accuracy	F1-Score
[34]	49.75	59.98
DeepLabv2	89.12	93.85

Boundaries Problem

We have observed that the trained segmentation model can localize very well the balloons in a comic page whether they are open (contour partially drawn) or closed. However, it cannot detect the tail of the balloons, and especially it can not identify precisely the boundaries of the balloons. Meanwhile, the traditional approaches in [31–34] detect very well the boundaries, although fail in difficult cases such as open and overlapping balloons and detect many false positive examples. Different from other detection tasks where we only need to detect a rectangle (a bounding box), balloon segmentation requires an algorithm able to detect the boundaries correctly. In this case, the traditional approaches give better results.

We have proposed a simple technique combining the results of the balloons extractor in [34] and the CNN balloon extractor (see Section 5.1). The false positive balloons by Rigaud et al. [34] are removed using the detections from the CNN model, and we can detect the closed balloons with the highest precision and still approximate open balloon as shown on Figure 11.

8.5. Text Recognition

In this section, we compare the performance of a pre-trained OCR system and then, when its output is post-processed by a subsequent OCR as presented in Section 6.

In the first OCR recognition round, we did not manually re-train Tesseract on the writing style used in the image, but we used its own pre-trained data for French and English. In the second round, a new model for OCRopus is automatically trained from scratch with validated text lines from the first step (text lines with a lexicality L = 100%, see Section 7.2.4). For the training of this new OCRopus model, we use the set of validated text lines as training set. Note that this training set may contain some errors since it has not been manually annotated. The resulting model is saved every 10,000 learning steps until 100,000. Each step consists in comparing one text line image and its associated ground-truth transcription from the testing set (http://www.danvk.org/2015/01/11/training-an-ocropus-ocr-model.html). When the training was completed, the model with the best accuracy is chosen for subsequent recognition tasks.

Table 12 shows the results of text recognition experiments using the pre-trained OCR (Tesseract) and the segmentation-free OCR system (OCRopus). Note that the training set can be easily extended by adding other images from the same scriptwriter because annotations are not required in the presented method. In Table 12, the number of extracted text lines is the output of the text line extraction algorithm presented in Section 6.1. The number of validated text lines is a subset of the extracted text lines that have a lexicality measure L = 100% (see Section 6.3).

The results of CER and WER of the both OCR systems (pre-trained and segmentation-free) are computed on the annotated text line images from the testing set. Segmentation-free OCR is automatically and only trained on validated text line images (see Section 6.3). The average results show that segmentation-free OCR improves by 12.41% the results from pre-trained OCR at character level (CER) and drops by 0.42% at word level as well (WER). This means that the quality of the validated text lines from pre-trained OCR is good enough to be used as pseudo ground-truth to automatically train a second OCR system that outperforms the initial OCR system at character level. Note that we also compared with the original English default model provided with OCRopus but we did not add them to Table 12 because the CER was always higher than 80% except for Albums 1–3, where it was 73.8%, 44.17%, and 69.85%, respectively.

When the proposed approach does not improve the results at character level (CER), it can be due to a tiny number of extracted, and, consequently, validated text lines (less than 10) which compose the training set of the segmentation-free OCR (Album 9). With such a limited training set, the segmentation-free OCR does not have enough data to learn how to recognize the writing style correctly. In other cases, it is because the writing style is lowercase which corresponds better to the type of text that the considered pre-trained OCR have been trained for (Albums 11 and 19). Sometimes it can be

because the writing style is only composed of clear and well-separated capital letters (Albums 20 and 28) which is also an easy case for the pre-trained OCR (no touching letters).

In Table 12, the Word Error Rate (WER) is often higher for the pre-trained OCR than the segmentation-free OCR. This can be explained by the fact that Tesseract uses a dictionary of frequent words from the language to improve its output [89].

The CER performance slightly changes according to the number of iterations used to train the model.It is hard to predict which number of iterations will be optimal for a given writing style but it seems to stabilize from 20,000 iterations so picked up results at iteration 50,000 in Table 12.

Note that the quality of the training set could not be measured in this experiment because its images are not annotated. This has the advantage to be easily extensible without requiring extra human effort.

Table 12. Pre-trained and segmentation-free OCR results on eBDtheque (Albums 1–11) and DCM772 (Albums 12–32) datasets. The first five columns show the album ID, the number of pages for the testing and training sets, and the number of text lines that have been extracted and validated, respectively. The next four columns indicate the average Character Error Rate (CER) and Word Error Rate (WER) of both OCR systems. The last column shows the number of iterations of the trained LSTM model that gives the lowest CER. The last row indicates the total or average of the values mentioned above.

