Data GroundTruth, Complexity and Evaluation Measures for Color Document Analysis

Leon Todoran, Marcel Worring, and Arnold W.M. Smeulders

Intelligent Sensory Information Systems, University of Amsterdam,
Kruislaan 403, 1098SJ Amsterdam, The Netherlands
{todoran, worring, smeulders }@science.uva.nl
http://www.science.uva.nl/~{todoran,worring,smeulders}

Abstract. Publications on color document image analysis present results on small, non-publicly available datasets. We propose in this paper a well defined and groundtruthed color dataset existing of over 1000 pages, with associated tools for evaluation. The color data groundtruthing and evaluation tools are based on a well defined document model, complexity measures to assess the inherent difficulty of analyzing a page, and well founded evaluation measures. Together they form a suitable basis for evaluating diverse applications in color document analysis.

1 Introduction

Color is now playing an important role in publishing everything from scientific journals, newspapers, magazines, to advertisements. The nature of documents in current applications is therefore rapidly shifting from simple black-and-white documents to complex color documents. Some tools for color documents as color OCR [3, 18], color document compression [1], and color string localization [12, 2, 4, 6] have been developed. However, whereas document analysis for black-and-white documents is mature, color document analysis is still in its infancy.

Two factors have been instrumental in advancing the field of black-and-white document analysis. Firstly, the existence of public domain data sets like the UW[9] and MTDB [15], freeing researchers from the labor intensive task of creating datasets to work on. Secondly, the availability of standard evaluation tools for OCR and page segmentation [10], [20], [14] allowing knowledge exchange between different researchers.

For color document image analysis, no such data set standardization has taken place. As a consequence, each developer now uses its own dataset for evaluating tools. Typically the data sets used are small as providing a ground truth for color documents is a time consuming task. In this paper we report on the creation of a large dataset with ground truth which could be a first step in standardizing the evaluation of color document analysis.

The dataset consists of over 1000 pages with a ground truth describing the document components, their layout and logical structure. As we focus on aspects specific to color documents, we leave out the document textual content

in the ground truth. In fact, we make the assumption that whenever a system can reliably decompose a document into its constituent components and their structure, that existing OCR methods can extract the content from a text zone.

The documents in the dataset show a great variety in complexity, ranging from simple one-column pages with one picture, to pages with several layers of document objects with multiple overlapping pictures. It is important to be able to quantify the complexity of a document in the collection prior to evaluation. If the complexity of documents in a dataset is known and well-defined, the complexity measures can be used to weight the evaluation results leading to evaluation independent of page difficulty [5]. It should be noted here that complexity is task dependent. A document can be simple for one task while being very difficult for another. Therefore, there is a need for a set of measures that collectively cover the whole document analysis process.

Such a set of complexity measures would rank the data, but evaluation measures are needed to assess the algorithm's performance on that data.

The existing evaluation methods for layout analysis can be grouped into two main categories: text-based and region-based evaluation. Text-based evaluation [7] uses textual ground truth and the edit distance to measure the errors in layout detection. Region-based evaluation methods [20, 9, 8, 10] compare the outline of the detected zones with the zone description in the ground truth.

For evaluating document analysis algorithms for color documents the region based methods are most suited as they can easily be applied to both text, pictures, and graphics. We do, however, have to extend them first to be able to evaluate color document analysis.

This paper is organized as follows. In Section 2 we describe the dataset and a model for its content. Section 3 defines the complexity of the documents with respect to the different tasks in color document analysis. For each of these tasks an appropriate evaluation measure is derived in Section 4. Finally, Section 5 discusses how the ground truth is generated .

2 Document dataset

In this section we will describe the documents that comprise the document dataset. We then define models to describe the content of each document.

2.1 Dataset content

A dataset for evaluation of color document analysis should be created following some guidelines. Firstly, to cover different applications, the dataset must be comprised of document pages of varying style and complexity. Secondly, color must be an essential component of the message the author wants to convey. Otherwise, the document is probably equivalent to a black-and-white document.

We found that commercial color magazines form the most representative category of color documents. Even inside a single issue the document pages show a great variety in style, ranging from simple pages containing text only, to

highly complex color advertisements. Especially in the latter category of pages, the color is chosen carefully to attract the readers' attention. A system tested well on such a dataset will perform well on most other applications.

