HAL open science

Robust frame and text extraction from comic books

Christophe Rigaud, Norbert Tsopze, Jean-Christophe Burie, Jean-Marc Ogier

To cite this version:

Christophe Rigaud, Norbert Tsopze, Jean-Christophe Burie, Jean-Marc Ogier. Robust frame and text extraction from comic books. Lecture Notes in Computer Science, 2013, 7423, pp.129-138. hal-00841493

HAL Id: hal-00841493

https://hal.science/hal-00841493v1

Submitted on 5 Jul 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Robust frame and text extraction from comic books

Christophe Rigaud[1], Norbert Tsopze[1,2], Jean-Christophe Burie[1], Jean-Marc Ogier[1]

[1] Laboratory L3i, University of La Rochelle
Avenue Michel Crépeau 17042 La Rochelle, France
[2] LAMOCA - Department of Computer Science
University of Yaoundé I, BP 812 Yaoundé - Cameroon
{christophe.rigaud, norbert.tsopze, jcburie, jmogier}@univ-lr.fr

Abstract. Comic books constitute an important heritage in many countries. Nowadays, digitization allows to search directly from content instead of metadata only (e.g. album title or author name). Few studies have been done in this direction. Only frame and speech balloon extraction have been experimented in the case of simple page structure. In fact, the page structure depends on the author which is why many different structures and drawings exist. Despite the differences, drawings have a common characteristic because of design process: they are all surrounded by a black line. In this paper, we propose to rely on this particularity of comic books to automatically extract frame and text using a connected-component labeling analysis. The approach is compared with some existing methods found in the literature and results are presented.

Keywords: comic books, comics frame extraction, comics text extraction, segmentation, connected-component labeling, k-means.

1 Introduction

Nowadays, comics represent an important heritage in many countries. Massive digitization campaigns have been carried out in order to enhance archives and contents. This work has been done by specific companies that index pages but not their content. If the "page only" limit could be exceeded then new usages of comics may become a reality such as the frame-per-frame reading [2, 9] on mobile devices, the search of specific items by content based image retrieval from an large amount of albums and even content analysis from text. Such applications are currently possible with e-comics because they are designed with specific software and they can be indexed throughout the design process. The aim of our work is to process digitized comics in order to extract and analyse the content for full content search purpose. Full content search is requested by some cultural organisations such as the International City of Comics and Images [3] for specific object retrieval.

To enhance comic books, some works have been done recently but they are not robust enough to be industrialised. These works concern the segmentation of the frames, speech balloon and text (inside speech balloon). This paper proposes a method to automatically segment the frames and all the text contained in comics pages (not only text included into speech balloon). The proposed method is based on connected-component labeling algorithm following by k-means [17] clustering and then filtering.

The paper is organised as follows. The section 2 presents the vocabulary of comics content. An overview of frame and text segmentation methods is given in section 3. Section 4 and 5 present respectively the proposed method and the experimentations. Finally, section 6 and 7 conclude this paper.

2 Comic books

According to [14], there are three categories of comic books created respectively in America, Asia (manga) and Europe. In this paper, only the two first categories are considered because mangas are very different in terms of strokes, frames [18] and text [2]. A careful observation of the page content shows that the main characteristic of comics drawing is the black line that surround each element (or almost). Because of this feature, a connected-component (CC) based method is used in order to extract frame content from its edges. This algorithm has two advantages in our study. First, it is well adapted for frame segmentation as presented above. Second, it can be also used for text segmentation [7]. Moreover, using a single algorithm to segment a page is time saving.

Comic books relate stories drawn into albums. In traditional comics, pages are split up into strips separated by white gutter. A strip is a sequence of frames. A frame is a drawing generally in a box. Note that sometimes frame doesn't have box, in this case the reading and the segmentation become harder. Moreover, extended contents (e.g. speech balloons, characters, comics art) can overlap two frames or more [13]. All these particularities may punctually disturb the image processing.

Comics contain different types of text (handwriting or typewritten) depending on the nature of the message to read. Most of the text is inserted for speech purposes between characters and written into speech balloons. Other categories concern the narrative text and onomatopoeia. The onomatopoeias represent the sounds in a textual way or a sequence of symbols.

