BOBCAT-DI:

Document Image Analysis Evaluation Literature Survey

Army Research Laboratory,
Adelphi MD

from

Laboratory for Language and Media Processing
University of Maryland, College Park, MD, USA

February 15, 2009

Table of Contents

List of Figures

No table of figures entries found.

BOBCAT-DI

1 Introduction

1.1 Background

Over the past five decades, evaluation has become increasingly important as the field of document image understanding has developed. A number of independent evaluations have been run by various academic organizations, all focusing on slightly different problems. In recent years, the University of Maryland has developed a number of tools aimed at supporting generic annotation and evaluation of document (and video) data. A set of three reports is being produced from this one year BOBCAT-DI research and development projects. The three reports include

- A *Segmentation and Evaluation Survey*- designed to identify major algorithms, tools and evaluation methodologies in the community,
- The *PETS Software Description*– a toolkit to evaluate segmentation, line detection and image enhancement algorithms, based on BOBCAT-DI requirements and as a response to the state of the art, and
- *Selected Evaluations using the PETS Environment* (This Document)– Evaluations designed to demonstrate the capabilities of the tools and provide a framework for use in operational environments.

It is the hope that this work will lead to a generic repository for evaluation which host data, tools, algorithms and evaluation results for community wide comparison.

1.2 Project Overview

The DoD Sequoyah Foreign Language Translation Program, managed by the interim Sequoyah Transition Mgt Office (STMO) under PEO IEWS, Ft Monmouth, NJ, is intended to address critical linguist shortfalls in US warfighting and intelligence operations through automated language translation capabilities (speech and text) and to provide document image processing and OCR capabilities for cases when material to be translated is paper or document images. To support unbiased, vendor-neutral assessment of technology candidates prior to field testing and deployment, the STMO has initiated a web-accessible, distributed "Best-of Breed Configurable Active Testbed" (BOBCAT) led and operated by ARL and distributed across NRL and AFRL. Yet to be incorporated into the testbed is the capability to assess OCR and other document image processing (DIP) tools. The STMO as well as the ODNI have tasked ARL with integrating document image (DI) processing assessment into BOBCAT, creating BOBCAT-DI . BOBCAT-DI will be used to assess a variety of document image processing capabilities and tools, with a focus on Arabic and other Southwest Asian languages. The image processing and analysis metrics and methods, particularly as applied to document images obtained from

cameras, scanners, etc., is needed to enable assessments that are reliable, robust, and scientifically defensible

1.3 Organization of Report

In this report, we focus primarily on pointers to the evaluation and related literature. However, one of the key contributions of PETS is the image differencing capability. This is best demonstrated on a technology that does document analysis and classification at the pixel level. One such problem is the problem of rule line detection and removal. Before surveying the literature on datasets and evaluation, we briefly highlight several techniques for rule line detection and removal.

2 Annotation and Ground Truthing tools

Publicly available datasets and tool for creating ground truth and metadata are two important aspects for the training and evaluation of document analysis systems in practice. In this section, we present an overview of document image analysis datasets and ground truth tools that are well-known to the document image analysis and understanding community and available in the public domain.

2.1 Datasets

The existence of public domain datasets free researchers from the labor-intensive task of data collection and provides a common ground for objective performance evaluation of different algorithms. The principal goal for constructing document image datasets is to reflect realities of everyday documents (widely varying layouts, complex entities, color, noise etc.) With persistent efforts from the document image analysis community, a number of datasets have been constructed and released.

Some well-known databases are the University of Washington database (UWDB), the MediaTeam database (MTDB), Medical Archive Records Groundtruth (MARG) database, UvA color document database, Tobacco database and IAM handwriting database. Capturing a blend of document categories and a great variety in complexity, these datasets have been widely used as performance benchmarks in the evaluation of document analysis systems.

The UWDB is a collection of three document image databases produced at the University of Washington, with a total about 1800 scanned pages in English and Japanese from a variety of conference proceedings and technical journals. Each document page has associated with it the bounding box information for each zone on the page, text ground truth data for each text zone, and finer level attributes (such as font size, alignment etc.) for each zone.

The MTDB consists of 171 scanned pages of technical journals, newspapers, magazines, and commercial ads. Similar to the UWDB, the MTDB contains block-level ground truth, where each document block has a logical label. However, the MTDB does not include text ground truth data for each text zone, such as OCR text and font information.

A representative domain-specific document collection is the MARG database, which is composed of page images drawn from biomedical journals. Besides the OCR text, its ground truth provides labels of text entities at finer granularity, such as article title, author names, institutional affiliations, abstracts.

Unlike the vast majority of document collections that are composed of binary or grayscale images, the UvA document database contains over 1000 high-quality color pages from full issues of magazines, along with associated geometric and logical region-

level attributes in the ground truth. However, the UvA color document database does not contain OCR text for the text zones.

Construction of test collection from large corpus of real-world document collections offers both realistic scope and complexity for research on document understanding and information retrieval. One initiative is the complex document information processing (CDIP) test collection project. The source repository of the CDIP database is the Tobacco document image library, which contains 42 million pages of documents (in 7 million multi-page TIFF images) obtained from UCSF and released by tobacco companies under the Master Settlement Agreement. It is a realistic dataset for document analysis and retrieval as these documents were collected and scanned using a wide variety of equipment over time, and the quality of OCR text in the ground truth vary significantly. In addition, a significant percentage of it are consecutively numbered multi-page business documents, making it a valuable testbed for various content-based document retrieval approaches. The CDIP document database has been used in TREC evaluations.