For the UvA Color Document Dataset, we have scanned (300dpi, color-24bits) full issues of the internationally available magazines listed in Table 1. These are representatives of scientific magazines, informative magazines, lifestyle magazines, and weekly news magazines. The issues together form a dataset of more than one thousand scanned pages.

The dataset set is made available via a website[1]. Access to this site is restricted to registered researchers. To use the images in publications each author should individually seek permission from the magazines' publication office.

2.2 The document model

For defining the ground truth, which provides the basis for evaluation, a document model is needed that captures all essential information in the document.

The model should be based on two different views of the document: the layout information - encoding the presentation of the document - and the logical information - encoding the meaning of the document.

The basic entities in both views are the n document objects in the document object set $\mathcal{O} = \{o_1, o_2, ..., o_n\}$, which hold the content of the document. Each document object is an entity in which the content has a uniform style expressing some intention of the author. So, an element in $\mathcal{O}$ can for example be a single picture used as illustration or a text line in bold acting as a header.

The two different views of the content of a document objects use different attributes to describe the content. Layout attributes are restricted to the geometric and color properties of the document objects. Logical attributes are functional labels expressing the function of the document object. The object sets $\mathcal{O}_g$ and $\mathcal{O}_l$ denote the set $\mathcal{O}$ with geometric and logical attributes added respectively.

An element in $\mathcal{O}$ does not appear in isolation, but an author adds structure to the set $\mathcal{O}$. A simple tree, often use for black-and-white documents, cannot describe all possible spatial relations between the document objects. Separate graphs are used to describe relations like overlap and inclusion. Thus the layout structure is given by a set of graphs where the vertices are the document objects $\mathcal{O}_g$ and the edges $\mathcal{R}_g$ denote a relation between the objects. The graphs can be directed or undirected and can have weights to encode attributes of the edges. As the vertices are the same for every graph, the layout structure is defined as follows: $\mathcal{G} = \langle \mathcal{O}_g, \mathcal{R}_g^1, \mathcal{R}_g^2, ... \rangle$. Similarly the logical structure is defined as: $\mathcal{L} = \langle \mathcal{O}_l, \mathcal{R}_l^1, \mathcal{R}_l^2, ... \rangle$.

Although logical structure (and sometimes layout) can span different pages, we use, for simplicity, a page based approach where every page receives a layout and logical structure. A full document $\mathcal{D}$ is represented by: $\mathcal{D} = \langle (\mathcal{G}_1, \mathcal{L}_1), (\mathcal{G}_2, \mathcal{L}_2) ... \rangle$. In the following subsections we describe how the generic model defined above is instantiated to describe the ground truth for the dataset.

[1] http://www.science.uva.nl/uva-doc

2.3 Geometric description

For the geometric description of a document we consider three major different categories of documents objects namely text, image, and graphics.

In the description of the outline of these objects we make a distinction between the *shape* and the *region* of a document object. With shape we denote the perceived shape of the object which in a layered document could be partly obscured by another document object. The object region is the true shape of the object. In the following, the object itself will be indicated as o, the shape of the object as $\overline{o}$ and the region of the object as $\hat{o}$. In a similar way $\hat{O}$ where O is a set of objects denotes the regions of all objects in the set.

To describe a geometric document object, the following attributes are used:

- geometric attributes;
  - category: {text, image, graphics};
  - shape: {line, rectangle, polygon, ellipse};
  - object region: set of polygons with possible holes;
  - orientation: horizontal, vertical, other;
- color attributes for text objects;
  - text: {uniform, mixture of uniform colors, texture}
  - background: {uniform, mixture of uniform colors, image, texture }

For later use, let us define notations for the following subsets of geometric document objects based on individual category and one mixed class for pictorial information: $T=\{o\in\mathcal{O}_g|category(o)=text\}$, $G=\{o\in\mathcal{O}_g|category(o)=graphics\}$, $I=\{o\in\mathcal{O}_g|category(o)=image\}$ and $P=\mathbf{G}\cup\mathbf{I}$. With respect to the shape of the document object we have: $\mathcal{O}_g^X=\{o\in\mathcal{O}_g|shape(o)=X\}$. For text document objects we introduce the generic notation $\mathcal{T}_f^b$ indicating a textobject with foreground type t and background type b. Choices for f and b are uniform (u), non-uniform ($\neg u$), graphic (g), image (i), or nothing (), the latter indicating that the foreground or background can be any of the given types.