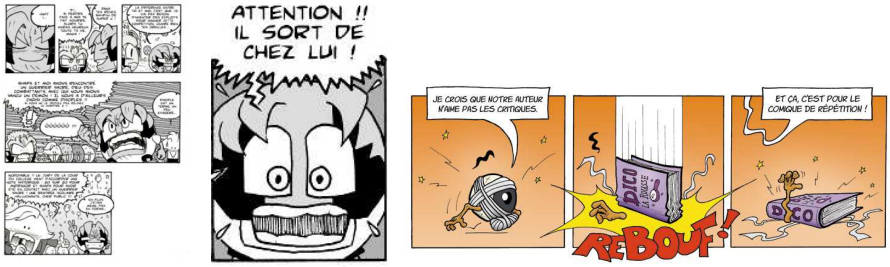
3 Existing methods

3.1 Frame segmentation

Frame segmentation has been mainly studied for reading comics on mobile device in order to display them frame by frame on a small screen. Here, our work concerns the indexing of a huge amount of albums that raises new issues in terms of variety of format, resolution and content.

Many segmentation methods have been studied to separate the background and the content as [9]. Most of them are based on white line cutting with Hough transform [6], recursive X-Y cut [8] or from gradient [16]. These methods doesn't handle empty area (case missing) [9] within a strip (figure 1a) or no full border frame (figure 1b). These issues have been corrected by connected-component approaches [1] but if some elements overlap (figure 1c), the frame segmentation process failed. The regions of interest (ROI) are often clustered by heuristic [2, 13] relative to the page size that is width and height dependent. A sequence of N erosions following by N dilatations has been proposed by [13] for cutting overlapping elements but it is time consuming and the choice of N

is unclear. [13] extracts the background of the pages by region growing algorithm, that is new in comparison with the binarisation applied by the other methods.



(a) Missing frame [4] (b) Partial box [4] (c) Overlapping between three frames [12]

Fig. 1: Examples of specific frames.

3.2 Text segmentation

In comics, most of the text is part of speech balloons. It is probably the reason why it is the only type of text studied so far. Previous works extract text from speech balloon [18, 1] or inversely speech balloon from text [13]. These approaches are really efficient but they suppose that text is written in black in a white balloon. We propose to enlarge this limitation: text background colour should be similar to page background.

4 Contribution

We propose a new method to extract frame and text area simultaneously from comics pages for indexation purpose. Our method processes page per page and begins by a pre-processing that binarise the page. Then, the ROI are defined as the set of the connected-component bounding boxes (rectangles). ROI are classified as “noise”, “text” and “frame” depending to their sizes, topological relations, and for the text, spatial relations. Note that only speech and narrative texts are considered in this study because they aren’t overlapped by object (e.g. line, drawing). The onomatopoeias will be studied in a future work. The originalities of this paper are frame segmentation, with or without box, and out-of-balloon text segmentation that can be extracted by CC algorithm.

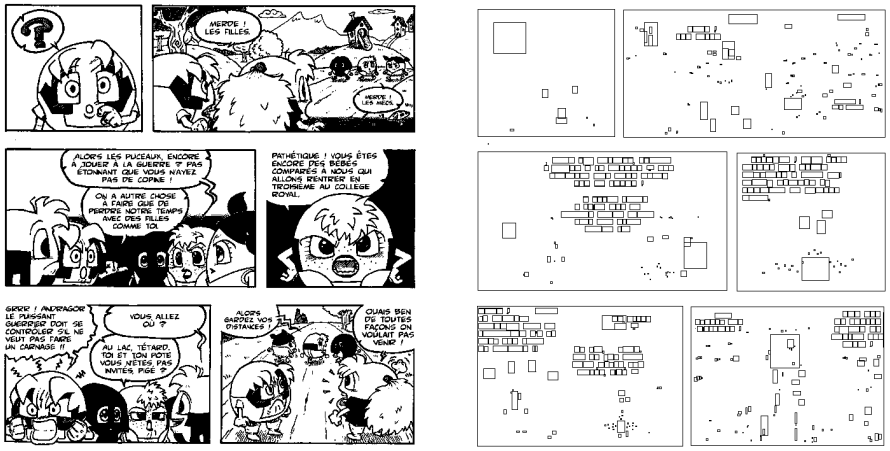
4.1 Pre-processing

The aim of the pre-processing step is to separate background and content of the page in order to focus on the content later. Several processing are implemented in order to apply CC algorithm, and then, to extract the bounding boxes. It can be resumed as follows:

