Compound Figure Separation Combining Edge and Band Separator Detection

Mario Taschwer[1] and Oge Marques[2]

[1] ITEC, Klagenfurt University (AAU), Austria
`mario.taschwer@aau.at`
[2] Florida Atlantic University (FAU), Boca Raton, FL, USA
`omarques@fau.edu`

Abstract. We propose an image processing algorithm to automatically separate compound figures appearing in scientific articles. We classify compound images into two classes and apply different algorithms for detecting vertical and horizontal separators to each class: the edge-based algorithm aims at detecting visible edges between subfigures, whereas the band-based algorithm tries to detect whitespace separating subfigures (separator bands). The proposed algorithm has been evaluated on two datasets for compound figure separation (CFS) in the biomedical domain and compares well to semi-automatic or more comprehensive state-of-the-art approaches. Additional experiments investigate CFS effectiveness and classification accuracy of various classifier implementations.
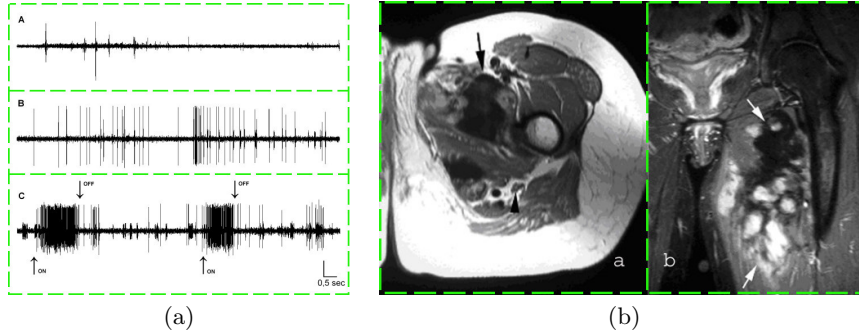
1 Introduction

Due to a substantial amount of compound figures in the biomedical literature[3], the automatic separation of these figures into subfigures has been recently identified as a relevant research problem for content-based analysis and image-based information retrieval in collections of biomedical articles [1, 4, 9, 11]. From the few approaches known from literature [1, 3, 6–8, 11], all but one [7] focus on the detection of homogeneous image regions separating subfigures, which we call *separator bands*, as illustrated by Fig. 1(a). These approaches fail for compound images where subimages are stitched together without separator bands, as shown in Fig. 1(b).

We therefore propose a method that provides separate algorithms for detecting separator bands and separator edges and selects the appropriate algorithm for a given compound image using the prediction of an image classifier. The classifier is trained to distinguish between graphical illustrations and other images in biomedical articles, based on the observation that compound images containing graphical illustrations almost always contain separator bands between subfigures, whereas most subfigures in other compound images show rectangular border edges.

---

[3] In recently published datasets drawn from open access biomedical literature, between 40% and 60% of figures occurring in articles are compound figures [1, 3, 4].

Fig. 1. Sample compound images (of the ImageCLEF 2015 CFS dataset [5]) suitable for two different separator detection algorithms. Subfigures are separated by (a) white-space, (b) a vertical edge. Dashed lines represent the expected output of CFS.

The proposed approach builds upon previous work [10] and adds the following new research contributions: (1) the proposed algorithm outperforms state-of-the-art automatic and semi-automatic CFS approaches on two recently published biomedical datasets; and (2) several implementation options for the illustration classifier have been evaluated with respect to effectiveness for CFS and classification accuracy.
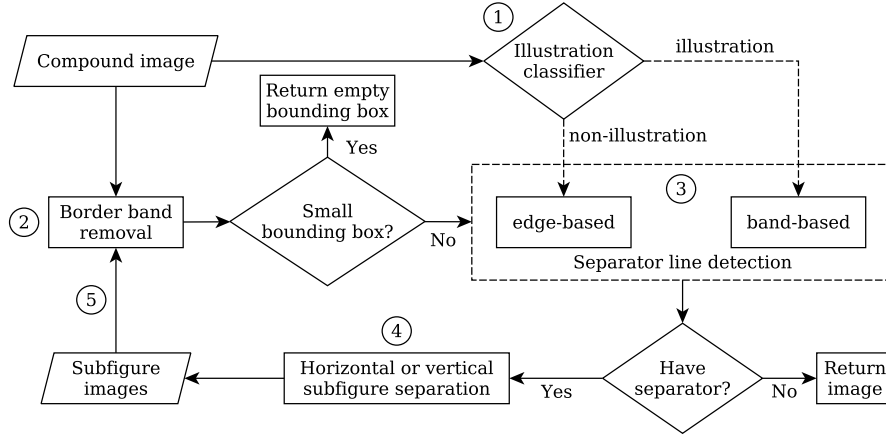
2 Related Work and Context