In the area of off-line handwriting recognition, a widely used public dataset is the IAM database, which includes 1,066 pages of handwritten full English sentences by approximately 400 different writers. Each page image is scanned as grayscale image and contains the machine printed sentences corresponding to the handwritten content on the page.

2.2 Ground Truth Tools

There are many annotation and visualization tools for editing and displaying document and video images with associated ground truth metadata. In this section we present a brief overview of a few representative tools well known to the document image analysis community.

2.2.1 PinkPanther

Pink Panther is an environment for creating segmentation groundtruth files and for page segmentation benchmarking. Pink Panther consists of two parts: Grounds-Keeper and Cluzo. Grounds-Keeper is a tool for creating ground truth metadata. It visualizes a document image and the corresponding metadata, and also allows users to zone the document image and specify the information for each zone. Groundtruth metadata created by Grounds-Keeper is stored in an ASCII format. Cluzo is a benchmarking tool for collecting the locations, types and severities of segmentation errors on a page as well as information on segmentation performance. Pink Panther was developed in C language. While Grounds-Keeper allows the user to enter segmentation groundtruth, entering text groundtruth is not possible.

2.2.2 ViPER

Video processing evaluation resource (ViPER) is a comprehensive ground truth tool for single-page document images and video clip. It consists of three main components: ViPER-GT, ViPER-PE, and ViPER-Viz. ViPER-GT contains modules for configuring and producing groundtruth information which describes a video sequence. The ViPER-PE module provides performance evaluation capabilities for comparing computed results with appropriate groundtruth information. ViPER-Viz enables a user to visualize groundtruth, analyze results, and evaluate performance. ViPER was developed in Java language.

2.2.3 TRUEViz

TrueViz is another tool developed at the University of Maryland for visualizing and editing document image groundtruth. It supports images in TIFF image format and stores hierarchical document ground truth information in XML format. TrueViz was implemented in Java and supports multilingual text.

2.2.4 GEDI

GEDI is the most recent tool and widely used in the government

3 Document Analysis Evaluation Metrics

3.1 Pink Panther

Yanikoglu et. al. introduced a complete system named Pink Panther (described above) for creating segmentation ground-truth files and bench-marking page segmentation algorithms. The performance was evaluated by comparing the output of a particular page segmentation system on large number of document images with the available groundtruth. Contrary to earlier evaluation approaches which was based on OCR evaluation, they took the region based approach. The region was represented by bounding polygon and different attributes like type, subtype and its parent zone. Using the region's On-pixel content, the overlap between segmentation regions and ground truth regions are determined and hence segmentation problems like missed, merge, split or misclassified ground truth regions and extraneous regions can be analyzed. The detailed shape of polygons or any other representation was ignored. The zoning information was saved in a ASCII format called RDIFF. They used region maps to determine the overlap between different ground truth and segmentation regions. They define match score between overlapping ground truth region **g** and segmentation region **s** as the percentage of On-pixels of the ground truth region covered by **s** minus percentage of On-pixel of **s** outside of **g**. They use region alignment if multiple segmentation regions correspond to a ground truth or vice-versa. Various kind of segmentation error detected by them - noise region, missed ground truth region, vertically split groundtruth region, horizontally split ground truth region, vertically merged ground truth region, horizontally merged ground truth region, mislabeled pixels.

A noise region is segmentation regions which do not overlap with any ground truth region and a missed region is a ground truth region which does not overlap with any

segmentation region. A ground truth region is declared split if no single segmentation region covers it. On the contrary, if multiple segmentation regions match with a ground truth region it is considered as merge. The severity of each mistake and the cost inolved is assessed in terms of ON-pixels, so that the penalty is proportional to the amount of mismatch. Using a startup file for indicating the weight of each error the overall page cost is computed along with individual errors. Their system also allows multiple ordering to handle complex document structure.

3.2 UMD: True-Vis

Mao and Kanungo defined the five-step methodology for quantitive comparison of segmentation algorithms. Their performance evaluation method firstly created mutually exclusive training and test data with groundtruth. Then they defined a performance metric which is easily computable. Firstly, they find the optimal parameters for each algorithm on the training data, using simplex method and then based on the results obtained, evaluation is done and finally statistical and error analysis is done. They used 4 types of textline based error measurements – missed ground truth textlines, Split in boundingbox of ground truth textlines, horizontally merged textlines and falsely detected noise zones.

3.3 Oulu

Okun and Pietikainen emphasized the importance of creating groundtruth for skewed images and proposed a new methodology for automatic generation of ground-truth for skewed images from the given upright images. Their method consists of partitioning the given ground truthed upright image in NXN pixel-blocks, followed by rotation of corner points of this image to obtain the skewed image. Then, block partitioning is again done on the skewed image, and class labels are assigned by finding the correspondence from the pixels in upright image.