1. Grayscale conversion

2. Binarisation threshold computation
3. Image inversion depending on the threshold
4. Binarisation
5. Connected-component extraction

The first step consists in a grayscale conversion as given in [15]. Then, a binarisation (figure 2a) is applied with a threshold computed from the median value of the border page pixels. We assume that the border pixels of the page are representative of the page background. If the median value is closer to "black" gray levels than "white" gray levels, then, image inversion is applied and we redo the complete process in order to always get a white background at the end of this step. This pre-processing is more robust than [2] who assumes that the page is always white and uses a constant threshold. Binarisation is very important for the rest of the method because the background part won't be considered anymore. Then, CC algorithm is used to extract, from connected components, the bounding boxes of all the elements (sequence of black pixels) of the image (figure 2b).



(a) Page after binarisation [5]

(b) Set of bounding boxes

Fig. 2: Pre-processing steps

4.2 ROI classification

ROI are defined as the connected-component bounding boxes. We define a set of regions $R = \{R_1, R_2, ..., R_n\}$. The classification is performed on ROI heights with k-means algorithm. The number of expected classes is 3 according to our experiments on several comics. Classes are labelled as "frame" (the highest), "text" (the most numerous) and "noise" (few pixels height) as shown on figure 3. This classification is performed dynamically on each page that makes our method invariant to page format and resolution. Indeed, ROI height classification is not page size dependent unlike [13, 2], and

the number of pixels for each ROI is proportional to the page resolution (do not bias the classification). This method assumes that the page contains text with background brightness similar to page background otherwise the binarisation and thus the classification may fail.

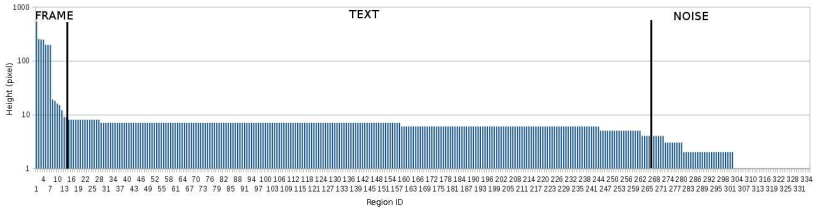


Fig. 3: Example of ROI classification on descendent histogram of the ROI height

Then, the variance of each class is computed to check the homogeneity of the ROI. If the variance of the "frame" class is high, a specific algorithm [13] is applied in order to improve the previous steps (binarisation and/or classification).

Example Figure 4 shows the frame segmentation of a page containing two frames overlapped by a black arrow (figure 4a and 4b). As shown in figure 4c, these two frames are detected as only one single frame by the CC algorithm (the biggest bounding box in figure 4c and the region 1 in figure 4d). The histogram figure 4d (log scale) shows that the first ROI is much higher than the others within the "frame" class. The variance of the "frame" class is therefore much higher than the two other classes that may due to an issue from the binarisation step. To fix this issue, a specific algorithm proposed by [13] can be used. It consists in frame segmentation by region growing applied on page background (frames become black blocks) followed by a sequence of erosions and dilatations in order to "disconnect" the black blocks (removes small overlapping elements). Then we redo pre-processing and classification steps for the frames only.

Note that the gap between the frame 7 and 8 in figure 4d is due to some objects (e.g. the top left big interrogation mark in figure 2a) higher than a character height. These ROI will be removed by a topological filtering process as explained bellow section 4.3.