The geometric structure of the document is the structure induced by the layers in the document. Edges in the geometric structure graph are defined by the *on top* relation, indicating that the object is in a higher layer. The relation is formally defined as $\mathcal{R}_g = \{(o_1, o_2) \in \mathcal{O}_g|\overline{o}_1 \cap \overline{o}_2 \cap \hat{o}_1 \neq \emptyset\}$.

2.4 Logical description

After an analysis of the magazines in the dataset, for each type of document object a set of possible representative logical labels were selected. Object classes which are not frequently appearing in the dataset receive the label "*Other*". Of course these could be refined later. It leads to the following logical attributes:

- logical label
  - text: {Author, Abstract, Bibliography, Caption, Equation, Header, Footer, Foot Note, List, Table, Title, Quote, Paragraph, Page Number, Advertisement, Note, Other};
  - image: {Advertisement, Image Containing Scene Text, Other};
  - graphics: {Separator, Border, Logo, Map, Barcode, Graph, Other};

All of the above document objects with their logical labels could be part of the logical structure of the document. As reading order is most important, we focus on this particular structure.

The reading order is based on the relation *before in reading* denoted by $<_r$. So the logical structure graph has as vertices the logical document objects O_l and there is a directed edge between $o_1, o_2 \in \mathcal{O}_l$ whenever $o_1 <_r o_2$. To be a proper reading order graph it should be a-cyclic. Then, any path in the graph is an independent reading order in the document. If there are multiple paths in the graph they are related to groups of document objects which can be read in arbitrary order. So for logical structure we have: $\mathcal{R}_l = \{(o_1, o_2) \in \mathcal{O}_l | o_1 <_r o_2\}$

3 Document Complexity

The performance of an algorithm on a given dataset depends on two things. Namely the quality of the algorithm itself and the complexity of the data. This complexity of the data is task dependent. When a ground truth is available the complexity can be computed beforehand. It can then be used to order the documents in the dataset so that one can choose a certain level of complexity for designing and testing the algorithm.

Before defining such a set of complexity measures we first consider which steps are performed when doing color document analysis.

3.1 Document analysis steps

We decompose color document analysis into four major steps:
- *page segmentation*: determination of the set of geometric document objects $\mathcal{O}_g$.
- *layout detection*: determination of the relation $\mathcal{R}_g$.
- *logical object classification*: determination of the logical document objects $\mathcal{O}_l$.
- *reading order detection*: determination of the relation $\mathcal{R}_l$.

For each of the steps a complexity measure will be derived: C_1 - Complexity for page segmentation; C_2 - Complexity for layout detection; C_3 - Complexity for logical object classification; C_4 - Complexity for reading order detection. These measures are all defined for a document page and can be computed from the ground truth graphs corresponding to the page. For a document, the complexity of each task is computed by averaging the complexities of individual pages.

3.2 Document complexity for page segmentation

Analyzing the difficulties of the page segmentation algorithms described in literature [11, 13, 19], we identified four main factors that influence the quality of the results. These factors are: *non-uniformity in color*, *shape irregularity*, *picture/text ratio* and the *amount of pictorial document objects containing text*.

Taking into account the above, we consider a document page containing only uniformly colored text objects, having rectangular shapes, on a uniform background to have complexity zero. An example of a document page of maximum complexity is one containing an image in the background, completely covering the page, with text objects with non-uniform color and irregularly shaped

boundaries placed on top of it. For each of the four factors we have designed a complexity measure which is normalized to the range [0,1].

