Bioinformatics, 2017, 1–8
doi: 10.1093/bioinformatics/btx611
Advance Access Publication Date: 10 October 2017
Original Paper

OXFORD

Data and text mining

Compound image segmentation of published biomedical figures

Pengyuan Li, Xiangying Jiang, Chandra Kambhamettu and Hagit Shatkay*

Department of Computer and Information Sciences, University of Delaware, Newark, DE 19716, USA

*To whom correspondence should be addressed.

Associate Editor: Robert Murphy

Received on March 10, 2017; revised on September 6, 2017; editorial decision on September 19, 2017; accepted on September 22, 2017

Abstract

Motivation: Images convey essential information in biomedical publications. As such, there is a growing interest within the bio-curation and the bio-databases communities, to store images within publications as evidence for biomedical processes and for experimental results. However, many of the images in biomedical publications are compound images consisting of multiple panels, where each individual panel potentially conveys a different type of information. Segmenting such images into constituent panels is an essential first step toward utilizing images.

Results: In this article, we develop a new compound image segmentation system, FigSplit, which is based on Connected Component Analysis. To overcome shortcomings typically manifested by existing methods, we develop a quality assessment step for evaluating and modifying segmentations. Two methods are proposed to re-segment the images if the initial segmentation is inaccurate. Experimental results show the effectiveness of our method compared with other methods.

Availability and implementation: The system is publicly available for use at: https://www.eecis.udel.edu/~compbio/FigSplit. The code is available upon request.

Contact: shatkay@udel.edu

Supplementary information: Supplementary data are available online at *Bioinformatics*.

1 Introduction

Images convey essential information in biomedical publications. As such, there is a growing interest within the bio-curation and the bio-databases communities, to store images from within publications as evidence for biomedical processes and for experimental results (Ahmed *et al.*, 2016; Kalpathy-Cramer *et al.*, 2015). To support this approach, recent efforts started exploring the use of image information for biomedical document classification/retrieval (Apostolova *et al.*, 2013; Müller *et al.*, 2012; Shatkay *et al.*, 2006). However, an obstacle toward automatically obtaining the information within published figures is the abundance of compound figures, i.e. images that consist of multiple panels, where each panel may carry a completely different type of information, obtained via diverse modalities. For instance, graphs microscopy and x-ray images may all be shown side-by-side as panels in one single figure (see e.g. Fig. 1). To display and to further process individual image panels, it is essential to first segment each compound image into its constituent panels.

Current compound image segmentation methods are primarily based on finding gaps between panels (Antani *et al.*, 2008; Apostolova *et al.*, 2013; Cheng *et al.*, 2011; Chhatkuli *et al.*, 2013; Murphy *et al.*, 2001; Taschwer and Marques, 2016; Yuan and Ang, 2014). Figure 1a shows a compound image whose panels are separated by gaps. The white bands in the image are detected as gaps and used for separation. However, in low quality images gaps are hard to detect, which leads to *under-segmentation*, that is, parts of the image may not be segmented into individual panels. To overcome this issue, the image can be transformed, for instance via edge-detection (Cheng *et al.*, 2011; Taschwer and Marques, 2016; Yuan and Ang, 2014), so that the gaps are more readily detected. Although current gap-based methods treat all solid bands within figures as gaps, such bands do not always serve as panel separators and can be an integral part of the image itself. As such, gap detection methods may erroneously split images into too many panels, to which we refer as *over-segmentation*. To address the under- and over-segmentation, captions and image

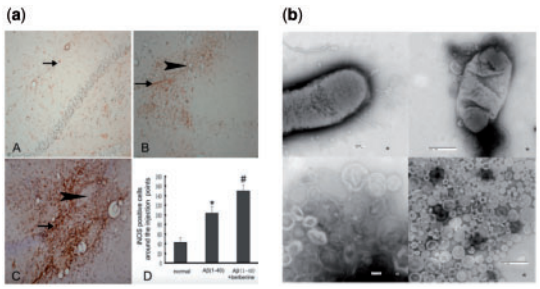

Fig. 1. Examples of compound images. **(a)** A compound image whose panels are separated by gaps. Taken from Figure 3 in Zhu and Qian (2006). **(b)** A compound image whose panels are not separated by visible gaps, taken from Figure 1 in Chooneea *et al.* (2010)

labels have been used to estimate the number of panels in compound images and to identify true gaps of separation (Antani *et al.*, 2008; Apostolova *et al.*, 2013; Cheng *et al.*, 2011). However, such methods are not effective, and may not even be applicable, when captions and labels are not available. Additionally, extracting labels from images requires optical character recognition, which can be a time-consuming operation. An alternative approach (Chhatkuli *et al.*, 2013), applies several rules to first eliminate gaps that are not panel separators thus aiming to avoid over-segmentation. Although this method does not utilize information from image captions and labels and is more efficient, it is still time consuming. Furthermore, its separation accuracy leaves much room for improvement.

Unlike the above methods that segment images using gaps, Shatkay *et al.* (2006) proposed a method based on first identifying connected contents within individual panels. They used connected components analysis (CCA) to detect individual panels in images. Follow-up work by Cheng *et al.* (2011), Kim *et al.* (2011) and Lopez *et al.* (2013) used the same method for panel separation as well. Similar to the gap-based technique discussed earlier, CCA also suffers from over-segmentation; unconnected small objects may be detected as individual panels and segmented off the main image-panel. Aiming to address a different task, namely the *identification of multi-paneled images*, Wang *et al.* (2015) used a post-processing step by setting a threshold on panel-size, to avoid fragmentation into very small panels. However, their work was not applied to the image-segmentation task, but rather aimed only to identify whether an image is compound or not.

Notably, none of the above methods can segment images whose panels are not separated by visible gaps. Figure 1b shows an example of such images, which are called *stitched compound images*. Santosh *et al.* (2015a,b) first proposed a method to separate stitched compound images based on straight lines detected in the images. Their method is applicable only to stitched compound images but does not offer a way to identify stitched images among other figures. As such, it was only applied to manually-identified stitched images.

