

A new pivoting and iterative text detection algorithm for biomedical images

Songhua Xu^{a,b}, Michael Krauthammer^{b,*}

^a Oak Ridge National Laboratory, One Bethel Valley Road, Oak Ridge, TN 37831, USA

^b Department of Pathology, Yale University School of Medicine, CT 06511, USA

ARTICLE INFO

Article history:

Received 10 November 2009

Available online 29 September 2010

Keywords:

Text detection

Histogram analysis for text detection

Pivoting and iterative text region detection

Biomedical image mining

ABSTRACT

There is interest to expand the reach of literature mining to include the analysis of biomedical images, which often contain a paper's key findings. Examples include recent studies that use Optical Character Recognition (OCR) to extract image text, which is used to boost biomedical image retrieval and classification. Such studies rely on the robust identification of text elements in biomedical images, which is a non-trivial task. In this work, we introduce a new text detection algorithm for biomedical images based on iterative projection histograms. We study the effectiveness of our algorithm by evaluating the performance on a set of manually labeled random biomedical images, and compare the performance against other state-of-the-art text detection algorithms. We demonstrate that our projection histogram-based text detection approach is well suited for text detection in biomedical images, and that the iterative application of the algorithm boosts performance to an F score of .60. We provide a C++ implementation of our algorithm freely available for academic use.

© 2010 Elsevier Inc. All rights reserved.

1. Background

1.1. Introduction

Biomedical literature mining is concerned with transforming free text into a structured, machine-readable format, to improve tasks such as information retrieval and extraction. Recent work indicates that there is much interest to also consider image information when mining research articles, as images often depict the results of experiments, and sum up a paper's key findings. There are several obstacles when mining image information. First, there are many different types of images, such as graphs, gel electrophoresis and microscopy images, diagrams or heat maps. There exists no image publication standard, neither with regard to image resolution, or image file format (images are stored at different resolutions, and in a variety of file formats, such as jpeg, tiff etc.). Also, there are no explicit image design guidelines, even though authors seem to follow some universally accepted norms when creating figures such as box plots, heatmaps or gel electrophoresis images.

A unifying element across all biomedical images is image text, i.e. text characters that are embedded in images. Text in images serves several purposes, such as labeling a graph, representing genes in a heat map images, or proteins in a pathway diagram. We have previously shown that extracting image text, and making

it available to image search, improves biomedical image retrieval [1]. In this work, we are concerned with optimizing the performance of a critical step in image text extraction—locating text regions in images, which is known as *text detection* in studies on image processing and Optical Character Recognition (OCR).

Generally speaking, text detection is a crucial step in processing textual information in biomedical images. For example, properly finding the text regions is the first stage of a standard OCR pipeline for extracting image text. Determining the location of text is also important for high-level image content understanding, as it is the text location that indicates the meaning of certain image text element, such as the label of the x - versus y -axis in a graph. Practical applications aside, in this paper, we are exclusively concerned with optimizing the performance of text detection, which is a fundamental research problem in image text processing.

In this work, we introduce a new text detection algorithm suited for biomedical images. We also discuss the methodological details in creating a gold standard biomedical image text detection corpus, and the use of the corpus for evaluating the performance of our algorithm. During the development of the corpus, we laid down clear guidelines on what exactly constitutes an image text region (or element) and how to manually mark the image region linked to the string. We then compared our algorithm against three existing state-of-the-art text detection methods. Even though our algorithm can in principle be applied for processing all images types, it is especially beneficial for images embedded in biomedical publications. Compared to other disciplines, biomedical authors tend to use distributed and nested text in their images in order to annotate experiment settings, conditions and results.

* Corresponding author at: Department of Pathology, Yale University School of Medicine, 300 Cedar Street, New Haven, CT 06510, USA. Fax: +1 203 785 3644.

E-mail addresses: xus1@ornl.gov (S. Xu), michael.krauthammer@yale.edu (M. Krauthammer).

1.2. Related work

1.2.1. Image text detection algorithms

First, we are going to briefly look at prior work on image processing algorithms for image text detection, which is concerned with separating image text elements from other elements in an image. Ohya et al. [2] presented an algorithm for text detection from scene images. In their work, they first detect character components according to gray-level differences and then match the results to standard character patterns captured in a database. Their method is very robust to the font, size and intensity variation in the image texts, but is not able to deal with color and orientation changes. To address the text detection problem for color images, Zhong et al. [3] introduced a connected component-based method for locating texts in a complex color image. Their method analyzes the color histogram of the RGB space to detect text regions. Jung [4] introduced a neural network based approach for identifying text in color images. To attack the text detection problem for texts with different orientations and other distortions, Messelodi and Modena [5] described the use of low level image features such as density and contrast to detect image texts, with the ability to deal with skew in the image text. Hasan and Karam [6] also proposed a morphological approach for image text detection, which is robust to the presence of noise, text orientation, skew and curvature.

There is a body of work using advanced texture and graph segmentation methods to detect text in images. For example, Jain and Karu [7] introduce a method for learning texture discrimination masks for image text detection. Jain and Zhong [8] used a learning based approach to detect image text through image texture analysis. Wu et al. [9] introduced a system for image text detection and recognition, which adopts a multi-scale texture segmentation scheme. In their method, a collection of second-order Gaussian derivatives are used to detect candidate text regions, followed by a *K*-means clustering process and a multi-resolutional stroke generation, filtering and aggregation process to further refine the detected text region. Felzenszwalb and Huttenlocher [10] proposed a graph-based image segmentation algorithm for efficiently separating textual elements from graphical elements in an image. Their algorithm can automatically adapt itself to the image structure variation. Liu et al. [11] proposed a novel method for text detection and segmentation through using stroke filters for text polarity assessment in analyzing features in local image regions.

There also exists a growing collection of work on text detection from videos or motion images, which are closely related to the image text detection problem studied in this paper. For example, Li et al. [12] used a hybrid neural network and projection profile analysis based approach to detect and track text regions in a video. Antani et al. [13] applied a variety of text detection methods and then fused the individual text detection results together to achieve a robust text detection for videos. Kim et al. [14] introduced a support vector machine based approach for image text detection in videos. Lyu et al. [15] proposed a coarse-to-fine localization scheme for detecting texts in multilingual videos. Recently, Qian et al. [16] proposed a discrete cosines transform coefficients based method for text detection in compressed videos. Despite the many commonalities between the video text and image text detection problems, one of the main differences between them is that frame images in a video demonstrate temporal coherence, which offer much useful information for text detection. Such clues are not present in still images, and hence make the image text detection problem more challenging than its counterpart in videos.

1.2.2. Biomedical image processing algorithms and systems

Our study is related to other projects in biomedical image processing. For example, Shatkay et al. [17] used image features for text categorization. Tulipano et al. [18] studied the use of natural

language processing to index and retrieve molecular images. Qian and Murphy [19] described an algorithmic system for accessing fluorescence microscopy images via image classification and segmentation.

In our own prior work [1], we discussed a novel approach for biomedical image search based on OCR. We have shown that the approach offers additional advantages compared to searching over image captions alone, notably the retrieval of additional and relevant images. The current study is closely linked to that project, discussing the algorithmic details for detecting image text regions.

2. Approach

2.1. Overview

An overview of our method is shown in Fig. 1. An input image (i.e. an image from a biomedical publication) undergoes detection of layout lines and panel boundaries, which are excluded from the image to increase text detection robustness. We implement the algorithm proposed by Busch et al. [20] for detecting these layout elements. The image is then converted to black and white, and subjected to an edge detection algorithm. The resulting edge image is then subjected to a pivoting text region detection (PTD) algorithm for extraction of text regions. PTD is repeated several times, in order to divide detected text regions into text sub-regions. If no more text regions are detected, the algorithm exits. Our algorithm is based on traditional histogram analysis-based text region detection, which takes edge images as input. We extend the traditional approach as follows: We perform a pivoting procedure while applying the histogram analysis, and repeat the procedure until no more text (sub)regions are detected.

2.2. Traditional histogram analysis-based text region detection

One of the most popular and well known text region detection methods is through analyzing the vertical and horizontal projection histograms of an image. More concretely, given an input image, we first detect the edge pixels in the image. Then a vertical and a horizontal projection histogram are derived. It is assumed that text regions generally exhibit higher density of edge pixels than non-text regions. The vertical and horizontal histograms will thus show the highest density of edge pixels in text areas. A density threshold defines the exact dimensions of the text area along the vertical and horizontal histogram. The elements of this basic procedure are discussed in more detail in the next section.

One distinct feature of many biomedical images is that they often employ a distributed and nested text layout. Figs. 3a and 4a show two typical examples, where text is distributed across many different image regions. Also, text regions often display some degree of nestedness. For example, the numbers along the x axis in Fig. 4b can be grouped in one large text area, or more correctly into separate (inner) text areas surrounding each individual number (Fig. 4d). The traditional histogram-based analysis technique does not cope well with distributed and nested text layout. To address this problem, we introduce a new iterative pivoting histogram analysis procedure for text region detection.

2.3. Pivoting text region detection (PTD)

We introduce a pivoting step into the classical histogram-based text detection algorithm in order to account for the distributed nature of biomedical image text. The pivoting procedure subdivides image regions into its text subcomponents, instead of identifying large text blocks. Our procedure is realized through analyzing the histograms of the input image region following the vertical and

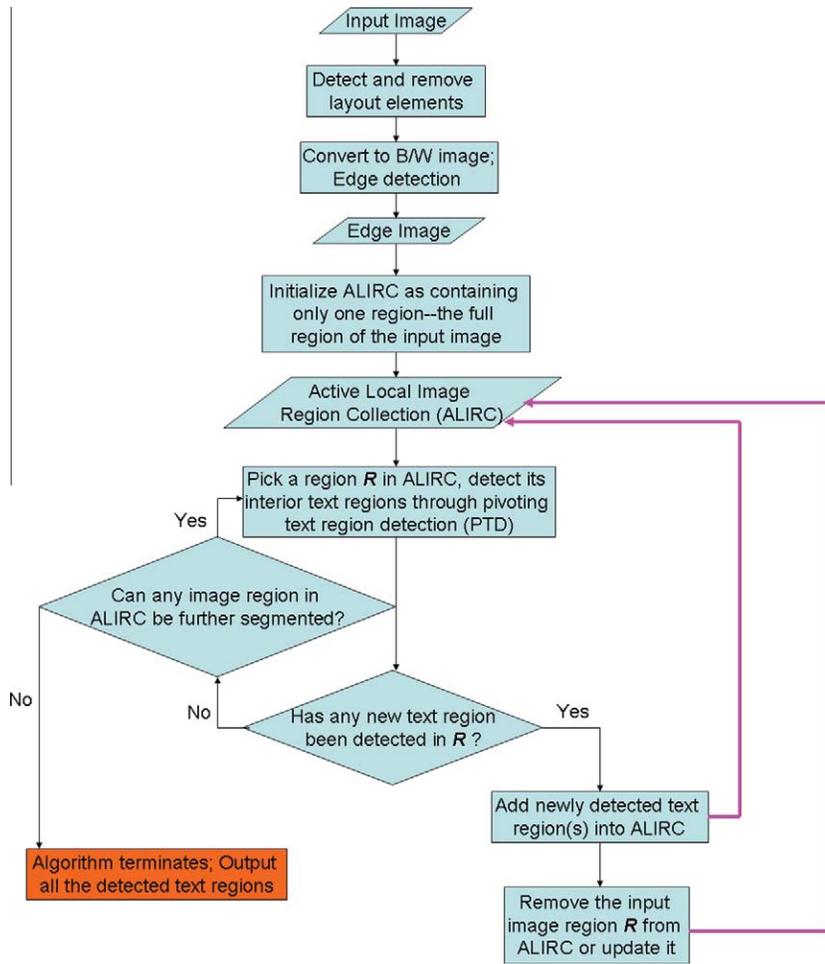


Fig. 1. Diagram illustrating the overall procedure of our new text detection algorithm.

horizontal directions alternatively, hence the name “pivoting”. Fig. 2 illustrates the key steps, and Fig. 1 in Appendix B (Supplementary Files) shows the working of the algorithm on a sample image. An input image is converted into black and white and

subjected to edges detection (Fig. 1d, Appendix B). For a specified region \mathcal{R} (the whole image in the first iteration of the procedure), to detect text areas in \mathcal{R} , we first vertically project all the edge pixels to derive the image region’s horizontal histogram \mathcal{H}_h (Fig. 1e,

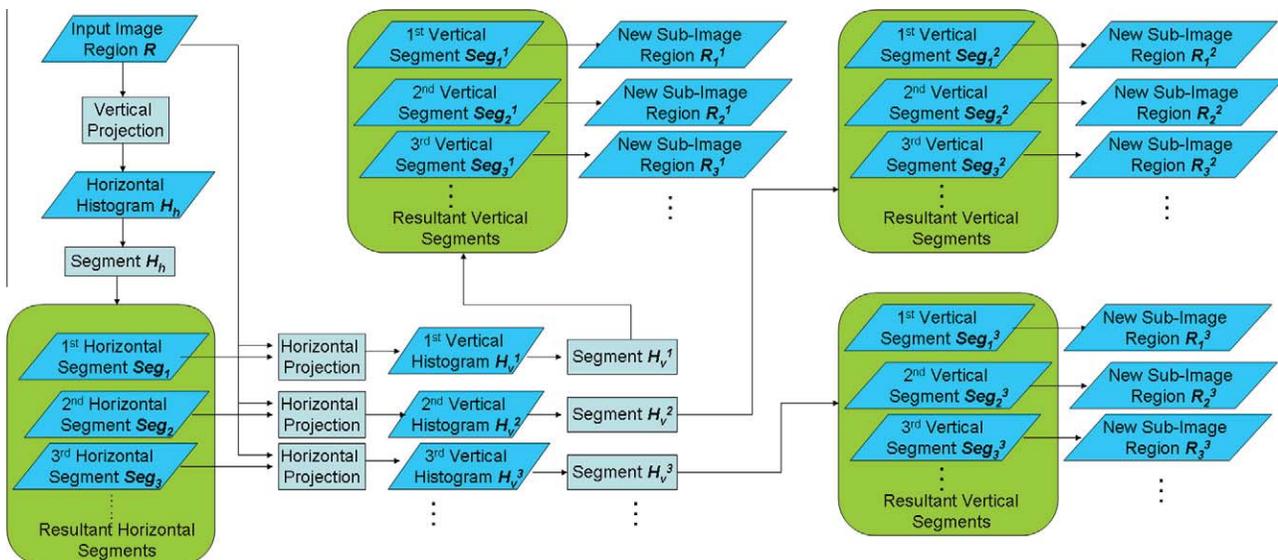


Fig. 2. Diagram illustrating one step of the PTD algorithm (Section 2.3).

Appendix B). We then segment the horizontal histogram into several segments, each corresponding to a horizontal region in the input image, denoted as Seg_1, Seg_2, \dots . The segments are defined by a threshold on the histogram densities. We then derive for each horizontal segment a vertical histogram through horizontally projection of all the edge pixels in the region. (This step is different from the traditional approach, where the horizontal projection is performed on the whole image). The resultant vertical histogram corresponding to the horizontal segment Seg_i of the image is denoted as \mathcal{H}_v^i (Fig. 1g, Appendix B). We then segment the vertical histogram \mathcal{H}_v^i the vertical segments Seg_1^i, Seg_2^i, \dots using a threshold on the densities (Fig. 1g1-3, Appendix B). Each such segment corresponds to a vertical region in the input image. Through pairing of a vertical segment Seg_j^i with its corresponding horizontal segment Seg_i , we are able to specify a rectangular region (bounding box) \mathcal{R}_j^i in \mathcal{R} (Fig. 1h1-3, Appendix B), corresponding to text regions.