Until now there has been little research on the CFS problem in the literature, probably because its relevance had not been recognized by the research community until ImageCLEF 2013, where a CFS task was introduced as one of the challenges in the biomedical domain [4]. Task organizers provided training and test datasets, and evaluated CFS results submitted by participants for the test dataset. The few approaches resulting from participation [1, 3, 6] as well as other proposed approaches [8, 11] focus on the detection of separator bands, and hence fail for compound images where subimages are stitched together without separator bands. Yuan and Ang [11] additionally used an edge-based approach involving Hough transform to separate overlayed zoom-in views from background figures, a case that is not considered in this work.

The next CFS task at ImageCLEF in 2015 [5] stimulated further work on the CFS problem. The approach of NLM (U.S. National Library of Medicine) [7] and our previous approach [10] independently proposed to address compound images without separator bands by processing edge detection results. Besides algorithmic differences in edge-based separator detection, our approach incorporates a classifier to automatically select edge- or band-based separator detection, whereas NLM's approach uses manual image classification for evaluation.[4]

---

[4] We therefore call NLM's approach [7] *semi-automatic*, although an automatic classifier could be easily integrated.

3   Proposed Algorithm

Our approach to compound figure separation is a recursive algorithm (see Fig. 2) comprising the following steps: (1) classification of the compound image as illustration or non-illustration image, (2) removal of border bands, (3) detection of separator lines, (4) vertical or horizontal separation, and (5) recursive application to each subfigure image. The *illustration classifier* is used to decide which of two separator line detection modules to apply: if the compound image is classified as an illustration image, the *band-based* algorithm is applied, which aims at detecting separator bands between subfigures. Otherwise, the image is processed by the *edge-based* separator detection algorithm, which applies edge detection and Hough transform to locate candidate separator edges. The algorithm selection is based on the assumption that edge-based separator detection is better suited for non-illustration compound images due to visible vertical or horizontal edges separating subfigures. Note that this assumption is not violated by non-illustration compound images with separator bands where subfigures have a visible rectangular border. The following four sections describe the illustration classifier, the main recursive algorithm, and the two separator detection modules in more detail.

Fig. 2. Recursive algorithm for compound figure separation. Numbers denote the main algorithmic steps described in the beginning of Section 3.

3.1   Illustration Classifier

The illustration classifier is used to decide which separator detection algorithm to apply to a given compound image. If the image is predicted to be a graphical

illustration with probability greater than `decision_threshold`, the band-based separator detection is applied, otherwise the edge-based separator module is used. This decision is made only once for each compound image, so all recursive invocations use the same separator detection algorithm.

For training the classifier, we use the dataset of the ImageCLEF 2015 multi-label image classification task [5]. The training dataset consists of 1071 images annotated with one or more labels of 29 classes (organized in a class hierarchy), which have been mapped to two meta classes: the *illustration* meta class comprises all "general biomedical illustration" classes except for chromatography images, screenshots, and non-clinical photos. These classes and all classes of diagnostic images have been assigned to the *non-illustration* meta class. About 36% of the images in the training set are labeled with multiple classes; most of them represent compound images.

Classifier training requires mapping the set of labels of a given image to a single meta class. We implemented four mapping strategies and selected the most effective one during parameter optimization (Section 4). All mapping strategies first assign each image label to the *illustration* or *non-illustration* meta class as described above, and then operate differently on the list L of meta labels associated with a given image: (1) the *first* strategy simply assigns the first meta label of L to the image; (2) the *majority* strategy selects the meta label occurring most often in L, dropping the image from the training set if both meta labels occur equally often; (3) the *unanimous* strategy only assigns a meta label to the image if all meta labels in L are equal, otherwise the image is dropped from the training set; and (4) the *greedy* strategy maps an image to the *illustration* label if L contains at least one such meta label, otherwise the image is assigned the *non-illustration* label. The *greedy* strategy is inspired by the assumption that a compound image containing an illustration subfigure can be processed more effectively by the band-based separator detection algorithm than by the edge-based algorithm. Whereas mapping strategies *first* and *greedy* could use all 1071 images in the original training set, *majority* and *unanimous* strategies resulted in reduced training sets of 895 and 867 images, respectively.