ID	# Pages		# Text Lines		Pre-Trained (Tesseract)		Segmentation-Free (OCRopus)		
	Test	Train	Extract [38]	Valid	CER (%)	WER (%)	CER (%)	WER (%)	# itera.
1	10	41	432	229	82.13	94.97	**26.25**	**53.10**	80,000
2	5	89	804	412	40.48	80.79	**20.55**	**53.42**	90,000
3	5	51	183	123	76.82	93.03	**21.49**	**39.80**	60,000
4	4	42	354	182	51.40	**78.77**	**40.62**	79.66	80,000
5	4	23	344	163	27.39	**49.13**	**22.45**	52.91	50,000
6	5	85	1433	1130	24.50	**52.92**	**19.45**	53.32	40,000
7	5	36	265	179	21.36	**42.94**	**18.25**	49.57	20,000
8	2	49	107	22	65.07	98.68	**54.57**	**97.90**	90,000
9	5	40	19	2	**84.95**	100.00	85.86	100	50,000
10	3	22	821	508	19.75	46.67	**12.02**	**42.73**	60,000
11	5	38	383	160	**20.52**	**61.58**	34.63	81.84	80,000
12	2	46	861	529	45.49	**69.13**	**28.00**	70.10	80,000
13	2	31	526	212	43.33	**73.30**	**32.61**	84.16	80,000
14	2	27	730	346	35.90	**63.51**	**29.36**	71.62	80,000
15	2	24	380	181	46.34	72.92	**31.60**	77.29	40,000
16	2	27	519	336	42.21	61.53	**23.39**	**59.40**	100,000
17	2	23	957	599	36.10	64.30	**14.68**	**46.56**	70,000
18	2	20	116	55	54.01	**84.02**	**51.01**	94.32	100,000
19	2	62	773	384	**22.71**	**50.79**	33.71	79.36	100,000
20	2	26	731	413	**8.46**	**35.64**	22.90	55.55	90,000
21	2	25	217	53	57.16	**85.09**	**55.36**	100.00	60,000
22	2	27	464	188	61.34	**85.03**	**59.24**	98.35	90,000
23	2	30	587	404	46.42	**68.02**	**41.88**	78.68	80,000
24	2	34	164	51	87.59	100.00	**72.49**	100.00	20,000
25	2	26	689	324	28.04	51.23	**19.61**	**50.99**	70,000
26	2	56	1026	279	40.74	**82.48**	**34.22**	86.13	100,000
27	2	21	348	198	49.57	**97.06**	**35.38**	98.04	20,000
28	2	28	802	580	**20.01**	**42.20**	22.96	56.06	100,000
29	2	14	82	30	52.40	**85.71**	**51.07**	98.85	80,000
30	2	16	421	160	63.35	91.58	**32.84**	**82.67**	90,000
31	2	17	448	227	69.53	91.62	**26.84**	**73.68**	100,000
32	2	26	622	277	71.63	89.97	**29.90**	**79.05**	60,000
	95	**1122**	**16,608**	**8936**	**47.07**	**72.86**	**34.66**	**73.28**	

8.6. Encoding and Indexing

According to the literature review in Section 2.6, we choose to encode all extracted elements using CBML format which best fits our need. However, because we are able to extract some information that is not part of the CBML customization reference (e.g., tail position and direction), we add some custom tags for this purpose.

The initial CBML file document generates a *teiHeader* tag containing the book metadata, and an empty text section where we place all encoded content information. The book metadata are usual bibliographic information such as title, description/synopsis, ISBN, publisher, publication date, contributors (role and full name), language, series title and volume number (see Listing 1). Note that this information is not mandatory for the indexing system, but they are usually available from the image provider. This CBML skeleton is used as a starting point to aggregate data extracted from the book. The extracted information is placed between <*text*> and </*text*> tags. Page changes are delimited by *div* tags associated with an ID attribute as shown in the introductory Figure 1. In this way, one or several page/image can be described in a single CBML file.

Listing 1: Initial CBML skeleton initialized with bibliographic tags and <*text*> tag where the automatic indexation system will add structured information about image content.

```
<!-- Namespaces -->
TEI: xmlns="http://www.tei-c.org/ns/1.0"
CBML: xmlns="http://www.cbml.org/ns/1.0"

<!-- Overview -->
<?xml version="1.0"?>
<TEI>
<teiHeader>
<titleStmt>...</titleStmt>
<fileDesc>
<publicationStmt>
<publisher>...</publisher>
<date>...</date>
<idno type="ISBN">...</idno>
</publicationStmt>
<notesStmt>...</notesStmt>
<seriesStmt>
<title>...</title>
<idno type="vol">...</idno>
</seriesStmt>
<sourceDesc>...</sourceDesc>
</fileDesc>
<profileDesc>...</profileDesc>
</teiHeader>
<text>...</text>
</TEI>
```

The CBML file, once completed by the proposed indexation system, contains all bibliographic information and content description. This file allows, for instance, an original comic book page image to be automatically split into several sub-images corresponding to panels with a precise visual and textual description of their respective contents. Proper names, verbs, etc. from the speech balloon can be indexed according to characters. Moods and situations such as dialogue, fight, love scene, and sport may also be determined and indexed by using natural language processing methods. Once indexed, complex queries such as "*character A* **and** *character B* **talking about** *something*" or "*keyword* **from** *character A*" can be achieved.