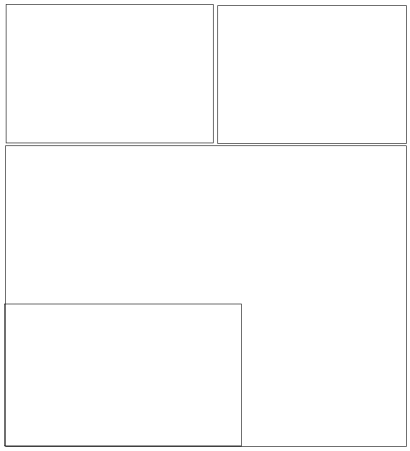
4.3 Filtering

After the classification stage, two filters are applied in order to remove false positive detection (region labelled mistakenly). The first filter is topological and keeps only the frames not fully contained in an other frame ($R_i \notin R_j \forall j, i \neq j$) (figure 5a and 5b).

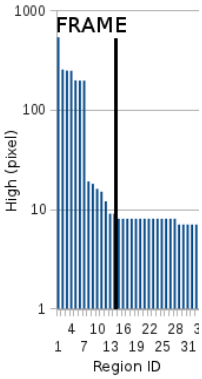
The second filter merges all the "text" ROI closer than two times the median "text" class height to define text areas (figure 6). Sometimes, detected text areas do not contain text but many small elements as high as text (figure 7). Thus, a text/graphic separation method [11] is applied to remove areas without text. This method compares vertical and horizontal projected histogram of each text area.

(a) Page with an overlapping element [10]

(b) Zoom of the overlapping element [10]