Here we present a new CCA-based scheme for segmenting compound images, including stitched compound images, while addressing both over- and under-segmentation issues. To do this, we first propose a pre-processing step to broaden and un-blur gaps in images so that more images can be segmented. We then extend our method by adding an assessment step to detect, evaluate and modify segmentation errors, and re-separate some of the images accordingly. The rest of the article is organized as follows: Section 2 describes the complete framework of our method; Section 3 presents experiments used to assess the performance of our method along with the results; Section 4 discusses and analyzes the results, while Section 5 concludes and outlines directions for future work.

2 Methods

Our goal is to segment compound images appearing in biomedical documents. As noted before, compound images consist of several panels, typically separated by gaps, which appear as vertical or horizontal light/dark bands; such gaps may be blurry or too thin to recognize. We first pre-process compound images by resizing, adjusting, and cropping them to make the gaps in the images clearer and broader. We then apply CCA to segment compound images into constituent panels. This approach eliminates small objects and keeps only the main components as individual panels. We assess separation quality of the extracted panels and modify them if the image segmentation quality appears to be low.

We note that CCA may not segment several kinds of images correctly, namely: individual panels whose contents are not well-connected, very blurry images, and stitched compound images. To handle the first two types of images, we first apply an edge detector to sharpen blurry components; we then dilate the edge image to increase connectivity within panels. To handle stitched compound images, we first apply an edge detector to sharpen panel boundaries. These boundaries, which are used for separating panels, can be detected by summing pixel values along horizontal and vertical directions. We finally assess the segmentation quality of the panels obtained from the previous step, and modify the segmentation if needed. The complete framework is shown in Figure 2. The rest of this section provides detail about the techniques used in each of the steps.

2.1 Image pre-processing

Gaps in compound images typically separate panels into clear individual components. However, some panels may be positioned too close to one another, and a thin gap may be noisy or blurred, making separation hard. To address this issue, we first scale the original image I of size $m \times n$, while employing bicubic interpolation (Keys, 1981; Lehmann *et al.*, 1999); this process produces a $2m \times 2n$ image in which the gaps are magnified, while the contrast between each gap and the image-region next to it is amplified. The gaps in the resulting image are thus broader and clearer.

As a second step toward improving gap clarity, we note that gaps are not always white or black, that is, the intensity of pixels in

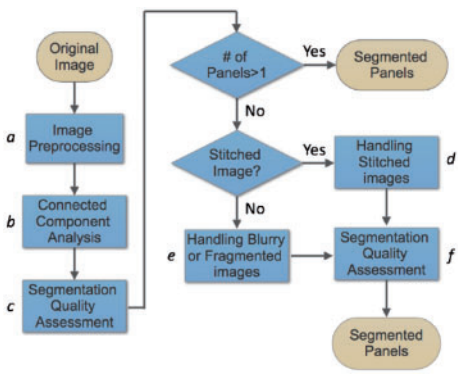

Fig. 2. The framework for our compound image segmentation method: In step *a*, we pre-process compound images to make the gaps between image panels clearer and broader. We then apply CCA to segment compound images into constituent panels in step *b*. Step *c* measures the separation between extracted panels and modifies them if the measurement suggests mis-segmentation. Individual panels whose contents are not well-connected, very blurry images, and stitched compound images may not be segmented by CCA. We separately handle the stitched compound images by Step *d*, and the other two types of images by Step *e*. Finally, in step *f* we assess the segmentation quality of the panels obtained from the previous step, and modify the segmentation if needed

gaps is non-binary. We therefore adjust the intensity of images. By setting a lower threshold T_{low} and an upper threshold T_{high} on the image, we re-map pixel intensities whose values are in the interval $[T_{\text{low}}, T_{\text{high}}]$ to the entire intensity interval [0, 1] by using *linear mapping*. This mapping enhances contrast within the image so that gaps, which are the lightest or the darkest bands in compound images, become clearer. In the experiments described here, we set T_{low} to 0.05 and T_{high} to 0.95.

Last, we note that it is hard to distinguish between the overall boundary of the whole image and boundaries of individual panels. Thus, to disambiguate image-boundaries, we crop the image borders by removing rows and columns of pixels whose maximum gradient value is 0. The image obtained by applying all the pre-processing steps is denoted by $I_{\text{processed}}$.

2.2 Connected component analysis

To segment a pre-processed image, we first detect connected components within it. We assume that gaps among image-panels are white (which can be reversed later by inverting pixel values). To identify gaps among panels, a binary mask M is generated as:

$$M_{(x,y)} = \begin{cases} 1 & \text{if} \quad I_{\text{processed}}(x,y) \leq t; \\ 0 & \text{if} \quad I_{\text{processed}}(x,y) > t, \end{cases} \tag{1}$$

where $I_{\text{processed}}(x,y)$ denotes the pixel at row x and column y in the pre-processed image $I_{\text{processed}}$. By setting the threshold t, each pixel located at coordinate (x, y) within the pre-processed image $I_{\text{processed}}$ for which $I_{\text{processed}}(x,y) > t$ is labeled as background ($M(x,y) = 0$) and is labeled as foreground ($M(x, y) = 1$) otherwise. In our experiments the threshold t is set to 0.95. Based on the mask M, we detect connected components by applying the *Connected Component Labeling* method (Gonzalez and Woods, 2002). This labeling works by scanning the mask M and assigning values to pixels as follows: For each pixel, pixels above and below it as well as to its left and right are considered adjacent; adjacent pixels sharing the same intensity are assigned the same label. A *connected component* is viewed as a set of pixels that have the same label value. A panel bounding box is set around the smallest rectangle that contains all pixels in each connected component. To detect panels separated by black or dark gaps within images, we generate complement images by replacing each pixel value v by $1 - v$.

Using CCA may generate some small bounding boxes due to small and unconnected objects, such as text, in the image. We thus set two thresholds to initially eliminate bounding boxes of very small box-height or box-width: $t_{\text{height}} = \text{height}/20$, $t_{\text{width}} = \text{width}/20$, where *width* and *height* are the total figure width and height. The relatively large bounding boxes, which typically correspond to the main components of the image, are kept and viewed as the main segmented panels within the compound image.