In Appendix A (Supplementary Files), we formally describe this procedure mathematically.

2.4. Iterative PTD procedure

Our algorithm iteratively constructs vertical and horizontal histograms to find nested text regions. As can be seen in Fig. 1h2, Appendix B, the first round of the PTD algorithm could not resolve the true text areas of the image region. In the image, region 1 groups distinct image text elements, and we propose to repeat the PTD step for separating these elements.

More concretely, our algorithm maintains an active local image region collection (ALIRC) during its running time (Fig. 1, main paper). Initially, the collection contains a single image region, which is the full image area of the input image. The algorithm then constructs pivoting vertical and horizontal histograms (see previous section) and detects text regions. Each detected text region is regarded as a new target region and added into ALIRC. The input image region is removed from ALIRC, with one exception: if, after subtracting the text regions from the input image, the input image is non-empty we populate ALIRC with an updated version of the input image, with the text areas subtracted. We iteratively apply our histogram-based text region segmentation procedure on all the image regions in the ALIRC until finer separation between text and non-text regions can be achieved. We will then output all the image regions maintained in the ALIRC. A final heuristic removes regions that are maintained in ALIRC but do not correspond to text regions. The heuristic evaluates the overall edge density, removing regions that exhibit a density that is too low or too high.

2.5. Formal description of the iterative PTD procedure

1. Assuming the height and width of the input image is h and w , respectively, we apply our pivoting text detection algorithm introduced in Section 2.3 to detect all the text regions in the full area of the input image. That is, we apply the PTD procedure onto \mathcal{I} with the text detection scope being $\mathcal{R} = (0, w, 0, h)$. We further assume the set of regions segmented from the input image are $\phi = \{\mathcal{R}_1, \mathcal{R}_2, \dots, \mathcal{R}_n\}$ where each \mathcal{R}_i specifies the scope of a rectangular image region resulting from the PTD process. We call ϕ the current active local image region collection.
2. For each image region \mathcal{R}_i in ϕ ($i = 1, \dots, n$), we apply the PTD procedure onto \mathcal{R}_i to further separate the text and non-text areas inside the region on a finer granularity. Assume this new round of text region detection produces k sub image regions, which are denoted as $\mathcal{R}_1^i, \mathcal{R}_2^i, \dots, \mathcal{R}_k^i$, respectively. Given such text and non-text region segmentation result, we first remove from ϕ the input region \mathcal{R}_i . And then we add all the resultant sub image regions $\mathcal{R}_1^i, \mathcal{R}_2^i, \dots, \mathcal{R}_k^i$ into ϕ . Lastly,

we also add into ϕ the smallest rectangular region that covers all the edge pixels belonging to the original input region \mathcal{R}_i but falling outside all the newly detected image regions $\mathcal{R}_1^i, \mathcal{R}_2^i, \dots, \mathcal{R}_k^i$.

3. We repeat the above process to recursively refine every image region maintained in the current active local image region collection ϕ until ϕ can be no longer changed through additional calls of our PTD procedure. We then output all the image regions in the final stage of the image region collection ϕ which are determined as text regions by our PTD process. These image regions constitute our final text region detection result for the input image \mathcal{I} .

In Appendix B (Supplementary Files), we show a step-by-step example of text region detection using our iterative and pivoting text detection algorithm for a biomedical image.

Appendices C and D (Supplementary Files) contain further examples of text detection results after applying our iterative PTD algorithm on biomedical images.

3. Evaluation method

In this section, we will first discuss the creation of a gold standard biomedical image text detection corpus. We will then discuss our evaluation strategy to measure the performance of our iterative PTD algorithm for detecting text regions in biomedical images.

3.1. Creation of a gold standard biomedical image text detection corpus

To objectively evaluate the performance of our algorithm, and to quantitatively compare the performance of our method to other peer methods, we created a gold standard corpus of biomedical images with manual markup of text regions. In order to create this corpus, we selected a two step approach. The first step dealt with the identification of the text regions in the image. We set up guidelines for manual identification of text regions (image text) in biomedical images, which are listed in Table 1. The guidelines define the nature of an image text region in a biomedical image, what to do about Greek letters and other special characters, and strings in super or subscript. After selecting 161 random images from biomedical articles indexed in PubMed Central, we used the guidelines to identify the image text regions. In the second step, we identified a minimum rectangular region (bounding box) for each detected text region. Such a bounding box is defined as the smallest rectangular region covering all character pixels of the text region. These image bounding boxes represent the gold standard image text regions.

3.2. Evaluation strategy

To evaluate the performance of our PTD text detection algorithm, we can proceed as follows: We compare the predicted text region bounding boxes with the bounding boxes of the gold standard corpus. In our study, we employ two approaches for measuring the degree of overlap between the predicted and gold standard text regions, looking at both the pixel overlap and the percentage of shared region.

3.2.1. Measuring recall, precision and F-rate from shared pixels

One approach for measuring the overlap of two text detection results is to measure the recall, precision and F-rate as determined by shared pixels. More concretely, *recall* is defined as the fraction of pixels in the gold standard text area that are contained in the (algorithmically) detected text region. *Precision* is defined as the fraction

Table 1

Guidelines for manual identification of image text regions.

1. Image texts that form a coherent entity such as “bcl-xl (+)”, “ $p < 0.01$ ”, and “S.E. RECOVER SOLUTION (i,j,k)” are considered individual text regions
2. Symbols that are attached to a word, such as brackets, forward slashes, and dashes, e.g. “MMC-transgenic” and “vif(+)” are part of the same term
3. Include Greek letters, and letters in subscript and superscript
4. Labels consisting of numbers or single letters are considered standalone image text regions

of pixels in the detected text region that are also contained in the groundtruth text area. And *F-rate* is defined as the harmonic mean of precision and recall, i.e. $F\text{-rate} = 2 \text{ Precision Recall} / (\text{Precision} + \text{Recall})$.

3.2.2. Measuring Modulated Overlapping Area

Another intuitive measure of overlap between two text detection results is to calculate the overlapping area modulated by the reciprocal of the area of the union of the two text detection results. Mathematically, this measurement can be formulated as:

$$\text{MOA} \triangleq \frac{\text{Area}(\text{Text Region}_{\text{groundtruth}} \cap \text{Text Region}_{\text{algorithm}})}{\text{Area}(\text{Text Region}_{\text{groundtruth}} \cup \text{Text Region}_{\text{algorithm}})} \quad (1)$$

In the above, $\text{Text Region}_{\text{groundtruth}}$ stands for text region in the gold standard corpus and $\text{Text Region}_{\text{algorithm}}$ stands for the algorithmically detected text region. The operator $\text{Area}(X)$ computes the area of the region X in pixels. The range of the MOA measurement as defined above is between 0 and 1. When $\text{Text Region}_{\text{groundtruth}}$ fully agrees with $\text{Text Region}_{\text{algorithm}}$, MOA reaches the maximum value of 1. When $\text{Text Region}_{\text{groundtruth}}$ is entirely disjoint from $\text{Text Region}_{\text{algorithm}}$, MOA reaches the minimum value of 0.

4. Results

4.1. Text detection performance in biomedical images

We start with a qualitative assessment on the performance of our text detection algorithm. To this end, we provide sample images along with automatically detected text regions (Figs. 3 and 4). The blue boxes outline the detected text regions, while the purple lines and areas indicate non-textual elements. A qualitative assessment of our approach is helpful for identifying the strength and weaknesses of our algorithm. For example, we see satisfactory text detection performance in Fig. 3b. However, two strings “the” and “number” in the bottom horizontal label of the image are mistakenly detected as one single text region “the number”. In Fig. 4, we show the intermediate text detection results of two rounds of the PTD algorithm, from which we can see that our algorithm progressively refines its text detection results.

4.2. Quantitative evaluation and performance comparison with peer text detection algorithms

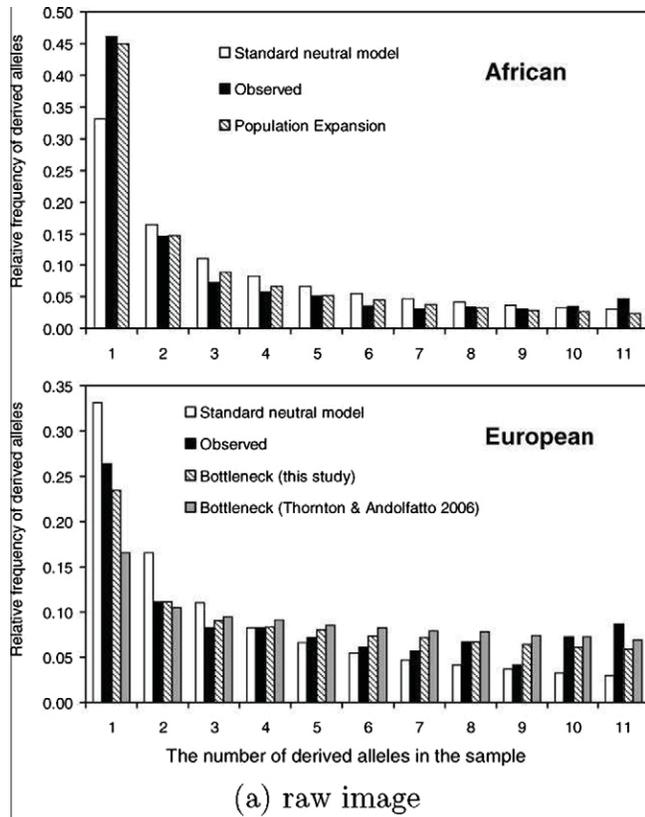
To explore the effectiveness and advantages of our approach, we also compare the performance of our algorithm with a few state-of-the-art text detection algorithms. To this end, we identified four recently published algorithms for text detection, including the DCT feature based text detection method proposed by Goto [21], the text particle based multi-band fusion method for text detection as proposed by Xu et al. [22], the visual saliency based and biologically inspired text detection method proposed by Fatma [23], and the fast text detection method proposed by Li et al. [24]. When conducting our experiments to quantitatively compare the performance of our algorithm with that of the four peer methods, we obtained either the source code or the executable program from the corresponding author(s). After that, we

worked directly with authors of these papers for running their algorithms, ensuring that each algorithm is correctly configured and properly executed during the evaluation experiment. Upon reaching the final comparison results, we shared our intermediate and final evaluation results with the authors.

Goto’s method [21] applies Fisher’s discriminant analysis to explore the frequency dependency of DCT-based features for identifying the optimal frequency band for text detection applications. Through their experiments, they empirically found out that a mixture of high frequency and lower-middle frequency bands is most effective for text detection from scene pictures. Based on this finding, they redefined a DCT-based feature for more accurately detecting text regions from natural scene images. Using their newly defined DCT-based text detection feature, they eventually employ an unsupervised thresholding based mechanism to differentiate text regions from background image regions. Since their method is targeted at detecting text from natural scene images, their algorithm does not pay special attention to the affect of having sophisticated text layout in an image. In fact, when text is arranged following a distributed and nested structure, as is often true in biomedical images, the overall spatial frequency features exhibited in the image region will be much affected. Under such circumstances, determining text regions according to the overall spatial frequency of a region tends not to be reliable. This fact limits the algorithm performance on biomedical images, as demonstrated in our evaluation results, shown below. In contrast to the cited DCT-based feature approach, our algorithm introduces a pivoting and iterative procedure, which progressively divides an image region into sub-regions. For each resultant image sub-region, we recompute the local image features in the form of vertical and horizontal histograms to adaptively analyze whether an image area contains text. Through this active sub-division based search process, our method can more thoroughly break down and capture the local image features to recover scattered and nested image text, as seen in images in biomedical publications.

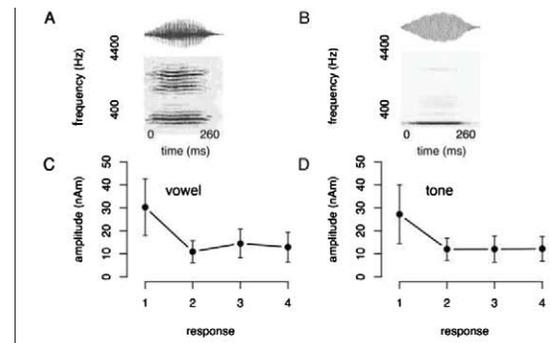
Xu et al.’s method [22] aims at detecting text from images with weak contrast and low text-background variance. To attain their goal, they first analyze an input image using the YUV color space via the Haar wavelet transformation. Given the derived wavelet coefficients, they introduce text particles, according to the local binary patterns demonstrated by the wavelet coefficients, which are known as Local Haar Binary Patterns. Once the concept of text particles is defined, their method further applies a density-based multi-band fusion procedure to generate the final text detection result. The main benefit of their approach for text detection is that their method can robustly detect text regions regardless of the image size, color, rotations, illuminations, and text-background contrasts. However, their method assumes that an input image does not exhibit a sophisticated text layout and uses a single pass analysis. Their assumption generally holds for street side images of signs and signages. However, for images in biomedical publications, which display distributed and nested text, the same assumption does not hold.

While our histogram-based analysis’ text discrimination capability is below the method of [22], our overall text detection capability is not much affected. This observation can be intuitively understood in that images carried in biomedical publications are

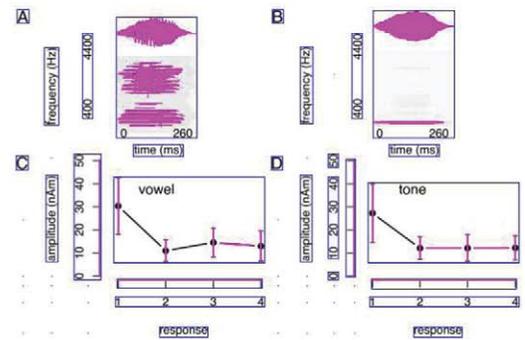


(b) text detection result by our algorithm

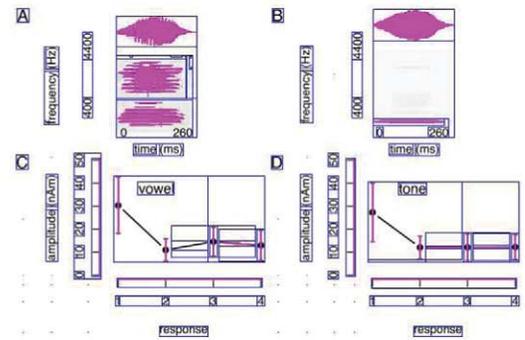
Fig. 3. A text detection example produced by our algorithm along with the original image. Image from [26].



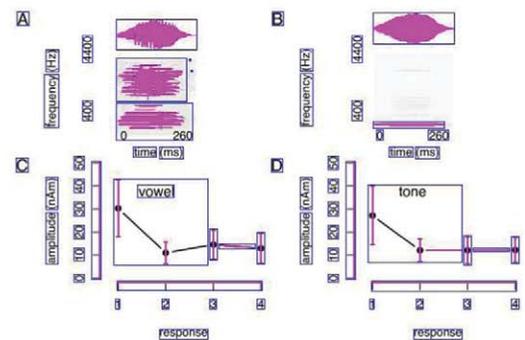
(a) The original image



(b) Text detection result after the first round



(c) Text detection result after the second round



(d) Final text detection result

Fig. 4. Text detection example with intermediate step-by-step text detection results. Image from [27].

usually carefully prepared so that they exhibit a good contrast and a sharp text background. Therefore, the strength of their more powerful text particle based multi-band fusion approach may not be advantageous in the biomedical domain.