The first measure is based on the textstrings that are either not uniformly colored or have a non-uniform background. Using |.| to denote the cardinality of a set and using the shorthands from section 2.3: $c_1^1 = \frac{|\mathcal{T}_u^g| + |\mathcal{T}_u^i| + |\mathcal{T}_{\neg u}|}{|\mathcal{T}|}$. The second measure considers the percentage of non-regular shapes: $c_1^2 = \frac{|\mathcal{O}_g^P| + |\mathcal{O}_g^E|}{|\mathcal{O}_g|}$. The third complexity measure considers the area of the geometric union of all the shapes corresponding to pictorial document objects, normalized by the width (w) and height (h) of the page: $c_1^3 = \frac{Area(\bigcup_{o \in P} \hat{o})}{w * h}$. Finally, the fourth measure considers the subset of graphics and image objects containing text, denoted by P^{ct} : $c_1^4 = \frac{|P^{ct}|}{|P|}$.

The complexity C_1 for page segmentation is defined as a linear combination of the four complexity features defined above.

$$C_1 = \frac{c_1^1 + c_1^2 + c_1^3 + c_2^4}{4} \tag{1}$$

3.3 Document complexity for layout detection

The problem of detecting multiple layers in color documents has, to our knowledge, not been addressed. The DjVu system [1] can be seen as an exception, however, the system is restricted to one foreground and one background layer, and more importantly the goal is compression not analysis.

As defined in section 2.3 the geometric structure is based on the observation that we perceive a regularly shaped objects as the full object even if it is partly occluded. Clearly the larger the occlusion the less clear this observation can be made. Therefore, to measure the complexity of the decision whether two elements overlap we consider the area of the intersection relative to the union of the two objects. Subsequently this is summed over all object pairs.

$$C_2 = \frac{1}{|\mathcal{R}_g|} \sum_{o_1 \neq o_2} \left\{ \frac{Area(\overline{o}_1 \cap \overline{o}_2)}{Area(\overline{o}_1 \cup \overline{o}_2)} \right\} \tag{2}$$

3.4 Document complexity for logical object classification

In general, logical object classification is based on layout features (visual appearance), content, and possible apriori information about the document class. Here, for deriving a complexity measure we use visual appearance only.

The complexity of the classification problem is determined by the similarity in visual appearance within a logical class and the dissimilarity between different logical classes. However, the variability and separability depend on the geometric features used and the classification method. As we want the complexity measure to be independent of the specific method used, we focus on the number of different classes on the page that have to be distinguished rather.

To be precise, let L_t denote the set of possible text labels for logical objects and let L_i and L_g be defined likewise for image labels and graphics labels. Furthermore, let L' denote the set of labels actually present on the page. Then the complexity measure for logical labeling is given as:

$$C_3 = \frac{1}{3} \left\{ \frac{|L'_t|}{|L_t|} + \frac{|L'_i|}{|L_i|} + \frac{|L'_g|}{|L_g|} \right\} \tag{3}$$

3.5 Document complexity for reading order detection

Analyzing existing methods for reading order detection [17, 16], it is observed that methods work well if document objects are nicely ordered e.g. in a column. Performance degrades if the reading order "jumps" from one object to the other in a non-regular way. To that end we will derive a complexity measure that measures the irregularity of the reading path when it is visiting the different text objects in the document.

Recall that the reading order is defined through the before in reading order relation $<_r$. Each maximal path in the graph with edges defined through this relation gives an independent reading path. Thus we can write the relation $\mathcal{R} = r_o, r_1, ...$ where each $r_i = (o_1, o_2, ..., o_{m(i)})$ is such a maximal path in the graph.

We now define a measure of irregularity for a path r_i. We consider the polyline with vertices p_j for $j = 1, m(i)$ that results if one connects the centres of gravity of the subsequent document objects in r_i. Now for analysis of reading order, based on geometric information, the simplest assumption one can make is that for finding p_{j+1} from p_j one continues in the direction of the vector from p_{j-1} to p_j. In general cases the point will be found in a different direction. Therefore, we define the turning angle α_j at p_j as the angle between the expected direction and the actual direction in which p_{j+1} can be found. The turning angle can be computed using the innerproduct as:

$$\alpha(j) = \cos^{-1} \frac{|\overrightarrow{p_{j-1}, p_j} \cdot \overrightarrow{p_j, p_{j+1}}|}{|\overrightarrow{p_{j-1}, p_j}||\overrightarrow{p_j, p_{j+1}}|} \tag{4}$$