Figure 3 illustrates the CCA method. Figure 3a shows a pre-processed image $I_{\text{processed}}$. Figure 3b is the binary mask generated according to Equation (1). By using the *Connected Component Labeling* method, we obtain the bounding boxes of connected components, shown as textured rectangles in Figure 3c. We then extract only the main components that are covered by large bounding boxes produced by the CCA process. Figure 3d shows the preliminary segmentation resulting from CCA.

After the CCA method is applied, some pieces of the original image may not be covered by the segmented panels, or the original image may be over-segmented. We next present our approach to assess and adapt segmentation results in order to address these shortcomings.

2.3 Segmentation quality assessment

We assess segmentation quality by employing the following five steps: (i) Merge overlapping panels; (ii) Temporarily eliminate small components; (iii) Recover missing panels; (iv) Check segmentation area; and (v) Recover small components. Steps (i)–(iii) above are used to evaluate and modify individual panels obtained by the segmentation methods discussed in this section. Steps (iv) and (v) are used to evaluate and adapt the segments.

1. Merge overlapping panels: Components within a panel may be erroneously detected by CCA as individual panels. As the largest connected component within a panel is typically indicative of the panel's boundary, the bounding boxes of smaller components within the same panel will typically overlap with the bounding box of the largest component. For example, the bounding box of legends may overlap the bounding box of corresponding line graph. We thus compute the ratio between the intersection area and the area of the minimum intersecting bounding box, and merge two bounding boxes when their overlap ratio exceeds a certain threshold t_{merge}. In the experiment described here, we set t_{merge} to 0.1. Figure 4 illustrates the utility of this merging step. Figure 4a shows the two detected bounding boxes around a graph and its respective legend; notably, the two bounding boxes overlap. Figure 4b shows the segmentation result obtained by using CCA. As the computed ratio between the intersection area and the area of the legend bounding box exceeds t_{merge}, we merge the bounding boxes surrounding the graph and the legend into one. Figure 4c shows the resulting merged panel.

2. Temporarily eliminate small components: Similar to the elimination step in CCA, we eliminate bounding boxes that are small (<1/5 in height or width) compared with the largest bounding box, thus reducing noise and removing text.

*3. Recover missing panels:*Due to blurriness or unconnected contents in compound figures, some panels will be omitted in our segmentation process. We thus introduce a recovery step, in which missing panels are detected and recovered. We assume that each missing panel is similar in size and symmetric in position to present panels. We thus check for each panel within the current image whether there is sufficient space for another bounding box to its left or right, as well as above or below it. The space available for a bounding box next to a present panel indicates the position of a candidate panel. We thus use the following two considerations to evaluate the position of a candidate panel: First, we note that pixels surrounding a panel boundary are typically part of gaps, and as such, all share the same color and intensity (given that gaps are bands of a single solid color). Second, the content area within the candidate panel boundary, calculated as the number of non-white pixels within it, should be similar to that of the

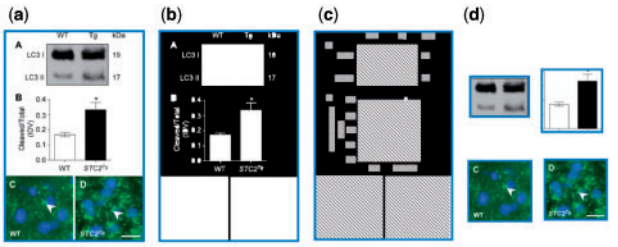

Fig. 3. Steps in CCA. **(a)** The pre-processed image. The original image is Figure 4 in Fazio *et al.* (2011). **(b)** The binary mask generated according to Equation (1). **(c)** The Connected Component Labeling result. Each textured rectangle represents a connected component. **(d)** By extracting only large connected components, we obtain the segmentation result of the CCA method

panel that is present. Figure 5 shows an example of such recovery. The thick frames in the figure indicate bounding boxes of detected panels. Figure 5a shows that panel A is not detected as the bounding box for panel A is missing. Figure 5b shows the panel introduced back into the image following the missing panel recovery step.

4. Check segmentation area: To detect incorrect segmentation, we compute the ratio between the sum of the areas of segmented panels and the area of the original image. If this ratio is lower than 0.5, we consider the segmentation to be incorrect. Incorrect segmentations are discarded and thus the image remains unsegmented. Figure 6 shows an example of incorrect segmentation through CCA. In Figure 6 some parts of image are detected as panels, as shown by the bounding boxes, but the segmentation misses many parts of the image. As the ratio between the sum of the areas bounded by boxes and the area of the original image falls below 0.5, the segmentation is identified as incorrect.

5. Recover small components: During the elimination of small bounding boxes, some essential parts, such as the text/note/legend within the image, may be erroneously eliminated. To re-adopt these small components into the image, we merge eliminated small bounding boxes into their nearest bounding box. To avoid merging bounding boxes that are not part of the same panel during the recovery process, we employ several rules to control the process:

- If merging changes both height and width of a qualified bounding box—do not merge.
- If merging changes more than 20% of the height or the width of a qualified bounding box - do not merge.
- If the height or width of a qualified bounding box changes more than 20% through the small components recovery step, this qualified bounding box keeps its original size and the adopted small bounding boxes will be left alone.
- An eliminated small bounding box will at most be merged once.

After executing the five steps above, some images may still remain unsegmented. Some of those are indeed single-paneled images, but others still comprise multiple panels. The panels may not have been correctly detected by the previous steps either because the image is *stitched*—lacking gaps among panels, or because the image is irregularly blurred or fragmented. These cases are discussed in the next subsection.