9. Conclusions

In this work, we have presented a complete indexing system for digital comic book images that can improve web image retrieval performance by automatically extracting and indexing textual and visual content. We have also compared state-of-the-art deep learning based and traditional approaches to give the reader as complete as possible overview of what are the current performances and limits of such techniques. We evaluate our contributions with online datasets and also propose a new public dataset called *DCM772* specially designed for comic character body and face detection.

From our empirical experimentation, we have found out that the recent deep learning approaches can be combined with traditional methods to improve the analysis of comic book images. In some tasks such as face detection, character detection, and text recognition, the use of neural network model to train a detector or recognizer can significantly improve the performance compared to traditional methods. In the task of panel detection, the deep learning model is better for a particular dataset, but the traditional methods generalize better for any dataset. For the balloon segmentation task, the deep learning model gives better accuracy compared to traditional methods but the latter has the advantage of detecting precisely the boundaries.

1 Face, character detection

The task of face detection and character detection are very challenging without learning from labeled data because it is hard to find features and heuristic rules which can well generalize faces/characters. Among machine learning methods, deep learning has the best detection model, and our empirical experiments show that deep learning models are capable of detecting face/character in comic images. We believe that further studies could improve the detection accuracy by benefiting from the domain knowledge.

2 Panel detection

Existing methods are still better for panel detection. The deep detection models have an advantage that they can detect some difficult panels with a complex background where the existing methods fail. However, they have two drawbacks. Firstly, they cannot generalize well to a new dataset which is different in style compared to the dataset used to train the model. Secondly, they cannot detect the panel boundaries with high precision.

3 Balloon segmentation

While the CNN model gives good accuracy for balloon segmentation, it still has the boundaries problem which is essential for the balloon segmentation task. With the boundaries problem, the CNN segmentation model does not give better results than traditional methods in our experiments. By combining the two methods, we can archive better results with fewer false positive detections and more true positive detections for open balloons.

4 Text recognition

We demonstrated that standard pre-trained OCR text line output associated with a good quality check process can feed another OCR system and overpass initial output quality. However, a minimum amount of carefully quality checked text line is required to train the subsequent OCR system efficiently. The issue could be overcome by extending (instead of replacing) the original training dataset with the newly validated text lines. This would be interesting especially in those cases where the number of validated text lines is very low. It would also ensure that the classifier has seen all symbol classes and not only those appearing on the validated text lines.

The subsequent OCR is highly dependent on the initial OCR. We experimented with only one initial OCR system, but the proposed approach can handle a bunch of them which will provide even more training data, and, consequently, improve final OCR transcriptions.

The word error rate (WER) is quite high for both OCR systems compared to usual results from the literature. This is partially due to the text line extractor algorithm that may cut or ignore some text lines. In addition, some pseudo ground-truth errors may not be filtered out in the automatic text line validation process. The error rates can also be reduced by using images with higher quality (OCR systems usually recommend 300 DPI). The post-processing of text extraction algorithm output could also improve the quality of the generated pseudo ground-truth (e.g., remove surrounding letter parts and separate letters from image border).

9.1. Encoding and Indexing

The proposed encoding format mainly relies on the CBML format with some custom extensions that are not (yet) part of the formalism. The proposed system generates one description file for each image or album and allows automatic sub image generation to return a more focused image to the user, according to its query. Many tags and attributes for describing balloon shape and tail are missing. In addition, localized contents such as general objects, animals, and scenes require proper semantic attributes to be best described.

9.2. Perspective

In the future, we will cluster detected comic characters to identify and link the detections of several instances of the same character throughout the comic book. This analysis can help contextual search and intra-book navigation. Another idea to investigate is to match the OCRed text and named entities related to character clusters to, for instance, retrieve their proper names. This analysis will facilitate search and navigation of comic characters across comic book images from different sources (inter-books).

Author Contributions: V.N. and C.R. both conceived, designed, and performed the experiments; analyzed the data; and wrote the paper. J.-C.B. gave precious advice throughout this research work and made significant review comments for the pre-final version of this paper.

Funding: This research received no external funding.