3.4 ICDAR - Page Segmentation

Antonacopoulos et. al. summarizes the evaluation methodology and results obtained for an international level page segmentation and region classification competition held at ICDAR 2005. The purpose of this competition was to evaluate the segmentation and region classification methods in realistic circumstances and which can be applied to variety of documents from different sources in real life. Their dataset consisted of scanned documents of commonly occurring publications. They also mentioned that the availability of ground truth for the evaluation of methods, analyzing complex layouts, is scarce. First of such kind of dataset was created at ICDAR 2003 competition. They used an updated dataset from PRIma research lab. For the competition, they chose a subset of documents, depending upon realism in occurrence of each type and general interest to analyze them. They decided to use 30% technical article and 70% magazine article. The ground truthing was done using isothetic polygons for different

types of regions. Evaluation was based on the count of the number of matches between the entities detected by the algorithm and the entities in the ground truth. They used a global match score table for all the entities, entries of which represent the intersection of ON-pixel sets of results and ground truth.

Using the count of elements in a entity of ground truth and result and predetermined weights they defined the detection rate and recognition rate for a particular entity, which sums all the possible mappings like one-to-one, one-to-many and many-to-one. These metric are further combined into single metric called Entity detection metric (EDM) by taking the harmonic average of detection rate and recognition accuracy. A global performance metric was devised by combining the values of entity's detection rate and recognition accuracy.

Discussion on the issues related to design, representation and creation of groundtruth for layout analysis and performance evaluation by the same authors can be found in related publications. They describe the importance of representing the groundtruth in such a way that it can be used in different evaluation contexts. In their view, the evaluation of framework should be able to summarize the performance of a method by providing scores at different level as required. They represented the GT information as an XML file. They build a semi-automatic groundtruth editor which can be used to ground truth ten different types of region. The GT created by them has been used for two international competition held at ICDAR.

3.5 GREC - Graphics

Two contests were held during GREC2005. The third *arc segmentation contest* was organized by Liu Wenyin and attracted 3 participants. Dr. Xavier Hilaire, from LORIA - Université Henri Poincaré, France, won the arc segmentation contest. The second *symbol recognition contest* was organized by Ernest Valveny and Philippe Dosch and had 4 participants. Mr. Feng Ming Feng, a MPhil student from the City University of Hong Kong, won the symbol recognition contest. The contests were a big success, and the inclusion of them has become a key issue in GREC workshops. Contests are useful not only to evaluate the state-of-the-art on algorithms related to different problems in graphics recognition, but also to provide evaluation databases to the community. This time, all the material used in the contests was distributed in a CD among GREC2005 delegates and is also available at the websites of the contests.

3.6 ICDAR 2009

In 2009, numerous isolated evaluations will take place including:

Arabic Handwriting Competition	Call for participation
Handwriting Segmentation Contest	Call for participation

Document Image Binarization Contest (DIBCO'09)	Call for participation
Book Structure Extraction Competition (JIIBSEC'09)	
Handwritten Farsi/Arabic Character Recognition Competition	Call for participation
Online Arabic Handwriting Recognition	Call for participation
Handwriting Competition	Call for participation
Signature Verification	Call for participation

Details can be found at: http://www.cvc.uab.es/icdar2009/competitions.html

4 Rule Line Detection and Removal

4.1 Introduction

Background lines which may also be referred to as “Ruled-Lines” are those lines that appear in a document or on a page to facilitate straight writing. They typically appear in notebooks and music scores, but are also present in graphic drawing and forms. As it used to guide and organized writing, these lines often touch the written content either text or graphic-like objects. Turning to automatic processing of these documents make it is necessary to detect and remove these lines.

Because of noise in the scanning or capture processes, background line detection and removal in binary image documents presents many challenges including:

- Content touching.
- Non-uniform width.
- A lack of continuity.
- Similarity to contents especially if the language’s characters written in connected form like Arabic.
- Dotted Lines.
- Interaction with other noise types like Slat-and-Pepper noise and borders.
- Curvature due to the capturing process.

The remainder of this section will provide a brief overview of some specific literature that addresses these problems. The UMD algorithm, detailed in the accompanying report, will be evaluated using PETS;

4.2 Algorithms

4.2.1 Finding Straight Lines in Drawings

Procedure:

- Find horizontal foreground runs that exceed character width.
- Connect the segments to construct unbroken longer lines.
- Get the estimated angles at which there are possible lines.
- Repeat steps (1) and (2).

Comments:

This paper discusses line detection in image documents. The main algorithm is used to find horizontal lines. All other lines in other directions are detected by the same algorithm after rotating the image document such that these lines are horizontal when processed. The horizontal detection is done by determining the foreground runs exceed the character followed by grouping to adjacent line segments to construct longer lines.

WebLink

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=620618&isnumber=13496

4.2.2 Line Removal and Restoration of Handwritten Strokes

Procedure:

- Make noise removal based on two threshold values for the connected components, namely for number of foreground pixels and aspect ratio for connected components. if the values less than threshold, then remove the connected component.
- The image is divided to blocks.
- Assume that the skew is in the range -10<angle<+10 degrees from the horizontal line. Horizontal projection is used to correct the skew by finding the angle that gives minimum entropy to be the skew angle.
- The Peaks of the horizontal projection in the corrected image are selected as the rows containing lines if it exceeds a threshold and simply deleted.
- The places deleted in each stroke is tried to be restored. The strokes are firstly detected by searching for foreground pixels in certain direction within search region. The restoration is done by filling the gaps that resulted from removal.