Due to promising effectiveness for CFS in early experiments, we use four sets of global image features as classifier input, computed after gray-level conversion: (1) *simple2* is a two-dimensional feature consisting of image entropy, estimated using a 256-bin histogram, and mean intensity; (2) *simple11* extends *simple2* by 9 quantiles of the intensity distribution; (3) *CEDD* is the well-known color and edge directivity descriptor [2] (144-dimensional); and (4) *CEDD_simple11* is the concatenation of *CEDD* and *simple11* features (155-dimensional).

As machine learning algorithms we consider support vector machines (SVM) with radial basis function kernel (RBF) and logistic regression. Although logistic regression is generally inferior to kernel SVM due to its linear decision boundary, it has the advantage of providing prediction probabilities, which allow us to tune the selection of separator detection algorithms using the `decision_threshold` parameter. Classifier performance and its effectiveness for CFS are evaluated in Section 5.

3.2 Recursive Algorithm

Before applying the main algorithm (Fig. 2) to a given compound figure image, it is converted to 8-bit gray-scale. *Border band removal* detects a rectangular bounding box surrounded by a maximal homogeneous image region adjacent to image borders (border band). If the resulting bounding box is empty or smaller than `elim_area` or if maximal recursion depth has been reached, an empty bounding box is returned, terminating recursion. The *separator line detection* modules are invoked separately for vertical and horizontal directions, so they deal with a single direction θ and return a list of corresponding separator lines. An empty list is returned if the respective image dimension (width or height) is smaller than `mindim` or if no separator lines are found. If the returned lists for both directions are empty, recursion is terminated and the current image (without border bands) is returned. The *decision about vertical or horizontal separation* is trivial if one of both lists of separator lines is empty. Otherwise the decision is made based on the regularity of separator distances: locations of separator lines and borders are normalized to the range [0,1], and the direction (vertical or horizontal) yielding the lower variance of adjacent distances is chosen. Finally, the current figure image is divided into subimages along the chosen separation lines, and the algorithm is applied recursively to each subimage.

3.3 Edge-based Separator Detection

The edge-based separator line detection algorithm aims at detecting full-length edges of a certain direction θ (vertical or horizontal) in a given gray-scale image. It comprises the following processing steps: (1) unidirectional edge detection, (2) peak selection in one-dimensional Hough transform, and (3) consolidation and filtering of candidate edges.

Edge detection is implemented by a one-dimensional Sobel filter und subsequent thresholding (`edge_sobelthresh`) to produce a binary edge map. The one-dimensional Hough transform counts the number of edge points aligned on each line in direction θ. So the peaks correspond to the longest edges, and their locations identify candidate separator edges. To make borders appear as strong Hough peaks, we add an artificial high-contrast border to the image prior to edge detection. Peaks are identified by an adaptive threshold t that depends on the recursion depth k (zero-based), the maximal value m of the current Hough transform, and the fill ratio f of the binary edge map (fraction of non-zero pixels, $0 \leq f \leq 1$), see (1). α and β are internal parameters (`edge_houghratio_min` and `edge_houghratio_base`).

$$h = \alpha * \beta^k \ , \qquad t = m * \left(h + (1-h) * \sqrt{f} \right) \ . \tag{1}$$

The rationale behind these formulas is to cope with noise in the Hough transform. Hough peaks were observed to become less pronounced as image size decreases (implied by increasing recursion depth) and as the fill ratio f increases (more edge points increase the probability that they are aligned by chance). Equation (1) ensures a higher threshold in these cases. Additionally, as recursion

depth increases, the algorithm should detect only more pronounced separator edges, because further figure subdivisions become less likely.

Hough peak selection also includes a similar regularity criterion as used for deciding about vertical or horizontal separation (see Section 3.2): the list of candidate peaks is sorted by their Hough values in descending order, and candidates are removed from the end of the list until the variance of normalized edge distances of remaining candidates falls below a threshold (`edge_maxdistvar`). Candidate edges resulting from Hough peak selection are then consolidated by filling small gaps (of maximal length given by `edge_gapratio`) between edge line segments (of minimal length given by `edge_lenratio`). Finally, edges that are too short in comparison to image height or width (threshold `edge_minseplength`), or too close to borders (threshold `edge_minborderdist`) are discarded.