Acknowledgments: This work was supported by the University of La Rochelle (France), the town of La Rochelle, the PIA-iiBD ("Programme d'Investissements d'Avenir") and the MEDIABD project from the French administrative region "Nouvelle Aquitaine". We are grateful to all authors and publishers of comic book images from eBDtheque and DCM772 datasets for allowing us to use and share their works.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cai, D.; He, X.; Li, Z.; Ma, W.Y.; Wen, J.R. Hierarchical Clustering of WWW Image Search Results Using Visual, Textual and Link Information. In Proceedings of the 12th Annual ACM International Conference on Multimedia (MULTIMEDIA '04), New York, NY, USA, 10–16 October 2004; ACM: New York, NY, USA, 2004; pp. 952–959, doi:10.1145/1027527.1027747. [CrossRef]
2. Feng, H.; Shi, R.; Chua, T.S. A Bootstrapping Framework for Annotating and Retrieving WWW Images. In Proceedings of the 12th Annual ACM International Conference on Multimedia (MULTIMEDIA '04), New York, NY, USA, 10–16 October 2004; ACM: New York, NY, USA, 2004; pp. 960–967, doi:10.1145/1027527.1027748. [CrossRef]
3. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–9, doi:10.1109/CVPR.2015.7298594. [CrossRef]
4. Halavais, A. *Search Engine Society*; Digital Media and Society; Wiley: Hoboken, NJ, UA, 2017.
5. Rigaud, C.; Guérin, C.; Karatzas, D.; Burie, J.C.; Ogier, J.M. Knowledge-driven understanding of images in comic books. *Int. J. Doc. Anal. Recognit. (IJDAR)* **2015**, *18*, 199–221, doi:10.1007/s10032-015-0243-1. [CrossRef]

6. Rigaud, C. Segmentation and Indexation of Complex Objects in Comic Book Images. Ph.D. Thesis, Université de La Rochelle, La Rochelle, France, 2014.
7. Chu, W.T.; Cheng, W.C. Manga-specific features and latent style model for manga style analysis. In Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 1332–1336, doi:10.1109/ICASSP.2016.7471893. [CrossRef]
8. Matsui, Y.; Ito, K.; Aramaki, Y.; Fujimoto, A.; Ogawa, T.; Yamasaki, T.; Aizawa, K. Sketch-based manga retrieval using manga109 dataset. *Multimed. Tools Appl.* **2017**, *76*, 21811–21838, doi:10.1007/s11042-016-4020-z. [CrossRef]
9. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016. Available online: http://www.deeplearningbook.org (accessed on 29 June 2018).
10. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778, doi:10.1109/CVPR.2016.90. [CrossRef]
11. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. 2016. Available online: http://xxx.lanl.gov/abs/1612.08242 (accessed on 29 June 2018).
12. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid Scene Parsing Network. *arXiv* **2016**, arXiv:1612.01105.
13. Augereau, O.; Iwata, M.; Kise, K. A survey of comics research in computer science. *arXiv* **2018**, arXiv:1804.05490.
14. Yamada, M.; Budiarto, R.; Endo, M.; Miyazaki, S. Comic Image Decomposition for Reading Comics on Cellular Phones. *IEICE Trans.* **2004**, *87*, 1370–1376.
15. In, Y.; Oie, T.; Higuchi, M.; Kawasaki, S.; Koike, A.; Murakami, H. Fast frame decomposition and sorting by contour tracing for mobile phone comic images. *Int. J. Syst. Appl. Eng. Dev.* **2011**, *5*, 216–223.
16. Duda, R.O.; Hart, P.E. Use of the Hough transformation to detect lines and curves in pictures. *Commun. ACM* **1972**, *15*, 11–15. [CrossRef]
17. Li, L.; Wang, Y.; Tang, Z.; Gao, L. Automatic comic page segmentation based on polygon detection. *Multimed. Tools Appl.* **2014**, *69*, 171–197. [CrossRef]
18. Han, E.; Kim, K.; Yang, H.; Jung, K. Frame segmentation used MLP-based X-Y recursive for mobile cartoon content. In Proceedings of the 12th International Conference on Human-Computer Interaction: Intelligent Multimodal Interaction Environments (HCI'07), Beijing, China, 22–27 July 2007; Springer: Berlin/Heidelberg, Germany, 2007; pp. 872–881.
19. Tanaka, T.; Shoji, K.; Toyama, F.; Miyamichi, J. Layout Analysis of Tree-Structured Scene Frames in Comic Images. In Proceedings of the 20th International Joint Conference on Artifical intelligence (IJCAI'07), Hyderabad, India, 6–12 January 2007; pp. 2885–2890.
20. Arai, K.; Tolle, H. Method for Automatic E-Comic Scene Frame Extraction for Reading Comic on Mobile Devices. In Proceedings of the Seventh International Conference on Information Technology: New Generations (ITNG), Las Vegas, NV, USA, 12–14 April 2010; IEEE Computer Society: Washington, DC, USA, 2010; pp. 370–375.
21. Rigaud, C.; Tsopze, N.; Burie, J.C.; Ogier, J.M. Robust Frame and Text Extraction from Comic Books. In *Graphics Recognition. New Trends and Challenges*; Lecture Notes in Computer Science; Kwon, Y.B., Ogier, J.M., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; Volume 7423, pp. 129–138, doi:10.1007/978-3-642-36824-0_13.
22. Ho, A.K.N.; Burie, J.C.; Ogier, J.M. Comics page structure analysis based on automatic panel extraction. In Proceedings of the GREC 2011: Nineth IAPR International Workshop on Graphics Recognition, Seoul, Korea, 15–16 September 2011.
23. Arai, K.; Tolle, H. Method for Real Time Text Extraction of Digital Manga Comic. *Int. J. Image Process. (IJIP)* **2011**, *4*, 669–676.
24. Ponsard, C.; Ramdoyal, R.; Dziamski, D. An OCR-Enabled digital comic books viewer. In *Computers Helping People with Special Needs*; Springer: Berlin/Heidelberg, Germany 2012; pp. 471–478.
25. Stommel, M.; Merhej, L.I.; Müller, M.G. Segmentation-free detection of comic panels. In *Computer Vision and Graphics*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 633–640.
26. Pang, X.; Cao, Y.; Lau, R.W.; Chan, A.B. A Robust Panel Extraction Method for Manga. In Proceedings of the 22nd ACM International Conference on Multimedia (MM '14), Orlando, FL, USA, 3–7 November 2014; ACM: New York, NY, USA, 2014; pp. 1125–1128, doi:10.1145/2647868.2654990. [CrossRef]
27. Wang, Y.; Zhou, Y.; Tang, Z. Comic frame extraction via line segments combination. In Proceedings of the 2015 13th International Conference on Document Analysis and Recognition (ICDAR), Tunis, Tunisia, 23–26 August 2015; pp. 856–860, doi:10.1109/ICDAR.2015.7333883. [CrossRef]