For a page, the average turning angle on any path is computed. Normalizing to [0,1] the complexity measure for reading order detection is given by:

$$C_4 = \sum_{i=1}^{|\mathcal{R}|} \Big(\frac{1}{(m(i)-2)\pi} \sum_{j=2}^{m(i)} \alpha(j) \Big) \tag{5}$$

For the four complexity measures, examples of increasing complexity are presented in Figure 1. To get an insight in the overall distribution of documents in the dataset, Table 1 gives the four complexity values averaged for each document of the UvA Data Set.

Table 1. The average complexity values for UvA Color Document Dataset.

Magazine	*Pages*	C_1	C_2	C_3	C_4
Cosmopolitan	362	0.29	0.09	0.11	0.05
Time	94	0.22	0.22	0.24	0.16
NewsWeek	64	0.22	0.20	0.25	0.29
National Geographic	160	0.20	0.04	0.09	0.02
IEEE Spectrum	106	0.10	0.15	0.26	0.27
The NewYorker	96	0.08	0.04	0.07	0.01
IEEE Computer	132	0.02	0.03	0.07	0.01

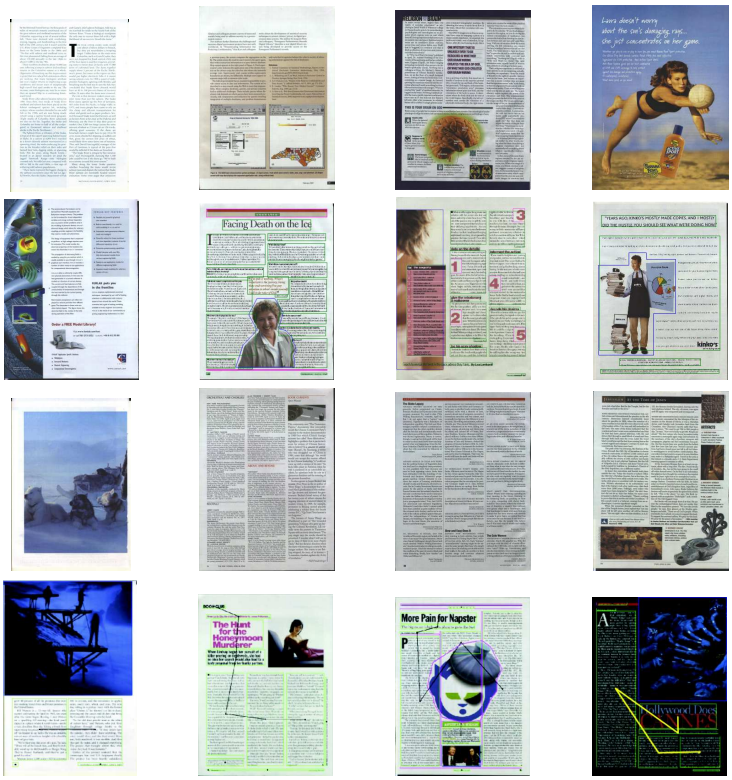

Fig. 1. Example images of various complexities. On the first row, complexity for page segmentation is ranging from a simple page containing only text, to a page with an image as background and polygonal text zones on top($C_1 = 0.0, 0.07, 0.66, 0.74$). On the second row, one can see documents with increasing complexity for layout detection ($C_2 = 0.0, 0.02, 0.12, 0.42$). The simplest example has document objects fully visible. In the most complex example the occluded area is large. Third row shows documents with increasing complexity ($C_3 = 0.03, 0.17, 0.27, 0.48.$) for logical classification. The first document has 1 logical label only, whereas the last document has 12 different labels. On the fourth row are presented documents with increasing complexity for reading order detection ($C_4 = 0.0, 0.20, 0.54, 0.91.$). Paths clearly range from regular to very irregular.

4 Evaluation measures

Complexity measures give an indication of the expected difficulty of a task based on the data, prior to the use of an algorithm. Evaluation measures are needed to compare different algorithms performing the task.