Comments:

This paper discusses detection and removal of nearly horizontal lines. The noise and skew removal is done as pre-processing steps. The noise removal is based on connected components analysis. The skew is detected through the rotation and horizontal projection, where the skew angle gives the minimum entropy. The peaks exceeds a threshold in the horizontal projection are considered as lines and its rows are simply deleted. The strokes are then tried to be restored through two stages: stroke detection by finding foreground pixels in certain direction. If any, the second stage is to fill the gaps.

WebLink:

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=04426369

4.2.3 Underline Detection and Removal in a Document Image Using Multiple Strategies

Procedure:

- They use some heuristic rules to decide if the text line has no underline, doubtful underline, or confirmed underline. Different measures are used to configure the heuristic rules such as bounding box height, width, top and bottom vertical edge positions...etc.
    - If there is no underlines, and then the text is the output.

    - If there is a doubtful underline, then it forwarded to further detection step.
    - If there is a confirmed underline, then it forwarded to the removal module.
- The further detection is done using bottom edge analysis.
- The underline removal method depends on the output of the detection process:
    - If there untouched underlines, it is simply deleted. The line width is chosen as the kth element in the descending ordered array of column's heights.
    - The height is the difference between the top and lower foreground pixels.
    - The bottom edge position is estimated as the local median value.
- If there touched underlines, mark the confirmed touched underline:
    - Check the width of pixels under this line, if it is meet some condition it kept as lower part of the character, otherwise it is deleted.
    - Using another heuristic rule to eliminate or keep other marked pixels.
- If there doubtful underlines, an artificial lines is made and use either method from A and B to remove it. But before removing such lines an OCR is run to be assure that it is really underline to reduce wrong removals.

Comments:

This paper discusses the detection and removal of the underlines in documents. The detection is based on some measurements of bounding boxes of the text line and its connected components, such as width, height, top edge, lower edge...etc. The whole process is divided into three stages: detection, removing, and disambiguation modules. The main part of the detection and removing modules is a set of heuristic rules. The removing part itself is divided into two sub-modules, one for untouched underlines that it is simply deleted. The second sub-module is for untouched underlines. The doubtful lines are completed artificially and introduced to one of the two sub-modules. An OCR is used latter to reduce wrong pixel deletion. The main assumption made is the lines always horizontal.

WebLink

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=01334314

4.2.4 Detection of Unknown Forms from Document Images

Procedure:

- Get line candidates that its length exceeds certain threshold horizontally and vertically using morphological operations.
- Divide each line candidate into connected components. Components which do not satisfy a minimum width and width: height ratio are removed.
- Apply some heuristic rules to reduce false alarms, for example remove horizontal lines that did not cross valid horizontal line(s).

- Apply another set of heuristic rules to detect different structures like rectangles ...etc.
- Classify the document based on the detected structures and amount of text.

Comments:

Paper discusses detecting forms in image documents and presents a method to discriminating between different types of forms and non-forms. The process starts with detecting constituting line segments and then using a set of rules to detect structures and forms.

WebLink

http://www.aprs.org.au/wdic2003/CDROM/141.pdf

4.2.5 Line detection in Soccer Video

Procedure:

- Extraction of field green color using color histogram peaks as the field color is the most dominant in the soccer video.
- Replace the logos and the result box by the field color as their locations are fixed for all frames.
- Replacing the advertisements and spectators by field colors, as their locations determined by tracking the boundaries of field's green color.
- Removing non-white pixels.
- Connect the broken lines through replacing any non-white pixel that has any white pixels in the 8-neighbors.
- Remove unwanted objects like the ball, the players T-shirts...etc that may have white color. These objects recognized by its size and length.

Comments:

This paper describes a method for finding the field lines in soccer video frames. The main discriminating feature is the color of the line as field lines has a fixed white color in all soccer videos. So, it is not suitable for binary images as the color is absent. Also, this method is not suitable for document analysis as text is usually touches the ruled lines.

WebLink

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1689104&isnumber=35625

4.2.6 Automatic Table Detection in Document Images

Procedure:

- Preprocessing: Binarization, borders removal, orientation and skew correction.
- A set of morphological operations is performed to connect line's breaks.
- Line detection based on horizontal and vertical foreground runs that exceed twice the average character height and its width is less than the average character height.
- Improving Line estimation using Image/text areas remove as it is surrounded by line segments greater than three times the average character height.
- Determine lines intersections using runs end points coordinates.

Comments:

This paper discusses table detection in image documents. Image is noise cleaning (like borders) and connecting line's breaks are required before table detection. The average character height is used as the main parameter value to decide upon the foreground runs. Also cross points grouping are used for table detection.

WebLink:

http://www.iit.demokritos.gr/~bgat/ICAPR2005.pdf

4.2.7 FAST LINE DETECTION USING MAJOR LINE REMOVAL MORPHOLOGICAL HOUGH TRANSFORM

Procedure:

- Divide the binary image to 32X32 Blocks to run or parallel architecture.
- Compute the Rho (the horizontal distance between the line start and the image reference point) and Alfa (the angle between the rho-axis and the y-axis).
- Make shearing to make the suspected major line segments vertical. The major line is the longest line segment in the 32X32 block.
- Make dilation followed by erosion only on the column containing the major line.
- Remove the major lines using "and" operation on the suspected column.
- Compute the column sum.