28. Wang, Y.; Zhou, Y.; Liu, D.; Tang, Z. Comic storyboard extraction via edge segment analysis. *Multimed. Tools Appl.* **2016**, *75*, 2637–2654, doi:10.1007/s11042-015-2680-8. [CrossRef]
29. Li, L.; Wang, Y.; Suen, C.Y.; Tang, Z.; Liu, D. A Tree Conditional Random Field Model for Panel Detection in Comic Images. *Pattern Recognit.* **2015**, *48*, 2129–2140, doi:10.1016/j.patcog.2015.01.011. [CrossRef]
30. Ho, A.K.N.; Burie, J.C.; Ogier, J.M. Panel and Speech Balloon Extraction from Comic Books. In Proceedings of the 2012 10th IAPR International Workshop on Document Analysis Systems, Gold Cost, QLD, Australia, 27–29 March 2012; pp. 424–428, doi:10.1109/DAS.2012.66. [CrossRef]
31. Kuboi, T. Element Detection in Japanese Comic Book Panels. Master's Thesis, California Polytechnic State University, San Luis Obispo, CA, USA, 2014.
32. Liu, X.; Wang, Y.; Tang, Z. A clump splitting based method to localize speech balloons in comics. In Proceedings of the 2015 13th International Conference on Document Analysis and Recognition (ICDAR), Tunis, Tunisia, 23–26 August 2015; pp. 901–905, doi:10.1109/ICDAR.2015.7333892. [CrossRef]
33. Liu, X.; Li, C.; Zhu, H.; Wong, T.T.; Xu, X. Text-aware balloon extraction from manga. *Vis. Comput.* **2016**, *32*, 501–511, doi:10.1007/s00371-015-1084-0. [CrossRef]
34. Rigaud, C.; Burie, J.C.; Ogier, J.M. Text-Independent Speech Balloon Segmentation for Comics and Manga. In *Graphic Recognition. Current Trends and Challenges, Proceedings of the 11th International Workshop (GREC 2015), Nancy, France, 22–23 August 2015*; Revised Selected Papers; Springer International Publishing: Cham, Switzerland, 2017; pp. 133–147, doi:10.1007/978-3-319-52159-6_10.
35. Rigaud, C.; Thanh, N.L.; Burie, J.C.; Ogier, J.M.; Iwata, M.; Imazu, E.; Kise, K. Speech balloon and speaker association for comics and manga understanding. In Proceedings of the 2015 13th International Conference on Document Analysis and Recognition (ICDAR), Tunis, Tunisia, 23–26 August 2015; pp. 351–355, doi:10.1109/ICDAR.2015.7333782. [CrossRef]
36. Eskenazi, S.; Gomez-Krämer, P.; Ogier, J.M. A comprehensive survey of mostly textual document segmentation algorithms since 2008. *Pattern Recognit.* **2017**, *64*, 1–14, doi:10.1016/j.patcog.2016.10.023. [CrossRef]
37. Su, C.Y.; Chang, R.I.; Liu, J.C. Recognizing Text Elements for SVG Comic Compression and Its Novel Applications. In Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR '11), Beijing, China, 18–21 September 2011; IEEE Computer Society: Washington, DC, USA, 2011; pp. 1329–1333, doi:10.1109/ICDAR.2011.267. [CrossRef]
38. Rigaud, C.; Karatzas, D.; Van de Weijer, J.; Burie, J.C.; Ogier, J.M. Automatic Text Localisation in Scanned Comic Books. In Proceedings of the 9th International Conference on Computer Vision Theory and Applications, Lisbon, Portugal, 5–8 January 2014.
39. Li, L.; Wang, Y.; Tang, Z.; Lu, X.; Gao, L. Unsupervised Speech Text Localization in Comic Images. In Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), Washington, DC, USA, 25–28 August 2013; pp. 1190–1194, doi:10.1109/ICDAR.2013.241. [CrossRef]
40. Rigaud, C.; Burie, J.; Ogier, J. Segmentation-Free Speech Text Recognition for Comic Books. In Proceedings of the 2nd International Workshop on coMics Analysis, Processing, and Understanding and 14th IAPR International Conference on Document Analysis and Recognition (ICDAR 2017), Kyoto, Japan, 9–15 November 2017; pp. 29–34, doi:10.1109/ICDAR.2017.288. [CrossRef]
41. Duc, B. *Du Scénario à la Réalisation Graphique, tout sur la Création des Bandes Dessinées*; Editions Glénat: Grenoble, France, 1997.
42. Lainé, J.M.; Delzant, S. *Le Lettrage des Bulles*; Eyrolles: Paris, France, 2010.
43. Medley, S. Discerning pictures: How we look at and understand images in comics. *Stud. Comics* **2010**, *1*, 53–70. [CrossRef]
44. Cohn, N. The limits of time and transitions: Challenges to theories of sequential image comprehension. *Stud. Comics* **2010**, *1*, 127–147. [CrossRef]
45. Ahmad, H.A.; Koyama, S.; Hibino, H. Impacts of Manga on Indonesian Readers' Self-Efficacy and Behavior Intentions to Imitate Its Visuals. *Bull. Jpn. Soc. Sci. Des.* **2012**, *59*, 3_75–3_84.
46. Sun, W.; Kise, K. Detection of Exact and Similar Partial Copies for Copyright Protection of Manga. *Int. J. Doc. Anal. Recognit.* **2013**, *16*, 331–349, doi:10.1007/s10032-013-0199-y. [CrossRef]
47. Sun, W.; Kise, K. Similar Partial Copy Detection of Line Drawings Using a Cascade Classifier and Feature Matching. In Proceedings of the 4th International Conference on Computational Forensics (IWCF'10), Tokyo, Japan, 11–12 November 2011; Springer: Berlin/Heidelberg, Germany, 2011; pp. 126–137.