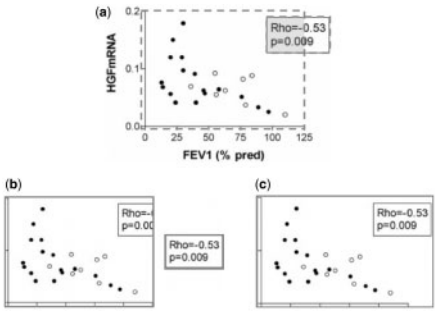

Fig. 4. The affect of merging overlapping panels. **(a)** The original image is taken from Figure 3, panel A, in Bonay *et al.* (2005). The bounding boxes, shown as dashed frames, of both the graph and its legend are detected using the CCA method. The intersection area is shown shaded. **(b)** Segmentation result based on the detected bounding boxes without merging overlapping panels. **(c)** Segmentation result after merging the two (partially overlapping) bounding boxes into one panel

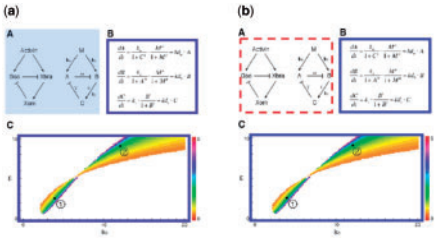

Fig. 5. (a) Segmentation result without *missing panel recovery* versus **(b)** The result after *missing panel recovery*. In **(a)** the bounding box of panel A, which is shown as shaded, is not detected using only the CCA method, resulting in panel A not being recovered. In part **(b)** the bounding box of Panel A is detected by employing the *missing panel recovery* step. The recovered panel is shown surrounded by a dashed frame in **(b)**. The original image is taken from Figure 5 in Saka and Smith (2007)

2.4 Handling stitched and other unsegmented images

Notably, stitched compound images differ from other compound images, because there is no gap-separation among the panels (e.g. Fig. 7a). To identify stitched images, we employ a classification step that separates stitched from non-stitched compound images. We define a *gap* as a row or a column whose minimum gray value is above 0.95. If a gap is found in a compound image, the image is classified as a *compound image with gaps*; otherwise, it is labeled as *stitched*.

2.4.1 Handling stitched images

Stitched compound images cannot be segmented by the CCA method because there is no gap separating panels in such images. Identifying panel boundaries is thus the main challenge. The SUSAN edge detector (Smith and Brady, 1997—*without applying edge-thinning*) is applied to identify pixels whose neighbors' intensity sharply changes as SUSAN has demonstrated the best performance in this context. The edge detector is applied to the pre-processed image $I_{\text{processed}}$ to generate a binary edge image in which the boundaries between panels are intensified (see e.g. Fig. 7b). The objective thus becomes that of detecting boundaries in the resulting edge image I_{edge}. Given an edge image I_{edge}, if pixel (x, y) is detected as a pixel along an edge we set $I_{\text{edge}}(x, y) = 1$. Summing the pixel value along the horizontal and the vertical directions gives rise to two projections: $Proj_{\text{horizontal}}$ and $Proj_{\text{vertical}}$, which are calculated as:

$$\begin{aligned} Proj_{\text{horizontal}} &= I_{\text{edge}}(x, y), \quad y \in 1 \ldots 2n; \\ Proj_{\text{vertical}} &= I_{\text{edge}}(x, y), \quad x \in 1 \ldots 2m. \end{aligned} \tag{2}$$

The values $2n$ and $2m$ are the width and height of the edge image, respectively. The panel segmentation takes place along the horizontal or the vertical line that goes through the highest projection position. For images with complex layout, the boundary between panels may not cross the whole image; in such cases we recursively segment the image along one direction at a time, where the

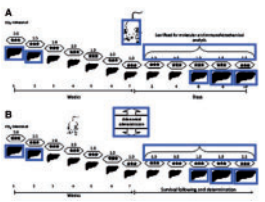

Fig. 6. An example of incorrect segmentation. The original image is taken from Figure 1 in Gálvez-Gastélum *et al.* (2010). The blue rectangles represent the segmentation result obtained through CCA. The part of the image not covered by blue rectangles is missing from result of the CCA method as the connected region in that part is too small to be detected

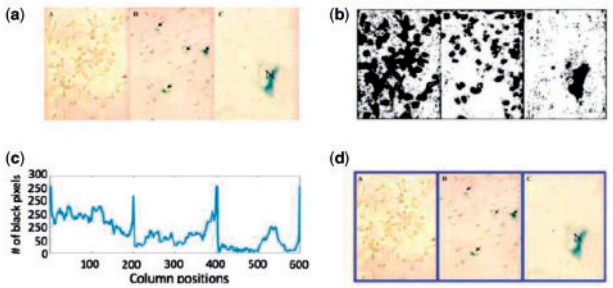

Fig. 7. Example illustrating the steps applied for handling stitched images. **(a)** The stitched compound image taken from Figure 5 in Sereno *et al.* (2015). **(b)** The edge image of the figure shown in (a). **(c)** The horizontal projection of (b) is calculated according to Equation (2). **(d)** Segmentation result obtained by choosing the peak position of the horizontal projection

projection peak value is at least 0.7 of the height or the width of the region currently considered for segmentation.

Figure 7 shows an example of the steps applied to stitched images. Figure 7a is the original stitched compound image. Figure 7b shows the edge image obtained by applying an edge detector. Panel boundaries are observed as straight black lines in the image. Figure 7c shows the horizontal projection plot $Proj_{\text{horizontal}}$ of Figure 7b. By applying the selection of a projection peak, we obtain the segmentation result shown in Figure 7d.

2.4.2 Handling potential blurry or fragmented images

To address the remaining unsegmented images, which are potentially *blurry, fragmented* or having many *fine details*, we first apply an edge-detector to to sharpen components in the pre-processed image $I_{\text{processed}}$. As with the case of stitched images, we use the SUSAN edge detector (*without applying edge-thinning*). The corresponding edge image, denoted I_{edge}, may still have poor connectivity. To enhance connectivity of components in I_{edge}, we dilate the connected regions within the edge image using the minimum gap-width in the image as the dilation factor. After dilation, the connectivity within the dilated edge image is increased. We then apply the CCA method on the dilated edge image again to obtain the segmentation.