Comments:

The paper discuses line detection and removal in documents containing clear lines (no attached text or so). The detection is done using Hough Transform. After detecting lines, they make shearing to make the major lines vertical. The main benefit from shearing to get vertical lines is that using fixed structure elements in morphological operations. The line's removal is done using AND morphological operation. The processing is done on 32X32 blocks of the binary image to exploit the parallel processing.

WebLink

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1199052&isnumber=26990

4.2.8 Line Removal and Restoration of Handwritten Characters on the Form Documents

Procedure:

- Determine junction classes.
- Get foreground runs greater a threshold as a line segments.
- Trace each segment from it top and down to determine the removable points and detect the junction points.
- Restore the broken characters.

Comments:

This paper discusses line detection, removal, and characters' restoration. The assumptions are straight horizontal lines, skew free and noise free documents. The line's detection is detected as foreground runs exceed a threshold. They assumed also that the line junction with text has a predefined set of certain junction points and shapes. The removal and restoration are done based on a set of heuristic rules based on junction points. Junction points are detected by tracing the candidate line segment from its top and bottom, if number of foreground pixels above or done the segment, it is considered as junction point.

WebLink

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=619827&isnumber=13483

4.2.9 Background Line Detection with A Stochastic Model

Comments:

The novelty here is the method adopted to get the lines positions, Hidden Markov Model (HMM) is used for this purpose. The number of states of the HMM is four states: one for start at the bottom, one to end at the top of the documents, and two representing the main core of the model. The observations are the lines projection histogram.

Probability distribution matrix is derived from groundtruth data that is smoothed firstly. Also duration probability is derived from groundtruth data. Vetribi algorithm is used to decode the HMM model. A simplex search method is used to optimize the HMM parameters to reduce errors. Post processing includes finding lines end points and reject lines that its length is less than a threshold.

WebLink

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4624281&isnumber=4624278

4.2.10 A Model-based Line Detection Algorithm in Documents

Procedure:

- Get the Directional Single-Connected chain that is connected vertical runs.
- Filtering DSCC using aspect ratio and skew angle.
- Merge DSCCs to get lines.
- Estimate model parameters:
    - A-Skew angle: the angle that have a line length greater than a threshold and then refined by defining a range around it and choosing the angle that gives maximum (Projection or) line length inside this local range.
    - B- Lines gap: is chosen as the distance between the first two peaks in autocorrelation of the line projections.
    - C- Vertical translation: search in the range (0, gap) and chose that gives you maximum.
- Refine the ending points of each line by choosing the two end points that have maximum

Comments:

This paper discusses the rule-lines detection. The main assumptions are that the lines are parallel and the distance between the lines are nearly equal. They try to find a model of these lines by finding the three parameters namely the skew angle of the lines, the gap between each two lines, and the translation of the first line vertically. The length of line/run is used as main factor in most of the method steps.

WebLink

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1227625&isnumber=27545

4.2.11 Form Frame Line Detection with Directional Single-Connected Chain

Procedure:

- Get the Directional Single-Connected chain that is connected vertical runs.

- Merge DSCCs using “CO -line Distance” and gap condition.
- Make two histograms one for characters height and one for characters width.
- Use the peaks in the two cases to filter the line segments.
- Remove lines that do not share in cells.

Comments:

This paper discusses forms' line detection. The main component of the method is the detection of the Directional Single-Connected chain that is connected vertical runs. Applying a set of rules heuristic rules to merge and filter these DSCC to get the forms lines.

WebLink

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=953880&isnumber=20622

4.2.12 ENERGY BASED LINE DETECTION

Comment:

This paper discusses mainly a method to detect global straight lines in an image. A Hough detector or such a method to detect lines is assumed to be done as a preprocessed step. The main objective is to concatenate small line segments to get the global lines. The problem is formulated as energy minimization, as line points have less energy values. Dynamic programming is used to optimize the energy.

WebLink

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4054911&isnumber=4054517

4.2.13 Detecting Wide Lines Using Isotropic Nonlinear Filtering

Comments:

This paper discusses the detection of wide lines. The detection is based mainly on isotropic nonlinear filter instead of derivatives.

WebLink

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4200762&isnumber=4200748

4.2.14 A Local Approach for Fast Line Detection

Comments:

This paper discusses horizontal and vertical lines in am image using local analysis. An edge detection has assumed to be the input for the algorithm. The image is divided into a number of disjoint blocks. The proposed method is then takes each block and divides it into MxN regions and sum all points in each region to be decided on the existence of line using an accumulator. A filtering method using line direction in neighboring blocks is used to assure the detected local lines are coherent.

WebLink

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1028286&isnumber=22090

4.2.15 A Bayesian Approach to the Hough Transform for Line Detection

Comments:

This paper discusses a probabilistic Hough Transform method to detect lines. The Hough space {Rho/Theta domain} is incremented by a probabilistic values rather than a regular counter. This makes edge points contribute in the Hough space accumulator by different values, as the claim is not all edge points have the same weight due to noise.