48. Viola, P.; Jones, M.J. Robust Real-Time Face Detection. *Int. J. Comput. Vis.* **2004**, *57*, 137–154.10.1023/B:VISI.0000013087.49260.fb. [CrossRef]
49. Kohei, T.; Henry, J.; Tomoyuki, N. Face detection and face recognition of cartoon characters using feature extraction. In Proceedings of the IEEE International Electric Vehicle Conference (IEVC '12), Greenville, SC, USA, 4–8 March 2012; Volume 18.
50. Khan, F.S.; Anwer, R.M.; van de Weijer, J.; Bagdanov, A.D.; Vanrell, M.; Lopez, A.M. Color attributes for object detection. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3306–3313, doi:10.1109/CVPR.2012.6248068. [CrossRef]
51. Ho, H.N.; Rigaud, C.; Burie, J.C.; Ogier, J.M. Redundant structure detection in attributed adjacency graphs for character detection in comics books. In Proceedings of the 10th IAPR International Workshop on Graphics Recognition, Bethlehem, PA, USA, 20–21 August 2013.
52. Sun, W.; Burie, J.C.; Ogier, J.M.; Kise, K. Specific Comic Character Detection Using Local Feature Matching. In Proceedings of the 2013 12th International Conference on Document Analysis and Recognition (ICDAR '13), Washington, DC, USA, 25–28 August 2013; IEEE Computer Society: Washington, DC, USA, 2013; pp. 275–279, doi:10.1109/ICDAR.2013.62. [CrossRef]
53. Iwata, M.; Ito, A.; Kise, K. A Study to Achieve Manga Character Retrieval Method for Manga Images. In Proceedings of the 2013 27th Brazilian Symposium on Software Engineering (SBES '13), Tours, France, 7–10 April 2013; IEEE Computer Society: Washington, DC, USA, 2013; pp. 309–313, doi:10.1109/DAS.2014.60. [CrossRef]
54. Le, T.N.; Luqman, M.M.; Burie, J.C.; Ogier, J.M. A Comic Retrieval System Based on Multilayer Graph Representation and Graph Mining. In *Graph-Based Representations in Pattern Recognition*; Liu, C.L., Luo, B., Kropatsch, W.G., Cheng, J., Eds.; Springer International Publishing: Cham, Switzerland, 2015; pp. 355–364.
55. Matsui, Y.; Ito, K.; Aramaki, Y.; Yamasaki, T.; Aizawa, K. Sketch-based Manga Retrieval using Manga109 Dataset. *arXiv* **2015**, arXiv:1510.04389.
56. Deng, L.; Yu, D. Deep Learning: Methods and Applications. *Found. Trends Signal Process.* **2014**, *7*, 197–387, doi:10.1561/2000000039. [CrossRef]
57. Xiong, W.; Droppo, J.; Huang, X.; Seide, F.; Seltzer, M.; Stolcke, A.; Yu, D.; Zweig, G. Achieving Human Parity in Conversational Speech Recognition. *arXiv* **2016**, arXiv:1610.05256.
58. Kumar, A.; Irsoy, O.; Su, J.; Bradbury, J.; English, R.; Pierce, B.; Ondruska, P.; Gulrajani, I.; Socher, R. Ask Me Anything: Dynamic Memory Networks for Natural Language Processing. *arXiv* **2015**, arXiv:1506.07285.
59. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems (NIPS'12), Lake Tahoe, NV, USA, 3–6 December 2012; Curran Associates Inc.: Red Hook, NY, USA, 2012; pp. 1097–1105.
60. Zeiler, M.D.; Fergus, R. Visualizing and Understanding Convolutional Networks. In Proceedings of the 13th European Conference on Computer Vision-ECCV 2014, Zurich, Switzerland, 6–12 September 2014; pp. 818–833, doi:10.1007/978-3-319-10590-1_53. [CrossRef]
61. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems 28*; Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2015; pp. 91–99.
62. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV '15), Santiago, Chile, 7–13 December 2015; IEEE Computer Society: Washington, DC, USA, 2015; pp. 1440–1448. doi:10.1109/ICCV.2015.169. [CrossRef]
63. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. *arXiv* **2016**, arXiv:1512.02325.
64. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.
65. Chen, L.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *arXiv* **2016**, arXiv:1606.00915. [PubMed]
66. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R.B. Mask R-CNN. *arXiv* **2017**, arXiv:1703.06870.
67. Breuel, T.M.; Ul-Hasan, A.; Al-Azawi, M.A.; Shafait, F. High-performance OCR for printed English and Fraktur using LSTM networks. In Proceedings of the 2013 12th International Conference on Document Analysis and Recognition, Washington, DC, USA, 25–28 August 2013; pp. 683–687.