Figure 8 illustrates these steps, applied to example blurred/fragmented images. Figure 8a shows an original compound image. By applying the SUSAN edge detector, we first unblur the blurry components, as shown in Figure 8b. We then find the gaps in the edge image along the horizontal and the vertical directions and use the width of the thinest gap as the dilation factor. Figure 8c is the dilated edge image, while Figure 8d shows the segmentation result obtained by using CCA on the dilated edge image. Two panels are detected and highlighted by bounding boxes in Figure 8d. The augmented method thus correctly handles this blurry and fragmented image and identifies the segments within it, unlike the original CCA method (see Fig. 6).

3 Experiments and results

To evaluate our method we conducted two sets of experiments, using datasets from the Figure Separation task in the ImageCLEF Medical shared task (De Herrera *et al.*, 2013, 2015, 2016). The first set of experiments aims to compare the separation accuracy obtained by the different steps of our segmentation method. We use the training and test datasets of ImageCLEF2015 to train our system and test its performance.

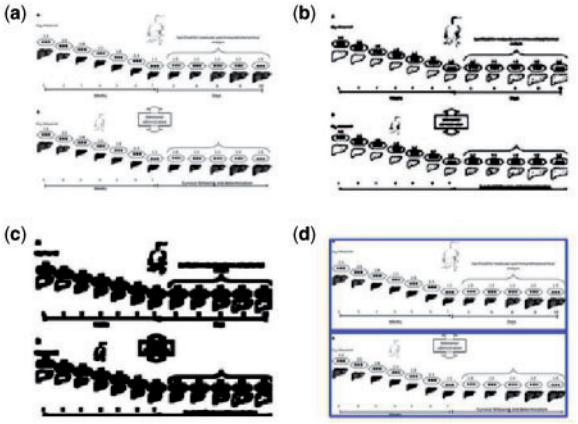

Fig. 8. Example illustrating the steps of *handling potential blurry and fragmented images*. **(a)** The compound image taken from Figure 1 in Gálvez-Gastélum *et al.* (2010). **(b)** The edge image of panel (a). **(c)** The dilated edge image. **(d)** The result of CCA on the dilated image

In the second set of experiments, we compare the separation accuracy of our comprehensive method against that of state-of-the-art systems using test datasets from ImageCLEF 2013, 2015 and 2016. Additionally, to demonstrate the general applicability of our method, we test the method we have developed using the ImageCLEF'15 dataset on the 2013 test dataset.

3.1 Datasets and evaluation

We used five ImageCLEF datasets in this study, two (from ImageCLEF'15 and '16) for training and three (from ImageCLEF'13, '15 and '16) for testing. The images in the datasets are first extracted from the biomedical publications stored in the PubMed Central and then identified as compound images by manually classification. The number of images and ground-truth panels in these datasets are shown in Table 1.

To evaluate our image separation performance, we use the tool provided by ImageCLEF Medical (De Herrera *et al.*, 2013). This tool computes the accuracy of the separation result for a compound image I_i, where the accuracy is defined as:

$$\text{ACC}_i = \frac{C_i}{\max\left(N_i^{\text{G}}, N_i^{\text{D}}\right)},$$

where C_i is the number of detected panels that overlap with at least two-third of the area of the ground-truth panel, N_i^{G} is the true number of panels in the image, and N_i^{D} is the total number of panels we detected. The overall accuracy for the dataset, denoted ACC^{DS}, is calculated by averaging the accuracy over all images:

$$\text{ACC}^{\text{DS}} = \frac{\sum_{i=1}^{i=n} \text{ACC}_i}{n},$$

where n is the number of images in the dataset.

3.2 Results

Table 2 shows the separation accuracies obtained in our first set of experiments using different combinations of steps within our method over the ImageCLEF2015 test dataset. The table shows both the accuracy obtained over the entire dataset as well as that calculated over only those images that were actually segmented into sub-panels by our method. The CCA method alone achieves 79.37% accuracy,

where 279 images remain unsegmented. For the 3102 images that were segmented by CCA, the accuracy is 86.51%. Proceeding the CCA method by a pre-processing step leads to an increase of 2.75% in accuracy. Table 2 also shows that 80 additional images are segmented when the pre-processing step is added. By combining the segmentation-quality-assessment step and the CCA method, 38 fewer images are separated compared with CCA-alone, but the overall accuracy increases by 3.32% (compared with the first row in the table). Moreover, the accuracy for the 3064 images that are segmented is 91.25%. Thus, the results suggest that segmentation quality assessment contributes to improved accuracy. Combining the image pre-processing step and the segmentation-quality-assessment step with the CCA method (Row 4 in the table), the overall accuracy reaches 85.82%, and the accuracy for the 3146 segmented images reaches 92.23%, but 235 images remain unsegmented.

To reduce the number of images that remain un-segmented, we utilize additional steps, as described in Section 2. Applying the step of handling stitched images, the accuracy for the whole dataset reaches 89.31% and the number of compound images that remain unsegmented decreases to 27. Similarly, applying the step of handling potential blurry or fragmented images, the accuracy reaches 89.83% and the number of compound images that remain unsegmented decreases to 71. Combining all the steps leads to the highest accuracy on the ImageCLEF2015 test dataset (90.65%) and on the segmented images (92.65%).

The combination of *all* steps similarly achieves the highest performance, compared with subsets of steps, when applied to other datasets, further demonstrating the effectiveness of our framework. Results on ImageCLEF'15 and '16 training datasets and ImageCLEF'13, and '16 test datasets are shown in Supplmentary Tables S2–S5.

Table 1. Datasets used for training and testing our compound image segmentation method

Figure separation dataset	No. of images	No. of panels
ImageCLEF'13 test	1429	5433
ImageCLEF'15 test	3381	12 789
ImageCLEF'15 training	3403	14 531
ImageCLEF'16 test	1615	8528
ImageCLEF'16 training	6782	27 315

Note: Names of datasets are shown on the left. The middle column shows the total number of images, while the rightmost column shows the number of ground-truth panels associated with each of the respective datasets.

In the second set of experiments, we compare the results achieved by our comprehensive method with those achieved by other systems submitted to ImageCLEF2015 Medical, using the 2015 test dataset. Santosh *et al.*'s (2015a,b) method [based on their previous work (Apostolova *et al.*, 2013; Santosh *et al.*, 2015a,b) discussed in Section 1] achieved an accuracy of 84.64%, while Taschwer and Marques (2016) achieved an accuracy of 84.90%. Our method performs significantly better than all other systems by achieving an accuracy of 90.65%.