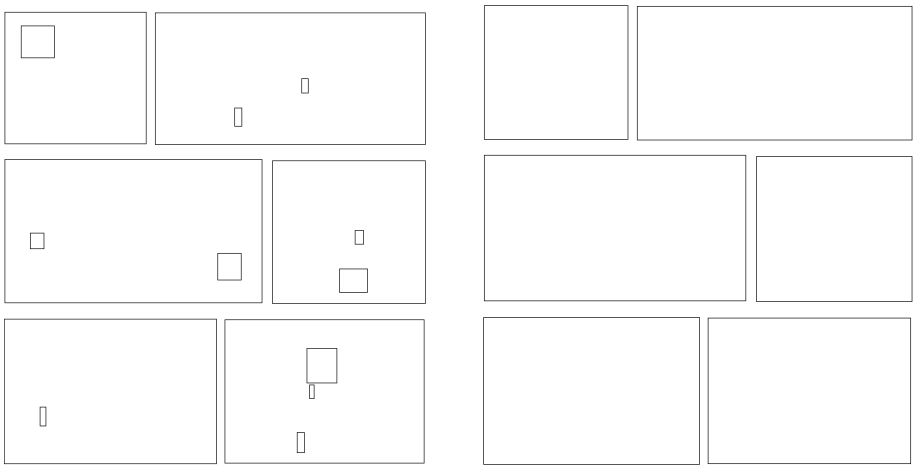


(c) Bounding boxes of connected-components



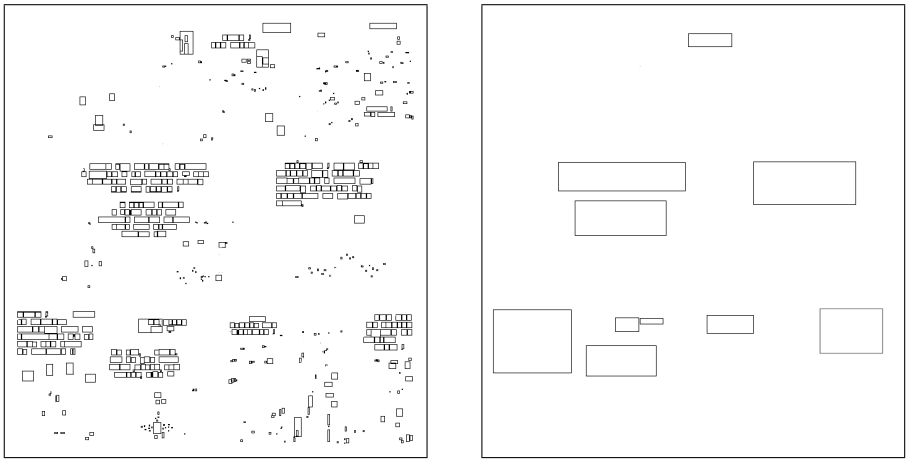
(d) Histogram zoomed on frames

Fig. 4: False positive frame detection



(a) Frame from k-means clustering (b) Frame after topological filter

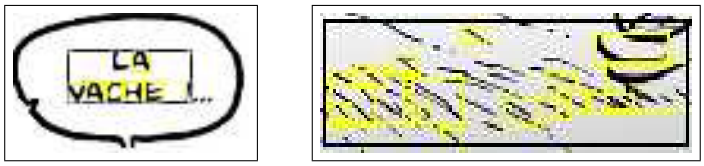
Fig. 5: Topological filtering of the frames



(a) Text from k-means clustering (b) Text area after filtering

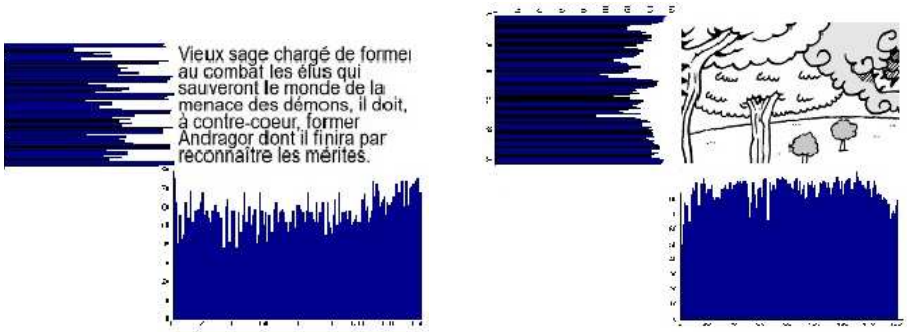
Fig. 6: Spatial filtering of text

Experimentally, we determined that to be a true text area, the variance of the horizontal projected histogram should be higher than the variance of the vertical projected histogram. The reason is that the horizontal projected histogram of a text area presents important variations due to the text and line spaces (figure 8a). This phenomenon isn't true for non-text areas (figure 8b).



(a) Correct text area (b) Wrong text area

Fig. 7: Example of text area detections (black rectangles)



(a) Histograms of a correct text areas (b) Histograms of a wrong text areas

Fig. 8: Example of projected histograms (number of white pixels)

5 Experimentation and results

5.1 Frame segmentation

Experiments were performed in the same conditions as [13] in order to compare the results. Namely, the same dataset and the comparison with same techniques found in the literature. The data set was composed of European and American comics: 42 pages from 7 different authors that contained 355 frames in total. This dataset is not publicly available because of copyright issues. To evaluate the results, the same two segmentation rates as [13] were computed. The first one is the success rate for page. A page was considered to be well segmented if ALL the frames of the page had been correctly extracted. This rate is used to estimate the quality of the extracted layout. The second is the success rate for frames. This rate gives the percentage of well extracted frames among the 355 frames of the data set.

Method	Tanaka [16]	Arai [1]	Ngo Ho [13]	Proposed method
Page (%)	42.8	47.6	64.3	66.7
Frame (%)	63.9	75.6	87.3	88.2

Fig. 9: Success rate comparison.

In comparison with [1, 13, 16], the proposed method is more efficient for frame segmentation because we handle border-free frames. Moreover, this method is 60% faster than [13]. This approach is faster because a time consuming process (specific algorithms) is applied only if the page contains overlapping elements (section 4.2). Nevertheless, the frame success rate does not bias the text success rate because text areas are extracted from the whole page and not from frames.

5.2 Text area segmentation

Text areas were extracted (section 4.3) from the same data set mentioned above. In order to be more accurate, speech text areas and narrative text areas were distinguished, namely 435 and 79 text areas respectively for the whole data set. We define:

- TP: the areas labelled as text areas that contain only text (true positive)
- FN: the areas ignored that contain text (false negative)

The text areas that were segmented partially or in many parts are considered as "false negatives".