68. Ul-Hasan, A.; Breuel, T.M. Can We Build Language-independent OCR Using LSTM Networks? In Proceedings of the 4th International Workshop on Multilingual OCR (MOCR '13), Washington, DC, USA, 24 August 2013; ACM: New York, NY, USA, 2013; pp. 9:1–9:5, doi:10.1145/2505377.2505394.
69. Jenckel, M.; Bukhari, S.S.; Dengel, A. anyOCR: A sequence learning based OCR system for unlabeled historical documents. In Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, Mexico, 4–8 December 2016; pp. 4035–4040, doi:10.1109/ICPR.2016.7900265. [CrossRef]
70. Springmann, U.; Fink, F.; Schulz, K.U. Automatic quality evaluation and (semi-) automatic improvement of mixed models for OCR on historical documents. *arXiv* **2016**, arXiv:1606.05157.
71. Simistira, F.; Ul-Hassan, A.; Papavassiliou, V.; Gatos, B.; Katsouros, V.; Liwicki, M. Recognition of historical Greek polytonic scripts using LSTM networks. In Proceedings of the 2015 13th International Conference on Document Analysis and Recognition (ICDAR), Tunis, Tunisia, 23–26 August 2015; pp. 766–770, doi:10.1109/ICDAR.2015.7333865. [CrossRef]
72. Iyyer, M.; Manjunatha, V.; Guha, A.; Vyas, Y.; Boyd-Graber, J.; Daumé, H., III; Davis, L. The amazing mysteries of the gutter: Drawing inferences between panels in comic book narratives. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; Volume 2.
73. Chu, W.T.; Li, W.W. Manga FaceNet: Face Detection in Manga based on Deep Neural Network. In Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval, Bucharest, Romania, 6–9 June 2017; ACM: New York, NY, USA, 2017; pp. 412–415.
74. W3C. *OWL 2 Web Ontology Language Direct Semantics*; Technical Report; W3C: Cambridge, MA, USA, 2012.
75. Klein, B.; Stroup, T. Grand Comics Database. 1994. Available online: https://www.comics.org/ (accessed on 29 June 2018).
76. McIntosh, J. ComicsML. 2011. Available online: http://comicsml.jmac.org/ (accessed on 29 June 2018).
77. Morozumi, A.; Nomura, S.; Nagamori, M.; Sugimoto, S. Metadata Framework for Manga: A Multi-paradign Metadata Description Framework for Digital Comics. In Proceedings of the International Conference on Dublin Core and Metadata Applications, Seoul, Korea, 12–16 October 2009; pp. 61–70.
78. Walsh, J.A. Comic Book Markup Language: An Introduction and Rationale. *Digit. Hum. Q. (DHQ)* **2012**, *6*, 1–50.
79. Text Encoding Initiative Consortium. *Text Encoding Initiative*; Text Encoding Initiative Consortium: Charlottesville, VA, USA, 2014.
80. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; doi:10.1109/CVPR.2016.91.
81. Everingham, M.; Eslami, S.M.; Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes Challenge: A Retrospective. *Int. J. Comput. Vis.* **2015**, *111*, 98–136, doi:10.1007/s11263-014-0733-5. [CrossRef]
82. Lin, T.; Maire, M.; Belongie, S.J.; Bourdev, L.D.; Girshick, R.B.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. *arXiv* **2014**, arXiv:1405.0312.
83. Shin, H.C.; Orton, M.R.; Collins, D.J.; Doran, S.J.; Leach, M.O. Stacked Autoencoders for Unsupervised Feature Learning and Multiple Organ Detection in a Pilot Study Using 4D Patient Data. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1930–1943, doi:10.1109/TPAMI.2012.277. [CrossRef] [PubMed]
84. Lee, H.; Grosse, R.; Ranganath, R.; Ng, A.Y. Unsupervised Learning of Hierarchical Representations with Convolutional Deep Belief Networks. *Commun. ACM* **2011**, *54*, 95–103, doi:10.1145/2001269.2001295. [CrossRef]
85. Salakhutdinov, R.; Hinton, G.E. Deep Boltzmann Machines. In Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS), Clearwater Beach, FL, USA, 16–18 April 2009; Volume 1, p. 3.
86. Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How Transferable Are Features in Deep Neural Networks? In Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS'14), Montreal, QC, Canada, 8–13 December 2014; MIT Press: Cambridge, MA, USA, 2014; pp. 3320–3328.
87. Zheng, S.; Jayasumana, S.; Romera-Paredes, B.; Vineet, V.; Su, Z.; Du, D.; Huang, C.; Torr, P.H.S. Conditional Random Fields as Recurrent Neural Networks. In Proceedings of the International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015.