We demonstrate the general applicability of our method by using parameters obtained by training over one of the datasets (ImageCLEF'15) to segment images provided in another dataset (ImageCLEF'13); our result shows 84.47% accuracy. The other three top performers (De Herrera *et al.*, 2013; Kitanovski *et al.*, 2013; Simpson *et al.*, 2013) achieved accuracy of 68.59, 69.27 and 84.64%, respectively. We note that while the performance of our method is slightly lower than that reported by De Herrera *et al.* (2013) using method proposed by Chhatkuli *et al.* (2013), the time our system requires to process one image is 0.74 s on average (wall-clock), which is much lower than that of the latter method, which is 2.4 s. Moreover, the ground truth of the ImageCLEFck) dataset was generated using a separation method proposed by Chhatkuli *et al.* (2013) as a basis and manually correcting the automatically generated results, thus biasing the ground truth provided by ImageCLEF toward those produced by the system described in De Herrera *et al.* (2013).

For ImageCLEF'16, as the only team participating in the Figure Separation task, we achieved an accuracy of 84.43% on the test dataset. The segmentation accuracy is similar to the best result obtained in ImageCLEF'15. This result is particularly noteworhy, given that the difficulty of the Figure Separation task was increased in 2016 by increasing the number of stitched compound images, as indicated in the task description (De Herrera *et al.*, 2016).

4. Discussion

The results demonstrate that our method segments compound images more accurately than other state-of-the-art methods (De Herrera *et al.*, 2013; Kitanovski *et al.*, 2013; Santosh *et al.*, 2015a,b; Simpson *et al.*, 2013; Taschwer and Marques, 2015). Due to the complex structure of published biomedical images, automatic detection of the correct separation gap is challenging. In contrast to these gap-detection based methods, our method segments compound images based on the connected contents within individual panels. Moreover, the quality assessment step evaluates separation quality, corrects for separation errors, and discards incorrect segmentations.

Table 2. Comparison of segmentation accuracies obtained by using different combinations of steps within our method

Steps used	Overall accuracy over the whole dataset	No. of images in the dataset	Accuracy over segmented images	No. of segmented images
CCA-alone	79.37%		86.51%	3102
Image pre-processing + CCA	82.12%		87.26%	3182
CCA + Segmentation quality assessment	82.69%		91.25%	3064
Image pre-processing + CCA + Segmentation quality assessment	85.82%		92.23%	3146
Image pre-processing + CCA + Segmentation quality assessment + Handling stitched images	89.31%	3381	90.03%	3354
Image pre-processing + CCA + Segmentation quality assessment + Handling potential blurry and fragmented images	89.83%		91.76%	3310
Combination of all steps	**90.65%**		**92.65%**	3308

Note: The leftmost column indicates the combination method used in the respective experiment. The next column shows the accuracy obtained over the entire dataset, which consists of 3381 compound images. The second column from the right shows the accuracy calculated only over those images that were actually segmented by the respective method. The number of images segmented by each method is shown on the right. The highest accuracies are shown in boldface.

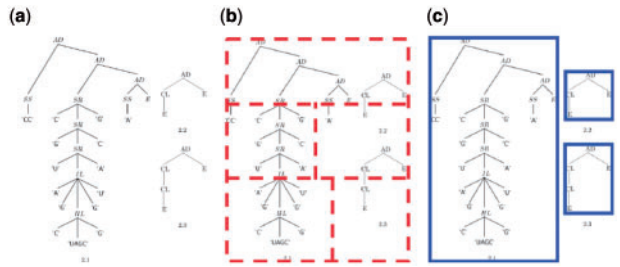

Fig. 9. Example comparing the segmentation result obtained by a gap-detection based method (Chhatkuli *et al.*, 2013) versus our system. **(a)** The compound image taken from Figure 2 in Voss *et al.* (2006). **(b)** The segmentation result on image (a) obtained using method proposed by Chhatkuli *et al.* (2013). **(c)** Our segmentation result on image (a)

Compound images that are hard to segment, including stitched, blurry or fragmented images are also handled well by our method. As an example, Figure 9 shows a tree image (a), along with its ground-truth segmentation (b), provided by ImageCLEFions The segmentation shown in 9(b) was obtained by using a gap-detection based separation method proposed by Chhatkuli *et al.* (2013) as a basis and manually correcting the automatically generated results (De Herrera *et al.*, 2013). The segmentation is clearly *incorrect* as it over-segments Figure 9a, which consists of only three panels, into five panels. Gaps within the image are erroneously detected as panel separators. In contrast, Figure 9c shows the correct segmentation obtained by our method. The quality-assessment step we use identifies the over-segmentation of the image (see e.g. Fig. 6) resulting from CCA, and applies instead the method we have described for handling potential blurry and fragmented images.

While our method successfully separates most compound images, there are a few cases where the separation is incorrect. For example, Figure 10a shows a case of under-segmentation, where the topmost part was not correctly identified as a panel. The method first detected the bounding boxes around two small disconnected gel images, within Panel A. As the height of these bounding boxes is $<1/10$ the height of the largest bounding box detected in the image, the method eliminated the small bounding boxes detected in panel A based on the *Segmentation Quality Assessment* step, leading to under-segmentation. As another example of under-segmentation, Figure 10b shows an image where the three stitched panels at the top as well as those at the bottom were incorrectly detected as two individual panels. As our method detected the two disconnected panels within the image it has pre-maturely stopped the separation process, without checking for stitched images. We plan to extend our method to re-assess the image and re-segment it if the segmented result contains compound images. Figure 10c illustrates a case of over-segmentation, where panels A and D are both split into two panels, as each of these panels consists of two well-separated components. Although the six panels we thus detect indeed do not match the four panels (A–D) annotated in the original figure, such over-segmentation may not necessarily negatively impact downstream processing. We shall investigate this issue in future studies, and possibly through the application of deep learning—if sufficient annotated data can be obtained.