Fatma's algorithm [23] treats the text detection problem as a texture classification task, where the task of locating text regions is formulated as finding regions whose texture is classified as text texture, rather than background image texture. Fatma chose the Support Vector Classification method to implement her texture classifier. The main novelty of her work is the introduction of a computational model of human visual attention for text detection, which examines image features at various spatial scales and modalities in a bottom-up manner. Like Xu et al.'s method [22], Fatma's algorithm also assumes that there is no nested text placement. When text placement is distributed and nested, image sub-regions are more likely to be misrecognized as carrying image texture, rather than text texture. More fundamentally, since Fatma's algorithm works with texture features of an image region, her algorithm is not directly concerned with how text strings are positioned inside a region; what matters to her algorithm is the overall texture exhibited by a region, regardless of whether it is composed by several text strings, by background image elements, or by a mixture of both. In fact, if some distributed and nested text happens to exhibit a texture that looks like a plausible image pattern, her algorithm would readily recognize the region as a background image region.

Li et al. introduced a stroke filter based approach for text detection [24]. Their algorithm applies stroke filters to examine text presence along the horizontal, vertical, left-diagonal, and right-diagonal directions. Their work focuses on detecting text from TV shots or video keyframes where almost all the text is placed following the four principal directions (vertical, horizontal, left-diagonal, and right-diagonal directions). This is probably because TV or video frames are designed to be rapidly parsed by the audience. Consequently, these frames, which typically carry only a very limited amount of text, are organized using quite simple and intuitive text layouts to facilitate viewer's rapid reading or scanning. In comparison, images in biomedical publications are meant to be read much more carefully, with image text being often distributed and nested. As a result, Li et al.'s four directional stroke filter based approach seems not sufficient to handle all the subtleties of extracting text from biomedical images.

We also implemented two simplified version of our algorithm to study the different components of our procedure. To distinguish between these different versions of our algorithm, we call the iterative text detection method introduced in Section 2.4 the multistep method, which is denoted as "multiple steps". We also study the

performance of our method when the number of iterations is limited to one round. We call this modification of our algorithm the one step iteration version, denoted as "one step". Finally, we also implemented the classical histogram-based analysis without pivoting where the vertical histogram is derived for the full image rather than for the segments from the horizontal histogram (see Section 2.2). We refer to this naive version as "naive".

The results of these evaluations are shown in Table 2. We observe the following: The naive method outperforms the other peer methods in terms of *F*-rate and MAO. The pivoting procedure improves upon the naive version, with a performance increase of 0.045 *F*-rate and .051 MAO. The iterative procedure further improves upon the pivoting result, both in terms of *F*-rate and MAO. There is no performance increase when conducting more than 2 iterations of our algorithm.

5. Discussion

5.1. Iterative PTD algorithm performance

Our evaluation showed that the iterative PTD algorithm performs well on the gold standard text detection corpus (Table 2). The naive (classical) version is outperformed by the pivoting algorithm, which performs the vertical histogram on each image text segment as determined by the horizontal histogram (Section 2.3 and Fig. 2). The pivoting algorithm subdivides image text regions into subcomponents, instead of identifying large text blocks as in the naive or classical approach. This sub-division into smaller units seems to cope better with the distributed nature of the biomedical image text. The iterative application of our algorithm results in further performance gains. As discussed, iteration ensures the detection of nested image regions. As can be seen in Table 2, performance seems to stabilize after one iteration. This can be understood as follows: Biomedical images seem to contain (on average) one level of text nesting, which can be recovered by one iteration of our PTD algorithm.

5.2. Comparison with prior work in text detection in images

We conducted an extensive comparison with existing text detection algorithms. None of the tested algorithms were able to outperform the histogram-based text detection approach. It should be noted that these algorithms are optimized for a particular text detection task, which might be different from the one encountered in biomedical images. Consequently, the performance of these algorithms as presented in the literature is higher than the numbers presented in Table 2. Our results indicate that it is difficult to use these algorithms on biomedical images without modifications.

For comparison, we quickly review the performance of the tested algorithms on other image sets. In [21], the author reports algorithm performance for two typical settings of his algorithm—a low frequency mode and a high frequency mode. The evaluation is performed on the ICDAR-set, which is from the TrialTrain data used in the ICDAR 2003 Robust Reading Competition, see [25]. For the low frequency mode, the average precision, recall and *F*-rate of his algorithm is 32.6%, 91.9%, and 43.4%, respectively. For the high frequency mode, the average precision, recall and *F*-rate is 35.6%, 88.6%, and 45.1%, respectively. It should be noted that the [21] algorithm performs well on our gold standard corpus in terms of recall. Precision is low, though, indicating many falls positive calls.

[22] evaluated their method on the Location Detection Database of ICDAR 2003 Robust Reading Competition Dataset, see [25]. The precision, recall and *F*-rate on the dataset is 60%, 81%, 69%, respectively. Finally, [24] reported the performance of their algorithm on an image set consisting of 308 images from the Web, recorded

Table 2
Performance comparison between different text detection methods.

(a) Performance of four existing text detection methods				
Measurement	Existing methods			
	[21]	[22]	[23]	[24]
Precision	0.291	0.110	0.116	0.457
Recall	0.980	0.464	0.528	0.210
<i>F</i> -rate	0.418	0.154	0.158	0.256
MOA	0.263	0.084	0.091	0.125
(b) Performance of our new text detection method				
	Our method			
	Naive	One step	Two steps	Multiple steps
Precision	0.528	0.598	0.637	0.637
Recall	0.626	0.655	0.672	0.670
<i>F</i> -rate	0.519	0.564	0.600	0.600
MOA	0.332	0.383	0.430	0.429

broadcast videos, and digital videos. The reported a recall and accuracy of 91.1% and 95.8%, respectively.

5.3. Applications of our algorithm

The general category of target images suitable for our algorithm's processing includes images with distributed and nested text use, as typically seen in biomedical images. Our biomedical image text detection algorithm can benefit a number of applications in the field of biomedical information processing and management. Below we list some immediate applications:

- A first benefit is the improvement of biomedical OCR performance.
- With accurate text region detection, we can highlight the locations of image text when retrieving biomedical images. With this functionality, users can easily see the positions of their querying text. This application is already available in EverNote (www.evernote.com).
- Once text regions in biomedical images are detected, we can more effectively compress these image regions by first recognizing the text contents, and then representing these text areas through vector graphics.
- Given the text region detection result, we can build some application where a user hovers his or her mouse over a text term in an image, to reveal more details related to the text term.
- Reliably detecting image text regions would allow accurate analysis of document layout. This will enable automatic retrieval of images or documents according to their layout. A reliable image/document layout analysis result provides many valuable features for image/document clustering and categorization.
- Given the text region detection result, we can design high-level image content understanding algorithms. For example, we may analyze the spatial relationships between text terms occurring in an image. This would provide clues to parse the graphical messages embedded in the images, leading to more advanced biomedical image retrieval and recommendation results.

6. Conclusions

Biomedical image search and mining is becoming an increasingly important topic in biomedical informatics. Accessing the biomedical literature via image content is complementary to text-based search and retrieval. A key element in unlocking biomedical image content is to detect and extract (via OCR) text from biomedical images, and making the text available for image search. In this paper, we are concerned with text detection, i.e. finding the precise areas of image text elements. We propose a new text detection algorithm which is ideally suited for this purpose. The key feature of our algorithm is that it searches for text regions in a pivoting and iterative fashion. The pivoting procedure allows for recovery of distributed image text, and the iterative procedure uncovers nested image information. We believe that these two algorithm features are crucial for detecting text in biomedical images.

Acknowledgments

This research has been funded by NLM Grant Nos. 5K22LM009255 and 1R01LM009956. Songhua Xu performed this research partly as a Eugene P. Wigner Fellow and staff member at the Oak Ridge National Laboratory, managed by UT-Battelle, LLC, for the US Department of Energy under Contract DE-AC05-00OR22725. We thank the authors of Ref. [21–24] for sharing their

source codes or executables for conducting the performance comparison studies.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at [doi:10.1016/j.jbi.2010.09.006](https://doi.org/10.1016/j.jbi.2010.09.006).

References

- [1] Xu S, McCusker J, Krauthammer M. Yale image finder (YIF): a new search engine for retrieving biomedical images. *Bioinformatics* 2008;24(17):1968–70.
- [2] Ohya J, Shio A, Akamatsu S. Recognizing characters in scene images. *IEEE Trans Pattern Anal Mach Intell* 1994;16(2):214–20. [doi:10.1109/34.273729](https://doi.org/10.1109/34.273729).
- [3] Zhong Y, Karu K, Jain AK. Locating text in complex color images. *Proc Third Int Conf Document Anal Recognit* 1995;1:146–9. [doi:10.1109/ICDAR.1995.598963](https://doi.org/10.1109/ICDAR.1995.598963).
- [4] Jung K. Neural network-based text location in color images. *Pattern Recognit Lett* 2001;22(14):1503–15. [doi:10.1016/S0167-8655\(01\)00096-4](https://doi.org/10.1016/S0167-8655(01)00096-4).
- [5] Messelodi S, Modena C. Automatic identification and skew estimation of text lines in real scene images. *Pattern Recognit* 1999;32(5):791–810.
- [6] Hasan Y, Karam L. Morphological text extraction from images. *IEEE Trans Image Process* 2000;9(11):1978–83. [doi:10.1109/83.877220](https://doi.org/10.1109/83.877220).
- [7] Jain AK, Karu K. Learning texture discrimination masks. *IEEE Trans Pattern Anal Mach Intell* 1996;18(2):195–205. [doi:10.1109/34.481543](https://doi.org/10.1109/34.481543).
- [8] Jain AK, Zhong Y. Page segmentation using texture analysis. *Pattern Recognit* 1996;29(5):743–70.
- [9] Wu V, Manmatha R, Riseman EM. Textfinder: an automatic system to detect and recognize text in images. *IEEE Trans Pattern Anal Mach Intell* 1999;21(11):1224–9. [doi:10.1109/34.809116](https://doi.org/10.1109/34.809116).
- [10] Felzenszwalb PF, Huttenlocher DP. Efficient graph-based image segmentation. *Int J Comput Vision* 2004;59(2):167–81. [doi:10.1023/B:VISI.0000022288.19776.77](https://doi.org/10.1023/B:VISI.0000022288.19776.77).
- [11] Liu Q, Jung C, Kim S, Moon Y, Yeun Kim J. Stroke filter for text localization in video images. In: *Proceedings of IEEE international conference on image processing*; 2006. p. 1473–6. [doi:10.1109/ICIP.2006.312560](https://doi.org/10.1109/ICIP.2006.312560).
- [12] Li H, Doermann D, Kia O. Automatic text detection and tracking in digital video. *IEEE Trans Image Process* 2000;9(1):147–56.
- [13] Antani S, Crandall D, Kasturi R. Robust extraction of text in video. In: *Proceedings of the 15th international conference on pattern recognition*, vol. 1; 2000. p. 831–4. [doi:10.1109/ICPR.2000.905537](https://doi.org/10.1109/ICPR.2000.905537).
- [14] Kim KI, Jung K, Park SH, Kim HJ. Support vector machine-based text detection in digital video. *Pattern Recognit* 2001;34(2):527–9.
- [15] Lyu M, Song J, Cai M. A comprehensive method for multilingual video text detection, localization, and extraction. *IEEE Trans Circuits Syst Video Technol* 2005;15(2):243–55. [doi:10.1109/TCSVT.2004.841653](https://doi.org/10.1109/TCSVT.2004.841653).
- [16] Qian X, Liu G, Wang H, Su R. Text detection, localization, and tracking in compressed video. *Image Commun* 2007;22(9):752–68. [doi:10.1016/j.image.2007.06.005](https://doi.org/10.1016/j.image.2007.06.005).
- [17] Shatkay H, Chen N, Blostein D. Integrating image data into biomedical text categorization. *Bioinformatics* 2006;22(14):e446–53. [doi:10.1093/bioinformatics/btl235](https://doi.org/10.1093/bioinformatics/btl235).
- [18] Tulipano PK, Tao Y, Millar WS, Zanzonico P, Kolbert K, Xu H, et al. Natural language processing and visualization in the molecular imaging domain. *J Biomed Inform* 2007;40(3):270–81. [doi:10.1016/j.jbi.2006.08.002](https://doi.org/10.1016/j.jbi.2006.08.002).
- [19] Qian Y, Murphy RF. Improved recognition of figures containing fluorescence microscope images in online journal articles using graphical models. *Bioinformatics* 2008;24(4):569–76. [doi:10.1093/bioinformatics/btm561](https://doi.org/10.1093/bioinformatics/btm561).
- [20] Busch A, Boles WW, Sridharan S, Chandran V. Detection of unknown forms from document images. In: *Proceedings of workshop on digital image computing*; 2003. p. 141–4.
- [21] Goto H. Redefining the DCT-based feature for scene text detection—analysis and comparison of spatial frequency-based features. *Int J Document Anal Recognit (IJ DAR)* 2008;11(1):1–8.
- [22] Xu P, Ji R, Yao H, Sun X, Liu T, Liu X. Text particles multi-band fusion for robust text detection. *Lect Notes Comput Sci Image Anal Recognit* 2008;5112:587–96.
- [23] Fatma IK. Visual saliency and biological inspired text detection. Master's thesis. Technical University Munich & California Institute of Technology; 2008.
- [24] Li X, Wang W, Jiang S, Huang Q, Gao W. Fast and effective text detection. In: *Proceedings of the 15th IEEE International Conference on Image Processing (ICIP)*; 2008. p. 969–72.
- [25] Lucas SM, Panaretos A, Sosa L, Tang A, Wong S, Young R. ICDAR 2003 robust reading competitions. In: *ICDAR'03: Proceedings of the seventh international conference on document analysis and recognition*. Washington, DC, USA: IEEE Computer Society; 2003. p. 682–7.
- [26] Li H, Stephan W. Inferring the demographic history and rate of adaptive substitution in *Drosophila*. *PLoS Genet* 2006;2(10):e166.
- [27] Sörös P, Michael N, Tollkötter M, Pfeleiderer B. The neurochemical basis of human cortical auditory processing: combining proton magnetic resonance spectroscopy and magnetoencephalography. *BMC Biol* 2006;4:25.

A New Pivoting and Iterative Text Detection Algorithm for Biomedical Images: Appendix A

Algorithm Details for our Pivoting Text Detection Algorithm

Inspired by the classical histogram analysis based text region detection methods ([2, 1]), we describe a procedure for locating text regions in an image through analyzing both the vertical and horizontal projection histograms of an image:

- **Input** An input image \mathcal{I} and a specified rectangular region $\mathcal{R} = (left, right, top, bottom)$ inside the region of \mathcal{I} .
- **Function of the Procedure** To detect all the text regions inside the interior region \mathcal{R} of the input image \mathcal{I} .
- **Output** A collection of text regions $\{\mathcal{R}_j^i\}$ so detected where each \mathcal{R}_j^i is a text region detected from within the region \mathcal{R} of the input image \mathcal{I} .

We can formally state the above procedure in the form of (1).

$$\text{Text Region Detection Procedure} : \mathcal{I}, \mathcal{R} \rightarrow \{\mathcal{R}_j^i\}. \quad (1)$$

We will now look at the details of our text detection procedure.

1. First, we convert the input image \mathcal{I} into black and white if it is originally a color image. We then apply a 3x3 median filter to blur the

image background in order to make our text detection procedure less sensitive to image noise.