3.4 Band-based Separator Detection

The band-based separator detection algorithm aims at locating homogeneous rectangular areas covering the full width or height of the image, which we call *separator bands*. Since this algorithm is intended primarily for gray-scale illustration images with light background, we assume that separator bands are white or light gray. The algorithm consists of four steps: (1) image binarization, (2) computation of mean projections, (3) identification and (4) filtering of candidate separator bands.

Initially, we binarize the image using the mean intensity value as a threshold. We then compute mean projections along direction θ (vertical or horizontal), that is, the mean value of each line of pixels in this direction. A resulting mean value will be 1 (white) if and only if the corresponding line contains only white pixels. Candidate separator bands are then determined by identifying maximal runs of ones in the vector of mean values that respect a minimal width threshold (`band_minsepwidth`). They are subsequently filtered using a regularity criterion similar to Hough peak selection (see Section 3.3), this time using distance variance threshold `band_maxdistvar`. Finally, selected bands that are close to the image border (threshold `band_minborderdist`) are discarded, and the center lines of remaining bands are returned as separator lines.

4 Parameter Optimization

The proposed CFS algorithm takes a number of internal parameters[5], whose initial values were chosen manually by looking at the results produced for a few training images. They were used during participation in ImageCLEF 2015 [10]. For parameter optimization, the CFS algorithm was evaluated for various parameter combinations on the ImageCLEF 2015 CFS training dataset (3,403

---

[5] Section 3 describes 17 internal parameters. A table with initial and optimized parameter values could not be included due to space constraints, but will be provided by authors upon request.

compound images, 14,531 ground-truth subfigures) using the evaluation tool provided by ImageCLEF organizers. Due to the number of parameters and the run time of a single evaluation run (about 17 minutes), a grid-like optimization evaluating all possible parameter combinations in a certain range was not feasible. Instead, we applied a hill-climbing optimization strategy to locate the region of a local maximum and then used grid optimization in the neighborhood of this maximum.

More precisely, we defined up to five different values per parameter, including the initial values, on a linear or logarithmic scale, depending on the parameter. Then a set of parameter combinations was generated where only one parameter was varied at a time and all other parameters were kept at their initial values, resulting in a feasible number of parameter combinations to evaluate (linear in the number of parameters). After measuring accuracy on the training set, the most effective value of each parameter was chosen as its new *optimal* value. For parameters whose optimal values differed from the initial ones, the range was centered around the optimal value. Other parameters were fixed at their latest value. The procedure was repeated until accuracy improved by no more than 5%, which happened after three iterations. Finally, after sorting parameter combinations by achieved accuracy, the five most effective parameters were chosen for grid optimization, where only two "nearly optimal" values (including the latest optimal value) per parameter were selected.

The effect of parameter optimization was surprisingly strong: whereas the initial parameter configuration achieved an accuracy of 43.5% on the training set, performance increased to 84.5% after hill-climbing optimization, and finished at 85.5% after grid optimization.

5 Evaluation

The proposed CFS algorithm is evaluated on two datasets and compared to state-of-the-art approaches. To shed light on the effectiveness of the illustration classifier, we evaluate the CFS algorithm without a classifier and with different classifier implementations on the ImageCLEF 2015 dataset and compare results with the best algorithm submitted to ImageCLEF 2015 (Section 5.1). To evaluate the generalization capability of our algorithm, we apply it to the dataset of Apostolova et al. [1] (U.S. National Library of Medicine, NLM) using the same parameters optimized on the ImageCLEF dataset, and compare it to the image panel segmentation approach of the authors (Section 5.2). Finally, we compare classification accuracy of the classifier implementations used before on a third dataset and confirm that classification performance is not a critical factor for CFS effectiveness (Section 5.3).