WebLink

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1519035&isnumber=32515

5 Additional Citations

5.1 Survey

1. Andersson, P. L. (1969). Optical Character Recognition: A Survey. Datamation. **15:** 43-48.
2. Chen, N. and D. Blostein (2007). A survey of document image classification: problem statement, classifier architecture and performance evaluation. IJDAR. **10:** 1-16.
3. Elliman, D. G. and I. T. Lancaster (1990). A Review of Segmentation and Contextual Analysis Techniques for Text Recognition. Pattern Recognition. **23:** 337-346.
4. Embley, D. W., M. Hurst, et al. (2006). Table-processing paradigms: a research survey. IJDAR. **8:** 66-86.
5. Govindan, V. K. and A. P. Shivaprasad (1990). Character Recognition: A Review. Pattern Recognition. **23:** 671-683.
6. Impedovo, S., L. Ottaviano, et al. (1991). Optical Character Recognition: A Survey. PRAI. **5:** 1-24.
7. Liang, J., D. Doermann, et al. (2005). Camera-based analysis of text and documents: a survey. IJDAR. **7:** 84-104.
8. Mori, S., C. Y. Suen, et al. (1992). Historical Review of OCR Research and Development. PIEEE. **80:** 1029-1058.
9. Tang, Y. Y., S. W. Lee, et al. (1996). Automatic Document Processing: A Survey. Pattern Recognition. **29:** 1931-1952.

5.2 Evaluation

1. Antonacopoulos, A., D. Karatzas, et al. (2006). Ground Truth for Layout Analysis Performance Evaluation. DAS06**:** 302-311.
2. Antonacopoulos, A. and H. Meng (2002). A Ground-Truthing Tool for Layout Analysis Performance Evaluation. DAS02**:** 236 ff.
3. Du, L., A. C. Downton, et al. (1997). Generalized Contextual Recognition of Hand-Printed Documents Using Semantic Trees With Lazy Evaluation. ICDAR97**:** Mo-4B.
4. Ferrer, M. and E. Valveny (2007). Combination of OCR Engines for Page Segmentation Based on Performance Evaluation. ICDAR07**:** 784-788.
5. Hull, J. J. (1996). Performance Evaluation for Document Analysis. IJIST. **7:** 357-362.
6. Kanai, J. (1996). Automated Performance Evaluation of Document Image-Analysis Systems: Issues and Practice. IJIST. **7:** 363-369.

7. Kanungo, T., H. S. Baird, et al. (2002). Special Issue on Performance Evaluation: Theory, Practice, and Impact. IJDAR. **4:** 139-139.
8. Kanungo, T. and S. Mao (2000). Software Architecture of PSET: A Page Segmentation Evaluation Toolkit. UMD.
9. Liang, J., I. T. Phillips, et al. (2001). Performance Evaluation of Document Structure Extraction Algorithms. CVIU. **84:** 144-159.
10. Mao, S. and T. Kanungo (1999). A Methodology for Empirical Performance Evaluation of Page Segmentation Algorithms. UMD.
11. Mao, S. and T. Kanungo (2001). Empirical Performance Evaluation Methodology and Its Application to Page Segmentation Algorithms. PAMI. **23:** 242-256.
12. Mao, S. and T. Kanungo (2002). Software Architecture of PSET: A Page Segmentation Evaluation Toolkit. IJDAR. **4:** 205-217.
13. Okun, O. and M. Pietikainen (2000). Automatic Ground-truth Generation for Skew-tolerance Evaluation of Document Layout Analysis Methods. ICPR00**:** Vol IV: 376-379.
14. Sankur, B., U. Baris, et al. (1997). The Performance Evaluation of Low Level Image Processing Techniques for Document Image Analysis and Recognition. ICDAR97**:** Poste.
15. Todoran, L., M. Worring, et al. (2002). Data GroundTruth, Complexity, and Evaluation Measures for Color Document Analysis. DAS02**:** 519 ff.

5.3 Ground Truth

1. Heroux, P., E. Barbu, et al. (2007). Automatic Ground-truth Generation for Document Image Analysis and Understanding. ICDAR07**:** 476-480.
2. Hobby, J. D. (1997). Matching Document Images with Ground Truth. ICDAR97**:** Tu-2B.
3. Hobby, J. D. (1998). Matching Document Images with Ground Truth. IJDAR. **1:** xx-yy.
4. Nartker, T. A., R. B. Bradford, et al. (1992). A Preliminary Report on UNLV/GT1: A Database for Ground-truth Testing in Document Analysis and Character Recognition. SDAIR92**:** 300-315.
5. Yacoub, S., V. Saxena, et al. (2005). PerfectDoc: a ground truthing environment for complex documents. ICDAR05**:** I: 452-456.
6. Yanikoglu, B. A. and L. Vincent (1995). Ground-Truthing and Benchmarking Document Page Segmentation. ICDAR95**:** 601-604.
7. Yanikoglu, B. A. and L. Vincent (1998). Pink Panther: A Complete Environment for Ground Truthing and Benchmarking Document Page Segmentation. Pattern Recognition. **31:** 1191-1204.