Notably, our system typically yields correct panel-separation; Figure 11 illustrates additional cases of such successful separation. Figure 11(a–c) were obtained by applying our method to compound images consisting of a single type of panel, e.g. graphs, medical images, or protein images. Compound images that consist of multiple panel types present further challenges. It is harder to identify

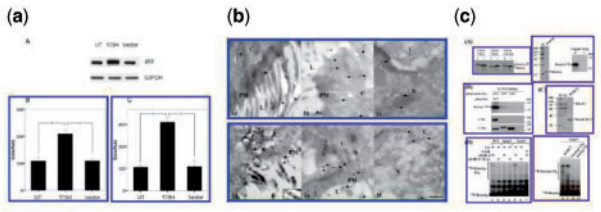

Fig. 10. Examples of inaccurate segmentation obtained by our method. **(a, b)** show two examples of under-segmentations, while **(c)** shows an example of over-segmentation. The original images **(a–c)** correspond to Figure 4 in Fazzino *et al.* (2010), Figure 3 in Piciucchi *et al.* (2011) and Figure 2 in Seibert *et al.* (2002), respectively

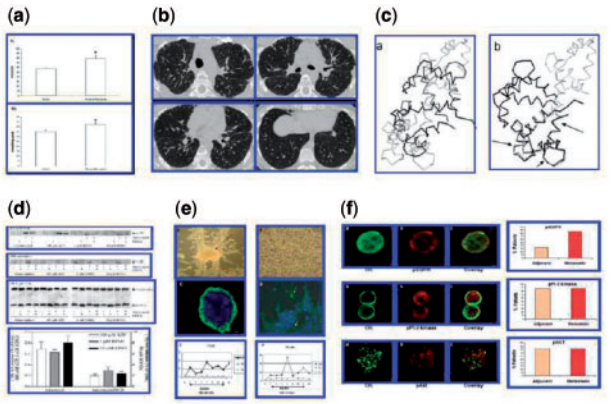

Fig. 11. Examples of successful panel separation obtained by our method. **(a–c)** show segmentation when panels are of a single modality. **(d–f)** show segmentation of panels stemming from multiple modalities. The original images **(a–f)** are taken from Figure 4 in Fazzino *et al.* (2010), Figure 3 in Piciucchi *et al.* (2011), Figure 6 in Liu *et al.* (2009), Figure 2 in Evans *et al.* (2002), Figure 2 in Koyanagi *et al.* (2010) and Figure 1 in Kallergi *et al.* (2008), respectively

the gaps for separation within such images, as the size of different type of panels and the content connectivity within different type of panels typically vary. The gap-detection based methods look for gaps throughout the whole image, and as such mis-identify some gaps as separation-gaps. In contrast, our method segments the image based on the connected components detected within the image, and as such is not affected by the variation in panel type within a compound image. Figure 11(d–f) demonstrates successful panel-separation in images consisting of multiple types of panels.

5 Conclusion

We presented a new scheme for segmenting compound images, including stitched compound images. We first proposed a pre-processing step to make gaps clearer in images so that more images are segmented. We then introduced a method based on CCA to segment images into panels. The segmentation errors were addressed by the step of segmentation quality assessment. Importantly, this step evaluates separation quality, modifies separation errors, and ensures that only correct panels are extracted from images. The over- and under-segmentation errors in very blurry images, fragmented images and stitched images are hard to correct. As such, we propose two advanced methods, namely, for handling potential blurry or fragmented images and for handling stitched images, to re-separate the three kinds of images accordingly. Handling potential blurry or fragmented images is done by dilating the content in images so that components within a panel become more connected

and the gaps between panels are used for separation. Handling of stitched images works by applying an edge detector and detecting boundaries between panels for separation. The results obtained demonstrate that our comprehensive method segments compound images more accurately than other reported methods.

As there is an increasing drive toward utilizing image information within biomedical documents, our method effectively addresses an essential need for extracting panels within published biomedical figures, while significantly improving over the state-of-the-art. As part of future work, we shall further develop methods to identify and address specific problem cases. We shall also continue to develop and improve on the interface to the FigSplit system, based on users' needs and input.

Funding

This work was partially supported by National Institutes of Health grants [R56LM011354, R01LM012527].

References

Ahmed,Z. *et al*. (2016) Mining biomedical images towards valuable information retrieval in biomedical and life sciences. *Database*, **2016**, baw118.

Antani,S. *et al*. (2008) Exploring use of images in clinical articles for decision support in evidence-based medicine. In *Proceedings of the Conference of the SPIE Document Recognition and Retrieval XV*, 6815, 68150Q.

Apostolova,E. *et al*. (2013) Image retrieval from scientific publications: Text and image content processing to separate multipanel figures. *J. Am. Soc. Inform. Sci. Technol*., **64**, 893–908.

Bonay,M. *et al*. (2005) Hepatocyte and keratinocyte growth factors and their receptors in human lung emphysema. *BMC Pulmon. Med*., **5**, 13.

Cheng,B. *et al*. (2011) Automatic segmentation of subfigure image panels for multimodal biomedical document retrieval. In *Proceedings of SPIE Document Recognition and Retrieval XVIII*, **7874**, 78740Z.

Chhatkuli,A. *et al*. (2013) Separating compound figures in journal articles to allow for subfigure classification. In *Proceding of SPIE medical imaging*, **8674**, 86740J.

Chooneea,D. *et al*. (2010) Elucidation of the outer membrane proteome of Salmonella enterica serovar Typhimurium utilising a lipid-based protein immobilization technique. *BMC Microbiol*., **10**, 44.

De Herrera,S. *et al*. (2013) The medGIFT group in ImageCLEFmed 2013. In *Working Notes of ImageCLEF'13*, Valencia, Spain.

De Herrera,A.G.S. *et al*. (2013) Overview of the ImageCLEF 2013 Medical Tasks. In *Working Notes of ImageCLEF'13*, Valencia, Spain.