5.1 Evaluation on ImageCLEF Dataset

The ImageCLEF 2015 CFS test dataset [5] contains 3,381 compound images with 12,789 ground-truth subfigures. The evaluation tool provided by ImageCLEF

organizers computes the accuracy of detected subfigures for a given compound figure as follows. A detected subfigure is associated with at most one ground-truth subfigure G of maximal overlap if the overlap ratio is greater than 2/3 and if G is not already associated with a different detected subfigure. The number C of such associations for a given compound figure represents the number of correctly detected subfigures (true positives), and accuracy is defined by $C/\max(N_G, N_D)$, where N_G and N_D are the numbers of ground-truth and detected subfigures, respectively. Accuracy on the test set is the average of accuracy values computed for each compound figure.

Experimental results are shown in Table 1. For comparison, we also included a previous version of our approach [10] that did not use optimized parameters, and the best approach submitted to ImageCLEF 2015 (by NLM). We evaluated the proposed algorithm with optimized parameters (see Section 4) and with different implementations and feature sets for the illustration classifier, as described in Section 3.1. Because logistic regression using *simple2* features was found to be most effective by parameter optimization when trained on the *greedy* set, we focused on this training set when evaluating other classifier implementations. Internal SVM parameters were optimized on the entire ImageCLEF 2015 multi-label classification test dataset (see Section 5.3) to maximize classification accuracy. The optimized `decision_threshold` parameter for deciding between edge-based and band-based separator detection is effective only for logistic regression classifiers, because SVM predictions do not provide class probabilities. To confirm the effectiveness of the illustration classifier, we also included results for algorithm variants where the classifier has been replaced by a random decision selecting band-based separator detection with probability p. The value $p = 0.741$ corresponds to the decision rate of the most effective classifier (LogReg,simple11,greedy). $p = 0$ and $p = 1$ represent algorithms that always use edge-based or band-based separator detection, respectively. Finally, we considered an algorithm variant (*SubfigureClassifier*) that applied the illustration classifier not only once per compound image, but also to each subimage that is to be further divided by recursive figure separation.

When comparing our results to NLM's approach, we note that the authors of [7] manually classified the test set into stitched (4.3%) and non-stitched (95.7%) images, whereas our approach uses automatic classification. Using band-based separator detection for all test images (no classifier, $p = 1$) works surprisingly well (82.2% accuracy), which can be explained by the low number of stitched compound images in the test set. On the other hand, using edge-based separator detection for all test images (no classifier, $p = 0$) results in modest performance (58% accuracy), which we attribute to a significant number of subfigures without rectangular borders (illustrations) in the test set. Selecting edge-based or band-based separator detection using the illustration classifier improved accuracy for all tested classifier implementations. In fact, it turned out to be effective to bias the illustration classifier towards band-based separator detection and apply edge-based separator detection only to high-confidence non-illustration images. This happened in two ways: by using the *greedy* training set, and by optimizing

Table 1. Experimental results on the ImageCLEF 2015 CFS test set. Illustration classifiers are described in Section 3.1 (LogReg = logistic regression). BB denotes the percentage of images (or decisions*) where band-based separator detection was applied.

Algorithm	Classifier	BB %	CFS Accuracy %
Previous [10]	LogReg,simple2,first		49.4
NLM [7]	manual	95.7	84.6
Proposed	LogReg,simple2,first	61.6	84.2
Proposed	LogReg,simple2,majority	61.1	84.1
Proposed	LogReg,simple2,unanimous	61.8	84.2
Proposed	LogReg,simple2,greedy	75.8	84.8
Proposed	LogReg,simple11,greedy	74.1	**84.9**
Proposed	SVM,simple2,greedy	58.6	83.5
Proposed	SVM,simple11,greedy	60.3	83.5
Proposed	SVM,CEDD,greedy	59.2	82.8
Proposed	SVM,CEDD_simple11,greedy	59.6	83.2
Proposed	random,p=0.741	74.7	75.4
Proposed	no classifier,p=0	0	58.0
Proposed	no classifier,p=1	100	82.2
SubfigureClassifier	LogReg,simple11,greedy	60.1*	84.0

the `decision_threshold` parameter for the logistic regression classifier. This explains why best results were obtained by logistic regression classifiers trained on the *greedy* training set.

To further analyze the effectiveness of separator detection selection, we partitioned the CFS test dataset into two classes according to decisions of the most effective CFS algorithm variant (LogReg,simple11,greedy) and evaluated detection results of this algorithm separately on the two partitions. Resulting accuracy values of 85.7% on the edge-based partition and 84.6% on the band-based partition show that the classifier was successful in jointly optimizing detection performance for both separator detection algorithms.