5.4 Datasets

1. Todoran, L., M. Worring, et al. (2005). The UvA color document dataset. IJDAR. **7:** 228-240.

Page Segmentation

2. Amin, A. and R. Shiu (2001). Page Segmentation And Classification Utilizing Bottom-up Approach. IJIG. **1:** 345-361.
3. Antonacopoulos, A. (1998). Page Segmentation Using the Description of the Background. CVIU. **70:** 350-369.
4. Antonacopoulos, A. and R. T. Ritchings (1994). Flexible page segmentation using the background. ICPR94**:** B:339-344.
5. Cantoni, V., L. Cinque, et al. (1997). Page Segmentation Using a Pyramidal Architecture. CAMP97**:** Session 6.
6. Chen, J. L. (1997). A Simplified Approach to the HMM Based Texture Analysis and Its Application to Document Segmentation. PRL. **18:** 993-1007.
7. Cinque, L., S. Levialdi, et al. (2002). Segmentation of page images having artifacts of photocopying and scanning. Pattern Recognition. **35:** 1167-1177.
8. Cinque, L., S. Levialdi, et al. (2003). A system for the automatic layout segmentation and classification of digital documents. CIAP03**:** 201-206.
9. Cinque, L., S. Levialdi, et al. (2002). DAN: An Automatic Segmentation and Classification Engine for Paper Documents. DAS02**:** 491 ff.
10. Cinque, L., L. Lombardi, et al. (1998). A Multiresolution Approach for Page Segmentation. PRL. **19:** 217-225.
11. Das, A. K., S. K. Saha, et al. (2002). An empirical measure of the performance of a document image segmentation algorithm. IJDAR. **4:** 183-190.
12. de Mello, C. A. B. (2004). Image Segmentation of Historical Documents: Using a Quality Index. ICIAR04**:** II: 209-216.
13. de Queiroz, R. L. and R. Eschbach (1997). Segmentation of Compressed Documents. ICIP97**:** III: 70-73.
14. de Queiroz, R. L. and R. Eschbach (1998). Fast Segmentation of the JPEG Compressed Documents. JEI. **7:** 367-377.
15. Etemad, K., D. Doermann, et al. (1994). Page Segmentation Using Decision Integration and Wavelet Packets. ICPR94**:** B:345-349.
16. Etemad, K., D. Doermann, et al. (1995). Multiscale Document Page Segmentation Using Soft Decision Integration. UMD.
17. Etemad, K., D. Doermann, et al. (1997). Multiscale Segmentation of Unstructured Document Pages Using Soft Decision Integration. PAMI. **19:** 92-96.
18. Fujisawa, H., Y. Nakano, et al. (1992). Segmentation Methods for Character Recognition: From Segmentation to Document Structure Analysis. PIEEE. **80:** 1079-1092.

19. Jain, A. K. and S. Bhattacharjee (1992). Text Segmentation Using Gabor Filters for Automatic Document Processing. MVA. **5:** 169-184.
20. Jain, A. K. and B. Yu (1997). Model-Based Document Representation: Application to Page Segmentation. ICDAR97**:** Mo-2B.
21. Jain, A. K. and Y. Zhong (1995). Page segmentation using texture discrimination masks. ICIP95**:** III: 308-311.
22. Jain, A. K. and Y. Zhong (1996). Page Segmentation Using Texture Analysis. Pattern Recognition. **29:** 743-770.
23. Kise, K., A. Sato, et al. (1998). Segmentation of Page Images Using the Area Voronoi Diagram. CVIU. **70:** 370-382.
24. Liu, J. M. and Y. Y. Tang (1998). Distributed Autonomous Agents For Chinese Document Image Segmentation. PRAI. **12:** 97-118.
25. Liu, J. M., Y. Y. Tang, et al. (1996). Adaptive document segmentation and geometric relation labeling: algorithms and experimental results. ICPR96**:** III: 763-767.
26. Mao, S. and T. Kanungo (2000). Automatic Training of Page Segmentation Algorithms: An Optimization Approach. ICPR00**:** Vol IV: 531-534.
27. Nadler, M. (1984). Document Segmentation and Coding Techniques. CVGIP. **28:** 240-262.
28. Patel, D. (1996). Page Segmentation for Document Image-Analysis Using a Neural-Network. OptEng. **35:** 1854-1861.
29. Pavlidis, T. (1991). Page Segmentation by White Streams. ICDAR91**:** 945-953.
30. Pavlidis, T. and J. Y. Zhou (1992). Page Segmentation and Classification. GMIP. **54:** 484-496.
31. Payne, J. S., T. J. Stonham, et al. (1994). Document segmentation using texture analysis. ICPR94**:** B:380-382.
32. Shih, F. Y. and S. S. Chen (1996). Adaptive Document Block Segmentation and Classification. SMC-B. **26:** 797-802.
33. Sylwester, D. and S. Seth (2001). Adaptive segmentation of document images. ICDAR01**:** 827-831.
34. Venkateswarlu, N. B. and R. D. Boyle (1995). New segmentation techniques for document image analysis. IVC. **13:** 573-583.
35. Yang, J. C. Y. and W. H. Tsai (2000). Document image segmentation and quality improvement by moiré pattern analysis. SP:IC. **15:** 781-797. Zlatopolsky, A. A. (1994). Automated Document Segmentation. PRL. **15:** 699-704.