De Herrera,A.G.S. *et al*. (2015) Overview of the ImageCLEF 2015 medical classification task. In *Working Notes of ImageCLEF'15*, Toulouse, France.

De Herrera,A.G.S. *et al*. (2016) Overview of the ImageCLEF 2016 Medical Tasks. In *Working Notes of ImageCLEF'16*, Evora, Portugal, pp. 219–232.

Evans,J.H. *et al*. (2002) Inhibition of the MEK1/ERK pathway reduces arachidonic acid release independently of cPLA 2 phosphorylation and translocation. *BMC Biochem*., **3**, 30.

Fazio,E.N. *et al*. (2011) Stanniocalcin 2 alters PERK signalling and reduces cellular injury during cerulein induced pancreatitis in mice. *BMC Cell Biol*., **12**, 17.

Fazzino,F. *et al*. (2010) Taurine and proliferation of lymphocytes in physically restrained rats. *J. Biomed. Sci*., **17**, S24.

Gálvez-Gastélum,F.J. *et al*. (2010) Combinatorial gene therapy renders increased survival in cirrhotic rats. *J. Biomed. Sci*., **17**, 42.

Gonzalez,R.C. and Woods,R.E. (2002) *Digital Image Processing*. Prentice Hall, NJ, USA.

Kalpathy-Cramer,J. *et al*. (2015) Evaluating performance of biomedical image retrieval systems—an overview of the medical image retrieval task at ImageCLEF 2004–2013. *Comput. Med. Imaging Graph*., **39**, 55–61.

Kallergi,G. *et al*. (2008) Phosphorylated EGFR and PI3K/Akt signaling kinases are expressed in circulating tumor cells of breast cancer patients. *Breast Cancer Res*., **10**, R80.

Keys,R. (1981) Cubic convolution interpolation for digital image processing. *IEEE Trans. Acoust. Speech Signal Process*., **29**, 1153–1160.

Kim,D. *et al*. (2011) Automatic figure classification in bioscience literature. *J. Biomed. Informatics*, **44**, 848–858.

Kitanovski,I. *et al*. (2013) FCSE at Medical Tasks of ImageCLEF 2013. In *Working Notes of ImageCLEF'13*, Valencia, Spain.

Koyanagi,M. *et al*. (2010) Diversifying selection and functional analysis of interleukin-4 suggests antagonism-driven evolution at receptor-binding interfaces. *BMC Evol. Biol*., **10**, 223.

Lehmann,T.M. *et al*. (1999) Survey: Interpolation methods in medical image processing. *IEEE Trans. Med. Imaging*, **18**, 1049–1075.

Liu,Y.S. *et al*. (2009) Using least median of squares for structural superposition of flexible proteins. *BMC Bioinformatics*, **10**, 29.

Lopez,L.D. *et al*. (2013) A framework for biomedical figure segmentation towards image-based document retrieval. *BMC Syst. Biol*., **7**, 1.

Müller,H. *et al*. (2012) Creating a classification of image types in the medical literature for visual categorization. In *Proceedings of SPIE Medical Imaging*, **8319**, 83190P.

Murphy,R.F. *et al*. (2001) Searching online journals for fluorescence microscope images depicting protein subcellular location patterns. In *Proceedings of the IEEE 2nd International Symposium on Bioinformatics and Bioengineering*, pp. 119–128.

Piciucchi,S. *et al*. (2011) High resolution CT and histological findings in idiopathic pleuroparenchymal fibroelastosis: features and differential diagnosis. *Respir. Res*., **12**, 111.

Saka,Y. and Smith,J.C. (2007) A mechanism for the sharp transition of morphogen gradient interpretation in Xenopus. *BMC Dev. Biol*., **7**, 47.

Santosh,K.C. *et al*. (2015a) NLM at ImageCLEF 2015: Biomedical Multipanel Figure Separation. In *Working Notes of ImageCLEF'15*, Toulouse, France.

Santosh, K.C. *et al*. (2015b) Stitched Multipanel Biomedical Figure Separation. In *Procceings of IEEE 28th International Symposium on Computer-Based Medical Systems*, pp. 54–59.

Seibert,V. *et al*. (2002) Combinatorial diversity of fission yeast SCF ubiquitin ligases by homo-and heterooligomeric assemblies of the F-box proteins Pop1p and Pop2p. *BMC Biochem*., **3**, 22.

Sereno,D. *et al*. (2015) Experimental study of the function of the excreted/secreted Leishmania LmSIR2 protein by heterologous expression in eukaryotic cell line. *Kinetoplastid Biol. Dis*., **4**, 1.

Shatkay,H. *et al*. (2006) Integrating image data into biomedical text categorization. *Bioinformatics*, **22**, e446–e453.

Simpson,M.S. *et al*. (2013) ITI's Participation in the 2013 Medical Track of ImageCLEF. In *Working Notes of ImageCLEF'13*, Valencia, Spain.

Smith,S.M. and Brady,J.M. (1997) SUSAN-a new approach to low level image processing. *Int. J. Comput. Vis*., **23**, 45–78.

Taschwer,M. and Marques,O. (2015) AAUITEC at ImageCLEF 2015: Compound Figure Separation. In *Working Notes of ImageCLEF'15*, Toulouse, France.

Taschwer,M. and Marques,O. (2016) Compound Figure Separation Combining Edge and Band Separator Detection. In *Proceedings of International Conference on Multimedia Modeling*, pp. 162–173.

Voss,B. *et al*.. (2006) Complete probabilistic analysis of RNA shapes. *BMC Biol*., **4**, 5.

Wang,X. *et al*. (2015) CIS UDEL working notes on Image-CLEF 2015: Compound figure detection task. In *Working Notes of ImageCLEF'15*, Toulouse, France.

Yuan,X. and Ang,D. (2014) A novel figure panel classification and extraction method for document image understanding. *Int. J. Data Mining Bioinformatics*, **9**, 22–36.

Zhu,F. and Qian,C. (2006) Berberine chloride can ameliorate the spatial memory impairment and increase the expression of interleukin-1beta and inducible nitric oxide synthase in the rat model of Alzheimer's disease. *BMC Neurosci*., **7**, 78.