4.1 Precision and recall

Precision and recall are well known evaluation measures. Let us first consider the general definition. Let S be a set of ground truth elements and let S' be the result of any task aiming at deriving the ground truth elements. Then precision and recall are given by: $p = \frac{|S' \cap S|}{|S'|}$ $r = \frac{|S' \cap S|}{|S|}$

When results are not discrete sets, but correspond to regions in the image, the same definitions can be used by using the area of the regions instead of counting the number of elements in a set.

To identify how elements contributed to the precision and recall measures, we can derive the following sets: $Correct = S \cap S'$, $Misdetection = S \backslash S'$ and $FalseAlarm = S' \backslash S$. In the following, the sets S and S' will be made specific for the evaluation of the different tasks.

4.2 Page Segmentation

For evaluation of page segmentation we are faced with the problem that there is no one-to-one correspondence defined between the areas found by the algorithm and the areas given in the ground truth. The same problem was encountered in evaluation of segmentation of a page into text lines by Liang et al. [9, 8]. We base our measures on the method proposed in the reference and extended by Mao and Kanungo in [10]. It is straightforward to use the definitions for the more general objects we consider.

So let us make this more precise. Whereas the ground truth objects are given by $\mathcal{O}_g$ let the result of the page segmentation be given by $\mathcal{O}'_g$. To find the likelihood of a match between elements in the two sets we consider the pairwise precision and recall between the object with index i in $\mathcal{O}'_g$ and the object with index j in $\mathcal{O}_g$ as follows:

$$p_l^{ij} = \frac{Area(\hat{o}'_i \cap \hat{o}_j)}{Area(\hat{o}'_i)} \qquad r_l^{ij} = \frac{Area(\hat{o}'_i \cap \hat{o}_j)}{Area(\hat{o}_j)} \tag{6}$$

Based on the analysis of the values for all possible pairs, Liang et.al. introduced six categories to measure the quality of detection. The first three are similar to the ones we encountered, but the imprecision of the match between two objects is taken into account.

To identify the correctly detected elements, let us define the approximate intersection $X \tilde{\cap} Y$ which gives the pairwise area intersection of all elements for which $r_1^{ij} \approx 1$ and $p_1^{ij} \approx 1$. Further categories are *misdetection* if for all $j : r_1^{ij} \approx 0$ and *false alarm* if for all $i : p_1^{ij} \approx 0$. In addition some more sets are identified to give the category of error, similar to [9, 8].

Note that the above definition requires two thresholds T_l and T_h to judge whether values are close to 0 or 1 respectively. The actual values for these two thresholds were selected by analyzing the p_1^{ij} and r_1^{ij} matrices, for 7 randomly selected pages from the dataset, groundtruthed twice. We found $T_h = 0.80$ and $T_l = 0.05$ to be the appropriate threshold values for the UvA data set.

The above-described measures give accurate local information. The definitions of global precision and recall for a page are:

$$p_l = \frac{Area(\hat{O}_g \cap \hat{O}'_g)}{Area(\hat{O}'_g)} \qquad r_l = \frac{Area(\hat{O}_g \cap \hat{O}'_g)}{Area(\hat{O}_g)} \tag{7}$$

After this task we assume that we have found the match between $\mathcal{O}$ and $\mathcal{O}'$ defined by the pairs of elements in the two sets for which $r_1^{ij} \approx 1$ and $p_1^{ij} \approx 1$. The objects in the matched graphs will be denoted by $\tilde{O}$ and $\tilde{O}'$ respectively. Likewise, relations between those objects in the result and the ground truth are

indicated by $\tilde{R}_g$ and $\tilde{R}'_g$. Further evaluation is restricted to those two object sets and relations to assure that errors made in the page segmentation do not propagate into further evaluation.

4.3 Evaluation of Layout Detection

In the layout detection for color documents, what needs to be evaluated is whether the geometric relations between document objects are found correctly. In our case this corresponds to evaluating whether the edges corresponding to pairs in the overlap relation $\mathcal{R}_g$ are correct.