5.2 Evaluation on NLM Dataset

Apostolova et al. [1] used a different criterion to evaluate the accuracy of detected subfigures. Each detected subfigure F is decided to be *true positive* or *false positive* according to the following rule. Let $\{G_i \,|\, i \in I\}$ be the set of ground-truth subfigures overlapping with F, let A_i be the area size of G_i, and O_i the size of the overlapping area between F and G_i. The subfigure F is considered true positive if and only if there is an index $j \in I$ with $O_j/A_j > 0.75$ and $O_i/A_i < 0.05$ for all $i \in I, i \neq j$. That is, subfigure F has a notable overlap with one ground-truth subfigure only.

Given the total number N of ground-truth subfigures in the dataset, the total number D of detected subfigures, and the number T of detected true positive subfigures, the usual definitions for classifier evaluation measures can be applied to obtain precision P, recall R, and F_1 measure, see (2). Note that accuracy is

not well-defined in this situation, because the number of negative results (not detected arbitrary bounding boxes) is theoretically unlimited.

$$P = \frac{T}{D} \ , \quad R = \frac{T}{N} \ , \quad F_1 = \frac{2 * P * R}{P + R} \ . \tag{2}$$

The dataset created by Apostolova et al. [1] contains 398 images with 1754 ground-truth subfigures. Table 2 shows the results of evaluating our proposed algorithm on this dataset using the measures described above. Note that we used the same parameter settings as in Section 5.1 to demonstrate the generalization capability of our algorithm. We selected the most effective illustration classifiers using logistic regression and SVM, respectively. They both use *simple11* features and the *greedy* training set. For convenience, we also included the results reported in [1] for a direct comparison with our approach.[6]

Table 2. Evaluation results on the NLM CFS dataset [1]. Precision, recall, and F_1 score are computed from the total number of detected (D) and true positive (T) subfigures.

Algorithm	D	T	Precision %	Recall %	F_1 %
Proposed (LogReg)	1646	1407	85.5	**80.2**	**82.8**
Proposed (SVM)	1681	1392	82.8	79.4	81.1
Apostolova et al. [1]	1482	1276	**86.1**	72.3	78.6

Results show that the relative performance of the proposed algorithm using different classifiers is consistent with evaluation results in Section 5.1. The proposed algorithm could detect 10% more true positive subfigures than the image panel segmentation algorithm of Apostolova et al. [1], leading to a higher recall rate. On the other hand, precision is only slightly lower. Note that algorithm [1] has been used as a component in NLM's CFS approach [7] referenced in Section 5.1.

5.3 Illustration Classifier Accuracy

To investigate the correlation of illustration classifier performance and effectiveness for CFS, we evaluated classification accuracy for the various classifier implementations considered in Section 5.1 on the test dataset of the ImageCLEF 2015 multi-label image classification task [5]. Labels of test images were mapped to binary meta classes using the same procedure as described in Section 3.1, resulting in 497 images for *first* and *greedy* test sets, 428 images for *majority*, and 398 images for *unanimous* test set. Evaluation results are shown in Table 3. The decision threshold for logistic regression was set to 0.5 to provide a fair comparison with SVM. Internal parameters of SVM (box constraint C and standard

---

[6] The dataset reported in [1] contains 400 images with 1764 ground-truth subfigures, so reported recall may be up to 0.4% higher if evaluated on the 398 images of the dataset available to us.

deviation σ of RBF kernel) were optimized using two-fold cross-validation on the test set.

Table 3. Classification accuracy on ImageCLEF 2015 multi-label image classification test dataset (497 images) for different implementation options of illustration classifier. Features and training sets are described in Section 3.1, LogReg = logistic regression.

Classifier	Features	Training Set	Accuracy %
LogReg	simple2	first	82.5
LogReg	simple2	majority	86.5
LogReg	simple2	unanimous	**88.2**
LogReg	simple2	greedy	84.7
LogReg	simple2	greedy	84.7
LogReg	simple11	greedy	83.7
SVM	simple2	greedy	84.3
SVM	simple11	greedy	84.3
SVM	CEDD	greedy	**87.1**
SVM	CEDD_simple11	greedy	86.7

The upper part of Table 3 tells us that *majority* and *unanimous* training sets improve classification performance, although we know from Section 5.1 that this does not help CFS effectiveness. From the lower part of Table 3 we note that, interestingly, SVM does not perform better on *simple2* features than logistic regression and causes only a modest improvement (around 3%) on CEDD features (144-dimensional). This may indicate the need to select more discriminative features for this classification task in future work, although results of Section 5.1 suggest that accuracy of the illustration classifier is not a critical factor of the proposed CFS algorithm.