88. Guérin, C.; Rigaud, C.; Mercier, A.; Ammar-Boudjelal, F.; Bertet, K.; Bouju, A.; Burie, J.C.; Louis, G.; Ogier, J.M.; Revel, A. eBDtheque: A Representative Database of Comics. In Proceedings of the 2013 12th International Conference on Document Analysis and Recognition, Washington, DC, USA, 25–28 August 2013; pp. 1145–1149, doi:10.1109/ICDAR.2013.232. [CrossRef]
89. Smith, R. An Overview of the Tesseract OCR Engine. In Proceedings of the 9th International Conference Conference on Document Analysis and Recognition (ICDAR), Parana, Brazil, 23–26 September 2007; pp. 629–633.
90. Tome, P.; Fierrez, J.; Vera-Rodriguez, R.; Ramos, D. Identification using face regions: Application and assessment in forensic scenarios. *Forensic Sci. Int.* **2013**, *233*, 75–83, doi:10.1016/j.forsciint.2013.08.020. [CrossRef] [PubMed]
91. Gales, M.J.F.; Liu, X.; Sinha, R.; Woodland, P.C.; Yu, K.; Matsoukas, S.; Ng, T.; Nguyen, K.; Nguyen, L.; Gauvain, J.L.; et al. Speech Recognition System Combination for Machine Translation. In Proceedings of the 2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP '07, Honolulu, HI, USA, 15–20 April 2007; Volume 4, pp. IV-1277–IV-1280, doi:10.1109/ICASSP.2007.367310.