Following the notation just introduced, this gives the following precision and recall measures for step 2 of the analysis process:

$$p_2 = \frac{|\tilde{R}'_g \cap \mathcal{R}_g|}{|\tilde{R}'_g|} \qquad r_2 = \frac{|\tilde{R}'_g \cap \mathcal{R}_g|}{|\tilde{R}_g|} \tag{8}$$

4.4 Evaluation of Logical Objects Classification

To evaluate the classification of objects into logical classes we have to find the objects in both the ground truth and the results with a specific label. We define: $\tilde{O}^i_l = \{o \in \tilde{O} | \text{logical label}(o) = i\}$ and $\tilde{O}^i_{l'} = \{o \in \tilde{O}' | \text{logical label}(o) = i\}$.

Furthermore, we need the intersection m of the objects in the result and the ground truth according to the labels: $m_{ij} = \tilde{O}^i_l \cap \tilde{O}^i_{l'}$.

By considering the cardinality of each m_{ij} we get the well known confusion matrix for classification. To evaluate the performance on the whole page we need to identify the set of objects M which were classified correctly i.e. all elements in m_{ii}. It leads to the following overall measures:

$$p_3 = \frac{|\bigcup_i (\tilde{O}^i_l \cap \tilde{O}^i_{l'})|}{|\bigcup_i (\tilde{O}^i_{l'})|} \qquad r_3 = \frac{|\bigcup_i (\tilde{O}^i_l \cap \tilde{O}^i_{l'})|}{|\bigcup_i (\tilde{O}^i_l)|} \tag{9}$$

4.5 Evaluation of Reading Order Detection

Evaluation of the final step in the analysis is similar to the layout detection as both are directly computed from the match between the edges of the graph. Again to avoid error propagation only the elements which received the correct label in the previous step are considered when matching the edges in the logical graph. Following the same notation conventions as earlier the relations between those objects in the result and the ground truth are indicated by $\tilde{R}_l$ and $\tilde{R}_{l'}$ respectively. So the final evaluation measures are given by:

$$p_4 = \frac{|\tilde{R}_{l'} \cap \tilde{R}_l|}{|\tilde{R}_{l'}|} \qquad r_4 = \frac{|\tilde{R}_{l'} \cap \tilde{R}_l|}{|\tilde{R}_l|} \tag{10}$$

5 Implementation

Groundtruthing a complex color document is a difficult task. Firstly, because of the many relations between the different objects. Secondly, some subjective choices have to be made. We have therefore defined a set of rules the groundtruther has to obey. These are included in the dataset distribution. Even

when the guidelines are strictly obeyed there will always be a variation between different evaluators as the boundary of an object has to be indicated by hand.

To measure the inherent variability in ground truth definition we performed a variability test. From each magazine we selected 4 document pages for each of the four complexity classes, thus 16 document pages. These were ground truthed 4 times in total by two different evaluators. The 4 ground-truth files obtained by the four evaluation runs are evaluated in pairs, each of them playing the role of ground-truth and result respectively. We use the same evaluation measures used before for each step to compute the variability. The evaluation results for all six possible pairs are averaged to get the variability measure. We concluded from these measures that the groundtruth in UvA-dataset is reproducible up to 97% -99% depending on task.

The ground truth was manually generated using the **GT-UvA** ground-truth editor software, implemented in VisualC++. The user interface allows the user to indicate the perceived shape of document objects by drawing rectangular, circular, elliptical, or polygonal shape around the document objects. The true shape is automatically computed. The layout and logical descriptions are then introduced via a property dialog box. The ground truth can be exported in a plain ASCII or in XML format. The evaluation measures described in Section 4 are implemented in a C++ program, called **Eval**. Eval takes as arguments two text files, one containing the ground truth information, the other the result description of a document page. It prints out the four precision and recall values.

6 Conclusion

To advance the field in color document analyis a well-defined dataset is essential. We have created the UvA color document dataset consisting of over 1000 document pages, ground truthed at the geometric and logical level.

To describe the document pages a graph based model is proposed. Based on the model the process of document analysis has been decomposed into four steps dealing with the vertices or edges of either the geometric graph or the logical graph describing the document.

As the variety of color documents ranges from very simple to complicated structures, we have defined four complexity measures which rank the document complexity for each of the four steps of the document image analysis.

For each of the four steps evaluation measures are defined. All of them are derived from the general evaluation measures precision and recall.