2. Next, we detect edges in the converted black and white image. Currently, we use the classical Sobel operator for this purpose due to its simplicity and satisfying performance in our experiments. Other edge detectors, such as Canny and Canny-Deriche edge detectors, can also be used without noticeably affecting the overall performance of our algorithm. We call the resultant image from this step the edge image of the original input image \mathcal{I} , which is denoted as $\widehat{\mathcal{I}}$.
3. We then compute the vertical projection for each pixel in the edge image $\widehat{\mathcal{I}}$ to derive $\widehat{\mathcal{I}}$'s horizontal histogram. More concretely, given the width w and height v (both in pixels) of $\widehat{\mathcal{I}}$, the horizontal projection histogram of the edge image $\widehat{\mathcal{I}}$ is denoted as $\mathcal{H}_h(i)$ ($i = 1, \dots, w$) where $\mathcal{H}_h(i)$ records the number of edge pixels on the vertical line that stays i pixels away from the left boundary of the image, i.e. $\mathcal{H}_h(i) = |\{pixel(i, y) | pixel(i, y) \text{ is an edge pixel in } \widehat{\mathcal{I}}; y = 1, \dots, v\}|$. Here $pixel(x, y)$ denotes the pixel whose horizontal and vertical coordinates are x and y respectively; and $|\mathbf{X}|$ returns the cardinality of the set \mathbf{X} . The overall horizontal histogram of the edge image $\widehat{\mathcal{I}}$ is thus represented as a w dimensional vector in the form of $\mathcal{H}_h \triangleq [\mathcal{H}_h(1), \dots, \mathcal{H}_h(w)]$.
4. We then segment the derived horizontal projection histogram \mathcal{H}_h according to a preset segmentation threshold τ_h . To carry out this segmentation, we first derive a binary sequence \mathcal{B}_h according to the horizontal projection histogram \mathcal{H}_h . Here we define \mathcal{B}_h to be a w dimen-

sional vector in the form of $\mathcal{B}_h \triangleq [\mathcal{B}_h(1), \dots, \mathcal{B}_h(w)]$. For each $\mathcal{B}_h(i)$, it is derived as follows:

$$\mathcal{B}_h(i) \triangleq \begin{cases} 1 & \text{if } \mathcal{H}_h(i) \geq \tau_h; \\ 0 & \text{otherwise.} \end{cases} \quad (i = 1, \dots, w) \quad (2)$$

We then detect all the segments of consecutive 1's in \mathcal{B}_h and denote the resultant sequence of segments as Seg_1, \dots, Seg_n where we assume there are n such resulting segments. Here Seg_i corresponds to the i -th segment of consecutive 1's in \mathcal{B}_h . For each such segment Seg_i , we represent its left and right boundaries in the binary sequence \mathcal{B}_h as $left_i$ and $right_i$ respectively. That is, Seg_i corresponds to the block of consecutive 1's which starts at the $left_i$ -th component in \mathcal{B}_h and ends at the $right_i$ -th component in \mathcal{B}_h . It is easy to see that $right_{i-1} < left_i$ as otherwise Seg_{i-1} should have been joined with Seg_i . Also, if any resultant segment's width is less than 3 pixels apart, i.e. $left_i - right_{i-1} < 3$, we will eliminate this segment, as such a segment probably correlates to an edge or a boundary in the input image rather than a text region since with this narrow width, texts are unlikely to be eligible.

5. For each segment Seg_i obtained from the previous step, we can locate a rectangular sub region $\mathcal{R}(i)$ in the edge image $\widehat{\mathcal{I}}$. The left, right, top, bottom boundaries of the region correspond to the lines $x = left_i$, $x = right_i$, $y = 1$, and $y = v$ in the image $\widehat{\mathcal{I}}$ respectively. And all the pixels falling between these boundaries constitute the region $\mathcal{R}(i)$, which is denoted as $\mathcal{R}(i) \triangleq \{pixel(x, y) | left_i \leq x \leq right_i; 1 \leq y \leq v\}$. For each so located region $\mathcal{R}(i)$, we then derive its vertical projection histogram, which is denoted as \mathcal{H}_v^i where the subscript v indicates

it is a vertical histogram and the superscript i indicates this vertical histogram corresponds to the region $\mathcal{R}(i)$. Such a vertical histogram \mathcal{H}_V^i is represented as a v dimensional vector in the form of $\mathcal{H}_V^i \triangleq [\mathcal{H}_V^i(1), \dots, \mathcal{H}_V^i(v)]$ where $\mathcal{H}_V^i(j)$ records the number of edge pixels on the horizontal line which stays j pixels above the bottom of the image, i.e. $\mathcal{H}_V^i(j) = |\{pixel(x, j) | pixel(x, j) \text{ is an edge pixel in } \widehat{\mathcal{I}}; left_i \leq x \leq right_i\}|$. This way of deriving the vertical histogram vector \mathcal{H}_V^i is very similar to the process for deriving the horizontal histogram vector \mathcal{H}_H as examined earlier in step 3.

6. Once the vertical projection histogram \mathcal{H}_V^i has been derived, we can then segment the image region $\mathcal{R}(i)$ following a similar routine as employed in step 4 in the above. That is, we first derive a binary sequence $\mathcal{B}_V^i \triangleq [\mathcal{B}_V^i(1), \dots, \mathcal{B}_V^i(v)]$ according to \mathcal{H}_V^i as follows:

$$\mathcal{B}_V^i(j) \triangleq \begin{cases} 1 & \text{if } \mathcal{H}_V^i(j) \geq \tau_V; \\ 0 & \text{otherwise.} \end{cases} \quad (j = 1, \dots, v) \quad (3)$$

where τ_V is a pre-selected segmentation threshold.

Our algorithm then detects segments of consecutive 1's in \mathcal{B}_V^i . The resultant sequence of such segments are denoted as $Seg_1^i, \dots, Seg_{m_i}^i$ assuming there are m_i segments of consecutive 1's detected from \mathcal{B}_V^i in total. For each Seg_j^i , the i -th segment of consecutive 1's in \mathcal{B}_V^i , we represent its left and right boundaries as $bottom_j^i$ and top_j^i respectively. That is, Seg_j^i corresponds to the block of consecutive 1's which starts at the $bottom_j^i$ -th component in \mathcal{B}_V^i and ends at the top_j^i -th component in \mathcal{B}_V^i . It is easy to see that $top_{j-1}^i < bottom_j^i$ as otherwise the two segments Seg_{j-1}^i and Seg_j^i should have been merged together. Similar

to the small segment elimination process in step 4, if any resultant segment's height is less than 3 pixels apart, i.e. $bottom_j^i - top_{j-1}^i < 3$, we will eliminate this segment, as such a segment probably correlates to an edge or a boundary in the input image rather than a text region since with this narrow height, texts are unlikely to be eligible.

7. Every pair of the segments Seg_i and Seg_j^i ($j = 1, \dots, m_i$) derived in step 4 and 6 in the above jointly defines a rectangular region \mathcal{R}_j^i inside the original image \mathcal{I} , whose left, right, top, bottom boundaries correspond to the lines $x = left_i$, $x = right_i$, $y = bottom_j^i$ and $y = top_j^i$ in \mathcal{I} respectively. That is, $\mathcal{R}_j^i \triangleq \{pixel(x, y) | left_i \leq x \leq right_i; bottom_j^i \leq y \leq top_j^i\}$. Each such region serves as a candidate text region. For every \mathcal{R}_j^i , we compute a corresponding minimum coverage bounding box, which is denoted as \mathcal{X}_j^i . Initially, the boundaries of the bounding box \mathcal{X}_j^i are set as the boundaries of the rectangular region \mathcal{R}_j^i . We then optimize positions of these boundaries through a two-stage expansion and shrinking process. In the first stage, the bounding box will be minimally expanded so that all the edge pixels which are connected to at least one edge pixel inside the text region \mathcal{R}_j^i will be covered by the expanded bounding box. And then in the second stage, the bounding box will be maximally shrunk so that the area of the bounding box is minimized without excluding any edge pixels originally contained in the region \mathcal{R}_j^i . After this two-stage process searching for optimal boundary positions for \mathcal{R}_j^i , we add the bounding box region \mathcal{X}_j^i into the result text region collection $\{\mathcal{R}\}$. All the text regions so derived constitute the result text region set \mathcal{R} , which are detected from the interior region

\mathcal{R} of the input image \mathcal{I} .

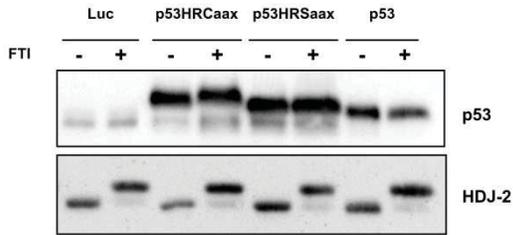
- [1] Lienhart, R. and Wernicke, A. (2002). Localizing and segmenting text in images and videos. *IEEE Transactions on Circuits and Systems for Video Technology*, **12**(4), 256–268.
- [2] Wu, V., Manmatha, R., and Riseman, E. M. (1999). Textfinder: an automatic system to detect and recognize text in images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **21**(11), 1224–1229.

A New Pivoting and Iterative Text Detection Algorithm for Biomedical Images: Appendix B

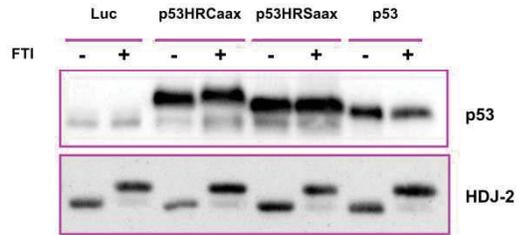
A Text Region Detection Example using our Algorithm

In Figure 1, we show the working of our algorithm on a sample image. Given a raw input image (a), our algorithm first detect the layout elements such as lines and panel boundaries for layout purpose (b). After removing these layout elements from the raw image (c), we apply a Sobel edge detector to derive the edge image (d). We then construct the horizontal histogram for the edge image (e). Through segmenting the horizontal histogram, we detect three horizontal text regions in the image, which are marked Region 1 to 3 (e). According to these horizontal ranges, we can correspondingly detect three text regions in the image (f). Starting with Region 1 in (f), we construct a vertical histogram, shown in (g-1). We proceed identically for Regions 2 and 3 in (f), obtaining vertical histograms as shown in (g-2) and (g-3), respectively. Segmenting the vertical histogram in (g-1), we obtain a vertical text region range. Then, the horizontal region range marked “Region 1” in (e) and the vertical region range marked “Region 1” in (g-1), are combined to construct a refined text region for Region 1 in (f), shown in (h-1). We proceed similarly for Regions 2 and 3 to receive refined image text regions (h-2 and h-3). Text regions shown in (h-1) and (h-3) can not be further refined and are flagged by the algorithm.

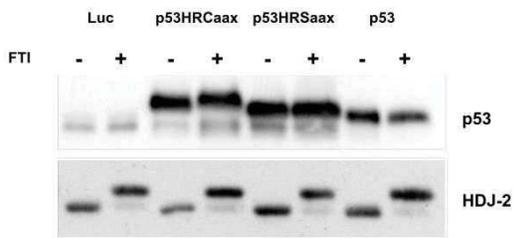
The subregion shown in (h-2) can be further refined through an additional round of our PTD algorithm (i-1). A horizontal histogram for Region 1 in (h-2) detects further image text segments (j-1). A similar result is obtained for region 2 in (h-2), see (i-2) and (j-2). Finally, we implemented a naive intensity based image block detection procedure, which marks non-text region at the end of each iteration. Running this procedure helps us remove the large none-areas inside Region 3 and Region 4 in (h-2). These areas are marked in pink in (k), and are subtracted in the final output image (l).



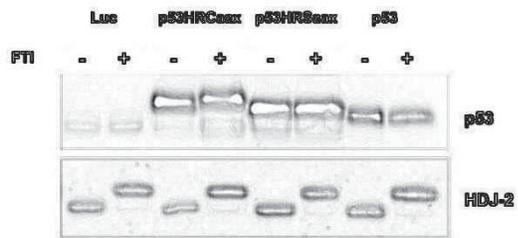
(a) Raw image



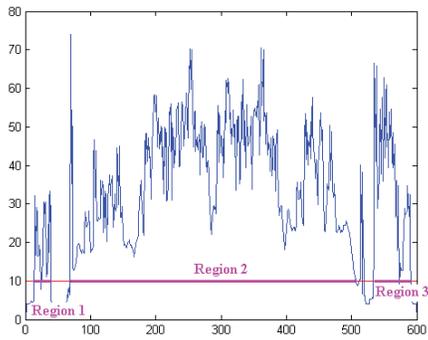
(b) Detected layout elements



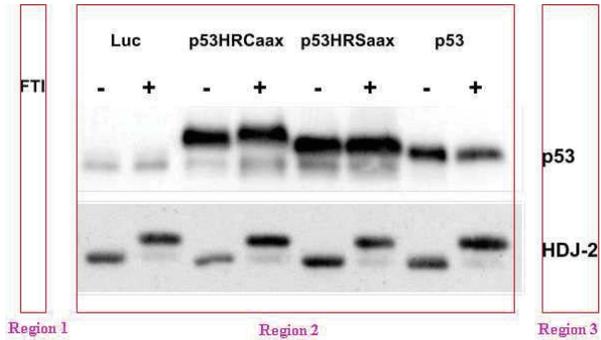
(c) After removing layout elements



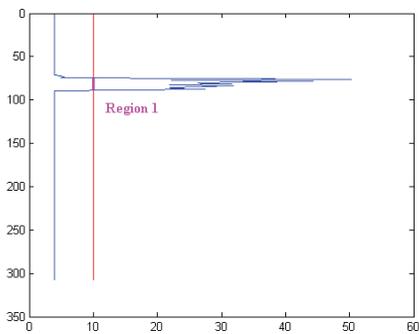
(d) Edge image for (c)



(e) Horizontal histogram for (d)

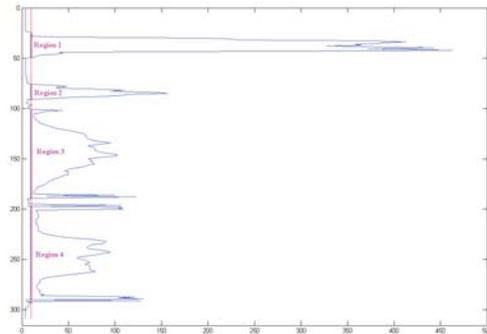


(f) Horizontal segmentation result for (c)



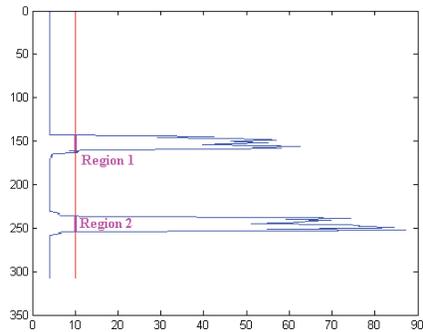
(g-1) Vertical histogram
for Region 1 in (f)

3



(g-2) Vertical histogram
for Region 2 in (f)

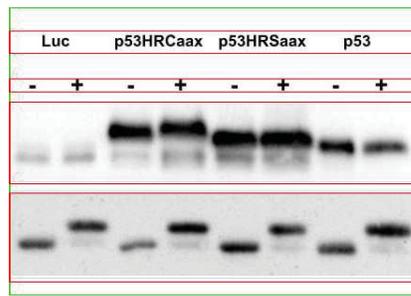
(to be continued on the next page)



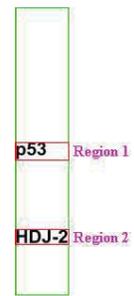
(g-3) Vertical histogram for Region 3 in (f)



(h-1) Segmentation result
for Region 1 in (f)

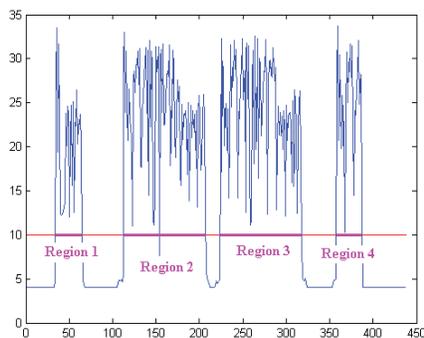


(h-2) Segmentation result
for Region 2 in (f)

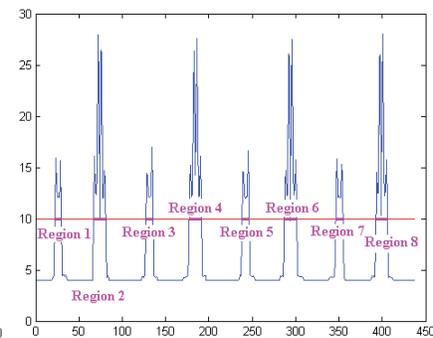


(h-3) Segmentation result
for Region 3 in (f)

(Image boundaries are indicated in green.)