6 Conclusion and Further Work

We proposed a recursive image processing algorithm for automatic separation of compound figures appearing in scientific articles. The algorithm has been evaluated on two recently published CFS datasets and achieved a detection performance slightly better than state-of-the-art approaches, even though it was compared to a semi-automatic approach [7] and not all known useful techniques (e.g. image markup removal and subfigure label recognition [1]) have been incorporated. Future work may therefore include the integration of such techniques into the proposed algorithm.

The use of the illustration classifier to select either edge-based or band-based separator detection for a given compound figure proved to be effective to improve CFS detection accuracy in conducted experiments. From evaluation of two classifier implementations, four image features, and four training sets, we conclude that classification accuracy is not a critical factor for CFS effectiveness, but biasing the classifier towards the illustration class improves CFS detection

accuracy. We explain these results by the observation that band-based separator detection works well for almost all compound figures in the dataset except for "high-confidence" non-illustration images, where edge-based separator detection is the more effective choice. Finding more discriminative image features and better training sets for the classifier to improve CFS effectiveness of the proposed algorithm may be an additional subject of future work.

References

1. Apostolova, E., You, D., Xue, Z., Antani, S., Demner-Fushman, D., Thoma, G.R.: Image retrieval from scientific publications: Text and image content processing to separate multipanel figures. J. Assoc. Inf. Sci. Technol. 64(5), 893–908 (2013)
2. Chatzichristofis, S.A., Boutalis, Y.S.: CEDD: Color and edge directivity descriptor: A compact descriptor for image indexing and retrieval. In: Computer Vision Systems, LNCS, vol. 5008, pp. 312–322. Springer (2008)
3. Chhatkuli, A., Foncubierta-Rodríguez, A., Markonis, D., Meriaudeau, F., Müller, H.: Separating compound figures in journal articles to allow for subfigure classification. Proc. SPIE 8674, 86740J–86740J–12 (2013)
4. García Seco de Herrera, A., Kalpathy-Cramer, J., Demner-Fushman, D., Antani, S., Müller, H.: Overview of the ImageCLEF 2013 medical tasks. In: CLEF 2013 Working Notes. CEUR Proc., vol. 1179 (2013), http://ceur-ws.org/Vol-1179/
5. García Seco de Herrera, A., Müller, H., Bromuri, S.: Overview of the ImageCLEF 2015 medical classification task. In: CLEF 2015 Working Notes. CEUR Proc., vol. 1391 (2015), http://ceur-ws.org/Vol-1391/
6. Kitanovski, I., Dimitrovski, I., Loskovska, S.: FCSE at medical tasks of ImageCLEF 2013. In: CLEF 2013 Working Notes. CEUR Proc., vol. 1179 (2013), http://ceur-ws.org/Vol-1179/
7. Santosh, K., Xue, Z., Antani, S., Thoma, G.: NLM at ImageCLEF 2015: Biomedical multipanel figure separation. In: CLEF 2015 Working Notes. CEUR Proc., vol. 1391 (2015), http://ceur-ws.org/Vol-1391/
8. Shatkay, H., Chen, N., Blostein, D.: Integrating image data into biomedical text categorization. Bioinformatics 22(14), e446–e453 (Jul 2006), http://dx.doi.org/10.1093/bioinformatics/btl235
9. Simpson, M.S., You, D., Rahman, M.M., Xue, Z., Demner-Fushman, D., Antani, S., Thoma, G.: Literature-based biomedical image classification and retrieval. Comput. Med. Imag. Graph. 39, 3–13 (2015)
10. Taschwer, M., Marques, O.: AAUITEC at ImageCLEF 2015: Compound figure separation. In: CLEF 2015 Working Notes. CEUR Proc., vol. 1391 (2015), http://ceur-ws.org/Vol-1391/
11. Yuan, X., Ang, D.: A novel figure panel classification and extraction method for document image understanding. Int. J. Data Min. Bioinformatics 9(1), 22–36 (Nov 2014), http://dx.doi.org/10.1504/IJDMB.2014.057779