Text type	TP	FN
Speech (%)	78	22
Narrative (%)	53	47

Fig. 10: Success rates of the text areas

The results are encouraging for the speech text category because most of the 22% of FN are text plus extra parts that need specific process. An adapted filtering will be developed to improve the detection. The narrative text extraction is harder because of its lower contrast with background (no white or light background). Nevertheless, it is difficult to compare our method with other approaches because we do not look for speech balloon only but for every single text area in the page, and as far as we know this hasn't been studied before in comics processing.

6 Conclusion and perspectives

A new method, to extract frames and texts simultaneously from comics, has been proposed and evaluated. The proposed approach is fast and especially robust to page format variations and border-free frames. Moreover, the method based on connected component analysis is able to extract all the text inside or outside the speech balloons.

The evaluation shows that more than 88% of the frames are correctly extracted. However, an effort has to be done to improve the results especially for large overlapping elements and narrative text extraction. The frame and text extraction was a first step. The main objective of our future work will be to analyse the content of the frame.

7 Acknowledgement

This work was supported by the European Regional Development Fund, the region Poitou-Charentes (France), the General Council of Charente Maritime (France) and the town of La Rochelle (France).

References

1. Arai, K., Tolle, H.: Method for automatic e-comic scene frame extraction for reading comic on mobile devices. In: Seventh International Conference on Information Technology: New Generations. pp. 370–375. ITNG, IEEE Computer Society, Washington, DC, USA (2010)
2. Arai, K., Tolle, H.: Method for real time text extraction of digital manga comic. International Journal of Image Processing (IJIP) 4(6), 669–676 (2011)
3. CIBDI: Cité internationale de la bande dessinées et de l'image [online], www.citebd.org
4. Cyb: La légende des Yaouanks. Studio Cyborga, Goven, France (2008)
5. Cyb: Bubblegôm. Studio Cyborga, Goven, France (2009)
6. Duda, R.O., Hart, P.E.: Use of the hough transformation to detect lines and curves in pictures. Commun. ACM 15, 11–15 (January 1972)
7. Fletcher, L., Kasturi, R.: A robust algorithm for text string separation from mixed text/graphics images. IEEE Transactions on Pattern Analysis and Machine Intelligence 10(6), 910–918 (Nov 1988)
8. 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. pp. 872–881. HCI'07, Springer-Verlag, Berlin, Heidelberg (2007)
9. 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. Internatinal journal of systems applications, engineering and development 5(2), 216–223 (2011)
10. Jolivet, O.: BostonPolice. Clair de Lune, Allauch, France (2010)
11. Khedekar, S., Ramanaprasad, V., Setlur, S., Govindaraju, V.: Text - image separation in devanagari documents. In: Proceedings of the Seventh International Conference on Document Analysis and Recognition. pp. 1265 – 1269 (August 2003)
12. Lamisseb: Les noeils Tome 1. Bac@BD, Valence, France (2011)
13. Ngo Ho, A.K., Burie, J.C., Ogier, J.M.: Comics page structure analysis based on automatic panel extraction. In: GREC 2011, Nineth IAPR International Workshop on Graphics Recognition. Seoul, Korea (September, 15-16 2011)
14. Ponsard, C., Fries, V.: An accessible viewer for digital comic books. In: ICCHP, LNCS 5105. pp. 569–577. Springer-Verlag Berlin Heidelberg (2008)
15. Pratt, K., W.: Digital image processing (2nd ed.). John Wiley & Sons, Inc., NY, USA (1991)
16. Tanaka, T., Shoji, K., Toyama, F., Miyamichi, J.: Layout analysis of tree-structured scene frames in comic images. In: IJCAI'07. pp. 2885–2890 (2007)
17. Tou, J., Gonzalez, R.: Pattern Recognition Principles. Addison-Wesley, USA (1974)
18. Yamada, M., Budiarto, R., Endo, M., Miyazaki, S.: Comic image decomposition for reading comics on cellular phones. IEICE Transactions 87-D(6), 1370–1376 (2004)