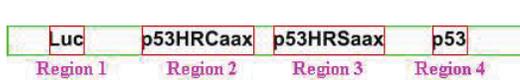


(i-1) Horizontal histogram
for Region 1 in (h-2)



(i-2) Horizontal histogram
for Region 2 in (h-2)

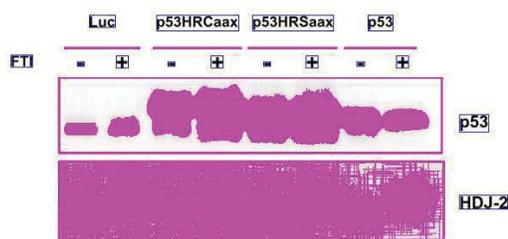
(to be continued on the next page)



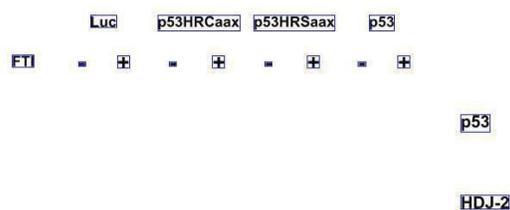
(j-1) Segmentation result
for Region 1 in (h-2)



(j-2) Segmentation result
for Region 2 in (h-2)



(k) The text detection result
after all iterations

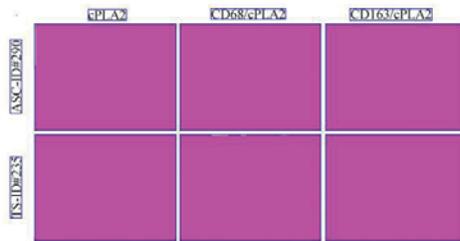
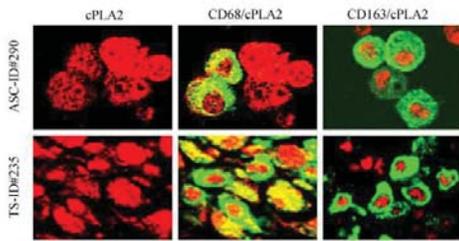


(l) The final text detection result

Figure 1: A step-by-step text detection example.

A New Pivoting and Iterative Text Detection Algorithm for Biomedical Images: Appendix C

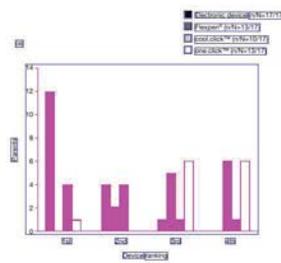
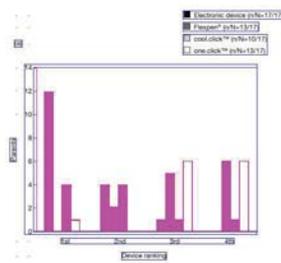
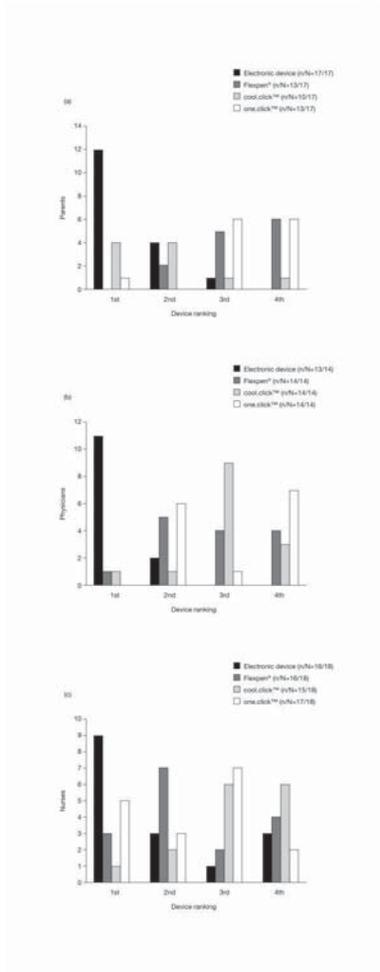
Below we will show more text detection examples.



(a) The original image

(b) Text detection result after the 1st round, i.e. the final result

Figure 1: A text detection example from [9].

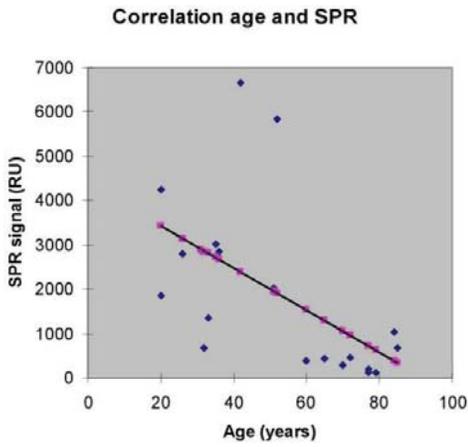


(a) The original image

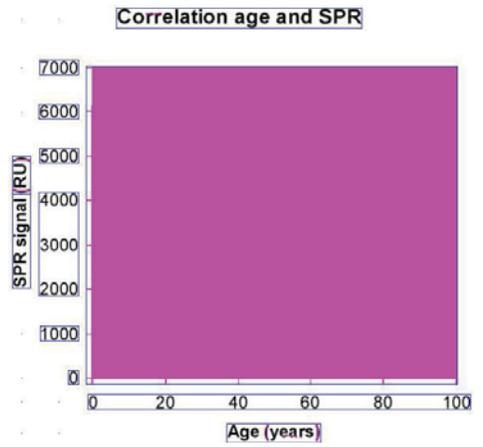
(b) Text detection result after the 1st round

(c) The final text detection result

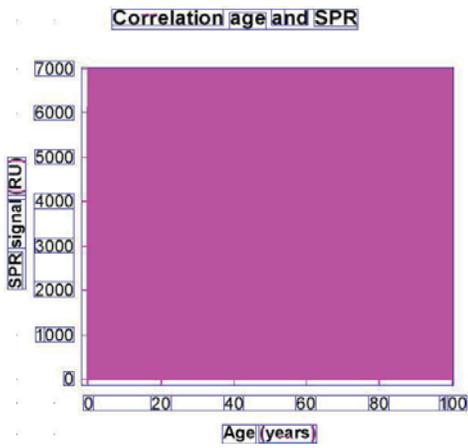
Figure 2: A text detection example from [3].



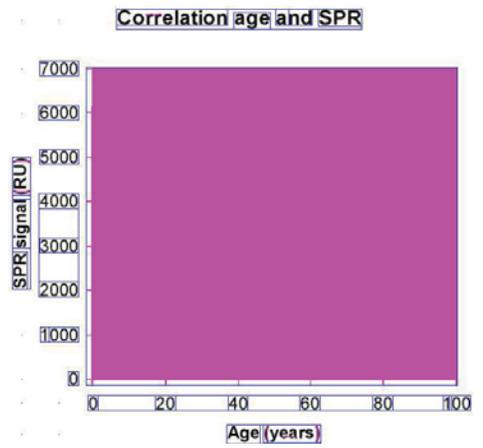
(a) The original image



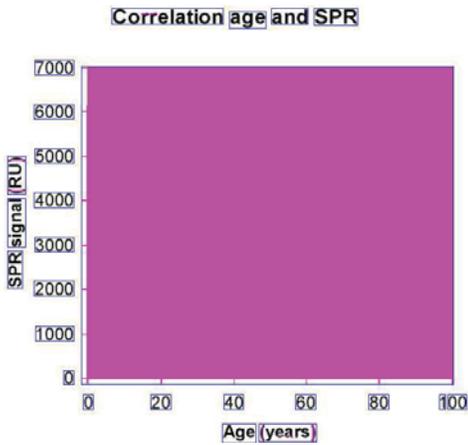
(b) Text detection result after the 1st round



(c) Text detection result after the 2nd round

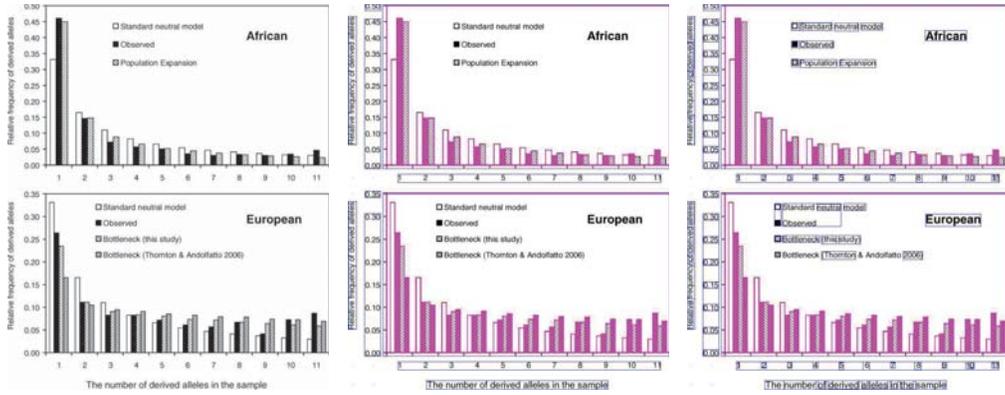


(d) Text detection result after the 3rd round

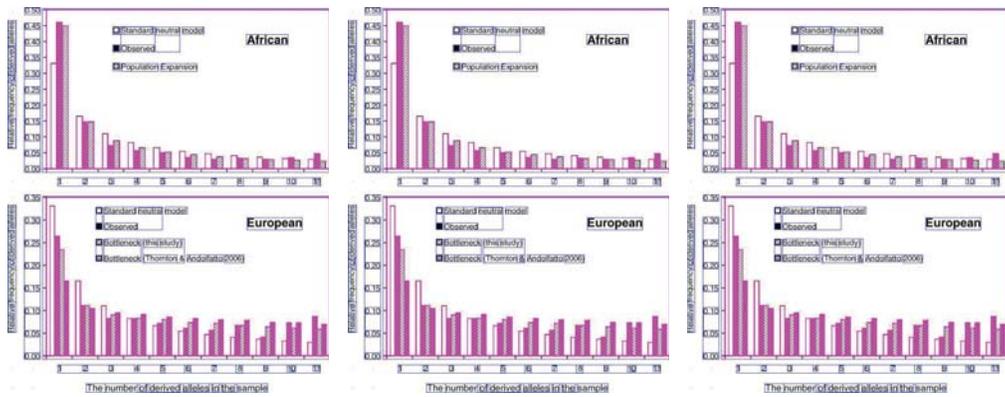


(e) The final text detection result

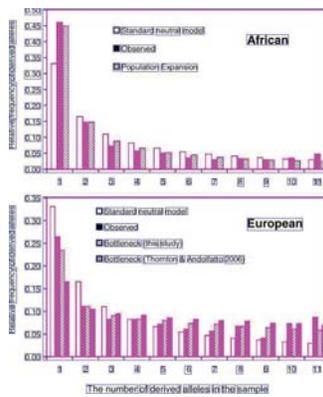
Figure 3: A text detection example from [7].



(a) The original image (b) Text detection result (c) Text detection result after the 1st round

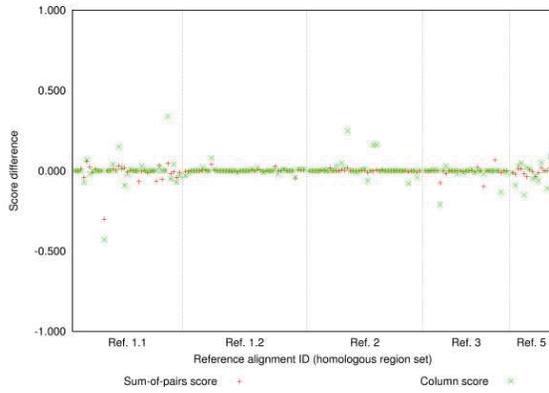


(d) Text detection result after the 3rd round (e) Text detection result after the 4th round (f) Text detection result after the 5th round

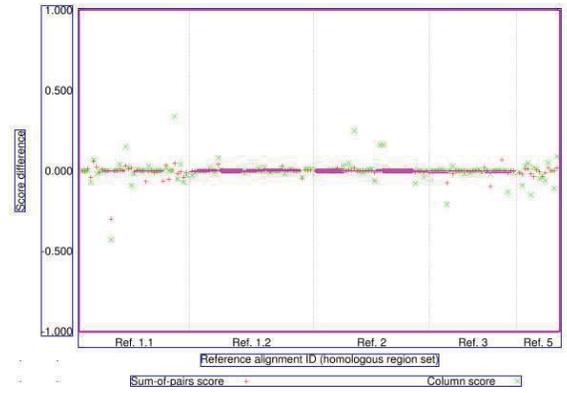


(g) The final text detection result

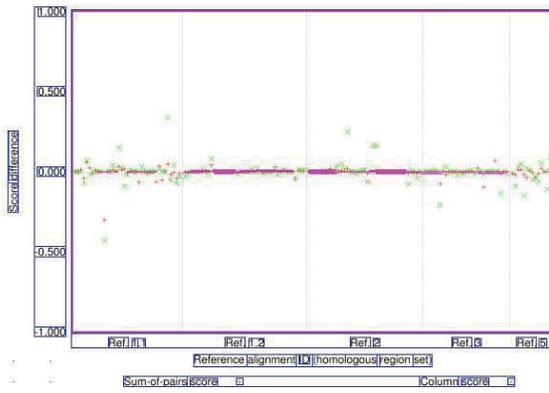
Figure 4: A text detection example from [4].