Finally, the documents and associated tools are available on a restricted basis to the research community via a special website.

References

1. L. Bottou, P. Haffner, P.G. Howard, and Y. LeCun. Djvu: Analyzing and compressing scanned documents for internet distribution. In *ICDAR'99, Bangalore, India*, pages 625–628, 1999.

2. W.Y. Chen and S.Y. Chen. Adaptive page segmentation for color technical journals' cover images. *Image and Vision Computing*, 16(3):855–877, 1998.
3. C. Garcia and X. Apostolidis. Text detection and segmentation in complex color images. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 75–78, Istanbul, 2000.
4. H. Hase, T. Shinokawa, M. Yoneda, M. Sakai, and H. Maruyama. Character string extraction from a color document. In *ICDAR'99*, pages 75–78, India, 1999.
5. X.S. Hua, L. Wenyin, and H.J. Zhang. Automatic performance evaluation for video text detection. In *Proc. of the 6th ICDAR'01*, pages 545–550, Seattle, USA, 2001.
6. A.K. Jain and B. Yu. Automatic text location in images and video frames. *Pattern Recognition*, 31(12):2055–2076, 1998.
7. J. Kanai, S.V. Rice, T.A. Nartker, and G. Nagy. Automated evaluation of ocr zoning. *IEEE Transactions on PAMI*, 17(1):86–90, 1995.
8. J. Liang, I.T. Phillips, and R. Haralick. An optimization methodology for document structure extraction on latin character documents. *IEEE Transactions on PAMI*, 23(7):719–734, 2001.
9. J. Liang, R. Rogers, R. Haralick, and I. Phillips. Uw-isl document image analysis toolbox: An experimental environment. In *Proc. of the 4th ICDAR, Ulm, Germany, August 1997.*, pages 984–988, 1997.
10. S. Mao and T. Kanungo. Empirical performance evaluation methodology and its application to page segmentation algorithms. *IEEE Transaction on PAMI*, 23(3):242–256, 2001.
11. G. Nagy. Twenty years of document image analysis in PAMI. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(1):38–62, 2000.
12. T. Perroud, K. Sobottka, H. Bunke, and L. Hall. Text extraction from color documents - clustering approaches in three and four dimensions. In *Proceedings of the 6th ICDAR'01*, pages 937–941, Seattle, USA, 2001.
13. D.S. Ryu, S.M. Kang, and S.W. Lee. Parameter-independent geometric document layout analysis. In *Proceedings of the 2000 International Conference on Pattern Recognition ICPR'00.*, pages 397–400, Barcelona, Spain, 2000.
14. J. Sauvola, S. Haapakoski, H. Kauniskangas, T. Seppanen, M. Pietiklainen, and D. Doermann. A distributed management system for testing document image analysis algorithms. In *Proceedings of the 4th ICDAR'97*, pages 989–995, Ulm, Germany, 1997.
15. J. Sauvola and H. Kauniskangas. MediaTeam Document Database II. CD-ROM collection of document images, University of Oulu, Finland. http://www.mediateam.oulu.fi/MTDB/index.html.
16. L. Todoran, M. Aiello, C. Monz, and M. Worring. Logical structure detection for hetrogeneous document classes. In *Proc. SPIE Vol. 3407, Document Recognition and Retrieval VIII.*, pages 99–111, San Jose, California, 2001.
17. S. Tsujimoto and H. Asada. Major Components of a Complete Text Reading System. *Proceedings of the IEEE*, 80(7):1133–1149, 1992.
18. V.Wu, R. Manmatha, and E.M. Riseman. Textfinder: An automatic system to detect and recognize text in images. *IEEE Trans. on PAMI*, 21(11):1224–1229, 1999.
19. T. Watanabe and T. Sobue. Layout analysis of complex documents. In *Proceedings of the 2000 International Conference on Pattern Recognition ICPR'00.*, pages 447–450, Barcelona, Spain, 2000.
20. B. Yanikoglu and L. Vincent. Pink panther: A complete environment for ground-truthing and benchmarking document page segmentation. *Pattern Recognition Letters*, 31(9):1191–1204, 1997.