5.5 Layout Analysis

1. Antonacopoulos, A. and D. Bridson (2007). Performance Analysis Framework for Layout Analysis Methods. ICDAR07**:** 1258-1262.
2. Baird, H. S. and D. Ittner (1993). Language-Free Layout Analysis. ICDAR93**:** 336-340.

3. Bulacu, M., R. van Koert, et al. (2007). Layout Analysis of Handwritten Historical Documents for Searching the Archive of the Cabinet of the Dutch Queen. ICDAR07**:** 357-361.
4. Bussi, S. and F. Mangili (1995). A semi-automatic method for form layout description. CIAP95**:** 539-544.
5. Chen, S., S. Mao, et al. (2007). Simultaneous Layout Style and Logical Entity Recognition in a Heterogeneous Collection of Documents. ICDAR07**:** 118-122.
6. Esposito, F., D. Malbera, et al. (1995). A Knowledge-Based Approach to the Layout Analysis. ICDAR95**:** 466-471.
7. Esposito, F., D. Malbera, et al. (1990). An Experimental Page Layout Recognition System for Office Document Automatic Classification: An Integrated Approach for Inductive Generalization. ICPR90**:** I: 557-562.
8. Ishitani, Y. (1997). Document Layout Analysis Based on Emergent Computation. ICDAR97**:** Mo-2B.
9. Kanungo, T. and S. Mao (2003). Stochastic language models for style-directed layout analysis of document images. IP. **12:** 583-596. Leung, M. and T. Twan (1998). Linear Layout Processing. ICPR98**:** Vol I: 403-405.
10. Liang, J. (2006). Processing Camera-Captured Document Images: Geometric Rectification, Mosaicing, and Layout Structure Recognition. Ph.D..
11. Liang, J. and D. Doermann (2002). Logical Labeling of Document Images Using Layout Graph Matching with Adaptive Learning. DAS02**:** 224 ff.
12. Liu, D., D. T. Chen, et al. (2006). Latent Layout Analysis for Discovering Objects in Images. ICPR06**:** II: 468-471.
13. Liu, D., D. T. Chen, et al. (2006). Unsupervised Image Layout Extraction. ICIP06**:** 1113-1116.
14. Watanabe, T. and T. Sobue (2000). Layout Analysis of Complex Documents. ICPR00**:** Vol IV: 447-450.

5.6 Line Detection and Removal

1. Andrea Bonci, T. L. and S. Longhi (2005). A Bayesian Approach to the Hough Transform for Line Detection. IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART A: SYSTEMS AND HUMANS. VOL. 35: 945-955.
2. Arias, J. F., A. Chhabra, et al. (1997). Finding straight lines in drawings. Document Analysis and Recognition, 1997., Proceedings of the Fourth International Conference on. 2: 788--791vol.2.
3. Arvind, K. R., J. Kumar, et al. (2007). Line Removal and Restoration of Handwritten Strokes. Conference on Computational Intelligence and Multimedia Applications, 2007. International Conference on. 3: 208--214.

4. Bai, Z.-L. and Q. Huo (2004). Underline detection and removal in a document image using multiple strategies. Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on. 2: 578--581Vol.2.
5. Busch, A. B. W. W. S. S. and V. Chandran (2003). Detection of unknown forms from document images. Workshop on Digital Image Computing: 141–144.
6. Cai, Z. Q. and J. Tai (2005). Line detection in Soccer Video. Information, Communications and Signal Processing, 2005 Fifth International Conference on: 538--541.
7. Gatos, B. D. D. P. I. P. S. J. (2005). Automatic table detection in document images. Third International Conference on Advances in Pattern Recognition (ICAPR’05), Lecture Notes in Computer Science (3686). 609–618.
8. Lefevre, S., C. Dixon, et al. (2002). A local approach for fast line detection. Digital Signal Processing, 2002. DSP 2002. 2002 14th International Conference on. 2: 1109--1112vol.2.
9. Liu, L., D. Zhang, et al. (2007). Detecting Wide Lines Using Isotropic Nonlinear Filtering. IEEE TRANSACTIONS ON IMAGE PROCESSING. VOL. 16: 1584 - 1595.
10. Rodrigo, R., W. Shi, et al. (2006). Energy Based Line Detection. Electrical and Computer Engineering, 2006. CCECE '06. Canadian Conference on: 2061--2064.
11. Sim, L. C., H. Schroder, et al. (2002). Fast line detection using major line removal morphological Hough transform. Neural Information Processing, 2002. ICONIP '02. Proceedings of the 9th International Conference on. 4: 2127--2131vol.4.
12. Yoo, J.-Y., M.-K. Kim, et al. (1997). Line removal and restoration of handwritten characters on the form documents. Document Analysis and Recognition, 1997., Proceedings of the Fourth International Conference on. 1: 128--131vol.1.
13. Zheng, Y., H. Li, et al. (2003). Background Line Detection with A Stochastic Model. Computer Vision and Pattern Recognition Workshop, 2003. CVPRW '03. Conference on. 3: 23--23.
14. Zheng, Y., H. Li, et al. (2003). A model-based line detection algorithm in documents. Document Analysis and Recognition, 2003. Proceedings. Seventh International Conference on: 44--48vol.1.
15. Zheng, Y., C. Liu, et al. (2001). Form frame line detection with directional single-connected chain. Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on: 699--703.