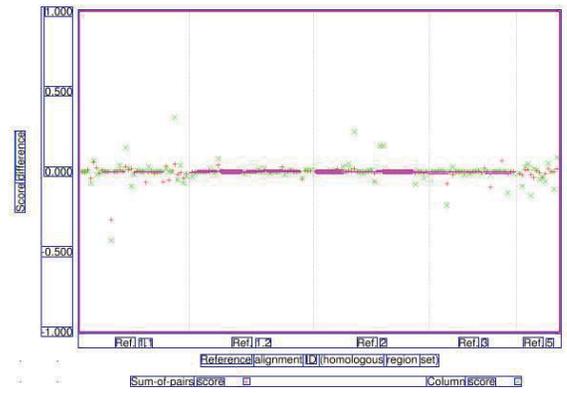
(a) The original image



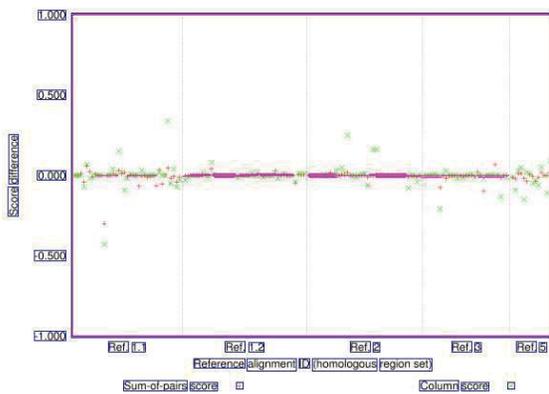
(b) Text detection result after the 1st round



(c) Text detection result after the 2nd round

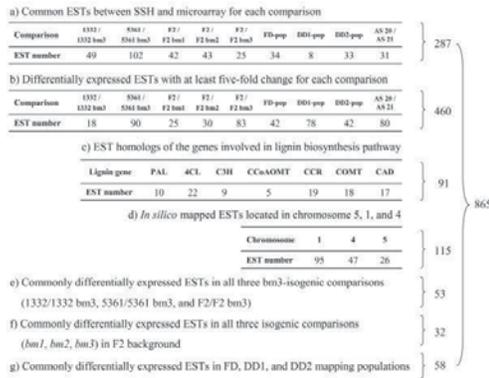


(d) Text detection result after the 3rd round

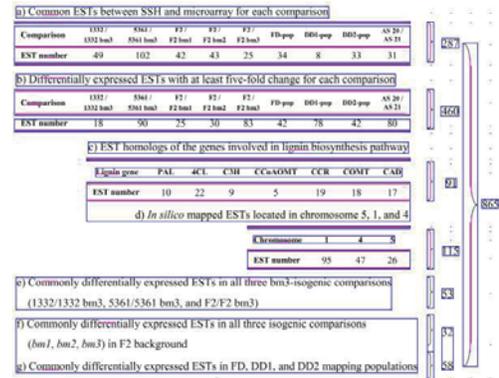


(e) The final text detection result

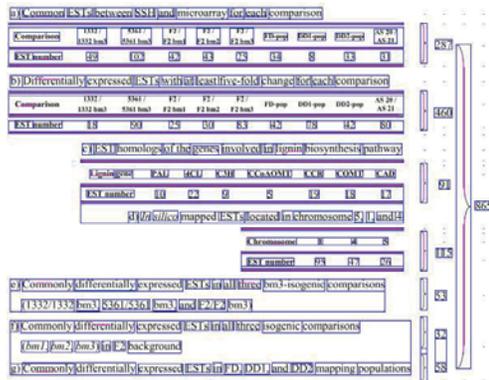
Figure 5: A text detection example from [10].



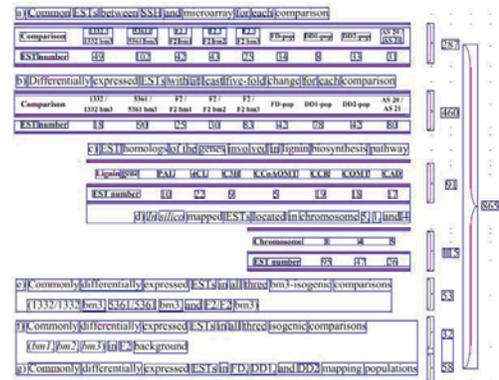
(a) The original image



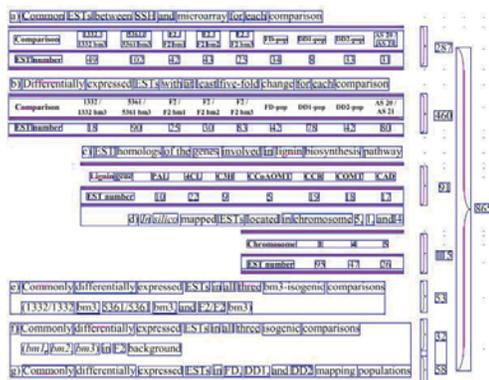
(b) Text detection result after the 1st round



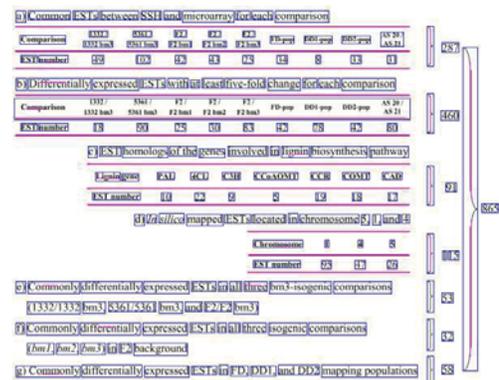
(c) Text detection result after the 2nd round



(d) Text detection result after the 3rd round

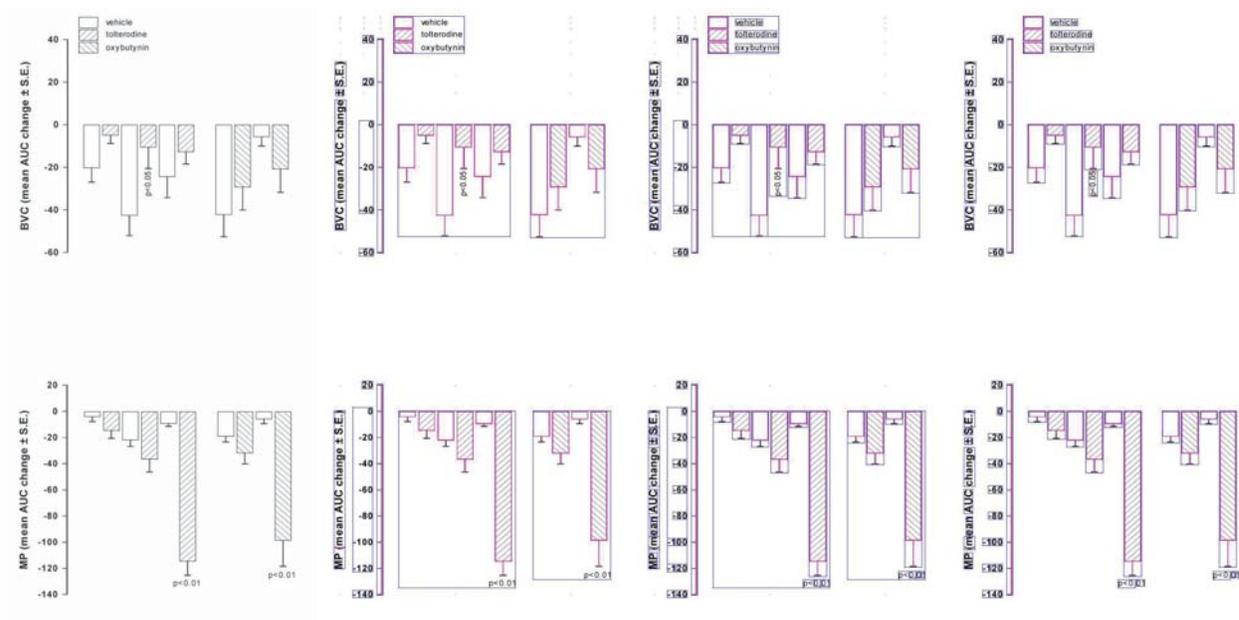


(e) Text detection result after the 4th round



(f) The final text detection result

Figure 6: A text detection example from [8].



(a) The original image (b) Text detection result (1st round) (c) Text detection result (2nd round) (d) Final text detection result

Figure 7: A text detection example from [1].

```
bcTopo IIIIn 601 VEMSEKMDFTGLHVESLEKKGSKPFTTKKVGSCKKDGDVIDKSTFYGCSNYNTTQDFTISKKILSKTISQKRSKLLK
bcTopo IIIIn 681 GERTDLIKGPKKGGERTFDKLEKWKDKINPVFEN
```

(a) The original image

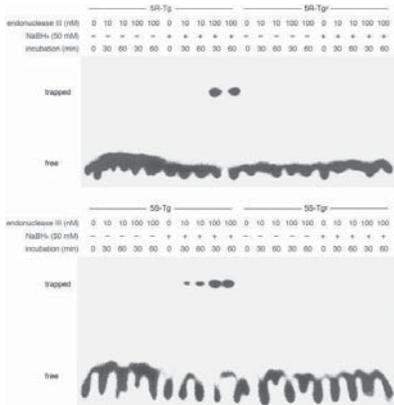
```
bcTopo IIIIn 603 VEMSEKMDFTGLHVESLEKKGSKPFTTKKVGSCKKDGDVIDKSTFYGCSNYNTTQDFTISKKILSKTISQKRSKLLK
bcTopo IIIIn 683 GERTDLIKGPKKGGERTFDKLEKWKDKINPVFEN
```

(b) Text detection result after the 1st round

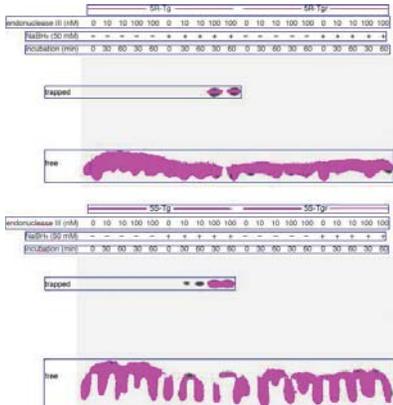
```
bcTopo IIIIn 503 VEMSEKMDFTGLHVESLEKKGSKPFTTKKVGSCKKDGDVIDKSTFYGCSNYNTTQDFTISKKILSKTISQKRSKLLK
bcTopo IIIIn 483 GERTDLIKGPKKGGERTFDKLEKWKDKINPVFEN
```

(c) The final text detection result

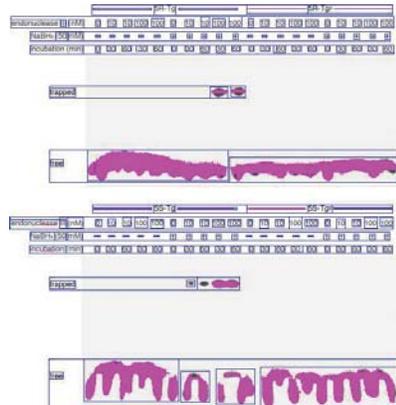
Figure 8: A text detection example from [5].



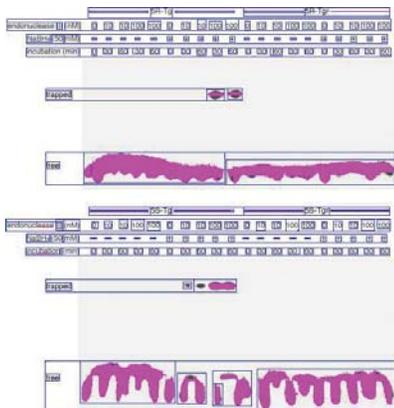
(a) The original image



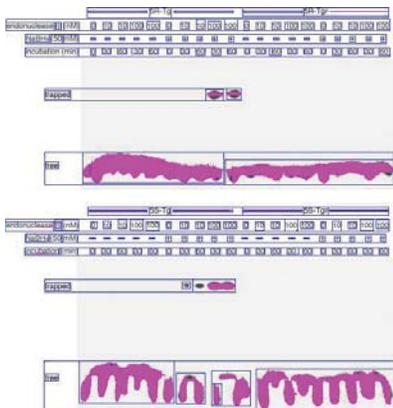
(b) Text detection result after the 1st round



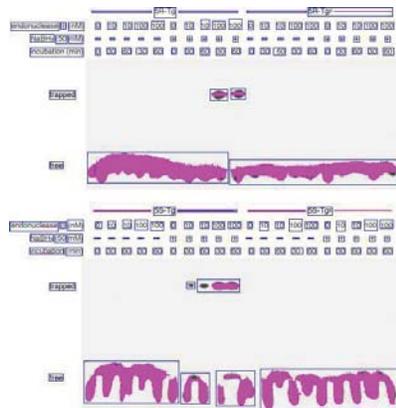
(c) Text detection result after the 2nd round



(d) Text detection result after the 3rd round

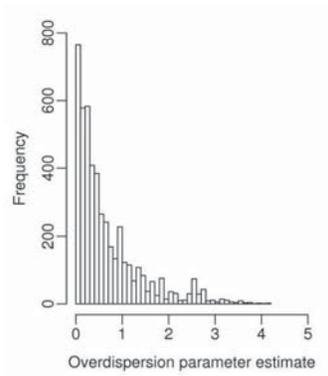


(e) Text detection result after the 4th round

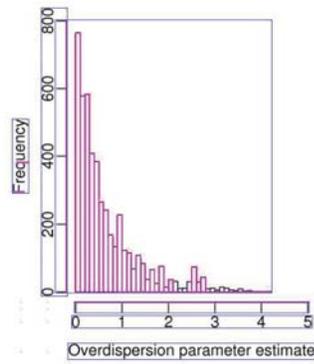


(f) The final text detection result

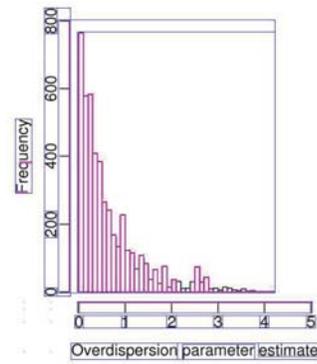
Figure 9: A text detection example from [2].



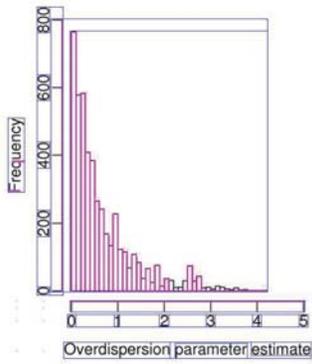
(a) The original image



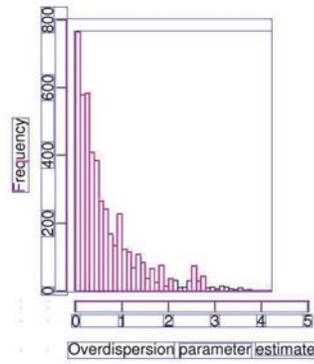
(b) Text detection result after the 1st round



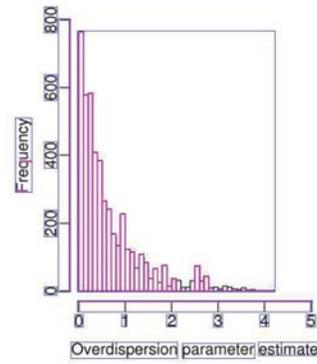
(c) Text detection result after the 2nd round



(d) Text detection result after the 3rd round



(e) Text detection result after the 4th round



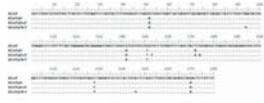
(f) The final text detection result

Figure 10: A text detection example from [6].

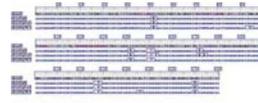
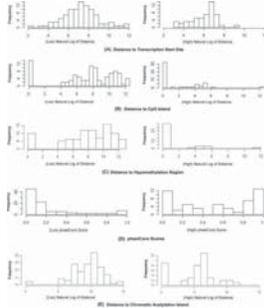
- [1] Angelico, P., Velasco, C., Guarneri, L., Sironi, G., Leonardi, A., and Testa, R. (2005). Urodynamic effects of oxybutynin and tolterodine in conscious and anesthetized rats under different cystometrographic conditions. *BMC Pharmacology*, **5**, 14–14. PMID: 16216132 PMCID: 1274333.
- [2] Doi, Y., Katafuchi, A., Fujiwara, Y., Hitomi, K., Tainer, J. A., Ide, H., and Iwai, S. (2006). Synthesis and characterization of oligonucleotides containing 2-fluorinated thymidine glycol as inhibitors of the endonuclease III reaction. *Nucleic Acids Research*, **34**(5), 1540–1551. PMID: 16547199 PMCID: 1409675.
- [3] Dumas, H., Panayiotopoulos, P., Parker, D., and Pongpaichana, V. (2006). Understanding and meeting the needs of those using growth hormone injection devices. *BMC Endocrine Disorders*, **6**, 5–5. PMID: 17034628 PMCID: 1618831.
- [4] Li, H. and Stephan, W. (2006). Inferring the demographic history and rate of adaptive substitution in drosophila. *PLoS Genetics*, **2**(10). PMID: 17040129 PMCID: 1599771.
- [5] Li, Z., Hiasa, H., and DiGate, R. (2005). Bacillus cereus DNA topoisomerase i and III: purification, characterization and complementation of escherichia coli TopoIII activity. *Nucleic Acids Research*, **33**(17), 5415–5425. PMID: 16192570 PMCID: 1236973.
- [6] Lu, J., Tomfohr, J. K., and Kepler, T. B. (2005). Identifying differential expression in multiple SAGE libraries: an overdispersed log-linear model approach. *BMC Bioinformatics*, **6**, 165–165. PMID: 15987513 PMCID: 1189357.
- [7] Nayeri, F., Aili, D., Nayeri, T., Xu, J., Almer, S., Lundstrm, I., kerlind, B., and Liedberg, B. (2005). Hepatocyte growth factor (HGF) in fecal samples: rapid detection by surface plasmon resonance. *BMC Gastroenterology*, **5**, 13–13. PMID: 15826299 PMCID: 1090571.
- [8] Shi, C., Uzarowska, A., Ouzunova, M., Landbeck, M., Wenzel, G., and Lbberstedt, T. (2007). Identification of candidate genes associated with cell wall digestibility and eQTL (expression quantitative trait loci) analysis in a flint flint maize recombinant inbred line population. *BMC Genomics*, **8**, 22–22. PMID: 17233901 PMCID: 1785377.
- [9] Wang, X., Deavers, M., Patenia, R., Bassett, R. L., Mueller, P., Ma, Q., Wang, E., and Freedman, R. S. (2006). Monocyte/macrophage and t-cell infiltrates in peritoneum of patients with ovarian cancer or benign pelvic disease. *Journal of Translational Medicine*, **4**, 30–30. PMID: 16824216 PMCID: 1550428.
- [10] Yamada, S., Gotoh, O., and Yamana, H. (2006). Improvement in accuracy of multiple sequence alignment using novel group-to-group sequence alignment algorithm with piecewise linear gap cost. *BMC Bioinformatics*, **7**, 524–524. PMID: 17137519 PMCID: 1769516.

A New Pivoting and Iterative Text Detection Algorithm for Biomedical Images: Appendix D

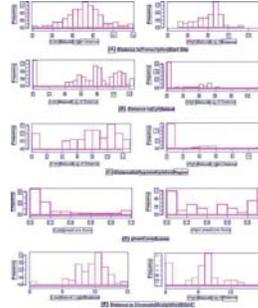
In Figures 1–8, we show some more example text detection results by our algorithm along with the original image.



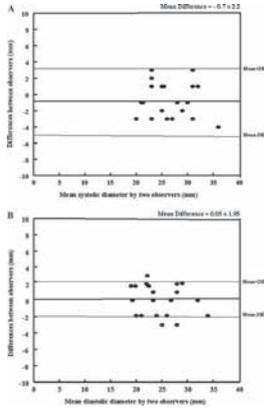
(A-I) raw image



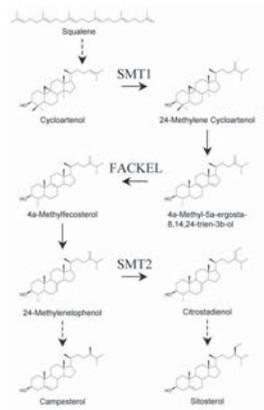
(A-II) text detection result



(B-I) raw image

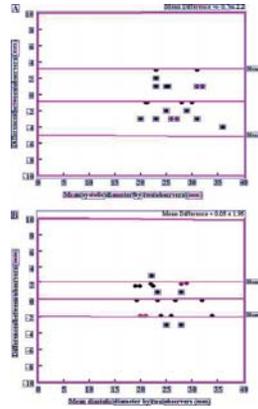


(C-I) raw image

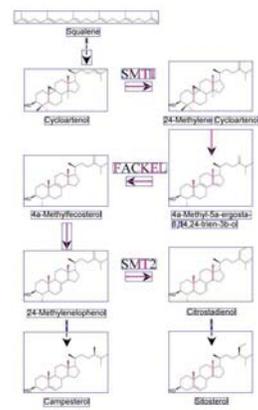


(D-I) raw image

(B-II) text detection result

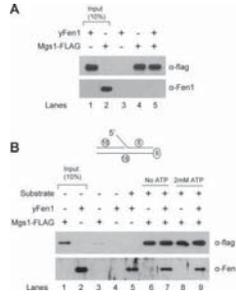


(C-II) text detection result



(D-II) text detection result

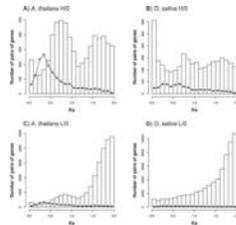
Figure 1: Some example text detection results by our algorithm along with the original image. (A) is from [26]; (B) is from [3]; (C) is from [16]; (D) is from [8].



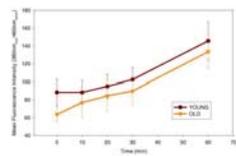
(A-I) raw image



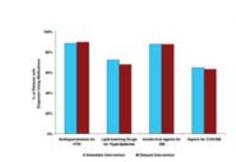
(B-I) raw image



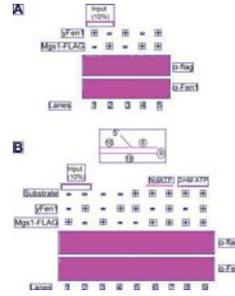
(C-I) raw image



(D-I) raw image



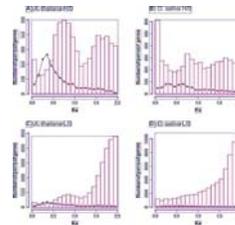
(E-I) raw image



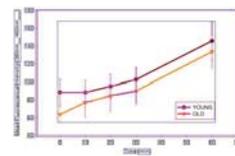
(A-II) text detection result



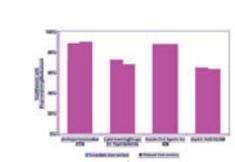
(B-II) text detection result



(C-II) text detection result

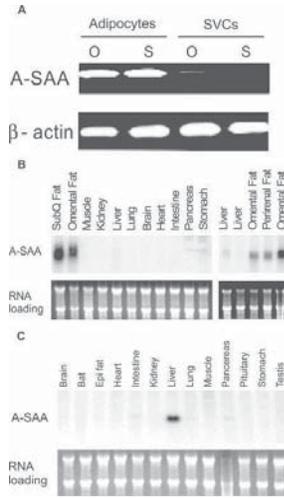


(D-II) text detection result

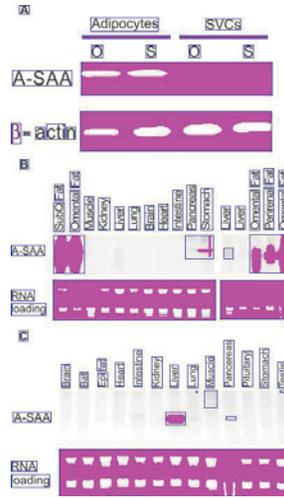


(E-II) text detection result

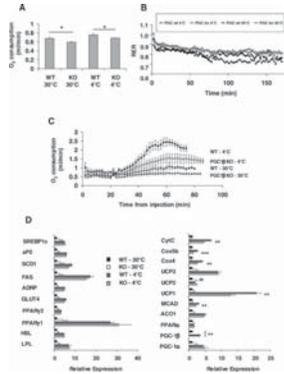
Figure 2: Some example text detection results by our algorithm along with the original image. (A) is from [13]; (B) is from [1]; (C) is from [23]; (D) is from [21]; (E) is from [15].



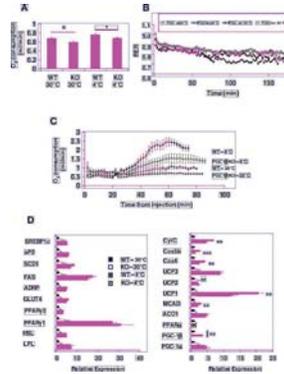
(A-I) raw image



(A-II) text detection result

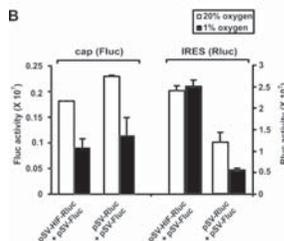
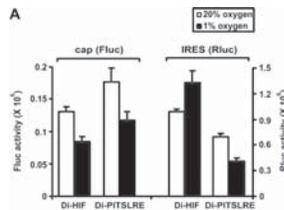


(B-I) raw image

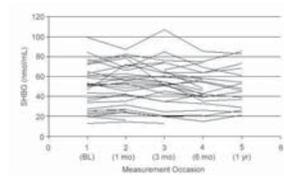


(B-II) text detection result

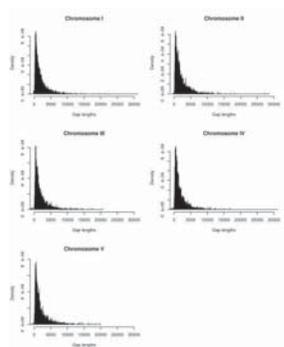
Figure 3: Some example text detection results by our algorithm along with the original image. (A) is from [32]; (B) is from [14].



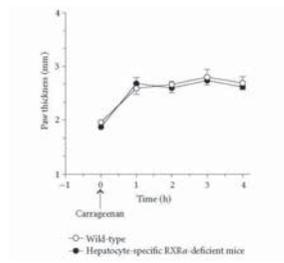
(A-I) raw image



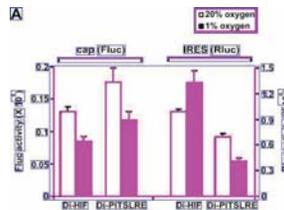
(B-I) raw image



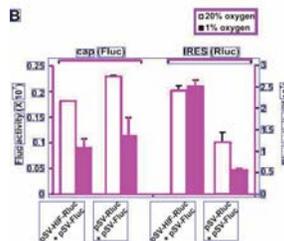
(C-I) raw image



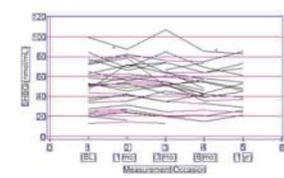
(D-I) raw image



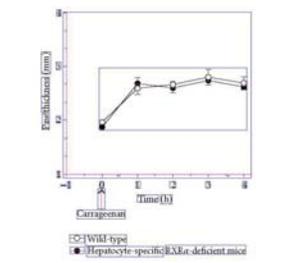
(A-II) text detection result



(B-II) text detection result

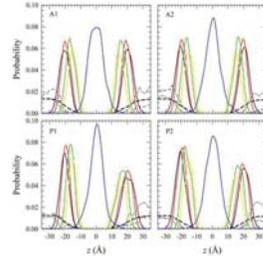


(C-II) text detection result

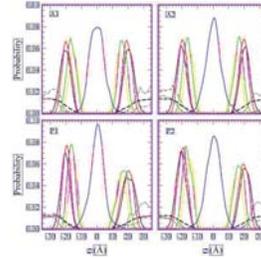


(D-II) text detection result

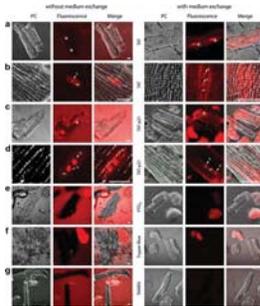
Figure 4: Some example text detection results by our algorithm along with the original image. (A) is from [24]; (B) is from [30]; (C) is from [22]; (D) is from [29].



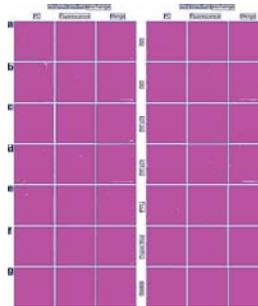
(A-I) raw image



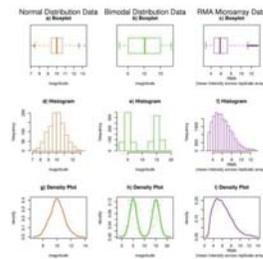
(A-II) text detection result



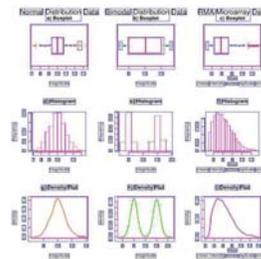
(B-I) raw image



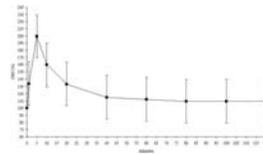
(B-II) text detection result



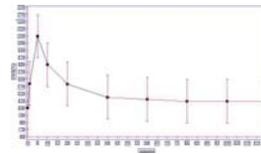
(C-I) raw image



(C-II) text detection result

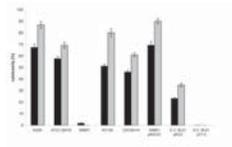


(D-I) raw image

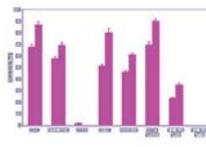


(D-II) text detection result

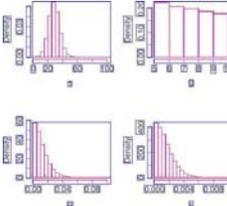
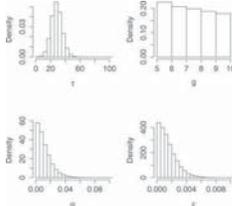
Figure 5: Some example text detection results by our algorithm along with the original image. (A) is from [10]; (B) is from [28]; (C) is from [31]; (D) is from [2].



(A-I) raw image

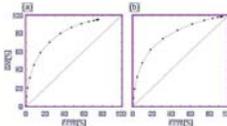
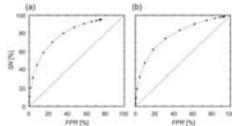


(A-II) text detection result



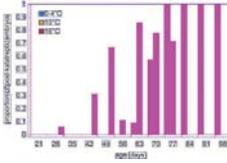
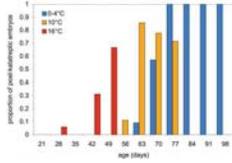
(B-I) raw image

(B-II) text detection result



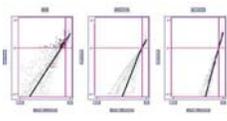
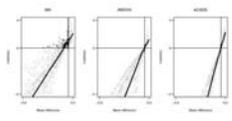
(C-I) raw image

(C-II) text detection result



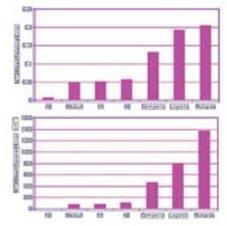
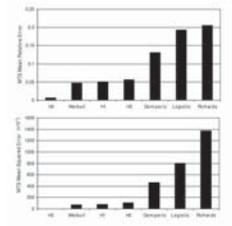
(D-I) raw image

(D-II) text detection result



(E-I) raw image

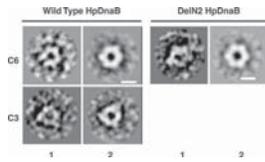
(E-II) text detection result



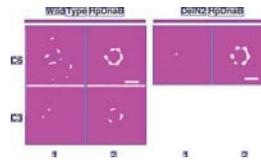
(F-I) raw image

(F-II) text detection result

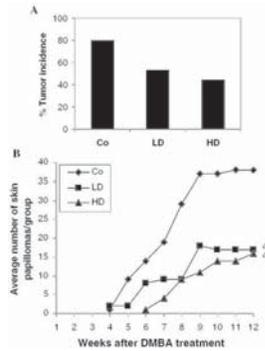
Figure 6: Some example text detection results by our algorithm along with the original image. (A) is from [7]; (B) is from [17]; (C) is from [6]; (D) is from [25]; (E) is from [4]; (F) is from [27].



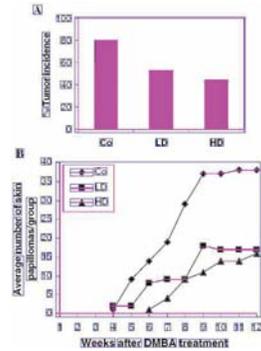
(A-I) raw image



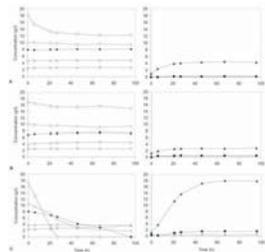
(A-II) text detection result



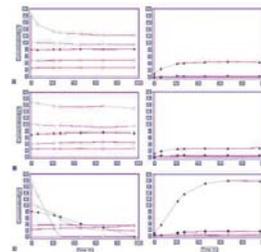
(B-I) raw image



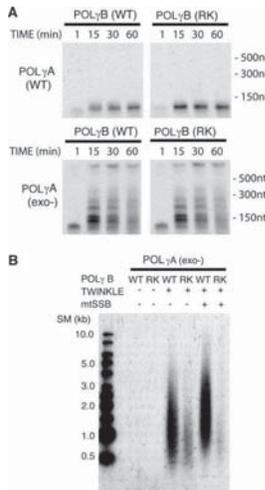
(B-II) text detection result



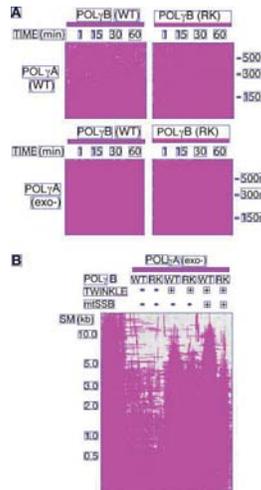
(C-I) raw image



(C-II) text detection result

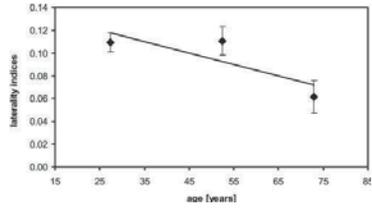


(D-I) raw image

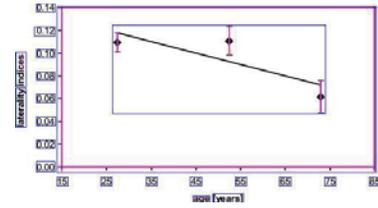


(D-II) text detection result

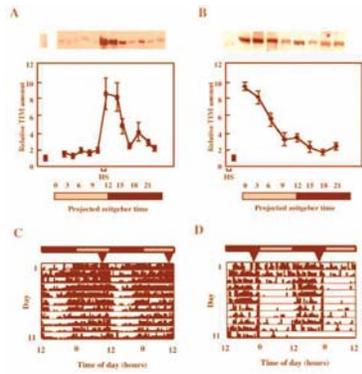
Figure 7: Some example text detection results by our algorithm along with the original image. (A) is from [19]; (B) is from [20]; (C) is from [12]; (D) is from [5].



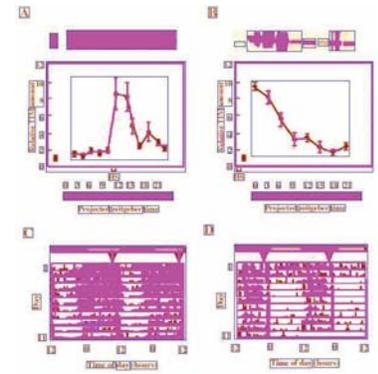
(A-I) raw image



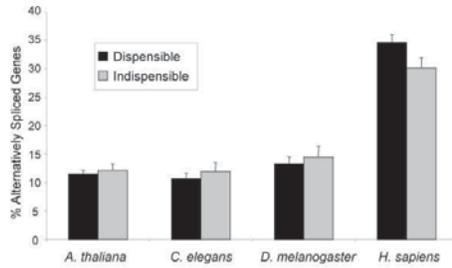
(A-II) text detection result



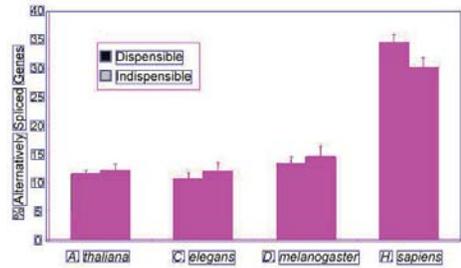
(B-I) raw image



(B-II) text detection result



(C-I) raw image



(C-II) text detection result

Figure 8: Some example text detection results by our algorithm along with the original image. (A) is from [11]; (B) is from [18]; (C) is from [9].

- [1] Alajez, N. M., Eghtesad, S., and Finn, O. J. (2006). Cloning and expression of human membrane-bound and soluble engineered t cell receptors for immunotherapy. *Journal of Biomedicine & Biotechnology*, **2006**(2), 68091. PMID: 16883054.
- [2] Budeus, M., Salibassoglu, E., Schymura, A. M., Reinsch, N., Wieneke, H., Sack, S., and Erbel, R. (2007). Effect of induced ventricular fibrillation and shock delivery on brain natriuretic peptide measured serially following a pre-discharge ICD test. *Indian Pacing and Electrophysiology Journal*, **7**(4), 195–203. PMID: 17957267.
- [3] Chen, Y., Blackwell, T. W., Chen, J., Gao, J., Lee, A. W., and States, D. J. (2007). Integration of genome and chromatin structure with gene expression profiles to predict c-MYC recognition site binding and function. *PLoS Computational Biology*, **3**(4), e63. PMID: 17411336.
- [4] Dabney, A. R. and Storey, J. D. (2007). Normalization of two-channel microarrays accounting for experimental design and intensity-dependent relationships. *Genome Biology*, **8**(3), R44. PMID: 17391524.
- [5] Farge, G., Pham, X. H., Holmlund, T., Khorostov, I., and Falkenberg, M. (2007). The accessory subunit b of DNA polymerase gamma is required for mitochondrial replisome function. *Nucleic Acids Research*, **35**(3), 902–911. PMID: 17251196.
- [6] Fujimori, S., Washio, T., and Tomita, M. (2005). GC-compositional strand bias around transcription start sites in plants and fungi. *BMC Genomics*, **6**(1), 26. PMID: 15733327.
- [7] Hertle, R. and Schwarz, H. (2004). *Serratia marcescens* internalization and replication in human bladder epithelial cells. *BMC Infectious Diseases*, **4**, 16. PMID: 15189566.
- [8] Houde, M., Belcaid, M., Ouellet, F., Danyluk, J., Monroy, A. F., Dryanova, A., Gulick, P., Bergeron, A., Laroche, A., Links, M. G., MacCarthy, L., Crosby, W. L., and Sarhan, F. (2006). Wheat EST resources for functional genomics of abiotic stress. *BMC Genomics*, **7**, 149. PMID: 16772040.
- [9] Irimia, M., Rukov, J. L., Penny, D., and Roy, S. W. (2007). Functional and evolutionary analysis of alternatively spliced genes is consistent with an early eukaryotic origin of alternative splicing. *BMC Evolutionary Biology*, **7**, 188. PMID: 17916237.
- [10] Jang, H., Ma, B., and Nussinov, R. (2007). Conformational study of the protegrin-1 (PG-1) dimer interaction with lipid bilayers and its effect. *BMC Structural Biology*, **7**, 21. PMID: 17407565.
- [11] Kalisch, T., Wilimzig, C., Kleibel, N., Tegenthoff, M., and Dinse, H. R. (2006). Age-related attenuation of dominant hand superiority. *PloS One*, **1**, e90. PMID: 17183722.
- [12] Karhumaa, K., Sanchez, R. G., Hahn-Hgerdal, B., and Gorwa-Grauslund, M. (2007). Comparison of the xylose reductase-xylytol dehydrogenase and the xylose isomerase pathways for xylose fermentation by recombinant *saccharomyces cerevisiae*. *Microbial Cell Factories*, **6**, 5. PMID: 17280608.

- [13] Kim, J., Kang, Y., Kang, H., Kim, D., Ryu, G., Kang, M., and Seo, Y. (2005). In vivo and in vitro studies of mgs1 suggest a link between genome instability and okazaki fragment processing. *Nucleic Acids Research*, **33**(19), 6137–6150. PMID: 16251400.
- [14] Lelliott, C. J., Medina-Gomez, G., Petrovic, N., Kis, A., Feldmann, H. M., Bjursell, M., Parker, N., Curtis, K., Campbell, M., Hu, P., Zhang, D., Litwin, S. E., Zaha, V. G., Fountain, K. T., Boudina, S., Jimenez-Linan, M., Blount, M., Lopez, M., Meirhaeghe, A., Bohlooly-Y, M., Storlien, L., Strmstedt, M., Snaith, M., Oresic, M., Abel, E. D., Cannon, B., and Vidal-Puig, A. (2006). Ablation of PGC-1beta results in defective mitochondrial activity, thermogenesis, hepatic function, and cardiac performance. *PLoS Biology*, **4**(11), e369. PMID: 17090215.
- [15] Ma, J., Lee, K., Berra, K., and Stafford, R. S. (2006). Implementation of case management to reduce cardiovascular disease risk in the stanford and san mateo heart to heart randomized controlled trial: study protocol and baseline characteristics. *Implementation Science: IS*, **1**, 21. PMID: 17005050.
- [16] Nemes, A., Caliskan, K., Geleijnse, M. L., Soliman, O. I. I., Anwar, A. M., and ten Cate, F. J. (2008). Alterations in aortic elasticity in noncompaction cardiomyopathy. *The International Journal of Cardiovascular Imaging*, **24**(1), 7–13. PMID: 17334818.
- [17] Nicolas, P., Kim, K., Shibata, D., and Tavar, S. (2007). The stem cell population of the human colon crypt: analysis via methylation patterns. *PLoS Computational Biology*, **3**(3), e28. PMID: 17335343.
- [18] Nishinokubi, I., Shimoda, M., and Ishida, N. (2006). Mating rhythms of drosophila: rescue of tim01 mutants by d. ananassae timeless. *Journal of Circadian Rhythms*, **4**, 4. PMID: 16522214.
- [19] Nitharwal, R. G., Paul, S., Dar, A., Choudhury, N. R., Soni, R. K., Prusty, D., Sinha, S., Kashav, T., Mukhopadhyay, G., Chaudhuri, T. K., Gourinath, S., and Dhar, S. K. (2007). The domain structure of helicobacter pylori DnaB helicase: the n-terminal domain can be dispensable for helicase activity whereas the extreme c-terminal region is essential for its function. *Nucleic Acids Research*, **35**(9), 2861–2874. PMID: 17430964.
- [20] Padmavathi, B., Rath, P. C., Rao, A. R., and Singh, R. P. (2005). Roots of withania somnifera inhibit forestomach and skin carcinogenesis in mice. *Evidence-Based Complementary and Alternative Medicine: eCAM*, **2**(1), 99–105. PMID: 15841284.
- [21] Ponnappan, S., Cullen, S. J., and Ponnappan, U. (2005). Constitutive degradation of IkappaBalpha in human t lymphocytes is mediated by calpain. *Immunity & Ageing: I & A*, **2**, 15. PMID: 16271147.
- [22] Riley, M. C., Clare, A., and King, R. D. (2007). Locational distribution of gene functional classes in arabidopsis thaliana. *BMC Bioinformatics*, **8**, 112. PMID: 17397552.
- [23] Rizzon, C., Ponger, L., and Gaut, B. S. (2006). Striking similarities in the genomic distribution of tandemly arrayed genes in arabidopsis and rice. *PLoS Computational Biology*, **2**(9), e115. PMID: 16948529.

- [24] Schepens, B., Tinton, S. A., Bruynooghe, Y., Beyaert, R., and Cornelis, S. (2005). The polypyrimidine tract-binding protein stimulates HIF-1 α IRES-mediated translation during hypoxia. *Nucleic Acids Research*, **33**(21), 6884–6894. PMID: 16396835.
- [25] Shingleton, A. W., Sisk, G. C., and Stern, D. L. (2003). Diapause in the pea aphid (*Acyrtosiphon pisum*) is a slowing but not a cessation of development. *BMC Developmental Biology*, **3**, 7. PMID: 12908880.
- [26] Styles, P. and Brookfield, J. F. Y. (2007). Analysis of the features and source gene composition of the AluYg6 subfamily of human retrotransposons. *BMC Evolutionary Biology*, **7**, 102. PMID: 17603915.
- [27] Tabatabai, M., Williams, D. K., and Bursac, Z. (2005). Hyperbolic growth models: theory and application. *Theoretical Biology & Medical Modelling*, **2**, 14. PMID: 15799781.
- [28] Tnnemann, G., Karczewski, P., Haase, H., Cardoso, M. C., and Morano, I. (2007). Modulation of muscle contraction by a cell-permeable peptide. *Journal of Molecular Medicine (Berlin, Germany)*, **85**(12), 1405–1412. PMID: 17717642.
- [29] Wan, Y. Y. and Badr, M. Z. (2006). Inhibition of Carrageenan-Induced cutaneous inflammation by PPAR agonists is dependent on Hepatocyte-Specific retinoid x ReceptorAlpha. *PPAR Research*, **2006**, 96341. PMID: 17259670.
- [30] Williams, A. E., Maskarinec, G., Franke, A. A., and Stanczyk, F. Z. (2002). The temporal reliability of serum estrogens, progesterone, gonadotropins, SHBG and urinary estrogen and progesterone metabolites in premenopausal women. *BMC Women's Health*, **2**(1), 13. PMID: 12498620.
- [31] Woody, O. Z. and Nadon, R. (2006). The shivplot: a graphical display for trend elucidation and exploratory analysis of microarray data. *Source Code for Biology and Medicine*, **1**, 6. PMID: 17147786.
- [32] Yang, R., Lee, M., Hu, H., Pollin, T. I., Ryan, A. S., Nicklas, B. J., Snitker, S., Horenstein, R. B., Hull, K., Goldberg, N. H., Goldberg, A. P., Shuldiner, A. R., Fried, S. K., and Gong, D. (2006). Acute-phase serum amyloid a: an inflammatory adipokine and potential link between obesity and its metabolic complications. *PLoS Medicine*, **3**(6), e287. PMID: 16737350.